



Industrial Electrical Event Data Recorder (IE-EDR): A Black-Box System for Electrical Panel Accident Investigation

Dr. D. D. Dhawale¹, Harshal Fulpatil², Abhay Thakare³, Arpit Ganorkar⁴, Ravindra Rahangdale⁵, Saish Badhe⁶

¹ Guide²⁻⁶ Student

Department of Electrical Engineering
Priyadarshini College of Engineering Nagpur

DOI : <https://doi.org/10.55248/gengpi.07.1026.3245>

ABSTRACT

Industrial electrical panels operate under a range of potentially hazardous conditions, such as excessive current, short circuits, poor electrical connections, elevated temperatures, arc faults, and unauthorized access. Although protective relays and monitoring systems help maintain operational safety, information required to investigate an electrical incident may be recorded by separate devices or may lack a common time reference. To address this limitation, this paper proposes an Industrial Electrical Event Data Recorder (IE-EDR), a compact monitoring system designed to preserve operational information from an electrical panel in a black-box format.

The proposed system integrates voltage and current measurement units, temperature and environmental sensors, door and switching-status inputs, a real-time clock (RTC), a microcontroller, and removable data storage. A rule-based event detection mechanism continuously evaluates the monitored parameters against predefined threshold values. Whenever a specified abnormal condition is identified, the system stores the corresponding measurements, status indicators, and timestamp to support subsequent analysis. This paper presents the system architecture, monitoring methodology, event-recording procedure, proposed data structure, fundamental decision logic, safety requirements, and experimental validation plan for a single-phase prototype. The design can potentially be extended to three-phase installations by incorporating suitable measurement devices and electrically isolated interfaces.

Advanced capabilities, including sensor-data fusion, TinyML- and LSTM-based anomaly analysis, power-quality monitoring, tamper-evident data logging, and industrial communication protocols, are identified as possible future developments rather than implemented or experimentally verified functions. As measured laboratory results are not currently available, no claims are made regarding detection accuracy, predictive capability, or overall system performance. Instead, a structured testing methodology is proposed to evaluate event-detection delay, completeness of recorded information, clock synchronization error, false-trigger frequency, and data-storage dependability. The proposed framework aims to provide a practical foundation for improving post-event analysis and fault investigation in industrial electrical panel applications.

Keywords: industrial electrical safety, event data recorder, electrical panel monitoring, embedded systems, fault investigation, event logging, predictive maintenance

1. Introduction

Electrical panels form an important interface between incoming supply, protection devices and industrial loads. Faults or abnormal conditions may arise from excessive current, insulation deterioration, poor electrical contacts, thermal stress, switching operations or environmental exposure. These conditions can lead to equipment damage, production interruption and safety hazards. Following an incident, investigators need to establish not only that a limit was exceeded, but also the sequence and context of events immediately before and after the occurrence.

An Industrial Electrical Event Data Recorder (IE-EDR) is proposed as a dedicated evidence-capture layer for an electrical panel. In the supplied project seminar, the intended parameters include voltage, current, temperature, gas/smoke indication, sound, fire indication, panel-door status and relay or breaker status. The current prototype is described as a single-phase demonstration model with a universal design intention; three-phase use would require suitable current/voltage transducers and channel scaling. Its purpose is complementary to—not a replacement for—protective relays, circuit breakers, interlocks or certified safety systems.

This paper organizes the project as an embedded monitoring and logging system. It distinguishes the presently described threshold-based design from proposed future intelligence features. This separation is important because a black-box recorder should prioritize trustworthy evidence capture, predictable operation and transparent limitations before advanced diagnosis is added.

2. Problem Statement and Motivation

A conventional threshold monitor can indicate that a measured parameter crossed a preset limit, but it may not preserve a synchronized history of multiple signals. Conversely, isolated logs from separate instruments may be difficult to align during an investigation. A dedicated recorder can reduce this gap by using a common time reference and a consistent event schema.

Evidence gap: the post-event record may not show the values and status changes immediately preceding the incident.

Context gap: a single sensor may be insufficient to distinguish a real equipment abnormality from sensor drift, environmental change or an expected operating transition.

Traceability gap: timestamps, sensor-health indicators and a consistent event identifier are needed to interpret stored evidence.

Practical constraint: an affordable prototype must work with constrained embedded hardware, limited storage and a clear safety boundary.

The core research question is: how can a low-cost embedded recorder capture a reliable, timestamped, multi-parameter history of abnormal electrical-panel events, and how should its performance be evaluated before deployment?

3. Objectives and Scope

Identify electrical and environmental parameters relevant to electrical-panel event reconstruction.

Develop an embedded architecture for continuous sensing, rule-based event detection and timestamped storage.

Define a structured event record that supports later review and sequence reconstruction.

Specify a test protocol for threshold detection, time accuracy, event completeness, storage recovery and false-alarm behaviour.

Identify practical extension paths for sensor fusion, edge anomaly detection, communications and tamper-evident audit trails.

Scope: the baseline discussed here is a single-phase demonstration prototype. It is not a certified protective device and must not be used to interrupt power or replace approved protection equipment. Three-phase measurement, waveform-grade power-quality analysis, predictive machine learning and third-party certification are outside the claims of the current prototype and are treated as future work.

4. Literature Review

4.1 Rule-based and learning-based fault diagnosis

Rule-based systems offer transparent decisions and are comparatively easy to implement on small microcontrollers. Their limitation is that fixed thresholds may not represent operating context or gradual changes. Kim et al. demonstrate a broader principle in transformer diagnosis: domain rules can be combined with neural models so that expert-derived information and learned patterns contribute together [1]. This does not directly validate an IE-EDR model, but it supports investigating hybrid rule-and-learning architectures.

4.2 Multi-sensor evidence fusion

Kang, Zhang and Du describe a multi-sensor information-fusion approach for fault early warning in smart-grid equipment [2]. Their work highlights the limitations of isolated sensor evidence and the value of combining evidence sources. For an IE-EDR, voltage/current trends can be considered alongside temperature, door status and relay state. Fusion must still account for sensor uncertainty, different sampling rates, missing values and conflicting indications.

4.3 Embedded monitoring and event records

The supplied project seminar identifies low-cost microcontroller monitoring as the implementation basis and describes an RTC and SD-card module for time-stamped storage. It also cites work on line-fault classification, LSTM-based microgrid fault classification, converter condition monitoring and adaptive relay coordination. These areas are related but have different objectives: fault classifiers identify or categorize faults, whereas the proposed IE-EDR emphasizes a persistent event timeline for post-incident investigation. The recorder can later consume diagnostic outputs, but its primary responsibility remains trustworthy data acquisition and preservation.

4.4 Standards and engineering context

IEC 60255-1 specifies common requirements for measuring relays and protection equipment, while IEC 61439-1 addresses general rules for low-voltage switchgear and controlgear assemblies [3], [4]. These standards provide relevant engineering context; mentioning them does not mean the

prototype complies with them. Any eventual product would need a formal assessment of electrical safety, electromagnetic compatibility, environmental robustness, insulation, enclosure, calibration and applicable local regulations.

5. Proposed System Architecture

Figure 1 shows the conceptual architecture. Sensors feed conditioned and validated signals to a microcontroller. The controller associates measurements with a common time reference, evaluates configured rules, and writes an event record to storage. A review process can then examine the logs. Optional AI and communications blocks are extensions, not requirements for the baseline recorder.

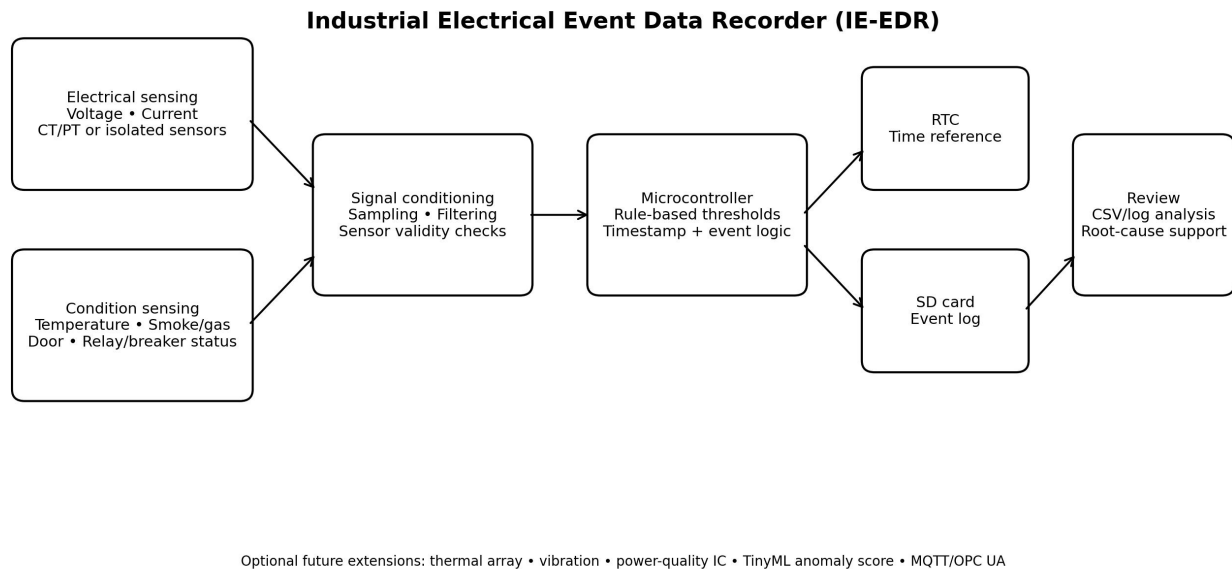


Figure 1. Conceptual architecture of the proposed IE-EDR.

5.1 Sensing layer

Voltage and current should be measured through appropriately rated, isolated transducers selected for the intended panel. Temperature sensing can identify abnormal thermal trends. Smoke/gas, sound, fire indication and door status may be included where the selected sensor and installation are suitable. Relay and breaker auxiliary contacts provide useful state information. Sensor range, accuracy, isolation, response time and calibration must be documented for every channel.

5.2 Processing and time reference

The microcontroller samples or polls the input channels, performs basic plausibility checks, applies configured thresholds and creates event records. An RTC provides calendar time, while a monotonic counter or elapsed-time counter is useful for ordering events even if the wall-clock time is corrected. The design should record clock quality or synchronization status when available.

5.3 Storage and evidence integrity

A removable SD card is suitable for a prototype, but write failures, card removal, filesystem corruption and power interruption must be considered. The implementation should use append-oriented records, sequence numbers, explicit error flags and a recovery strategy. A checksum can detect accidental corruption; it does not by itself provide cryptographic authenticity. A future secure design could use a message authentication code or digital signature with protected keys and a tamper-evident audit process.

6. Event Detection and Recording Method

Proposed event-capture workflow

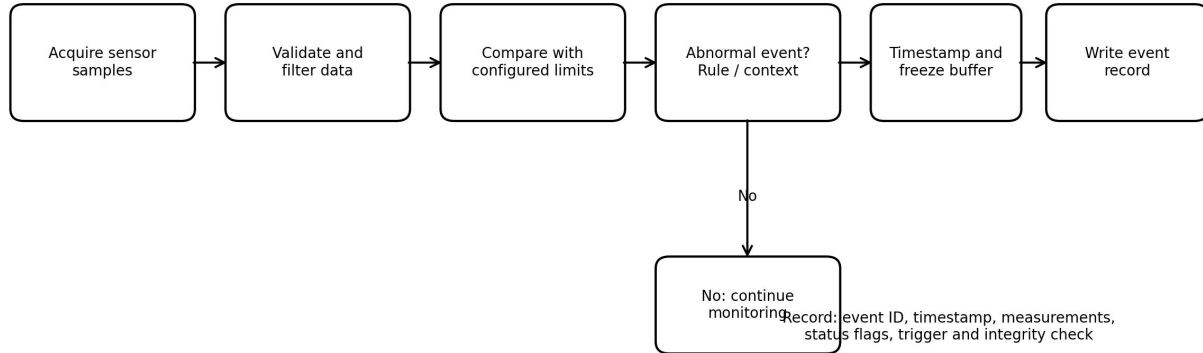


Figure 2. Baseline event-capture workflow.

6.1 Threshold logic

For a measured channel $x_i(t)$, a basic high-limit trigger can be represented as:

$$E_i(t) = 1, \text{ if } x_i(t) > H_i; \text{ otherwise } 0$$

where H_i is a configured high limit. A corresponding low-limit rule may be defined using L_i . Limits must be selected from the intended equipment ratings, sensor calibration and engineering requirements; this paper does not prescribe universal values. Hysteresis and a persistence interval can reduce repeated triggers near a boundary. For example, an event may be confirmed only when a limit remains exceeded for a configured duration, while the raw samples around the trigger are retained for investigation.

6.2 Pre-event and post-event buffering

A circular buffer can preserve recent samples before the trigger. Once an event is confirmed, the recorder can save the buffered pre-event segment and continue recording for a defined post-event interval. This provides more useful context than storing only the first threshold-crossing sample. The buffer size and sample rate must be chosen from available RAM, channel count and desired event duration.

6.3 Suggested event record schema

Header: schema version, device identifier, firmware version and event sequence number.

Time: RTC timestamp, monotonic time and clock-validity flag.

Trigger: event code, source channel, configured threshold, observed value and persistence duration.

Context: all available channel readings, relay/breaker/door states and sensor-health flags.

Integrity: record length, checksum, storage status and optional authenticated digest.

Closure: event end time, recovery state and any missing-sample or write-error indication.

6.4 Event classification

The baseline should use descriptive labels such as HIGH_CURRENT, LOW_VOLTAGE, HIGH_TEMPERATURE, SMOKE_INDICATION, DOOR_OPEN or SENSOR_FAULT. Labels should not imply a proven root cause. For example, high current alone does not prove a short circuit; diagnosis requires additional evidence and suitable investigation.

7. Implementation Considerations

7.1 Hardware selection

The seminar identifies a microcontroller, voltage/current sensors, temperature and other condition sensors, RTC and SD-card storage. A selected sensor must be matched to the expected electrical range and installation environment. The Arduino Mega may be adequate for a low-rate demonstration, but higher sampling rates, waveform analysis or embedded neural inference may require a more capable processor and external analog front end. Direct connection of hazardous mains voltages to a development board is not appropriate; measurement interfaces must provide suitable isolation, protection and enclosure design, and construction/testing should be supervised by qualified personnel.

7.2 Firmware structure

Initialize sensors, storage, RTC and configuration; validate each subsystem before declaring the recorder ready.

Acquire readings at defined intervals and attach quality flags to missing, stale or out-of-range data.

Apply filtering only where appropriate; preserve raw or minimally processed values needed for forensic review.

Evaluate rules with hysteresis and persistence settings; prevent duplicate records from one sustained event unless repeated records are explicitly intended.

Write the event and surrounding buffer to storage, verify the write result, and record any failure.

Expose health status for sensor disconnection, RTC failure, low storage, restart and watchdog reset.

7.3 Reliability and safety boundary

The IE-EDR should be observational and electrically independent of safety-critical trip logic unless a separately engineered and certified interface is designed. Loss of recorder power must not inhibit protection. Power-failure behaviour can be improved with a hold-up supply or orderly shutdown, but such measures require design validation. Sensor saturation, disconnected wiring and stuck values should be treated as faults rather than silently accepted as normal readings.

8. Proposed Evaluation Plan

No numerical experimental results were supplied in the seminar material. Therefore, the following is a test plan and not a report of completed measurements. Testing should begin on a safe, isolated low-voltage test setup and progress only under appropriate supervision.

Test	Method	Metric / acceptance evidence
Threshold correctness	Apply known low-voltage test inputs around configured boundaries	Detection decision agrees with configured rule; document boundary behaviour
Timestamp error	Compare RTC event times against a trusted reference clock	Report mean, maximum and percentile timestamp error
Event completeness	Generate known sequences of sensor changes and compare with saved logs	Measure missing, duplicated and out-of-order records
Detection latency	Measure time between controlled trigger condition and event confirmation	Report latency by channel and configured persistence interval
False alarms	Run representative normal-operation profiles	Count false events per operating hour and identify causes
Storage robustness	Test near-capacity storage, interrupted writes and restart recovery	Document recoverable records, corruption and recovery behaviour
Sensor fault handling	Disconnect or freeze a sensor input in a safe test setup	Confirm explicit SENSOR_FAULT or invalid-data indication
Repeatability	Repeat each controlled scenario multiple times	Report trial count and variability; retain raw test records

Recommended reporting metrics include detection rate, false-alarm rate, event-record completeness, timestamp error, detection latency, storage-write failure rate and recovery success rate. If labeled test cases are available, precision, recall and F1-score may be computed for each event class. Any reported metric should state the test conditions, sample count, threshold settings and sensor calibration status.

9. Future Enhancements

9.1 Multi-sensor fusion

A thermal array such as the MLX90640, a vibration sensor such as the ADXL345 and a dedicated power-quality measurement IC such as the ADE9000 could add complementary information. These devices measure different physical quantities and have different sampling, calibration and installation requirements. Their addition should be justified by the target fault scenarios and validated against reference instruments. Sensor fusion can be rule-based initially, with explicit handling of missing and conflicting evidence.

9.2 TinyML, LSTM and autoencoders

A compact model could learn normal operating patterns from time-series features and produce an anomaly score. An LSTM may capture temporal dependencies, while an autoencoder may identify inputs that are poorly reconstructed by a model trained on normal data. Quantization and TensorFlow Lite for Microcontrollers may help deploy small models on supported microcontrollers, but feasibility depends on RAM, flash, inference latency, sampling rate and model size. An Arduino Mega-class platform may be too constrained for some models; model deployment should be benchmarked before hardware selection. AI should supplement rather than silently override deterministic safety logic.

9.3 Context-aware models and incident triage

A lightweight Isolation Forest or one-class SVM could provide contextual anomaly scores if an appropriate embedded implementation and feature pipeline are available. Operating schedules, load state and ambient temperature may help distinguish expected behaviour from unusual patterns. A triage assistant could summarize evidence and rank likely hypotheses, but it should label outputs as suggestions, show the supporting signals and leave final diagnosis to a qualified investigator.

9.4 Communications and auditability

MQTT or OPC UA interfaces could support integration with supervisory control and data acquisition (SCADA), building management and maintenance systems. Network connectivity should be optional and secured with authentication, authorization, protected updates and network segmentation. A permissioned ledger may offer tamper-evident audit records in multi-party settings, but it adds operational complexity and cannot prove that the original sensor reading was correct. A simpler signed log with protected keys may be a more proportionate starting point.

9.5 Standards collaboration and deployment

Future work may map event fields to applicable industrial data models, consult relevant IEC/IEEE working groups and seek independent testing. Collaboration with vocational training institutions could support maintenance training and interpretation of event logs. Compliance or certification should only be claimed after formal assessment by the appropriate bodies.

10. Limitations

The present project description identifies prototype development and initial testing but does not provide raw data, test counts, measured accuracy, latency or reliability statistics.

Sensor models, ranges, calibration certificates, sample rates and circuit details must be finalized for a reproducible hardware implementation.

Threshold-based monitoring detects configured conditions but cannot guarantee detection of every fault or establish root cause by itself.

AI-based anomaly detection, multi-sensor fusion, blockchain logging and industrial APIs are proposed extensions, not demonstrated capabilities of the current prototype.

Safety and standards compliance have not been established by this design paper.

11. Conclusion

This paper has presented a design-oriented description of the Industrial Electrical Event Data Recorder, a black-box-style monitoring layer intended to preserve a timestamped history of abnormal electrical-panel conditions. The baseline approach combines isolated sensing, microcontroller-based threshold logic, a real-time clock and persistent event storage. Its principal contribution is the focus on reconstructable event evidence rather than fault

indication alone. A structured event schema, pre/post-trigger buffering, sensor-health flags and storage-integrity checks can make the record more useful for maintenance and post-incident investigation. The proposed evaluation plan provides a way to measure detection behaviour and logging reliability without inventing results. Future research can add multi-sensor fusion, edge anomaly detection and secure interoperability, provided each addition is benchmarked and does not compromise the recorder's reliability or the independent operation of certified protection systems.

References

- [1] S. Kim, J. Park, W. Kim, S.-H. Jo, and B. D. Youn, "Learning from even a weak teacher: Bridging rule-based Duval method and a deep neural network for power transformer fault diagnosis," *International Journal of Electrical Power & Energy Systems*, vol. 136, art. 107619, 2022. doi: 10.1016/j.ijepes.2021.107619.
- [2] Z. Kang, Y. Zhang, and Y. Du, "Application of multi-sensor information fusion technology in fault early warning of smart grid equipment," *Energy Informatics*, vol. 7, art. 123, 2024. doi: 10.1186/s42162-024-00433-0.
- [3] IEC, IEC 60255-1:2022, Measuring relays and protection equipment—Part 1: Common requirements. International Electrotechnical Commission, 2022. <https://webstore.iec.ch/en/publication/59762>
- [4] IEC, IEC 61439-1:2020, Low-voltage switchgear and controlgear assemblies—Part 1: General rules. International Electrotechnical Commission, 2020. <https://webstore.iec.ch/en/publication/32338>
- [5] A. D. Devi, N. Sinha, L. C. Saikia, and S. Rhison Singh, "Detection and Classification of Faults in an Islanded Microgrid Using LSTM Model and its Real Time Validation," 2023 IEEE SILCON, 2023. Bibliographic details are reproduced from the supplied seminar slides and should be checked against the publisher record before submission.
- [6] S. K. Maurya, R. K. Panda, A. Mohapatra, and A. Sharma, " ΔV - ΔI Plane-Based Line Faults Detection and Classification in DC Microgrid," *IEEE Transactions on Power Delivery*, 2023. Bibliographic details are reproduced from the supplied seminar slides and should be checked against the publisher record before submission.
- [7] T. P. Meher, S. N. Singh, and A. Mohapatra, "Condition Monitoring of Power Electronic Converters: Present Status and Key Challenges," 2021 IEEE CATCON, 2021. Bibliographic details are reproduced from the supplied seminar slides and should be checked against the publisher record before submission.
- [8] A. K. Soni, A. Kumar, R. K. Panda, A. Mohapatra, and S. N. Singh, "Adaptive Coordination of Relays in AC Microgrid Considering Operational and Topological Changes," *IEEE Systems Journal*, 2023. Bibliographic details are reproduced from the supplied seminar slides and should be checked against the publisher record before submission.
- [9] R. Dora, M. G. Basha, and K. Chenchireddy, "Fault detection in distribution system using Arduino micro-controller," *AIP Conference Proceedings*, vol. 3361, art. 030070, 2025. doi: 10.1063/5.0299535. This reference was included in the user-provided literature list; verify publication details before final submission.
- [10] K. Thakur, D. Dwivedi, and M. Pal, "Advancements in arc fault detection for electrical distribution systems: A comprehensive review from artificial intelligence perspective," *arXiv:2311.16804*, 2023/2024. <https://arxiv.org/abs/2311.16804>
- [11] TensorFlow, "TensorFlow Lite for Microcontrollers," official documentation. <https://www.tensorflow.org/lite/microcontrollers>