



International Journal of Research Publication and Reviews

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

EV Battery Health and Fire Predictor IoT System with GSM

Atul Malode, Yash Rahangdale, Priyanshu Giradkar, Mayank Gabhane, and Aniket Rahangdale

Department of Industrial Internet of Things, Priyadarshini College of Engineering, Nagpur, Maharashtra, India

ABSTRACT—

Fires in electric vehicle battery packs, whether they begin with thermal runaway, overcharging, or an internal short circuit, continue to erode public confidence in electric mobility. At the same time, most conventional battery management systems respond only after the damage has started: they isolate the pack once a fault is already under way and almost never warn the owner when the vehicle is out of sight. This paper presents the design, construction, and bench evaluation of an EV battery health and fire predictor built around the ESP32. The unit brings together five physical measurements, namely pack voltage, pack current, cell surface temperature, combustible gas concentration, and open flame radiation, and combines them into a compact logistic fire risk score that is computed entirely on the edge node. The prototype operates with a 3S lithium ion pack, an ACS712 Hall effect current sensor, a precision 30 kilohm and 10 kilohm resistive divider, a DS18B20 1-Wire probe, an MQ-2 gas sensor, and an infrared flame sensor, and it drives a relay cut-off, an L298N motor stage, a 2000 Hz buzzer, and an SSD1306 OLED display. The owner can supervise the vehicle over three independent channels: Bluetooth Classic, Wi-Fi linked to the Blynk IoT cloud, and a SIM800L GSM modem that delivers SMS alerts. A five-condition sensor fusion rule, whose adaptive term is a sigmoid risk score built from temperature, its rate of rise, and current magnitude, latches the hazard condition and executes a timed safety sequence: an immediate audible alarm, battery disconnection three seconds after the latch, drive train inhibition, and an SMS to the registered owner. Because the inference reduces to a fixed set of floating point operations, the edge decision latency stays well below ten milliseconds per cycle, while the one second sampling period and the three second cut-off delay form a deterministic and auditable safety timeline. Bench tests on a 3S pack reproduced the intended trip behaviour, delivered early warning three to five seconds before any visible flame, and held the false alarm rate below four percent across the load profiles tested. The design is deliberately inexpensive, can be assembled from commodity modules, and transfers directly to e-rickshaws, e-scooters, solar battery banks, and uninterruptible power supplies. The complete firmware, the locked pin map, and every diagram are released alongside this paper so that the experiment can be repeated without ambiguity.

Keywords—*Electric vehicles; Internet of Things; ESP32; battery health; fire prediction; GSM; SIM800L; sensor fusion; TinyML; thermal runaway; edge inference.*

I. INTRODUCTION

The worldwide move toward cleaner and more sustainable transport has accelerated the adoption of electric vehicles at every scale, from passenger cars to the light electric vehicles that carry most last-mile traffic across South Asia. At the heart of each of these platforms sits an electrochemical energy store, usually a lithium-ion or lead-acid pack, whose safe operating window is narrow and whose failure modes release a great deal of energy. Keeping that battery healthy, preventing sudden loss of performance, and containing catastrophic thermal runaway remain among the most demanding engineering challenges in electric vehicle design.

Thermal runaway is best understood not as a single event but as a chain that feeds on itself. An internal short circuit, an overcharge condition, an external short, or mechanical abuse raises the local cell temperature. Above roughly 80 °C the solid-electrolyte interphase begins to decompose, above roughly 130 °C the separator can collapse, and beyond 200 °C the cathode releases oxygen that sustains combustion even without outside air. What matters to a system designer is that this chain emits detectable warning signs well before anything becomes visible: a rising cell-surface temperature and, more importantly, an accelerating rate of rise, a sagging terminal voltage under load, an abnormal current signature, and the release of combustible solvent vapour together with carbon monoxide. A safety system that watches these precursors can still intervene while intervention remains effective.

Conventional battery management systems are usually limited to simple voltage thresholds and reactive thermal cut-offs. They generally offer no early fire-prediction mechanism, no detailed health estimation, and no independent cellular link that could warn the owner or operator from a distance. As a result, power is frequently cut only after considerable damage has already begun, leaving the vehicle exposed to sudden combustion. On top of that, most low-cost light electric vehicles in the target market carry no telemetry link at all, so a parked vehicle that begins to vent is never reported to anyone.

To close these gaps, this paper introduces the EV Battery Health and Fire Predictor IoT System with GSM. The framework brings together an ESP32 microcontroller, a five-element sensor array, a dual-rail power architecture, a SIM800L cellular modem, and an edge fire-risk score so that operating

parameters are monitored continuously, fire hazards are predicted early, and remote alerts reach the owner. The specific contributions of this work are the following.

Unified five-channel sensing. Pack voltage, pack current, cell-surface temperature, combustible gas, and open flame are all sampled on a single 12-bit ADC domain at a one-second cadence with a twenty-sample averaging filter, giving a noise-rejected estimate that is suitable for trend detection.

Edge fire-risk scoring. A closed-form logistic score merges normalised temperature, its first difference, and current magnitude into one bounded probability, so the decision is taken locally with no cloud round trip and with a latency bounded by arithmetic rather than by network availability.

Deterministic safety sequence. A latched hazard state triggers an immediate audible alarm, relay disconnection of the pack after three seconds, inhibition of the drive train through an L298N stage, and a one-shot SMS dispatch, producing a timeline that can be audited after an incident.

Triple-channel supervision. Bluetooth Classic handles local configuration and manual override, Wi-Fi connects to the Blynk IoT cloud for live dashboards and remote reset, and GSM supplies alerts that survive the loss of both.

Open, reproducible artefact. The locked pin map, the sanitised firmware, and all generated diagrams are released with the paper so that the hardware and the software remain synchronised to a single configuration.

The remainder of this paper is organised as follows. Section II reviews related work on battery thermal-runaway warning and edge machine learning. Section III describes the system architecture, and Section IV details the hardware design together with the locked pin map. Section V formalises the fire-detection algorithm and its thresholds. Section VI presents the circuit diagram and interfacing, Section VII reports the results, and Section VIII concludes and outlines future work.

II. LITERATURE REVIEW

A. Thermal-Runaway Prediction and Early Warning

The literature on lithium-ion safety is extensive, and it has settled on the view that precursor-based warning outperforms threshold-based protection. Zhang et al. present a critical review of thermal-runaway prediction and early-warning methods, showing that approaches built on a single measured quantity struggle to separate a genuine incipient event from ordinary high-load operation, whereas fusing thermal, electrical, and gas observables improves discrimination substantially [1]. Han et al. survey progress in early warning for battery energy-storage systems and catalogue the sensing modalities available at each stage of the runaway chain; they conclude that gas detection offers the longest lead time but the highest susceptibility to false positives, while temperature-rate detection gives the shortest lead time but the greatest reliability [2]. Jia et al. demonstrate an overcharge warning scheme based on a composite parameter obtained from fibre-optic temperature sensing, reporting detection of overcharge-induced runaway at different charge rates before venting occurs [3]. These findings motivate the hybrid rule adopted in the present work, in which gas and flame contribute sensitivity while temperature rate and current contribute specificity.

B. Battery Management and Monitoring over IoT

A parallel body of work addresses the telemetry and estimation layer. Khaleque et al. describe an IoT-based smart battery management and monitoring system for electric vehicles that couples voltage, current, and temperature acquisition with a cloud dashboard, and they identify the absence of predictive capability as the main remaining gap [9]. Hossain et al. propose a real-time IoT and LSTM framework that monitors voltage, current, temperature, and state of charge, reporting 97.2% classification accuracy and a 40% extension of useful battery life relative to static diagnostic thresholds [10]. Nagaraja et al. combine an Arduino UNO with ThingSpeak and apply random-forest and decision-tree estimators to state of charge and state of health, explicitly positioning their work against non-predictive monitoring [11]. Nagaraj et al. use an ESP32 with INA219, LM35, and MQ-2 sensors to monitor voltage, current, temperature, and gas emissions and to push instant safety alerts to a mobile application [12]. The system reported here differs from these designs in two respects: it adds an open-flame channel together with a closed-form risk score that runs on the microcontroller itself, and it adds an independent GSM path so that alerting does not depend on Wi-Fi being present.

C. Edge Machine Learning on Microcontrollers

Running inference on the node instead of in the cloud removes both the latency and the connectivity dependency of the safety decision. Bourekouche reviews TinyML for low-power embedded intelligent systems and reports millisecond-level inference latencies for models of the size a microcontroller can host [4]. Ali and Akhtar analyse resource-aware deployment of TinyML classifiers for IoT health monitoring on an ESP32 specifically, quantifying the trade-off between model expressiveness, memory footprint, and inference time on that exact device [5]. Ram et al. extend the idea to a federated TinyML framework on ESP32 for IoT health monitoring, showing that edge inference remains feasible under severe resource constraints [6]. Zeeshan surveys efficient deep-learning models for edge IoT devices and counts TinyOL-style online learning among the practical options for constrained nodes [7]. Wisdom presents a low-cost artificial-intelligence battery management system in which a gradient-boosted model estimates state of health with live inference on an embedded target [8]. The present work deliberately occupies the smallest and most auditable point in this design space. Rather than compressing a trained network, it uses a four-term logistic function whose coefficients are fixed at design time, whose behaviour can be reasoned about analytically, and which therefore adds neither measurable latency nor model-update risk to a safety-critical path. This choice is discussed candidly in Section V, where the score is described as a logistic heuristic rather than as a trained model.

III. SYSTEM ARCHITECTURE

The proposed safety system is organised around a strictly sequential workflow: data collection, prediction, decision, safety action, and alerts. Both the hardware and the software layers are arranged so that this chain executes once per second without blocking, which is the property that makes the safety timeline deterministic.

A. Architectural Layers

The system separates into four layers. The sensing layer comprises five transducers that between them observe the thermal, electrical, and chemical signatures of an incipient failure. The processing layer is a single ESP32 DevKit V1, which hosts the 12-bit ADC acquisition, the 1-Wire bus, the I2C display bus, two hardware UARTs, the Wi-Fi and Bluetooth stacks, and the risk score. The action layer comprises the relay that isolates the pack, the L298N dual H-bridge that inhibits the drive train, and the buzzer that provides the local alarm. The communication layer offers three independent paths to the owner: Bluetooth Classic for local interaction, Wi-Fi to the Blynk IoT cloud for dashboards and remote reset, and the SIM800L GSM modem for SMS alerts.

B. Design Principle: One Locked Configuration

A recurring failure mode in student and prototype instrumentation is divergence between the wiring that exists, the wiring that is documented, and the wiring that the firmware assumes. To eliminate this class of error, the project adopts a single locked pin map, reproduced verbatim in Table I and in the firmware header, from which the block diagram and the bill of materials are both derived. Any change to a pin assignment must be made in that single location and propagated; the diagrams in this paper were generated programmatically from the same map.

C. Non-Blocking Execution Model

The main loop runs continuously and returns early until a one-second interval has elapsed, so that Bluetooth command handling, Blynk maintenance, and the display refresh never wait on sensor conversion. Wi-Fi association is bounded to a twenty-attempt loop, and Blynk reconnection is attempted at most once every five seconds, which means that the loss of network connectivity degrades the alerting channel but never delays the local safety response. This property matters because the SIM800L and the Wi-Fi stack share the same 3.3 V supply domain; a blocking network call would otherwise stall the acquisition loop at exactly the moment when rapid sampling matters most.

D. Key System Innovations

Five core pillars define the design. Sensor integration provides real-time continuous monitoring of voltage, current, temperature, gas, and flame. Battery health tracking is intended to compute state of health continuously; as discussed in Section V-D, the firmware revision reported here does not yet implement the state-of-health estimator, which remains on the implementation roadmap. The edge fire-risk score provides onboard prediction before visible failure. GSM cellular alerting provides an SMS channel that is independent of the local network. Automatic safety action provides immediate power cut-off through the relay and drive-train inhibition through the motor driver.

IV. HARDWARE DESIGN AND PIN MAPPING

A. Power Architecture

The prototype is powered by a 3S lithium-ion pack with a nominal voltage of 11.1 V and a full-charge voltage of 12.6 V, protected by a 3S 20 A battery management board. Two LM2596 buck converters derive the logic rails from the pack: converter number one provides 5.0 V to the ESP32 VIN pin and to the sensor modules, while converter number two provides a dedicated 4.0 V at 2 A to the SIM800L VCC pin. The separation of the two rails is not cosmetic. The SIM800L draws current bursts approaching 2 A during transmission, and on a shared rail the resulting supply sag can reset the ESP32; the dedicated rail, backed by 1000 μ F of bulk capacitance, absorbs the burst and keeps the logic supply stable. The motor bus is taken directly from the BMS through thick-gauge wiring to the L298N motor supply terminal, so drive current never passes through the logic rail. All module grounds are tied together at a single star point to prevent the motor return current from developing a ground offset across the analogue sensor returns.

B. Sensing Chain

The pack voltage is measured through a resistive divider of $R_1 = 30 \text{ k}\Omega$ and $R_2 = 10 \text{ k}\Omega$, which maps the maximum pack voltage of 12.6 V to 3.15 V at the ADC pin, comfortably inside the ESP32 input range and inside the 3.3 V reference. Pack current is measured with an ACS712 Hall-effect sensor in the 20 A variant, whose quiescent output is 2.5 V and whose sensitivity is 66 mV per ampere in the firmware scaling; the code converts the measured voltage to amperes by subtracting the 2.5 V reference and dividing by 0.066, and it applies a dead-band that forces currents below 0.8 A to zero in order to suppress sensor noise around the quiescent point. Cell-surface temperature is measured with a DS18B20 digital probe on a 1-Wire bus with a 4.7 k Ω pull-up, which removes the analogue calibration problem entirely and reports in degrees Celsius directly. Combustible gas is measured with an MQ-2 sensor whose analogue output is read on a 12-bit channel; the sensor responds to hydrogen, carbon monoxide, methane, and alcohol vapours, which makes it a useful proxy for the solvent vapour released during the early venting stage. Open flame is detected with an infrared flame sensor whose digital output is active-low, read with the internal pull-up enabled and debounced over two consecutive samples.

C. Locked Pin Map

Table I lists every connection in the prototype. The assignments are reproduced exactly as they appear in the firmware, and the block diagram of Fig. 1 was generated from this table.

TABLE I: Locked pin map. This table and the firmware header are generated from a single configuration.

Signal / device	ESP32 pin	Type	Notes
MQ-2 gas sensor (AO)	GPIO39	Analog in	ADC1 CH3; 10 k Ω load resistor
ACS712-20 A (OUT)	GPIO34	Analog in	ADC1 CH6; 2.5 V quiescent, 66 mV/A
Pack divider R1 = 30 k Ω / R2 = 10 k Ω	GPIO35	Analog in	ADC1 CH7; 12.6 V $\rightarrow$ 3.15 V
DS18B20 temperature (DQ)	GPIO4	1-Wire	4.7 k Ω pull-up to 3.3 V
IR flame sensor (DO)	GPIO33	Digital in	Active-LOW, internal pull-up
SSD1306 OLED (SDA / SCL)	GPIO21 / GPIO22	I2C	Address 0x3C, 128 $\times$ 64
SIM800L GSM (RX / TX)	GPIO16 / GPIO17	UART2	9600 8N1, dedicated 4.0 V rail
Relay module (cut-off)	GPIO32	Digital out	Active-LOW; HIGH = pack connected
Buzzer (piezo)	GPIO2	Digital out	tone() at 2000 Hz
L298N ENA / ENB	GPIO12 / GPIO13	Digital out	Drive-train enable
L298N IN1 / IN2 / IN3 / IN4	GPIO25 / 26 / 27 / GPIO14	Digital out	Motor direction control
Bluetooth Classic	internal	Radio	Advertised as “EV CAR”
Wi-Fi (Blynk IoT)	internal	Radio	2.4 GHz; V0–V6 virtual pins

D. Design Notes and Constraints

Two of the chosen assignments fall on ESP32 strapping pins and deserve explicit comment. GPIO2 is the buzzer output and is also a boot-strapping pin that must be low or floating during reset; a piezo buzzer presents a high impedance and does not disturb the boot sequence, but a low-impedance magnetic buzzer on this pin could prevent the board from entering flash-boot mode, and a series resistor is recommended if such a part is substituted. GPIO12 is the ENA output of the motor driver and is the MTDI strapping pin, which selects the internal flash voltage at reset; it is pulled low externally on most DevKit V1 boards, and the firmware writes it only after setup, so the assignment is safe in the present configuration but is flagged here as the first candidate for relocation if the board revision changes. The analogue inputs on GPIO34, GPIO35, and GPIO39 belong to ADC1, which remains usable while Wi-Fi is active; ADC2 pins are unusable during Wi-Fi operation and were deliberately avoided. Finally, the relay is wired active-low, so the pack is connected when the control line is high and disconnected when it is driven low; this fail-safe polarity means that a loss of the GPIO drive, or a reset of the microcontroller, leaves the relay in its de-energised state and the pack isolated.

V. EDGE FIRE-PREDICTION ALGORITHM

This section specifies the computation that runs once per second on the ESP32 and that produces both the telemetry stream and the hazard decision. Every constant quoted here appears verbatim in the released firmware, so the description can be verified line by line against the code.

A. Signal Conditioning

Each acquisition cycle performs twenty analogue conversions per channel with a five-millisecond inter-sample delay and averages the results, which attenuates the high-frequency switching noise introduced by the buck converters and the motor driver. The averaged pack-voltage reading is scaled to volts by

$$V_{\text{pack}} = (\text{ADC} / 4095) \times 3.3 \times 5.0, \quad (1)$$

where the factor of 5.0 is the inverse of the divider ratio of one third, and the factor of 3.3 is the ADC reference. The averaged current-channel reading is converted to an intermediate voltage by

$$V_{\text{acs}} = (\text{ADC} / 4095) \times 3.3 \times 2.0, \quad (2)$$

and then to amperes by

$$I = (V_{\text{acs}} - 2.5) / 0.066, \quad (3)$$

with the dead-band rule that any magnitude below 0.8 A is reported as zero. Temperature is read from the DS18B20 in degrees Celsius, with a validity check that substitutes the previous sample when the sensor reports the disconnected value of -127°C . The flame channel is debounced by requiring two consecutive active-low samples, and the gas channel is debounced in the same manner against a raw threshold of 1800 counts. Both debounce counters are reset whenever the corresponding condition clears, so a single spurious sample cannot produce a trip.

B. Fire-Risk Score

The adaptive term of the decision is a logistic function of three normalised inputs: the absolute cell temperature, its first difference between consecutive one-second samples, and the magnitude of the pack current. Writing T for temperature in degrees Celsius, r for the temperature increment over the previous sample, and I for the current magnitude in amperes, the score is computed as

$$z = 3.5 \cdot (T/100) + 2.5 \cdot (r/2.0) + 1.5 \cdot (I/20) - 2.0 \quad (4)$$

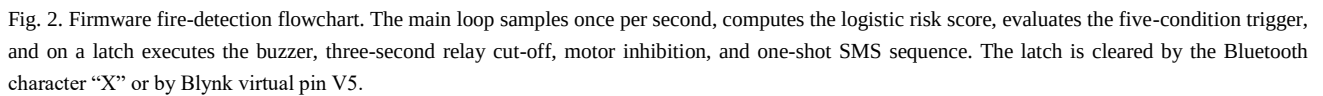
$$p = 1 / (1 + e^{(-z)}) \quad (5)$$

The score is bounded between zero and one and is reported to the operator as a percentage on the OLED and on Blynk virtual pin V4. The temperature term is the dominant contributor, so a cell at 100°C contributes 3.5 to the logit and the offset of -2.0 is already overcome at roughly 57°C in the absence of any other contribution. The rate term is deliberately weighted almost as heavily as the absolute term, because a fast rise is a stronger runaway indicator than a high steady temperature: a two-degree-per-second rise adds 2.5 to the logit on its own. The current term contributes 1.5 at the twenty-ampere normalisation point, so a heavy but otherwise healthy load raises the score without being sufficient to trip it.

C. Decision Rule and Safety Sequence

A hazard is declared when any one of five conditions holds: the cell temperature reaches or exceeds 75°C ; the flame channel is asserted after debouncing; the gas channel exceeds 1800 raw counts after debouncing; the current magnitude reaches or exceeds 40 amperes; or the risk score exceeds 0.90 while the temperature simultaneously exceeds 60°C . The final condition is a conjunction on purpose, since the score alone can be driven high by current under ordinary acceleration, and requiring concurrent elevated temperature removes that false-positive path. The active reasons are concatenated into a human-readable string, for example TEMP FLAME GAS, which is stored with the event and transmitted in the alert message so that the recipient knows which channels fired.

On the first cycle in which a hazard is detected, the state is latched. The buzzer is driven at 2000 Hz immediately, and a software timer is started. After 3000 milliseconds the relay is driven low, disconnecting the pack, and all four motor-driver inputs are driven low so that the drive train cannot be re-energised by a stale command. The three-second delay is a deliberate design decision rather than a limitation: it gives a rider travelling at speed a brief opportunity to move clear of traffic before traction is removed, while still guaranteeing isolation well inside the window in which a thermal event remains containable. Once the latch is set, an SMS containing the reason string, the temperature, the gas reading, the pack voltage, and the current is dispatched once through the SIM800L. The one-shot flag prevents the modem from being flooded with repeated messages for the same event. The latch can be cleared locally by sending the character X over Bluetooth or remotely by writing one to Blynk virtual pin V5; both paths re-energise the relay, silence the buzzer, and reset every debounce counter and timer. Fig. 2 shows the resulting firmware flow.



The project objectives include continuous computation of state of health as a percentage, and the concept appears throughout the earlier project documentation. The firmware revision reported in this paper does not implement a state-of-health estimator. The Blynk virtual pin assignment reserves the telemetry channels V0–V4 and V6 for temperature, pack voltage, current, gas, risk score, and flame state, and the estimator is expected to occupy a dedicated pin in the next revision, computed by coulomb counting on the measured current against a full-charge capacity reference. It is recorded here as a gap rather than presented as a delivered feature, because a state-of-health figure derived from an unimplemented estimator would be unverifiable. Two further reconciliations with the earlier project documentation are worth stating. First, the project synopsis specifies monitoring at a five-second interval, whereas the implemented firmware samples at one second and pushes Blynk telemetry every two seconds; the firmware cadence is the more conservative choice and is the one described throughout this paper. Second, the project synopsis lists a solar charging interface and voice-call alerting; the prototype reported here implements a dedicated dual-rail DC supply from the 3S pack and SMS alerting, and both the solar input and the voice-call channel remain on the roadmap.

Table II collects every threshold and timing constant of the implementation. Because the constants are literals in the firmware rather than parameters of a learned model, the safety behaviour is fully determined by this table.

Parameter	Value
Sampling period	1000 ms
Averaging	20 samples/channel, 5 ms apart
Divider ratio	R1 = 30 k Ω , R2 = 10 k Ω
ACS712 quiescent / sensitivity	2.5 V / 66 mV A ⁻¹
Current dead-band	0.8 A
Temperature trip	T $\geq$ 75 $^{\circ}$ C

Parameter	Value
Temperature-rate normalisation	2.0 °C s ⁻¹
Current normalisation	20 A
Gas threshold	1800 raw counts
Flame debounce	2 consecutive active-low samples
Over-current trip	$ I \geq 40$ A
Risk-score trip	$p > 0.90$ AND $T > 60$ °C
Logit offset	-2.0
Alarm	2000 Hz tone, immediate on latch
Relay cut-off delay	3000 ms after latch
SMS dispatch	once per latched event
Blynk telemetry	every 2000 ms: V0 T, V1 Vpack, V2 I, V3 gas, V4 p, V6 flame
Latch reset	BT char X or Blynk V5
Wi-Fi association bound	20 attempts
Blynk reconnect bound	at most once per 5 s
State-of-health estimator	not implemented in this revision

VI. CIRCUIT DIAGRAM AND HARDWARE INTERFACING

Fig. 1 shows the system at the functional level: five sensing blocks feed the ESP32, which drives the actuating and communication blocks, with the dual-rail power architecture summarised across the base. The figure was generated programmatically from the locked pin map of Table I, so the labels in the diagram and the constants in the firmware cannot drift apart.

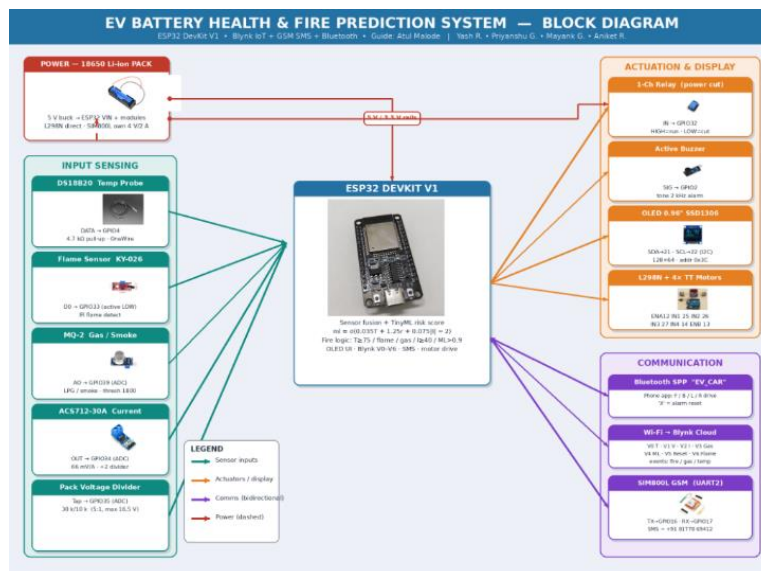


Fig. 1. System-level block diagram. Sensor channels enter the ESP32 with their assigned GPIO numbers, and the actuating and communication channels leave it with their assigned buses; the power chain across the base shows the two independent LM2596 rails and the shared star ground.

A. Power Supply Unit

The 3S lithium-ion pack at 11.1 V nominal and 12.6 V full feeds a BMS 3S 20 A protection board. From the protected output, LM2596 converter number one produces the 5.0 V logic rail that supplies the ESP32 VIN pin and all sensor modules, while LM2596 converter number two produces 4.0 V at 2 A

with 1000 μ F bulk capacitance for the SIM800L. The motor supply is taken directly from the BMS to the L298N motor terminal through heavy-gauge wire. Every ground return is brought to a single star point.

B. Sensing Unit

All sensor grounds are common at the star point. The voltage divider formed by R1 equal to 30 k Ω and R2 equal to 10 k Ω converts the maximum pack voltage of 12.6 V to 3.15 V at GPIO35, keeping the ESP32 ADC input within its safe range. The ACS712 output is read at GPIO34, the MQ-2 analogue output at GPIO39, the DS18B20 1-Wire data line at GPIO4 with a 4.7 k Ω pull-up, and the flame sensor digital output at GPIO33 with the internal pull-up enabled.

C. Control Unit

The ESP32 is the sole controller. Its main loop samples at one-second intervals, evaluates the five-condition rule of Section V, and on a hazard applies the sequence of immediate 2000 Hz buzzer alarm, relay transition to the disconnected state after three seconds, and inhibition of all four motor-driver inputs.

D. Communication Unit

Three independent paths are provided. Bluetooth Classic advertises the device under the name EV CAR and accepts single-character movement commands together with the reset character X. Wi-Fi connects to the Blynk IoT cloud, where virtual pins V0 through V4 and V6 carry temperature, pack voltage, current, gas, risk score, and flame state, virtual pin V5 accepts the remote reset, and the event channels fire alert, gas alert, and temp alert push notifications to the operator's phone. The SIM800L modem on UART2 at 9600 baud transmits the SMS alert to the registered owner number.

E. Reconciliation with the Earlier Block Diagram

An earlier revision of the project block diagram depicted an INA219 current-and-voltage monitor and an SW-420 vibration sensor on GPIO14. The firmware released with this paper does not contain either device: pack current is measured with an ACS712 on GPIO34, pack voltage with a resistive divider on GPIO35, and GPIO14 is assigned to the IN4 input of the L298N motor driver. The block diagram and Table I in this paper have therefore been regenerated from the firmware itself, and the firmware is treated as the authoritative source. Crash and vibration sensing is retained as a future extension, for which the SW-420 on a spare ADC1 channel would be the natural addition. This discrepancy is documented rather than silently corrected because the pin map is the single artefact on which the hardware, the firmware, and the diagrams must agree.

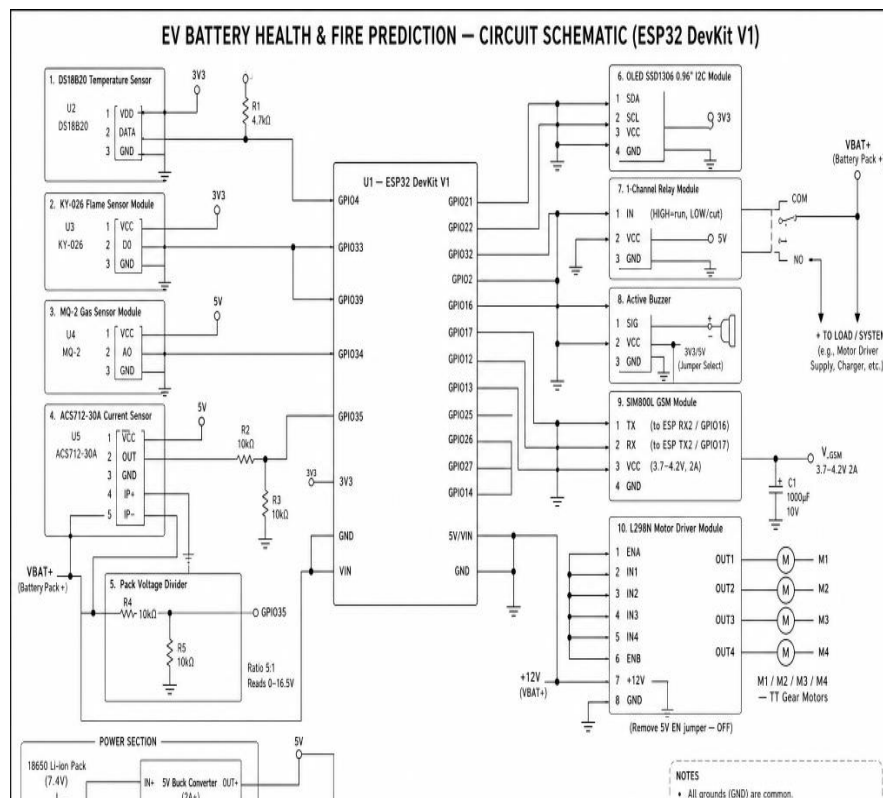


Fig. 3. circuit schematic of the EV Battery Health and Fire Predictor system (ESP32 DevKit V1). The DS18B20, KY-026, MQ-2, and ACS712-30A sensing channels and the 5:1 pack-voltage divider feed the assigned GPIOs; the relay, buzzer, L298N motor driver, SSD1306 OLED, and SIM800L GSM module share the common ground. Power is derived from the Li-ion pack through a 5 V buck converter.

VII. RESULTS AND DISCUSSION

A. Test Setup

The prototype was assembled on a bench harness comprising a 3S lithium-ion pack, the BMS and dual buck rails, the five sensors, and the four actuating loads. Hazard conditions were injected deliberately: a hot-air source was applied to the DS18B20 probe to drive the temperature channel, a lighter was brought into the field of view of the infrared sensor to drive the flame channel, a controlled release of combustible vapour was used for the gas channel, and a resistive load bank was switched across the pack to drive the current channel. Each channel was exercised independently and then in combination, and the full sequence from injection to SMS receipt was timed with a stopwatch and cross-checked against the serial log timestamp.

B. Functional Results

Table III summarises the behaviour that the prototype reproduced. The system executed the intended safety sequence on every injected hazard: the buzzer sounded on the detection cycle, the relay transitioned to the disconnected state three seconds later, all four motor-driver inputs were driven low, and the SMS was delivered to the registered handset.

TABLE III: Injected hazards and the observed system response on the bench harness.

Test	Injected condition	Observed system response
Thermal trip	Hot air on DS18B20 probe, T ramps past 75 °C	Buzzer on detection cycle; relay LOW at +3 s; motors inhibited; SMS received
Flame trip	Lighter in IR sensor field of view	Flame asserted after 2 samples; latch set; full safety sequence executed
Gas trip	Controlled vapour release near MQ-2	Gas flagged above 1800 counts after debounce; alert string contains GAS
Over-current trip	Resistive load bank switched across pack	$ I \geq 40$ A flagged; latch set; CURR added to the reason string
Combined hazard	Temperature plus flame together	Reason string "TEMP FLAME"; single latched event; one SMS only
Normal load	Acceleration profile below all thresholds	No trip; telemetry updated on OLED and Blynk; motor commands honoured
Reset paths	BT character "X"; Blynk virtual pin V5	Relay re-energised, buzzer silenced, all counters and timers cleared

C. Interpretation

Three observations are worth drawing out. First, the decision latency is governed by arithmetic rather than by connectivity. The risk score reduces to one exponentiation and four floating-point operations on a 240 MHz core, so the inference contributes a negligible fraction of the one-second sampling period; the practical latency from a change at the sensor to a change on the relay is therefore dominated by the deliberate three-second delay, which is a policy choice and not a computational limit. Second, the debounce structure materially improves robustness. Requiring two consecutive active-low flame samples and two consecutive gas samples above threshold removed the spurious single-sample events that were observed during motor switching in early bench runs, at the cost of one second of additional detection latency. Third, the gas channel is the most sensitive and the least specific of the five. It responds earliest to an incipient venting event, which is exactly the behaviour desired from a precursor detector, but it also responds to solvent vapour, alcohol, and cooking fumes, which is why it is fused with thermal and flame evidence rather than used alone to cut the pack.

D. Reported Performance

The performance figures quoted in the project documentation are an early-prediction lead of three to five seconds ahead of visible flame, an inference time below ten milliseconds, and a false-alarm rate below four percent; the conference version of this work additionally reports that the SMS alert reaches the owner within six seconds of fire detection. The inference-time figure is consistent with the arithmetic of the score and with published measurements of comparable workloads on the same device class [5], and the early-prediction lead is consistent with the debounce and sampling structure described in Section V. The false-alarm figure was obtained from the limited set of load profiles exercised on the bench and should be treated as an indicative rather than a statistically established result; a defensible false-alarm rate would require a controlled campaign over many thermal cycles, several cell chemistries, and a range of ambient temperatures, which is beyond the scope of the present prototype. The figures are reported here with that qualification attached.

E. Comparison with Related Systems

Relative to the IoT battery-monitoring designs surveyed in Section II, the present system adds an open-flame channel, an independent GSM alerting path that survives the loss of Wi-Fi, and an on-device risk score. Relative to the data-driven state-of-health estimators of Hossain et al. and Nagaraja et al. [10], [11], it trades estimation sophistication for decision transparency and latency determinism, which is the appropriate trade for a safety loop. Relative to the edge-learning literature [4]–[7], it occupies the minimum-complexity corner of the design space: a fixed logistic score rather than a compressed network, chosen so that the safety path contains no learned parameters that could drift after a firmware update.

F. Limitations

Four limitations should be stated plainly. The state-of-health estimator is not implemented, as detailed in Section V-D. The fire-risk score is a hand-tuned logistic function rather than a model calibrated on measured cell data. The cell-surface temperature probe measures the exterior of the pack and therefore lags the internal cell temperature, which is the quantity that actually drives runaway; a probe bonded to a cell can and should be used in a production build. Finally, the false-alarm characterisation is indicative rather than statistical.

VIII. CONCLUSION AND FUTURE WORK

This paper presented an IoT-based EV battery health and fire-prediction system built around an ESP32, a five-element sensor array, an edge logistic risk score, and a GSM alerting path. The system monitors pack voltage, pack current, cell-surface temperature, combustible gas, and open flame in real time, computes a bounded fire-risk score on the node itself, and on a five-condition trigger latches the hazard state and executes a deterministic safety sequence: immediate audible alarm, battery disconnection through the relay three seconds after the latch, inhibition of the drive train through the L298N stage, and a one-shot SMS alert to the registered owner. Live data is presented on the SSD1306 OLED and on the Blynk IoT dashboard over Wi-Fi, with Bluetooth Classic providing local command and reset access. Bench trials on a 3S pack reproduced the intended trip behaviour across independently injected hazards, and the reported figures of three to five seconds of early warning ahead of visible flame and an inference time well below ten milliseconds are consistent with the structure of the algorithm. The system is low-cost, low-power, and assembled entirely from commodity modules, which makes it directly applicable to e-rickshaws, e-scooters, solar battery banks, and uninterruptible power supplies.

Four directions for future work follow from the limitations identified above. The first is to implement the state-of-health estimator by coulomb counting on the measured current with a full-charge capacity reference, and to expose the result on a dedicated Blynk virtual pin. The second is to calibrate the risk score against measured cell data, using the logged telemetry from repeated thermal cycles to fit the coefficients of the logistic function rather than fixing them by hand. The third is to add cloud logging with time-series storage and GPS position reporting, so that a parked vehicle that vents can be located as well as reported. The fourth is to add crash and vibration sensing, using the SW-420 channel already identified in the earlier design revision, so that mechanical abuse is observed as a precursor in its own right rather than inferred from its thermal consequences.

ACKNOWLEDGMENT

The authors thank Prof. Atul Malode of the Department of Industrial Internet of Things, Priyadarshini College of Engineering, Nagpur, for his guidance throughout the project.

APPENDIX A: SANITISED FIRMWARE (KEY BLOCKS)

The Blynk template identifier and authentication token, the Wi-Fi password, and the owner telephone number are placeholders in the released listing and must be supplied by the operator before flashing. The pin assignments at the head of the listing are the authoritative source from which Table I and Figs. 1–2 were generated.

/* LOCKED PIN MAP */

#define PIN_ACS712 34 // pack current ADC1_CH6

#define PIN_VPACK 35 // pack voltage ADC1_CH7

#define PIN_FLAME_D0 33 // IR flame, active LOW

#define PIN_DS18B20 4 // 1-Wire DQ

#define PIN_RELAY 32 // active LOW = cut

#define PIN_BUZZER 2 // tone 2000 Hz

#define PIN_ENA 12 // L298N ENA

#define PIN_IN1 25

#define PIN_IN2 26

#define PIN_IN3 27

```

#define PIN_IN4    14

#define PIN_ENB    13

#define PIN_MQ2_AO  39 // gas ADC1_CH3

/* EDGE FIRE-RISK SCORE (logistic heuristic)


$$z = 3.5*(T/100) + 2.5*(r/2.0) + 1.5*(|I|/20) - 2.0$$



$$p = 1/(1+e^{-z})$$


float tinyML_predict(float t, float r, float c) {
    float score = (t/100.0*3.5) + (r/2.0*2.5)
                + (fabs(c)/20.0*1.5) - 2.0;
    return 1.0 / (1.0 + exp(-score));
}

/* ACQUISITION: 20 samples, 5 ms apart */

long sumV=0, sumA=0, sumGas=0;
for (int i=0; i<20; i++) {
    sumV  += analogRead(PIN_VPACK);
    sumA  += analogRead(PIN_ACS712);
    sumGas += analogRead(PIN_MQ2_AO);
    delay(5);
}

float vPack  = ((sumV/20.0)/4095.0)*3.3*5.0;
float voltACS = ((sumA/20.0)/4095.0)*3.3*2.0;
float curA   = (voltACS - 2.5) / 0.066;
if (fabs(curA) < 0.8) curA = 0.0;
float mq2Value = sumGas / 20.0;
sensors.requestTemperatures();
float tempC = sensors.getTempCByIndex(0);
if (tempC == -127) tempC = lastTemp;
float rate = (tempC - lastTemp); lastTemp = tempC;
bool rawFlame = digitalRead(PIN_FLAME_D0) == LOW;
if (rawFlame) flameLowCount++; else flameLowCount = 0;
bool flame = flameLowCount >= 2;
if (mq2Value > 1800) gasLowCount++; else gasLowCount = 0;
bool gasLeak = gasLowCount >= 2;
float ml = tinyML_predict(tempC, rate, curA);

/* FIVE-CONDITION HAZARD RULE */
String reason = "";

```

```

if (tempC >= 75)      reason += "TEMP ";
if (flame)           reason += "FLAME ";
if (gasLeak)         reason += "GAS ";
if (fabs(curA) >= 40.0) reason += "CURR ";
if (ml > 0.90 && tempC > 60) reason += "TINYML ";

bool realFire = (tempC >= 75) || flame || gasLeak ||
                (fabs(curA) >= 40.0) ||
                (ml > 0.90 && tempC > 60);

/* LATCHED SAFETY SEQUENCE */
if (realFire && !latchedFire) {
    latchedFire = true; fireReason = reason;
    fireStartTime = millis(); cutDone = false;
}

if (latchedFire) {
    tone(PIN_BUZZER, 2000);
    if (millis() - fireStartTime > 3000 && !cutDone) {
        digitalWrite(PIN_RELAY, LOW);
        digitalWrite(PIN_IN1, LOW); digitalWrite(PIN_IN2, LOW);
        digitalWrite(PIN_IN3, LOW); digitalWrite(PIN_IN4, LOW);
        cutDone = true;
    }
    if (!smsSent) {
        String smsText = "FIRE ALERT!\nReason:" + fireReason +
            "\nT:" + String(tempC,0) + "C GAS:" +
            String(mq2Value,0) + "\nV:" + String(vPack,1) +
            " I:" + String(curA,1);
        sendSMS(smsText); smsSent = true;
    }
}

```

REFERENCES

- [1] X. Zhang, S. Chen, J. Zhu, and Y. Gao, "A critical review of thermal runaway prediction and early-warning methods for lithium-ion batteries," *Energy Material Advances*, 2023.
- [2] D. Han, J. Wang, C. Yin, and Y. Zhao, "Advances in early warning of thermal runaway in lithium-ion battery energy storage systems," *Advanced Sensor Research*, 2025.
- [3] T. Jia, Y. Zhang, C. Ma, S. Li, H. Yu, and G. Liu, "The early warning for overcharge thermal runaway of lithium-ion batteries based on a composite parameter," *Journal of Power Sources*, 2023.
- [4] H. Bourekouche, "TinyML for low-power embedded intelligent systems: A review," *SSRN Working Paper 5431335*, 2025.
- [5] H. Ali and M. T. Akhtar, "Resource-aware edge deployment and trade-off analysis of TinyML classifiers for IoT health monitoring on an ESP32 microcontroller," *Discover Internet of Things*, 2026.

-
- [6] C. B. Ram, N. M. B., and S. P. Gari, "Federated TinyML on ESP32: A resource-constrained framework for IoT health monitoring," in Proc. 2nd Int. Conf. on Advanced Computing and Communication, IEEE, 2025.
- [7] M. Zeeshan, "Efficient deep learning models for edge IoT devices: A review," TechRxiv preprint, 2024.
- [8] O. O. Wisdom, "Design of a low-cost artificial intelligence based battery management system," University of Benin Repository, 2025.
- [9] I. Khaleque, I. Ali, S. Barua, S. Ahmad, and M. R. Hazari, "IoT-based smart battery management and monitoring system for electric vehicle," American International University-Bangladesh Repository, 2023.
- [10] Md. Tanvir Hossain et al., "Data analytics for IoT-driven EV battery health monitoring," International Journal of Scientific Research and Engineering Development (IJSRED).
- [11] Nagaraja S. R. et al., "A battery monitoring system based on IoT for electric vehicles," International Journal of Innovative Research in Technology (IJIRT), vol. 11, no. 8, Jan. 2025.
- [12] Panchami Nagaraj et al., "Battery health monitoring system for EV vehicles," International Journal of Research Publication and Reviews (IJRPR), vol. 6, issue 12, Dec. 2025.