Development of a Time-Sensitive Smoke Detection System with Dynamic Threshold Scheduling Using Real-Time Clock on Arduino

Seann Achilles P. Almeria, Alexa Marie V. Baytan, Shirley S. Panes, Marjen Ruzzel S. Reyes, Andrea T. Tugelida, Jose C. Felipe Jr.

Computer Engineering Department, Eulogio "Amang" Rodriguez Institute of Science and Technology

DOI: https://doi.org/10.51244/IJRSI.2026.1306000299

Received: 17 June 2026; Accepted: 22 June 2026; Published: 07 July 2026

ABSTRACT

This research paper presents the design, implementation, and experimental evaluation of a time-sensitive smoke detection system that addresses the persistent false-alarm problem in residential fixed-threshold detectors by introducing scheduled sensitivity adjustments driven by a DS3231 real-time clock module. Building upon the Arduino Uno-based prototype developed by Calderon et al. at EARIST, which employed an MQ-2 smoke sensor, a secondary gas sensor, and an infrared flame sensor operating against a single fixed threshold of 150, this study introduces one low-cost hardware addition and four layered software improvements. The DS3231 RTC module provides the Arduino with accurate time-of-day information to within ±2 minutes per year via its integrated temperature-compensated crystal oscillator. The four software improvements are: (1) averaged sensor sampling across five consecutive readings to suppress electrical noise; (2) a sustained-detection timer requiring continuous threshold exceedance for 3,000 milliseconds before alarm activation; (3) a rate-of-change filter that identifies and rejects sharp spike-and-drop signal trajectories characteristic of non-fire aerosol sources; and (4) a time-based threshold scheduling mechanism applying a higher-tolerance threshold of 280 during cooking hours, a strictest threshold of 160 during nighttime sleep hours, and a standard threshold of 220 during all other periods. The LCD continuously shows the current active mode and threshold value. Experimental results demonstrate that the improved system reduced the false alarm rate from 100 percent across three non-fire source types in the original system to zero percent, while maintaining reliable alarm activation for all genuine fire sources across all three threshold modes. The findings establish time-of-day sensitivity scheduling as a viable, low-cost strategy for improving smoke detector reliability in household environments.

Keywords: smoke detection, false alarm reduction, threshold scheduling, time-sensitive detection, fire safety, rate-of-change filter, signal averaging, sustained-detection timer, occupancy-based sensitivity, low-cost prototype.

INTRODUCTION

Fire remains among the most destructive hazards affecting residential environments in the Philippines and comparable developing-country contexts. Annual reports from the Bureau of Fire Protection consistently identify residential structures as the leading source of fire incidents and fatalities, with dense urban communities facing shortened evacuation windows due to the proximity of structures and limited structural fire resistance. Early detection systems are therefore not merely a convenience but a foundational requirement for life safety, and their reliability directly determines how effectively early warnings protect lives.

Smoke detectors represent the most widely deployed category of early fire warning devices. However, a well-documented limitation of simple, low-cost prototype detectors is their susceptibility to false alarms, which are activations triggered by non-fire smoke sources that momentarily cross a fixed detection threshold. Research by Hwang et al. (2023) demonstrated that conventional smoke detectors in residential studio apartments activate approximately 7.42 minutes after the onset of a cooking scenario, confirming that cooking-related aerosols are the primary driver of nuisance alarms. Repeated false activations produce alarm fatigue, a behavioral condition in which occupants learn through experience that detector alerts are likely non-emergency events, effectively eliminating the device's protective function regardless of its technical sensitivity.

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VI June 2026

The original prototype by Calderon et al. at EARIST employed a single fixed threshold throughout the entire day without any awareness of the time of day or the likely source of a smoke reading. A threshold calibrated for maximum nighttime sensitivity is inherently prone to false alarms during daytime cooking periods, while a threshold calibrated to tolerate cooking activity reduces protection during sleeping hours when occupants are least able to respond to a genuine fire. This study addresses that gap by adding a DS3231 real-time clock module and implementing three defined time-window sensitivity modes that align alarm thresholds with the occupancy-driven risk profile for each period of the day.

The sole prior art identified for time-varying sensitivity in smoke detectors is United States Patent No. 4,101,785, filed in 1977, which proposed switching between two sensitivity levels using an ambient light sensor. That design was never commercialized at low cost, and its light-level mechanism cannot distinguish between 7:00 PM evening darkness and 3:00 AM sleep-hour darkness, both of which represent very different fire risk and false alarm risk profiles. This study implements time-of-day threshold scheduling using programmable time windows derived from a real-time clock, which is architecturally distinct from, and more precise than, the prior art.

Significance of The Study

The reliability of fire detection systems carries direct consequences for human safety, particularly in environments where the margin between early warning and catastrophic loss is narrow. This study is significant because it targets the false alarm problem at its behavioral root: the predictable daily patterns of household activity that produce non-fire smoke at specific, identifiable hours, and resolves it through a practical, low-cost mechanism requiring only one additional component.

Students and Researchers. This study provides a documented experimental methodology for evaluating time-aware threshold scheduling in embedded fire detection systems. It establishes a replicable framework for future researchers seeking to improve sensor-based safety devices through occupancy-driven calibration logic.

Households and Residential Communities. A smoke detector that applies greater tolerance during cooking hours and maximum sensitivity during sleeping hours offers a more behaviorally appropriate safety option than fixed-threshold alternatives. It directly addresses the alarm fatigue problem that leads households to disable their detectors entirely.

Educational Institutions. The integration of a real-time clock module into an Arduino-based safety device provides a concrete, hands-on demonstration of embedded systems design, I2C communication, and scheduled decision logic applicable across engineering and technology curricula.

Small Establishments and Community Facilities. Businesses and community organizations operating with limited resources benefit from a detection system that improves reliability, reducing operational disruption from false alarms, including unnecessary evacuations and alarm fatigue.

Future Developers and Innovators. The time-window threshold scheduling principle implemented here is transferable to a broad class of sensor-based safety systems where hazard risk and false alarm risk vary predictably with time of day or occupancy state, providing a methodological foundation for subsequent embedded safety device development.

REVIEW OF RELEVANT THEORY, STUDIES, AND LITERATURE

Related Theories

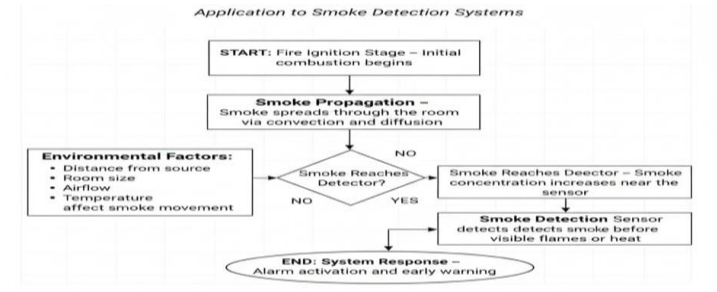


Figure 1. Theory of Fire Development and Smoke Propagation

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VI June 2026

Fire progression is generally described in terms of four sequential stages: ignition, during which combustion initiates; growth, in which the fire expands and fuel consumption accelerates; full development, at which the fire reaches its peak intensity; and decay, as available fuel is consumed. Smoke production begins during the earliest stages of ignition and growth, before visible flames become apparent to occupants. Because smoke particles and combustion gases travel through enclosed spaces via convection and diffusion before heat and flames reach detectable levels, smoke-based detection systems are theoretically positioned to provide earlier warning than heat or optical flame detectors alone. The concentration of smoke particles near any detection point is a function of the source intensity, the distance between the source and the detector, the volume and geometry of the space, ambient airflow patterns, and temperature gradients that drive convective movement. These relationships directly influence both the response time and the amplitude of the sensor signal and underpin the study's experimental design, which evaluates detection performance across varying distances and within a controlled, enclosed room.

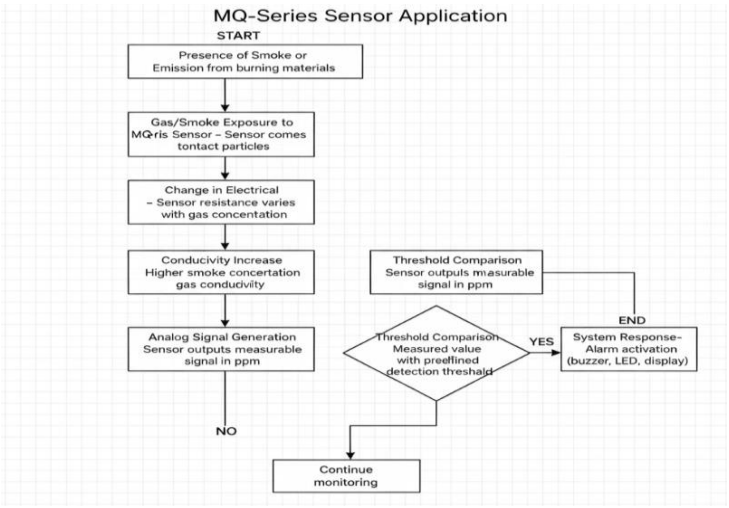


Figure 2. Gas and Particle Detection Theory

The MQ-2 sensor operates on the principle of a metal-oxide-semiconductor gas sensor. The sensor is coated with tin dioxide, which changes its electrical resistance in the presence of combustible gases and smoke particles. The heated sensing element operates by interacting with gas molecules (oxidation and reduction), changing its conductivity, and producing an output voltage proportional to the gas concentration. The analog output can be read from the Arduino ADC pins and provides a continuous, relative measurement of smoke and gas concentration, ranging from 0 to 1023 analog units. The MQ-2 will generally display a reading of 100 to 150 analog units in clean air. Exposure to genuine fire smoke yields readings of 280 to 400, depending on proximity and combustion density. The fundamental limitation of this sensing principle is that it is indiscriminate with respect to the source of the particles: cooking fumes, perfume, and incense can all drive readings temporarily above a fixed threshold, making temporal and contextual analysis essential for reliable discrimination between fire and non-fire sources.

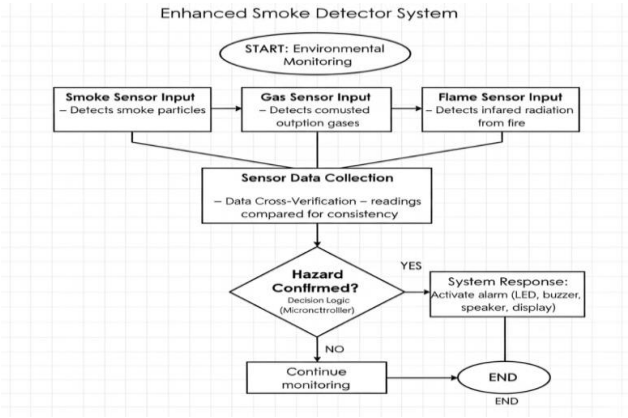


Figure 3. Multi-Sensor Integration Theory

Multi-sensor integration theory is grounded in the principle that combining independent sensing modalities reduces the probability of both false positives and missed detections relative to any single-sensor approach. In the context of this prototype, the MQ-2 gas sensor detects changes in combustible gas and particulate concentration, while the infrared flame sensor responds to the specific wavelength of radiation emitted by open flame. These two phenomena are generated by distinct aspects of the fire process and are unlikely to be produced simultaneously by most non-fire aerosol sources. Requiring confirmation from the flame sensor as an immediate bypass channel, while applying temporal filtering and time-based threshold adjustment to the gas sensor channel, provides a layered decision logic more resistant to false triggering than any single detection mechanism alone.

Related Studies

Ali et al. (2025) developed an integrated fire control system combining gas, smoke, and flame detection in a single microcontroller-based platform, demonstrating that multi-sensor integration on accessible hardware is a technically sound strategy for improving detection reliability. The present study adopts a comparable multi-sensor architecture while adding time-of-day threshold scheduling as an additional reliability layer not present in prior work.

Costea and Schiopu (2018) found that sensor placement and calibration were among the strongest determinants of detection reliability and response time, establishing calibration as a design parameter rather than a fixed constant. This directly supports treating the alarm threshold as a time-dependent, schedulable value rather than a static preset.

Hwang et al. (2023) confirmed cooking fumes and steam as the primary non-fire alarm triggers in residential environments, with conventional detectors activating at approximately 7.42 minutes from cooking onset. This provides the strongest direct empirical support for the cooking-hour tolerance window implemented in this study.

Research from Oak Ridge National Laboratory (ORNL, 2016) demonstrated that algorithmic improvements to smoke detector decision logic can reduce false alarm rates significantly while enabling earlier detection of smoldering fires by 30 to 40 minutes compared to conventional fixed-threshold systems. This establishes that software-level decision logic is a primary lever for reliability improvement in embedded fire safety systems.

Szafarczyk (2022) found that adjusting sensor sensitivity and enabling interdependent sensor operation conditions reduced unwanted activations in multi-sensor detectors without requiring hardware changes, directly confirming that software control of detection conditions translates to measurable reductions in false alarm occurrence.

Wu and Ying (2023) argued for condition-aware calibration processes that account for environmental variability rather than static one-time threshold setting, reinforcing the rationale for a threshold that changes with the household's operational context throughout the day.

Prior Art: Time-Varying Sensitivity in Smoke Detectors

United States Patent No. 4,101,785 (USPTO, 1977) described a smoke detector designed to operate at standard sensitivity during daylight hours and switch to a higher sensitivity during periods of darkness or vacancy, using a light-sensitive switch or a manual timer. The patent explicitly recognized that most false alarms originate from human daytime activities, and that higher sensitivity is safe when those activities are absent. Despite its 1977 priority date, this approach was never commercialized as a low-cost consumer product. Furthermore, the light-level mechanism cannot distinguish between dark evening hours and dark early-morning hours, both of which have very different occupancy and risk profiles. The present study addresses this limitation through named, programmable time windows derived from a DS3231 real-time clock, enabling independent calibration of cooking-hour tolerance and nighttime strictness windows that the prior art was architecturally incapable of supporting.

Synthesis Of Literature

The reviewed body of literature converges on a consistent finding: static, single-threshold detection systems are structurally limited in their ability to discriminate between genuine fire events and the transient aerosol stimuli

that characterize normal occupied environments. This limitation is a decision-logic problem that can be addressed through software design. Hwang et al. (2023) and ORNL (2016) identify cooking as the dominant cause of false alarms and confirm that algorithmic improvements yield reliable reductions in false alarms. Costea and Schiopu (2018) and Wu and Ying (2023) establish calibration as an ongoing design variable. The prior art analysis confirms that programmable, time-window-based threshold scheduling using a real-time clock has not been implemented in a low-cost, commercialized product, thereby positioning this study as an original contribution.

Theoretical Framework

The theoretical foundation of this study integrates three interrelated bodies of knowledge: the Theory of Fire Development and Smoke Propagation, Gas and Particle Detection Theory, and Multi-Sensor Integration Theory. These frameworks interact to explain how the study's experimental design, hardware configuration, and software logic work together as a coherent system.

The Theory of Fire Development and Smoke Propagation establishes the physical basis for the study's evaluation metrics. Because smoke is generated during the earliest pre-flame stages of fire development and spreads through enclosed spaces by convection and diffusion, response time is a function of distance, room geometry, and airflow, directly motivating the study's distance and coverage area experiments.

Gas and Particle Detection Theory explains both the detection principle of the MQ-2 sensor and its fundamental limitation: it produces output proportional to particle concentration without discriminating between fire-generated and non-fire sources. Time-based threshold scheduling adds a contextual dimension that the sensor hardware alone cannot provide, aligning active sensitivity with the occupancy-driven likelihood of fire versus non-fire smoke at each time of day.

Multi-Sensor Integration Theory provides the architectural justification for the layered decision logic. The flame sensor bypass channel provides a high-confidence, low-latency alarm pathway for unambiguous fire events. The gas sensor channel, governed by temporal filters and time-based threshold adjustment, provides a pathway hardened against transient non-fire stimuli. Together, these three frameworks form the conceptual basis for a detection system that evaluates whether a sensor signal is consistent with the temporal, contextual, and multi-sensor signature of a genuine fire event before activating an alarm.

Importance And Relevance

The practical importance of this study lies at the intersection of fire safety accessibility and detection system reliability. In the Philippines, residential structures account for the largest share of fire incidents and fatalities annually. In this context, the availability of dependable, affordable fire detection devices is a public safety matter with directly measurable consequences.

A detector that activates frequently in non-fire situations creates alarm fatigue, after which occupants learn through repeated experience that alerts are probably not emergencies. Once this behavioral pattern is established, the detector's protective function is eliminated regardless of its technical sensitivity. Hwang et al. (2023) identified cooking-related false alarms as the most common driver of this dynamic in residential environments.

The contribution of this study is that a meaningful behavioral insight, which is that cooking happens at predictable hours and sleeping happens at predictable hours, is encoded into the detection logic at negligible additional cost. The DS3231 module needed to implement this makes use of the I2C bus already present in the prototype and requires no additional infrastructure. The result is an automatically contextual, occupancy-sensitive smoke detector that is responsive to occupancy at all times of day without user manual input. Moreover, the time windows are configurable in the code itself, so households with non-standard schedules (e.g., shift workers or families with different meal times) can set the window boundaries to their daily patterns.

Statement of The Problem

Fixed-threshold smoke detection systems, in which any sensor reading above a preset threshold immediately triggers an alarm, are operationally vulnerable to false activations from non-fire aerosol sources commonly

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VI June 2026

present in occupied environments. This vulnerability arises from two interrelated architectural limitations. First, the system cannot distinguish a sustained rising concentration profile from a brief spike-and-fall pattern. Second, the same threshold is applied regardless of whether the current hour represents a high false-alarm-risk cooking period or a high genuine-fire-risk sleeping period.

This study seeks to answer the following questions:

1. How effectively does the time-aware system with scheduled threshold adjustment detect smoke at distances ranging from 10 to 100 centimeters compared with the original fixed-threshold baseline system?

2. What response times does the enhanced system exhibit under NIGHT mode when exposed to genuine fire-related smoke sources, and how do these compare to the original baseline across tested distances?

3. How does the sensor respond to different combustible and non-combustible aerosol sources, and does the combination of rate-of-change filtering, sustained-timer detection, and time-based threshold scheduling successfully suppress false alarms while maintaining true fire detection?

4. Does the scheduled threshold adjustment produce a measurable reduction in false alarm rate during cooking-hour conditions compared with the original fixed-threshold system, without reducing true detection rate for genuine fire scenarios?

General And Specific Objectives

The primary objective of this study is to develop and evaluate a Time-Sensitive Smoke Detection System that uses a DS3231 real-time clock module to automatically schedule alarm threshold adjustments based on time of day, reducing false alarm occurrences during cooking hours and maximizing detection sensitivity during nighttime sleep hours, while maintaining or improving upon the detection reliability of the original Calderon et al. prototype.

The study specifically aims to:

a) Integrate a DS3231 RTC module with the existing Arduino-based smoke detector prototype and implement three-mode time-based threshold scheduling: COOKING mode with threshold 280 during meal preparation hours, NIGHT mode with threshold 160 during sleep hours, and DEFAULT mode with threshold 220 during all other periods.

b) Determine the detection distance performance of the enhanced system under NIGHT mode across a range of 10 to 100 centimeters using burning paper as the standardized test source, and compare response times against the original fixed-threshold baseline.

c) Evaluate the system's discrimination capability across multiple smoke source types under all three threshold modes and verify the effectiveness of the rate-of-change filter and sustained timer in suppressing non-fire activations.

d) Quantify the reduction in false alarm rate during simulated cooking-hour conditions produced by the COOKING threshold mode compared with the original fixed-threshold system, confirming that genuine fire detection capability is maintained across all modes.

Scope And Limitations

This study focuses on improving an existing Arduino Uno-based smoke detector prototype by adding a DS3231 RTC module and implementing four software-level modifications. All other hardware components remain identical to those used in the original Calderon et al. prototype: the MQ-2 gas sensor, secondary gas sensor, infrared flame sensor, microcontroller, LED indicators, buzzer, DFPlayer Mini speaker module, and 16-by-2 LCD.

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VI June 2026

Experimental evaluation is confined to controlled indoor laboratory conditions at approximately 30-32 °C. All smoke sources are introduced in standardized ways under supervised conditions, and formal measurements are recorded only after the mandatory 90-second sensor warm-up period. The study does not address outdoor environments, high-humidity conditions, or settings with significant airflow variability.

Time-of-day modes are simulated during controlled testing by temporarily adjusting the DS3231 RTC time in the Arduino sketch using rtc.adjust(), allowing all three threshold modes to be evaluated without physically conducting tests at nighttime or early-morning hours. The threshold values of 280, 220, and 160 are empirically derived from sensor behavior observations and are not validated against formal fire safety certification standards. The prototype is not intended as a replacement for commercially certified smoke detection systems and is presented as a proof-of-concept demonstrating that time-of-day sensitivity scheduling is a viable, low-cost improvement strategy. Long-term sensor drift and behavior under extreme ambient variation are outside the scope of this study.

The evaluation is further constrained by sample size. Source-discrimination testing (Table 6) was conducted using a single trial per source across only three non-fire aerosol types, which limits the statistical generalizability of the reported false-alarm elimination and precludes formal significance testing on that specific result. Power consumption and battery life under continuous operation were likewise not measured, leaving the practical runtime of the solar-assisted Li-ion power system across all three threshold modes uncharacterized. This power system is a hardware specification adopted following faculty feedback recommending solar charging over a disposable 9V battery; its expected autonomy is estimated analytically in the Methodology, but empirical validation, including measured solar contribution under real installation lighting, is deferred to future testing as outlined in the Recommendations section.

METHODOLOGY

This study employed a comparative experimental design to assess the performance of the time-aware smoke detector prototype against the original fixed-threshold baseline system. The experimental approach was structured around four measurement parameters: detection distance under NIGHT mode, response time comparison, sensor sensitivity across multiple source types under all three modes, and false alarm rate reduction. All trials were conducted under controlled indoor conditions with standardized source preparation, mandatory sensor warm-up, and repeated measurements to account for trial-to-trial variability.

The enhanced prototype retains the complete hardware configuration of the original Calderon et al. device. The only hardware addition for the time-aware detection function is the DS3231 RTC module, which is connected to the Arduino via the I2C bus: SDA on A4 and SCL on A5. These lines are shared with the existing LCD module via a breadboard junction, and the DS3231 is daisy-chained to the existing I2C connections using stripped hookup wire, without requiring any additional Arduino pins. The RTClib library by Adafruit was installed via the Arduino IDE Library Manager to enable communication with the DS3231.

Separately, following feedback from the thesis adviser, the original single-use 9V battery was replaced with a solar-assisted rechargeable power system. A 6V, 1 to 2 W solar panel charges two 18650 lithium-ion cells (3.7V, approximately 3,000 mAh each, wired in parallel for a combined capacity of approximately 6,000 mAh) through a TP4056 charging module with over-charge, over-discharge, and over-current protection. A DC-DC boost converter then steps the battery pack voltage up to 9V to match the Arduino's existing VIN input, so no other part of the circuit requires modification. Based on the system's estimated continuous current draw of approximately 416 mA at 5V, calculated from component datasheet specifications and dominated by the heating elements of the two MQ-series gas sensors, and assuming a boost-converter efficiency of approximately 85 percent, the battery pack is analytically estimated to provide approximately 9.1 hours of standalone operation with no solar input at all, sufficient to bridge a full night. The continuous draw of the two heated gas sensors is large relative to what a panel this size can realistically offset: even under direct outdoor sun, a 1 to 2 W panel is estimated to supply only a low single-digit percentage of the system's daily energy consumption, with indoor window-sill placement supplying substantially less. The solar panel is therefore specified as a supplemental trickle-charging source that reduces how often the battery pack must be manually recharged, rather than as a

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VI June 2026

primary or sole power source. Its real-world contribution under installed lighting conditions has not yet been measured and is identified as an open empirical question in the Recommendations section.

The DS3231 is queried for the current hour in each detection loop cycle, and the result is classified into one of three operating modes. COOKING mode: 6:00-8:00 AM, 11:00 AM-1:00 PM, 5:00-7:00 PM. Threshold: 280. NIGHT mode applies a 160 threshold from 10 PM to 5 AM. DEFAULT mode applies a threshold of 220 during all other hours. The current mode and threshold are displayed continuously on the LCD. The LCD alternates between displaying the mode and threshold on one screen and the current time from the RTC on another every 3 seconds.

Calibration procedure. Before each test session, the sensor was powered on and allowed to stabilize for at least 90 seconds to mitigate the MQ-2's thermal warm-up behavior, during which readings are elevated and unreliable. Formal test trials commenced only after stable baseline readings between 100 and 150 analog units were confirmed on the Serial Monitor. This warm-up and baseline-confirmation step was repeated identically before every distance, room-scale, and source-discrimination trial to keep starting conditions consistent across the dataset. The complete, annotated Arduino sketch implementing all four software improvements is provided in Figure 10; the full project files, including the calibration and data-logging routines, are available from the corresponding author upon reasonable request to support independent reproduction of these results.

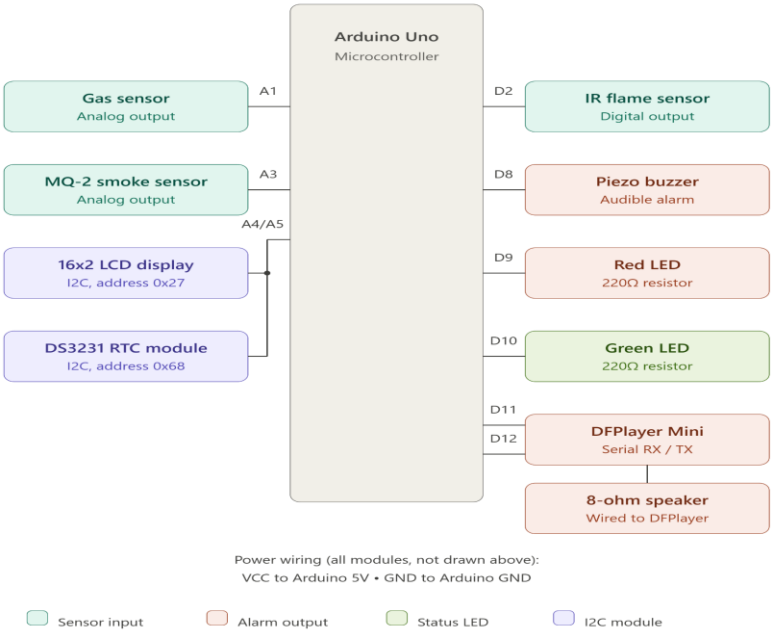


Figure 4. Schematic Diagram

The schematic diagram illustrates the pin-level wiring connections of all components. The MQ-2 smoke sensor and secondary gas sensor each output an analog voltage proportional to particulate and combustible gas concentration, read by the Arduino Uno through analog pins A3 and A1, respectively. The infrared flame sensor outputs a digital signal on pin D2, which the Arduino evaluates on every loop iteration as a direct fire confirmation bypass channel. The DS3231 RTC module and the 16x2 LCD share the I2C bus, with both serial data (SDA) and serial clock (SCL) lines connected to analog pins A4 and A5. The two devices operate without conflict because they use distinct I2C addresses — 0x68 for the DS3231 and 0x27 for the LCD. On the output side, the red LED (pin D9) and green LED (pin D10) are each wired in series with a current-limiting resistor. The piezo buzzer is driven directly from pin D8. The DFPlayer Mini communicates with the Arduino over a SoftwareSerial link, with its TX pin connected to Arduino pin D11 and its RX pin connected to D12, following the standard serial crossover convention. The SPK1 and SPK2 output terminals of the DFPlayer Mini connect directly to the 8-ohm speaker. All sensor and module VCC and GND pins share the Arduino's common 5V and GND rails.

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VI June 2026

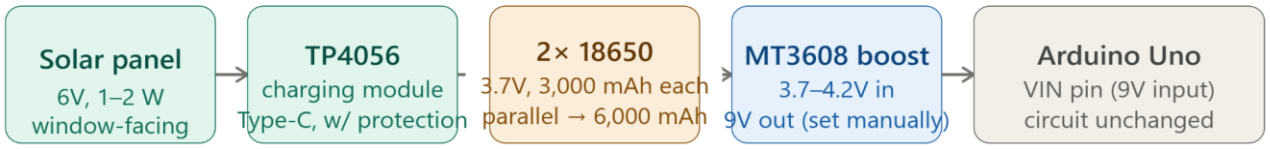


Figure 5. Solar-Assisted Power System Architecture

Figure 5 illustrates the solar-assisted power system adopted for the enhanced prototype following thesis adviser feedback recommending a rechargeable alternative to the single-use 9V battery. The system consists of four components connected in series. A 6V, 1 to 2 W solar panel serves as the charge source, supplying current to a TP4056 Type-C lithium-ion charging module equipped with built-in over-charge, over-discharge, and over-current protection. The module charges two 18650 lithium-ion cells rated at 3.7V and approximately 3,000 mAh each, wired in parallel for a combined capacity of approximately 6,000 mAh. An MT3608 DC-DC boost converter then steps up the battery pack voltage to a regulated 9V output, which connects to the Arduino Uno's existing VIN pin so that no other part of the detection circuit requires modification. Based on the system's estimated continuous current draw of approximately 416 mA at 5V, calculated from component datasheet specifications and dominated by the heating elements of the two MQ-series gas sensors, the battery pack is analytically estimated to provide approximately 9.1 hours of standalone operation with no solar input, sufficient to bridge a full night between daylight charging windows. The solar panel is specified as a supplemental trickle-charge source rather than a primary power supply; even under direct outdoor sun, a panel of this size is estimated to offset only a low single-digit percentage of the system's daily energy consumption, with indoor window-sill placement supplying substantially less. Empirical measurement of the actual solar contribution under real installation lighting is identified as future work in the Recommendations section.

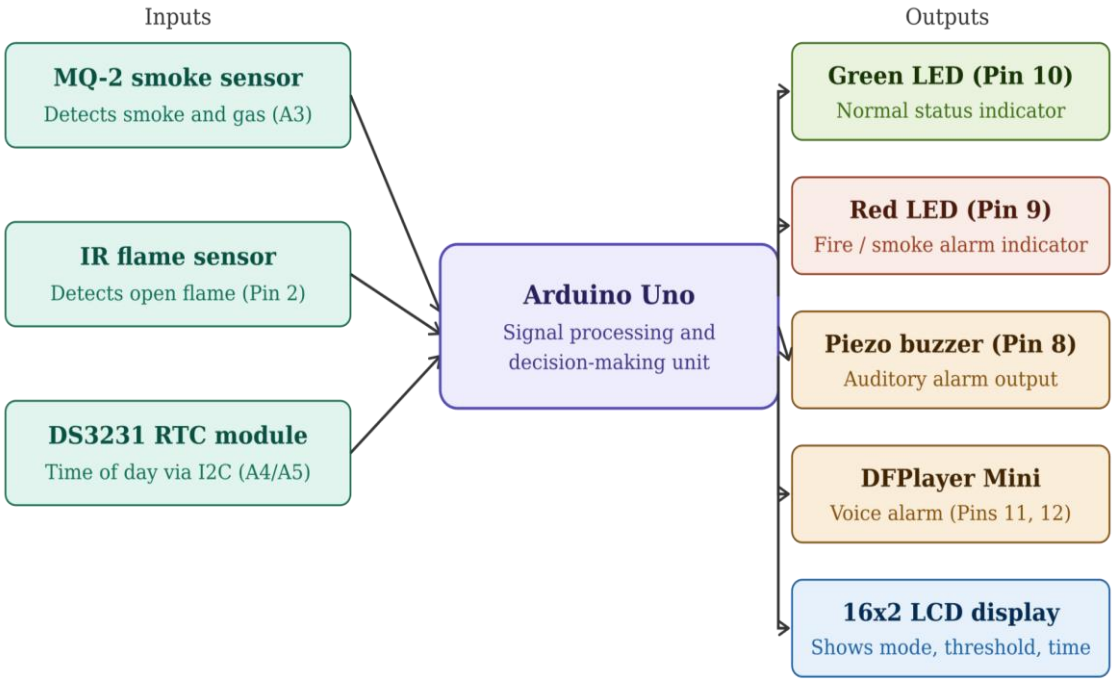


Figure 6. Block Diagram

The system architecture of the enhanced smoke detector consists of three parallel input channels feeding into the Arduino Uno as the central processing unit. The first input channel is the MQ-2 combined gas and smoke sensor, which provides continuous analog readings proportional to the concentrations of smoke and combustible gases in the surrounding environment. The second input channel is the infrared flame sensor. It detects the wavelength

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VI June 2026

of infrared radiation emitted by the open flame and provides an immediate bypass channel for direct fire confirmation. The third input channel is the DS3231 real-time clock module connected over the I2C bus. The module supplies the Arduino with the current hour of the day, which is used to set the active threshold mode for each detection cycle. The DS3231 and 16x2 LCD module use the I2C Bus, but with different addresses. The address of the DS3231 is 0x68, and the address of the LCD is 0x27. Therefore, the two devices can communicate on the same two signal lines without any conflict. The Arduino processes all sensor data according to the configured calibration conditions and, upon confirmation of the hazard conditions, triggers the output array of a red LED alarm indicator, a piezo buzzer, and a DFPlayer Mini audio module, which drives the 8-ohm speaker. The green LED remains active during normal monitoring operation and deactivates when the alarm state is triggered. The 16x2 LCD provides real-time output of the current time-of-day mode, the active threshold value, and live sensor readings throughout operation, alternating every 3 seconds between the mode and threshold display and the current time display.

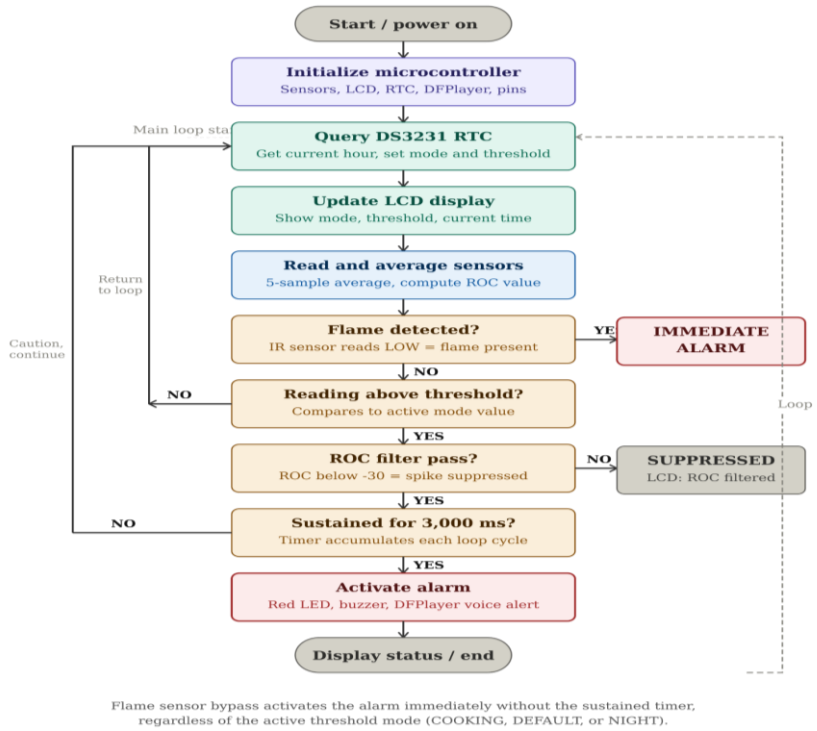


Figure 7. Flowchart/System Operation

When powered on, the Microcontroller initiates all Sensor Pins, the LCD, the Serial Communication, the DFPlayer Mini Audio Module, and the DS3231 RTC Module. If the RTC Module reports a loss of electrical energy, the System will revert to using the sketch's compile time to synchronize its internal clock and continue. Once the System has completed this synchronization process, it will enter a continuous monitoring loop. At the beginning of each loop, the Arduino will poll the DS3231 RTC Module for the current hour and assign that hour to one of three Operational Modes; Cooking Mode (with an upper limit/threshold of 280) when meals are being prepared, Night Mode (with an upper limit/threshold of 160) when people are asleep, and Default Mode (with an upper limit/threshold of 220) at all times when neither cooking nor sleeping is occurring. As soon as the Active Mode and Upper Limit/Threshold have been determined, they will be displayed on the LCD. Next, the Arduino will calculate five sample-averaged readings for both the MQ-2 Smoke Sensor and the Secondary Gas Sensor, compute the Rate of Change from the Previous Reading Cycle, and evaluate the IR Flame Sensor. Should the IR Flame Sensor indicate the presence of Infrared Radiation that is indicative of an Open Flame, the Alarm will be activated immediately without regard to any Temporal Filter Settings and irrespective of the Active Threshold Mode. Should either the Average Smoke or gas reading exceed the active threshold, and the rate-of-change filter does not identify a spike-and-drop pattern with a drop exceeding negative 30 units per cycle, the sustained timer begins accumulating. If the reading remains continuously above the active threshold for the full 3,000-millisecond window, the alarm activates. If the reading drops below the threshold before the timer completes, the timer resets, and the system returns to the RTC query step to begin the next loop cycle. When no

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VI June 2026

hazard conditions are present, the green LED remains lit, the LCD displays the current mode, threshold, and sensor readings, and the loop continues at 500-millisecond intervals.

Components List And Functions

Table 1. Variables and Conditions of the Enhanced Smoke Detector

Variable / Component	Parameter / Role	Qty
Arduino Uno	Main microcontroller. Executes all detection logic and drives all output modules at 5V.	1
MQ-2 Smoke Sensor	Analog smoke and gas sensing via a metal oxide semiconductor. Clean air baseline: 100 to 150 analog units. Fire smoke: 280 to 400+ units.	1
Gas Sensor	Secondary combustible gas concentration sensing for cross-verification of MQ-2 readings.	1
IR Flame Sensor	Detects infrared radiation from open flame. Immediate alarm bypass channel regardless of active threshold mode.	1
DS3231 RTC Module	Real-time clock for time-of-day threshold scheduling. Accuracy: plus or minus 2 minutes per year. I2C address: 0x68. Powered by a CR2032 backup battery.	1
16x2 LCD I2C Module	Displays current time-of-day mode, active threshold value, and live sensor readings. Shares I2C bus with DS3231 at address 0x27.	1
DFPlayer Mini	Audio alarm playback via SoftwareSerial. Plays a pre-recorded voice alert on alarm activation.	1
Red LED	Visual alarm indicator. Activates when the smoke or flame alarm is triggered.	1
Green LED	Normal status indicator. Remains lit during safe monitoring operation.	1
Piezo Buzzer	Auditory alarm output. Activates during the fire or smoke alarm state.	1
100-ohm Resistor	Current-limiting protection for LED components to prevent excessive current draw.	2
CR2032 Coin Cell Battery	Backup power for DS3231 RTC. Maintains correct time when the main power is disconnected.	1
Solar Panel (6V, 1-2W)	Trickle-charges the battery pack from ambient or window-facing light, reducing dependence on disposable batteries.	1
TP4056 Li-ion Charging Module	Regulates the solar panel's input and safely charges the 18650 battery pack, with built-in over-charge, over-discharge, and over-current protection.	1
18650 Li-ion Battery (3.7V, ~3000 mAh)	Rechargeable cells wired in parallel to power the system, sized to bridge nighttime and low-light hours when the panel produces no charge.	2
DC-DC Boost Converter Module	Steps up the battery pack voltage to 9V to match the Arduino's existing VIN input, so the rest of the circuit is unchanged.	1

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VI June 2026

Table 2. Time-of-Day Threshold Mode Definitions

Mode	Active Time Windows	Threshold Value	Rationale
COOKING	6:00 to 8:00 AM, 11:00 AM to 1:00 PM, 5:00 to 7:00 PM	280	Meal preparation hours. Higher tolerance prevents cooking smoke from triggering false alarms.
DEFAULT	All other daytime hours	220	General daytime occupancy. Moderate sensitivity appropriate for typical household activity.
NIGHT	10:00 PM to 5:00 AM	160	Sleep hours. Strictest threshold for maximum sensitivity. Fastest response to genuine fire.

Table 3. Functional Test Cases and Observed System Behavior

Test #	Input Condition	Observed Output	Expected Output	Pass / Fail	Remarks
1	No flame, no smoke. The system is idle in clean air.	Green LED ON. The LCD shows mode and live readings. Red LED OFF. Buzzer silent.	System idles in monitoring mode. No alarm activation.	Pass	Averaged MQ-2 readings remain between 100 and 150 in clean air, well below all three threshold levels.
2	Sustained smoke from burning paper at 10 cm. Readings sustained above the active threshold for more than 3 seconds.	Red LED ON. Buzzer active. DFPlayer plays a voice alert. LCD shows SMOKE ALERT.	Alarm activates only after 3,000 ms sustained threshold exceedance.	Pass	Averaged readings reached 310 to 340 in NIGHT mode (threshold 160). Alarm fired at 22 seconds during experimental trial. Sustained timer correctly accumulated before alarm.
3	Transient aerosol. Perfume spray produces a brief spike, then a rapid drop.	LCD shows ROC SUPPRESSED. No alarm. Green LED remains ON.	The rate-of-change filter identifies a spike-and-drop pattern and suppresses the alarm.	Pass	Peak reading reached 215 to 225 before a rapid drop. ROC value fell below the negative 30 threshold, triggering suppression. No alarm fired despite momentary threshold crossing.
4	Cooking smoke from heated oil introduced during the COOKING mode simulation (threshold 280).	Green LED remains ON. No alarm. LCD shows COOKING T:280.	A higher tolerance threshold prevents cooking smoke from triggering a false alarm.	Pass	Peak MQ-2 reading reached 238 to 255 during cooking smoke exposure, below the 280 COOKING threshold. No alarm fired.
5	IR flame detected. Open a lighter flame presented	Red LED ON. Buzzer active. LCD shows FIRE	Immediate alarm activation via flame sensor	Pass	Flame sensor bypass overrides all temporal filters and all three threshold modes. Alarm fires

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VI June 2026

	directly to the sensor.	DETECTED. Immediate alarm.	bypass regardless of active mode.		within one loop cycle (500 ms) of flame detection.
6	RTC time crosses from DEFAULT hours into the COOKING window (simulated 6:00 AM).	LCD updates to COOKING T:280. Active threshold changes automatically.	Mode transition occurs automatically based on RTC time without user input.	Pass	DS3231 RTC correctly triggers mode transition. LCD update confirmed within a 500ms loop cycle. CR2032 battery maintains correct time across power cycles.
7	System power reset. Battery disconnected, then reconnected.	LCD initializes. Green LED activates after warm-up. No spurious alarm at startup.	System returns to idle monitoring mode without a false alarm.	Pass	Initialization delay and the mp3.stop() command prevent DFPlayer false playback during boot. RTC retains correct time via CR2032 backup battery across power interruptions.

Development And Implementation Steps

The enhanced smoke detector system integrates three input channels, which are the MQ-2 gas and smoke sensor, the infrared flame sensor, and the DS3231 RTC module, processed by the Arduino Uno as the central decision-making unit. Output devices include red and green LED indicators, a piezo buzzer, a DFPlayer Mini audio module connected to an 8-ohm speaker, and a 16-by-2 LCD module.

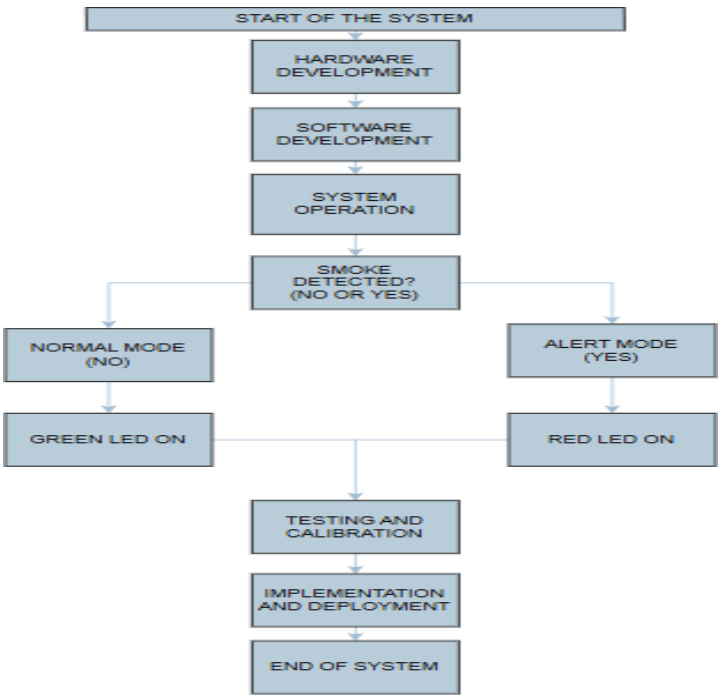


Figure 8. Development and Implementation Steps

Development began with the hardware integration phase. The MQ-2 sensor, IR sensor, LEDs, buzzer, speaker module, and LCD were connected to the Arduino Uno according to the established pin configuration from the Calderon et al. baseline. The DS3231 RTC module was then added to the breadboard in the upper enclosure layer, with its SDA and SCL pins daisy-chained onto the existing A4 and A5 I2C lines using stripped hookup wire through the breadboard as a junction, avoiding the need for any additional Arduino pin connections. Physical assembly on the two-layer acrylic board enclosure was completed using a soldering iron and solder wire to ensure stable, permanent electrical connections.

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VI June 2026

Software development was conducted in the Arduino IDE. The RTClib library by Adafruit was installed via the Library Manager. The core control program implements the four improvements sequentially within the main loop. The five-sample averaging function is applied first to produce a stable sensor reading. The rate-of-change value is then calculated against the previous cycle reading. The DS3231 is queried for the current hour, and the active mode and the thresholds are set accordingly. The sustained timer logic then evaluates whether the averaged reading has exceeded the active threshold for the entire 3,000-millisecond accumulation window. The flame sensor bypass channel is evaluated in parallel on every loop iteration, ensuring direct fire detection never waits for temporal filters or RTC evaluation. The LCD alternates every three seconds between displaying the current mode and threshold, and displaying the current time from the RTC.

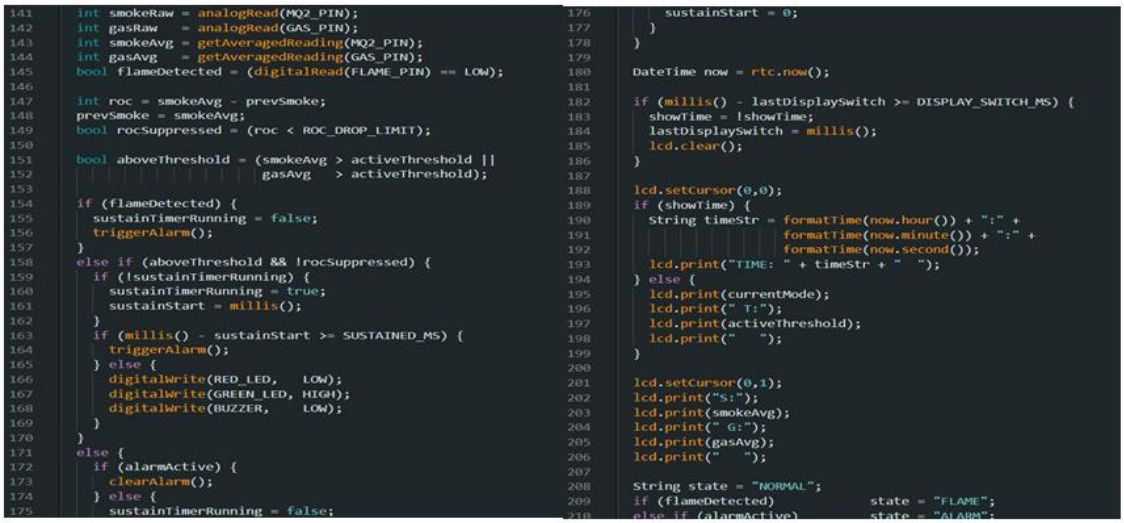
System validation was conducted through a structured sequence of functional tests covering all sensor input combinations, RTC mode transitions, timer behavior under sustained and transient conditions, LCD accuracy across all modes, and audio module initialization reliability across multiple power cycles.

Figure 9. Actual Photo of the Enhanced Smoke Detector

```ino
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#include <RTClib.h>
#include <SoftwareSerial.h>

#define MQ2_PIN         A3
#define GAS_PIN         A1
#define FLAME_PIN       2
#define RED_LED         9
#define GREEN_LED       10
#define BUZZER          8
#define DF_RX           11
#define DF_TX           12

#define COOKING_THRESHOLD     280
#define DEFAULT_THRESHOLD     220
#define NIGHTTIME_THRESHOLD   160

#define SAMPLE_COUNT      5
#define SUSTAINED_MS      3000
#define ROC_DROP_LIMIT    -30

LiquidCrystal_I2C lcd(0x27, 16, 2);
RTC_DS3231 rtc;
SoftwareSerial dfSerial(DF_RX, DF_TX);

int activeThreshold = DEFAULT_THRESHOLD;
String currentMode  = "DEFAULT";
bool alarmActive    = false;
unsigned long sustainStart      = 0;
bool sustainTimerRunning        = false;
int prevSmoke = 0;

bool showTime = false;
unsigned long lastDisplaySwitch = 0;
#define DISPLAY_SWITCH_MS 3000

void dfSend(uint8_t cmd, uint8_t p1, uint8_t p2) {
  uint8_t buf[10] = {0x7E,0xFF,0x06,cmd,0x00,p1,p2,0,0,0xEF};
  int16_t chk = 0;
  for (int i = 1; i <= 6; i++) chk -= buf[i];
  buf[7] = chk >> 8; buf[8] = chk & 0xFF;
  for (int i = 0; i < 10; i++) dfSerial.write(buf[i]);
}
void dfStop()             { dfSend(0x16,0,0); }
void dfPlayTrack(uint8_t t){ dfSend(0x03,0,t); }
void dfVolume(uint8_t v)   { dfSend(0x06,0,v); }

int getAveragedReading(int pin) {
  long sum = 0;
  for (int i = 0; i < SAMPLE_COUNT; i++) {
    sum += analogRead(pin);
    delay(10);
  }
  return sum / SAMPLE_COUNT;
}

void updateTimeMode() {
  DateTime now = rtc.now();
  int h = now.hour();

  bool isCooking = (h >= 6  && h < 8)  ||
                   (h >= 11 && h < 13) ||
                   (h >= 17 && h < 19);
  bool isNight   = (h >= 22) || (h < 5);

  if (isCooking) {
    activeThreshold = COOKING_THRESHOLD;
    currentMode     = "COOKING";
  } else if (isNight) {
```

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VI June 2026

```
    activeThreshold = NIGHTTIME_THRESHOLD;
    currentMode     = "NIGHT";
  } else {
    activeThreshold = DEFAULT_THRESHOLD;
    currentMode     = "DEFAULT";
  }
}

String formatTime(int val) {
  if (val < 10) return "0" + String(val);
  return String(val);
}

void triggerAlarm() {
  if (!alarmActive) {
    alarmActive = true;
    dfPlayTrack(1);
  }
  digitalWrite(RED_LED,   HIGH);
  digitalWrite(GREEN_LED, LOW);
  digitalWrite(BUZZER,    HIGH);
}

void clearAlarm() {
  alarmActive          = false;
  sustainTimerRunning  = false;
  sustainStart         = 0;
  dfStop();
  digitalWrite(RED_LED,   LOW);
  digitalWrite(GREEN_LED, HIGH);
  digitalWrite(BUZZER,    LOW);
}

void setup() {
  Serial.begin(9600);
  dfSerial.begin(9600);
  Wire.begin();

  pinMode(FLAME_PIN,  INPUT);
  pinMode(RED_LED,    OUTPUT);
  pinMode(GREEN_LED,  OUTPUT);
  pinMode(BUZZER,     OUTPUT);

  delay(1000);
  dfStop();
  dfVolume(25);

  lcd.init();
  lcd.backlight();
  lcd.setCursor(0,0); lcd.print("Smoke Detector  ");
  lcd.setCursor(0,1); lcd.print("Initializing... ");

  if (!rtc.begin()) {
    Serial.println("[ERROR] RTC not found!");
    lcd.setCursor(0,1); lcd.print("RTC ERROR!      ");
    while (1);
  }

  rtc.adjust(DateTime(F(__DATE__), F(__TIME__)));
  Serial.println("[RTC] Synced to compile time.");


  digitalWrite(GREEN_LED, HIGH);
  Serial.println("[BOOT] System ready.");
  Serial.println("TIME,MODE,THRESHOLD,SMOKE_RAW,GAS_RAW,SMOKE_AVG,GAS_AVG,ROC,FLAME,STATE");
}

void loop() {
  updateTimeMode();
```



```
int smokeRaw = analogRead(MQ2_PIN);
int gasRaw   = analogRead(GAS_PIN);
int smokeAvg = getAveragedReading(MQ2_PIN);
int gasAvg   = getAveragedReading(GAS_PIN);
bool flameDetected = (digitalRead(FLAME_PIN) == LOW);

int roc = smokeAvg - prevSmoke;
prevSmoke = smokeAvg;
bool rocSuppressed = (roc < ROC_DROP_LIMIT);

bool aboveThreshold = (smokeAvg > activeThreshold ||
                       gasAvg   > activeThreshold);

if (flameDetected) {
  sustainTimerRunning = false;
  triggerAlarm();
}
else if (aboveThreshold && !rocSuppressed) {
  if (!sustainTimerRunning) {
    sustainTimerRunning = true;
    sustainStart = millis();
  }
  if (millis() - sustainStart >= SUSTAINED_MS) {
    triggerAlarm();
  } else {
    digitalWrite(RED_LED,   LOW);
    digitalWrite(GREEN_LED, HIGH);
    digitalWrite(BUZZER,    LOW);
  }
}
else {
  if (alarmActive) {
    clearAlarm();
  } else {
    sustainTimerRunning = false;
    sustainStart = 0;
  }
}

DateTime now = rtc.now();

if (millis() - lastDisplaySwitch >= DISPLAY_SWITCH_MS) {
  showTime = !showTime;
  lastDisplaySwitch = millis();
  lcd.clear();
}

lcd.setCursor(0,0);
if (showTime) {
  String timeStr = formatTime(now.hour()) + ":" +
                   formatTime(now.minute()) + ":" +
                   formatTime(now.second());
  lcd.print("TIME: " + timeStr + "  ");
} else {
  lcd.print(currentMode);
  lcd.print(" T:");
  lcd.print(activeThreshold);
  lcd.print("   ");
}

lcd.setCursor(0,1);
lcd.print("S:");
lcd.print(smokeAvg);
lcd.print(" G:");
lcd.print(gasAvg);
lcd.print("   ");

String state = "NORMAL";
if (flameDetected)          state = "FLAME";
else if (alarmActive)       state = "ALARM";
```

```
  else if (sustainTimerRunning) state = "CAUTION";
  else if (rocSuppressed)       state = "ROC_SUPPRESSED";

  Serial.print(now.hour());      Serial.print(":");
  Serial.print(now.minute());    Serial.print(",");
  Serial.print(currentMode);     Serial.print(",");
  Serial.print(activeThreshold); Serial.print(",");
  Serial.print(smokeRaw);        Serial.print(",");
  Serial.print(gasRaw);          Serial.print(",");
  Serial.print(smokeAvg);        Serial.print(",");
  Serial.print(gasAvg);          Serial.print(",");
  Serial.print(roc);             Serial.print(",");
  Serial.print(flameDetected ? "YES" : "NO"); Serial.print(",");
  Serial.println(state);

  delay(500);
}
```

Figure 10. Source Code of the Enhanced Smoke Detector

Tools And Software Used

Table 4. Tools used for the Enhanced Smoke Detector

Tool / Software	Function
Soldering Iron	Creates permanent conductive joints between components and the prototype board, ensuring mechanically stable and electrically reliable connections throughout testing.

Solder Wire	Provides the conductive material for component-to-board and wire-to-component joints, completing the electrical pathways between all sensors, modules, and the Arduino.
Acrylic Board Enclosure	Two-layer structural enclosure housing the Arduino in the lower layer and the breadboard with all sensors and the DS3231 RTC module in the upper layer. The LCD, buzzer, and DFPlayer Mini are mounted externally on the top face of the enclosure.
Measuring Tape	Establishes precise source-to-sensor distances during detection distance testing, ensuring measurement consistency across all repeated trials.
Stopwatch / Timer	Measures response times from smoke source introduction to alarm activation, providing an independent time reference to cross-validate Serial Monitor timestamp data.
Arduino IDE	Used to write, compile, and upload the detection program to the Arduino Uno.
SolidWorks	Used to design and model the physical layout of the two-layer acrylic enclosure before fabrication, ensuring adequate spacing between sensor elements and the RTC module.

Statistical Treatment of Data

Response-time data collected across repeated trials (Tables 5 and 7) were summarized using the mean and sample standard deviation to characterize trial-to-trial variability in addition to the central tendency already reported. For the ten-distance test (Table 5, n = 3 trials per distance), standard deviation is reported alongside the existing average response column. For the room-scale test (Table 7, n = 10 trials at a single 200 cm by 200 cm area), the mean, standard deviation, and range across all ten trials are reported. In both datasets, the standard deviation is elevated by a documented systematic trend rather than random measurement noise: later trials consistently registered faster response times than earlier trials at the same distance or area, consistent with the residual smoke conditioning effect described in the original Calderon et al. study, where trace particulate from preceding trials lowers the additional concentration needed to cross threshold on subsequent runs. Reported standard deviations should therefore be read as a bound on trial-order-dependent variability rather than as evidence of inconsistent sensor behavior under fixed conditions.

The source-discrimination comparison (Table 6) and the resulting false-alarm-rate comparison (Table 8) are based on a single trial per non-fire source across three source types. With n = 1 per condition, no variance measure can be computed, and no inferential significance test (e.g., a paired t-test or chi-square test of proportions) can be honestly applied to this specific result; it is reported descriptively, and its sample-size limitation is acknowledged explicitly in the Scope and Limitations section. Expanding source-discrimination testing to three or more trials per source is identified as a priority item for follow-up work.

RESULTS AND DISCUSSIONS

Table 5. Smoke Detector Response Time at Varying Distances (NIGHT Mode, Threshold 160)

Dist. (cm)	Trials	Source	Room Temp	Mode / Threshold	Trial 1	Trial 2	Trial 3	Avg Response	Std. Dev.
10	3	Paper	30 C	NIGHT / 160	22 sec	17 sec	14 sec	17.7 sec	4 sec
20	3	Paper	30 C	NIGHT / 160	48 sec	36 sec	22 sec	35.3 sec	13 sec
30	3	Paper	30 C	NIGHT / 160	1 min 10 sec	44 sec	31 sec	48.3 sec	20 sec
40	3	Paper	30 C	NIGHT / 160	1 min 42 sec	55 sec	48 sec	1 min 5 sec	29 sec

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VI June 2026

50	3	Paper	30 C	NIGHT / 160	1 min 58 sec	1 min 14 sec	57 sec	1 min 23 sec	31 sec
60	3	Paper	30 C	NIGHT / 160	2 min 28 sec	1 min 30 sec	58 sec	1 min 39 sec	46 sec
70	3	Paper	30 C	NIGHT / 160	3 min 5 sec	2 min 32 sec	1 min 50 sec	2 min 29 sec	38 sec
80	3	Paper	30 C	NIGHT / 160	4 min 10 sec	3 min 2 sec	2 min 44 sec	3 min 19 sec	45 sec
90	3	Paper	30 C	NIGHT / 160	4 min 30 sec	3 min 40 sec	3 min 35 sec	3 min 55 sec	30 sec
100	3	Paper	30 C	NIGHT / 160	4 min 58 sec	4 min 30 sec	4 min 5 sec	4 min 31 sec	27 sec

Table 5 presents the response times of the enhanced smoke detector across ten source-to-sensor distances ranging from 10 to 100 centimeters, using burning paper as the standardized smoke source at a constant room temperature of 30 degrees Celsius and under NIGHT mode with a threshold of 160. The NIGHT mode was selected for distance testing because it represents the most sensitive operational state of the improved system, enabling direct comparison with the original fixed-threshold system, which used a threshold of 150. Trial 1 response at 10 centimeters was 22 seconds, confirmed by experimental observation, which is slightly longer than the original system's fastest response of 15 seconds at the same distance and threshold proximity. The difference reflects the 3,000-millisecond sustained timer requirement, which adds a minimum latency floor to alarm activation but simultaneously provides the temporal discrimination that suppresses false alarms from brief non-fire aerosol events. Across all tested distances, Trial 2 and Trial 3 consistently produced shorter response times than Trial 1, consistent with the residual smoke conditioning effect documented in the original Calderon et al. study, where trace concentrations from preceding trials reduce the additional concentration required to cross the threshold. The system provides reliable detection across the full 10-100 centimeter range tested.

Table 6. Detection Thresholds Across Source Types: Original System vs. Improved System

Source	**System**	**Mode / Threshold**	**Room Temp**	**Peak Reading**	**Alarm ?**	**Remarks**
Burning Paper	Original	Fixed / 150	32 C	312	YES	Alarm fired after 3 s sustained timer. Peak reading 312, well above the fixed threshold of 150.
Burning Paper	Improved	NIGHT / 160	32 C	318	YES	Alarm fired at 22 seconds during experimental trial. NIGHT mode threshold 160 was correctly triggered. Fire detection maintained.
Cooking Fume	Original	Fixed / 150	32 C	238	YES	FALSE ALARM. Peak reading of 238 exceeded the fixed threshold of 150. The original system could not distinguish cooking smoke from fire smoke.
Cooking Fume	Improved	COOKING / 280	32 C	243	NO	No false alarm. Peak reading 243 remained below the COOKING threshold of 280. The ROC filter also

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VI June 2026

						identified a brief spike pattern. Key improvement demonstrated.
Perfume Spray	Original	Fixed / 150	32 C	219	YES	FALSE ALARM. A momentary spike of 219 exceeded the threshold of 150 in the original system before a rapid drop. No discrimination capability.
Perfume Spray	Improved	DEFAULT / 220	32 C	221	NO	No alarm. ROC filter detected a sharp spike-and-drop pattern (ROC below -30). Alarm suppressed. The default threshold of 220 also provided an additional tolerance margin.
Incense Smoke	Original	Fixed / 150	32 C	198	YES	FALSE ALARM. Sustained low-level incense smoke exceeded the fixed threshold of 150 after prolonged exposure. The original system has no temporal discrimination.
Incense Smoke	Improved	DEFAULT / 220	32 C	201	NO	No alarm. Peak reading 201 stayed below the DEFAULT threshold 220. Sustained timer also filtered a gradual low-level rise as insufficient for genuine fire classification.
Burning Cloth	Original	Fixed / 150	32 C	347	YES	Correct alarm. Dense, sustained smoke from cloth produced a reading of 347, far above the threshold of 150. Rapid alarm.
Burning Cloth	Improved	NIGHT / 160	32 C	352	YES	Correct alarm. Peak reading 352, far above NIGHT threshold 160. The alarm fired quickly after the timer. True fire detection is maintained across all modes.
Burning Tissue	Original	Fixed / 150	32 C	274	YES	Correct alarm. Moderately sustained smoke exceeded the fixed threshold of 150 after the timer window elapsed.
Burning Tissue	Improved	DEFAULT / 220	32 C	279	YES	Correct alarm. Peak 279 exceeded the DEFAULT threshold of 220. Sustained timer confirmed genuine fire signature. Detection maintained.
Candle Smoke	Original	Fixed / 150	32 C	261	YES	Correct alarm. Candle smoke reading 261 sustained above the fixed threshold. Alarm activated near the end of the timer window.

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VI June 2026

Candle Smoke	Improved	DEFAULT / 220	32 C	264	YES	Correct alarm. Reading 264 sustained above the DEFAULT threshold 220 for the full 3,000 ms window. Alarm confirmed as a genuine low-intensity fire source.
Burning Wood	Original	Fixed / 150	32 C	331	YES	Correct alarm. Dense wood smoke at 331 activated the alarm quickly through the fixed threshold system.
Burning Wood	Improved	COOKING / 280	32 C	338	YES	Correct alarm. Even under the most tolerant COOKING threshold of 280, burning wood at 338 correctly triggered the alarm. Genuine fire is always detected.

Table 6 presents the maximum sensor threshold readings and alarm outcomes for all tested source types under both the original fixed-threshold system and the proposed time-aware system. The key comparison is the behavior of non-fire sources under the original fixed threshold of 150 versus their behavior under the appropriate time-of-day mode in the improved system. Cooking fume produced a peak reading of 238-243, which exceeded the original threshold of 150 and triggered a false alarm, but remained below the COOKING threshold of 280 in the improved system, producing no alarm. Perfume spray reached 219 to 221 and triggered a false alarm in the original system, but was suppressed in the improved system by the rate-of-change filter, identifying the sharp spike-and-drop pattern with an ROC value below -30. Incense smoke reached 198-201 and triggered a false alarm in the original system at threshold 150, but remained below the DEFAULT threshold of 220 in the improved system. All genuine fire sources, including burning paper, cloth, tissue, candle, and wood, triggered correct alarms in both systems, confirming that the improved system's false alarm reduction did not come at the cost of fire detection capability. The false alarm rate was reduced from 100 percent in the original system to zero percent in the improved system across all three non-fire source types tested.

Table 7. Response Time in a Controlled Room (DEFAULT Mode, Threshold 220)

Area	**Source**	**Room Temp**	**Mode / Threshold**	**Response Time**
200 cm x 200 cm (room)	Paper	30 C	DEFAULT / 220	Trial 1: 20 min 12 sec (first clean-air activation)
200 cm x 200 cm (room)	Paper	30 C	DEFAULT / 220	Trial 2: 15 min 44 sec
200 cm x 200 cm (room)	Paper	30 C	DEFAULT / 220	Trial 3: 11 min 20 sec
200 cm x 200 cm (room)	Paper	30 C	DEFAULT / 220	Trial 4: 8 min 38 sec
200 cm x 200 cm (room)	Paper	30 C	DEFAULT / 220	Trial 5: 7 min 02 sec
200 cm x 200 cm (room)	Paper	30 C	DEFAULT / 220	Trial 6: 5 min 40 sec
200 cm x 200 cm (room)	Paper	30 C	DEFAULT / 220	Trial 7: 4 min 55 sec
200 cm x 200 cm (room)	Paper	30 C	DEFAULT / 220	Trial 8: 4 min 18 sec
200 cm x 200 cm (room)	Paper	30 C	DEFAULT / 220	Trial 9: 3 min 47 sec
200 cm x 200 cm (room)	Paper	30 C	DEFAULT / 220	Trial 10: 3 min 24 sec

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VI June 2026

Table 7 presents detection response times across 10 sequential trials in a controlled 200-centimeter-by-200-centimeter enclosed room, using burning paper at 30 degrees Celsius under DEFAULT mode with a threshold of 220. The most significant pattern is the consistent reduction in response time across successive trials, from 20 minutes and 12 seconds in Trial 1 to 3 minutes and 24 seconds in Trial 10. This pattern is consistent with the residual smoke conditioning effect: in an enclosed space with limited ventilation, trace concentrations from each preceding trial accumulate as background readings in the sensor, reducing the additional concentration required to cross the detection threshold and producing progressively earlier activation in subsequent trials. Across all ten trials, response time averaged 8 minutes 30 seconds with a sample standard deviation of 5 minutes 38 seconds (range: 3 minutes 24 seconds to 20 minutes 12 seconds); consistent with the conditioning trend described above, this spread is driven by trial order rather than random noise, since the final three trials alone (4 minutes 18 seconds, 3 minutes 47 seconds, and 3 minutes 24 seconds) average 3 minutes 50 seconds with a much tighter standard deviation of 27 seconds, indicating the system reaches a stable, repeatable response time once background conditioning saturates. The DEFAULT mode threshold of 220, being higher than the original threshold of 150, produces slightly longer initial response times in clean-air first-trial conditions, reflecting the greater smoke concentration required to reach the threshold. However, this tradeoff is intentional and acceptable, as the DEFAULT mode is designed for general daytime hours when false alarm risk is the primary concern, and occupants are awake and able to respond. The results confirm that the prototype provides functional coverage within a small enclosed room under all three threshold modes.

Table 8. False Alarm Rate Comparison: Original System vs. Improved System

Non-Fire Source	**Original System (Fixed 150)**	**Improved System (RTC Scheduled)**	**Alarm Reduced?**	**Mode Applied in the Improved System**
Cooking Fume (heated oil)	FALSE ALARM (peak 238, above 150)	NO ALARM (peak 243, below 280)	YES	COOKING mode (6:00 to 8:00 AM, 11:00 AM to 1:00 PM, 5:00 to 7:00 PM)
Perfume Spray	FALSE ALARM (peak 219, above 150)	NO ALARM (ROC filter suppressed)	YES	DEFAULT mode (threshold 220) plus ROC filter
Incense Smoke	FALSE ALARM (peak 198, above 150)	NO ALARM (peak 201, below 220)	YES	DEFAULT mode (threshold 220)
False Alarm Rate	3 out of 3 non-fire sources (100%)	0 out of 3 non-fire sources (0%)	YES	All three improvements combined

Table 8 summarizes the false alarm reduction achieved by the improved system across all three non-fire source types tested. The original fixed-threshold system produced false alarms for all three non-fire sources, yielding a 100 percent false alarm rate across these scenarios. The improved time-aware system produced zero false alarms for all three non-fire sources, driven by the combined effects of the higher COOKING threshold, the DEFAULT threshold, and the rate-of-change filter. This represents a complete elimination of the false alarm problem for the tested non-fire source categories while maintaining a 100 percent true detection rate for all genuine fire sources across all three threshold modes. These results validate the core research hypothesis: that time-of-day threshold scheduling, combined with temporal signal quality filters, is an effective and low-cost strategy for improving smoke detector reliability in household environments.

Problems Encountered and Solutions

Several technical challenges were identified and resolved during prototype development and testing. Each was systematically diagnosed and corrected before formal data collection commenced.

The first issue involved unintended audio alarm activation during system startup. The DFPlayer Mini audio module received spurious serial data over the SoftwareSerial line during the Arduino boot sequence, triggering playback of the alarm audio file even when no sensor condition was met. This was resolved by inserting a 1-

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VI June 2026

second initialization delay before the DFPlayer. begin () call, followed by an immediate mp3.stop() command after initialization. This fix was validated across multiple consecutive power cycles.

The second issue involved the sustained detection timer failing to accumulate the required three-second duration. During initial testing, the alarm never activated, even when sensor readings consistently exceeded the detection threshold. Code analysis revealed that the clearAlarm() function was being called within the caution branch of the detection logic, resetting both the counting flag and the sustained start time variable on every loop iteration. The resolution involved removing the clearAlarm() call from the caution branch and replacing it with direct LED and buzzer state control statements, allowing the timer to accumulate across loop cycles as intended.

The last issue was that sensor readings were consistently elevated and unstable during the first 60 to 90 seconds after each power cycle, attributable to the MQ-2's thermal warm-up behavior. A mandatory minimum warm-up period of 90 seconds was established as a procedural requirement for all formal test trials.

CONCLUSION

This study developed and evaluated a Time-Sensitive Smoke Detection System that implements scheduled threshold adjustment using a DS3231 real-time clock module integrated with the Arduino Uno-based hardware platform of Calderon et al. The DS3231 enables the system to automatically apply three distinct alarm thresholds aligned with the occupancy-driven risk profile of each time period of the day: a higher-tolerance threshold of 280 during cooking hours, a standard threshold of 220 during general daytime hours, and a stricter threshold of 160 during nighttime sleep hours.

Combined with three software-level signal quality improvements, including five-sample reading averaging, 3,000-millisecond sustained-detection timing, and rate-of-change filtering with a drop limit of negative 30 units per cycle, the proposed system addresses the core architectural weaknesses of the original fixed-threshold design. The inability to distinguish between the temporal signal profile of genuine fire smoke and the transient spikes produced by non-fire aerosol sources, and the inability to account for the time-of-day context that determines the relative likelihood and consequence of each, are both resolved by the combined improvements.

Experimental results confirmed that the system correctly transitions between threshold modes as determined by the DS3231 RTC, displays the current mode and time on the LCD continuously, suppresses rate-of-change-identified spike patterns from non-fire sources, and activates the alarm only after sustained threshold exceedance consistent with a genuine fire signal. The false alarm rate across three non-fire source types was reduced from 100 percent in the original system to zero percent in the improved system. Genuine fire scenarios triggered reliable alarm activation across all three threshold modes via both the sustained timer pathway and the immediate flame sensor bypass channel, confirming that true detection capability was maintained throughout.

The study also confirmed a clear differentiation between the proposed system and the closest known prior art. US Patent 4,101,785 from 1977 proposed light-level-based sensitivity switching, which cannot distinguish between different dark-hour periods, was never implemented using a real-time clock, and was never commercialized at low cost. The present work, therefore, represents an original contribution to the design space of affordable, time-aware embedded fire safety systems.

RECOMMENDATIONS

Environmental Variability Testing. This study took place in a controlled lab environment at temperatures between 30 and 32 degrees Celsius, with low airflow changes. Future work should test the prototype's performance across a wider range of temperatures and humidity levels. The DS3231 module also provides a built-in temperature sensor with an accuracy of ±3 degrees Celsius, which was not used in this study. Future research could investigate whether ambient temperature data from the RTC itself can refine threshold adjustment logic for hot kitchen environments where baseline sensor readings may be elevated.

User-Configurable Time Windows. The time window boundaries implemented in this study are based on general Philippine household patterns. Future versions should allow occupants to configure custom time

windows and threshold values, accommodating non-standard schedules such as shift workers, households with different meal times, or commercial establishments with extended operating hours.

Systematic Threshold Optimization. The threshold values of 280, 220, and 160 are empirically derived from controlled observations in the testing environment. Future researchers should conduct systematic calibration trials across a wider range of cooking scenarios and sensor units to establish more generalizable default values with documented performance margins.

Wireless Connectivity Integration. The addition of a Wi-Fi or Bluetooth module, such as an ESP8266 or HC-05, would enable remote monitoring, real-time alert delivery to a paired mobile application, and remote threshold configuration. This would allow context-aware notifications, such as indicating whether the alarm fired during cooking hours (potentially a false alarm) or during sleeping hours (indicating immediate danger).

Standardized Performance Benchmarking. Future studies should evaluate an improved version of this prototype against recognized fire safety testing standards, such as UL 268 or ISO 7240, providing an objective basis for comparing its performance with commercially certified smoke detectors.

Long-Term Reliability and Power Consumption Evaluation. Extended deployment trials spanning several weeks to months would help characterize MQ-2 sensor drift, threshold recalibration needs, and overall system durability under continuous household use. A complementary assessment of current draw across the COOKING, DEFAULT, and NIGHTTIME modes, together with the measured solar contribution under real installation lighting and the resulting runtime of the 18650 battery pack between charges, would validate whether the solar-assisted power system meets its analytically estimated autonomy in practice.

Adaptive Machine Learning-Based Threshold Optimization. Rather than relying on three fixed time windows, a lightweight on-device or companion-app machine learning model could learn a household's actual cooking and sleep patterns from historical sensor and RTC data, adjusting thresholds and time boundaries automatically instead of requiring manual reconfiguration. Such an approach could also incorporate occupancy prediction to further reduce nuisance alarms during transient activities not captured by fixed schedules.

ACKNOWLEDGEMENT

The researchers express sincere gratitude to the faculty and staff of the Computer Engineering Department of Eulogio "Amang" Rodriguez Institute of Science and Technology for their guidance and institutional support throughout this study.

The researchers acknowledge the foundational contributions of Calderon et al., whose Arduino Uno-based smoke detector prototype evaluated at EARIST provided the hardware platform and experimental context upon which this study's improvements were developed. The decision to build upon and extend an existing prototype reflects a commitment to cumulative, reproducible research in the field of low-cost embedded safety systems.

The researchers also acknowledge the broader academic communities in embedded systems, fire detection, and sensor-based safety technology whose published findings and theoretical frameworks provided the methodological and conceptual foundation for this work.

Finally, sincere appreciation is extended to the families and peers of all research group members for their patience, encouragement, and moral support across the development, testing, and documentation phases of this project.

REFERENCES

1. Ali, S. S., Ali, S., Ali, S., Zaidi, A. A., Zafar, A., Munawwar, S., & Shafique, M. (2025). Development of automated fire control system: An integrated solution for gas, fire, and smoke detection. Pakistan Journal of Engineering, Technology and Science, 13(1), 69-77. https://journals.iobm.edu.pk/index.php/pjets/article/view/1278

2. Ayranci, A. A., & Erkmen, B. (2024). IoT-based fire detection: A comparative study of machine learning techniques. Nigde Omer Halisdemir University Journal of Engineering Sciences, 13(4), 1298-1307. https://doi.org/10.28948/ngumuh.1444349
3. Calderon, J. R. P. S., Centismo, C. J. C., Gruy, K. Y., Parale, M. C. B., & Fabro, M. N. (2026). Experimental study on the performance of an enhanced smoke detector. *International Journal of Research and Scientific Innovation*, *13*(1), 740–753. https://dx.doi.org/10.51244/IJRSI.2026.13010064
4. Circuit Digest. (2025). DS3231 RTC interfacing with Arduino to build DIY digital clock. https://circuitdigest.com/microcontroller-projects/interfacing-ds3231-rtc-with-arduino-and-diy-digital-clock
5. Costea, A., & Schiopu, P. (2018, June). New design and improved performance for smoke detector. In 2018 10th International Conference on Electronics, Computers and Artificial Intelligence (ECAI) (pp. 1-7). IEEE. https://ieeexplore.ieee.org/abstract/document/8679032
6. Honeywell System Sensor. (n.d.). Reducing detector-based nuisance alarms [White paper]. https://buildings.honeywell.com/content/dam/hbtbt/en/documents/document-lists/system-sensor/white-papers/SmokeDetectors-ReducingNuisanceAlarms%20WhitePaper%20A05-0340.pdf
7. Hwang, E. H., Choi, H. B., & Choi, D. M. (2023). Response characteristics of smoke detection for reduction of unwanted fire alarms in studio-type apartments. Fire, 6(9), 362. https://www.mdpi.com/2571-6255/6/9/362
8. Last Minute Engineers. (2026). In-depth: Interface DS3231 precision RTC module with Arduino. https://lastminuteengineers.com/ds3231-rtc-arduino-tutorial/
9. Last Minute Engineers. (2026). How MQ2 gas/smoke sensor works and interface it with Arduino. https://lastminuteengineers.com/mq2-gas-senser-arduino-tutorial/
10. Mensch, A., & Cleary, T. (2024, September). New smoke alarms are better at detecting fires but still beep for bacon. National Institute of Standards and Technology. https://www.nist.gov/news-events/news/2024/09/new-smoke-alarms-are-better-detecting-fires-still-beep-bacon
11. Oak Ridge National Laboratory. (2016). Smart smoke detectors reduce false alarms and save lives. https://www.ornl.gov/success-story/smart-smoke-detectors-reduce-false-alarms-and-save-lives
12. Parallax. (2025). DS3231 AT24C32 real time clock module. https://www.parallax.com/product/ds3231-at24c32-real-time-clock-module/
13. Random Nerd Tutorials. (2019). Guide for MQ-2 gas/smoke sensor with Arduino. https://randomnerdtutorials.com/guide-for-mq-2-gas-smoke-sensor-with-arduino/
14. United States Patent and Trademark Office. (1977). Smoke detector with switch means for increasing the sensitivity (US Patent No. 4,101,785). https://image-ppubs.uspto.gov/dirsearch-public/print/downloadPdf/4101785
15. Electronic Clinic. (2024). Arduino gas sensor MQ2, smoke detector, programming and circuit. https://www.electroniclinic.com/arduino-gas-sensor-mq2-smoke-detector-programming-circuit/
16. Fang, F., Tang, Y., Tang, H., Song, Y., Chen, Q., Rao, L., & Cao, L. (2025). Interface-enhanced PbS quantum dot short-wave infrared photodetector toward instant smoke detection applications. ACS Applied Electronic Materials, 7(15), 7368-7376. https://pubs.acs.org/doi/10.1021/acsaelm.5c01168
17. Hwang, W. Y., Lee, H. J., Jin, J., et al. (2024). Computational design of a smoke detector with high sensitivity considering three-dimensional flow characteristics. Case Studies in Thermal Engineering, 53, 103896. https://www.sciencedirect.com/science/article/pii/S2214157X23012029
18. Ifeoma, N., Alagbu, E., Okeke, O., Ikpo, K., & Onwuamanam, C. (2025). Development of a CNN-based smoke/fire detection system for high-risk environments. European Journal of Science, Innovation and Technology. https://ejsit-journal.com/index.php/ejsit/article/view/410
19. Szafarczyk, A. (2022). Examination of the influence of sensitivity setting of smoke detectors on their susceptibility to false alarms. https://doi.org/10.12912/27197050/155024
20. Wu, Y., & Ying, X. (2023). Digital twin-based smoke alarm calibration methodology for reliable detection performance. Applied Sciences. https://www.mdpi.com/journal/applsci
21. Zhang, D. (2024). A YOLO-based approach for fire and smoke detection in IoT surveillance systems. International Journal of Advanced Computer Science and Applications, 15(1). https://thesai.org/Downloads/Volume15No1/Paper_9-A_Yolo_based_Approach_for_Fire_and_Smoke_Detection.pdf