Bluetooth-controlled smart home automation with adaptive lighting and security

Kutay DAMAR[1]
Mohammad Ruhul Amin BHUIYAN[2]
Hayati MAMUR[3]

Geliş tarihi / *Received:* 13.02.2026
Düzeltilerek geliş tarihi / *Received in revised form:* 07.05.2026
Kabul tarihi / *Accepted:* 26.06.2026
DOI: 10.17932/IAU.ABMYOD.2006.005/abmyod_v21i73009

Abstract
Smart home technologies play a crucial role in energy efficiency and user comfort. However, commercial systems are often expensive and lack customizable security features. This study presents a cost-effective, hybrid-controlled smart home automation prototype designed with microcontroller (MCU) and Bluetooth communication. The system integrates environmental sensing such as temperature, humidity, light, and motion with a Finite State Machine (FSM) control algorithm. A unique "Black Box" feature has been developed using EEPROM to record critical security events and timestamps. The system is controlled via a custom Android application developed with Mobile Application Interface (MIT App Inventor), allowing for both manual override and autonomous operation. Experimental results show that the adaptive lighting algorithm, which employs an exponential smoothing filter, prevents flickering, and the security recording system successfully logs all unauthorized attempts.The proposed system offers a robust application model for Internet of Things (IoT) and embedded systems.

Keywords: *Smart home, IoT, Bluetooth, FSM, EEPROM logging, Adaptive lighting*

[1] Manisa Celal Bayar University, damarqkutkiq@gmail.com, ORCID:0009-0007-6281-2660
[2] Islamic University, mrab_eee@iu.ac.bd, ORCID:0000-0001-7335-4158
[3] Sorumlu Yazar/ Corresponding Author: Manisa Celal Bayar Üniversitesi, Mühendislik Fakültesi, Elektrik-Elektronik Mühendisliği hayati.mamur@cbu.edu.tr, ORCID:0000-0001-7555-5826

Bluetooth kontrollü uyarlanabilir aydınlatma ve güvenlik özellikli akıllı ev otomasyonu

Öz

Akıllı ev teknolojileri, enerji verimliliği ve kullanıcı konforunda çok önemli bir rol oynamaktadır. Bununla birlikte, ticari sistemler genellikle yüksek maliyetlidir ve özelleştirilebilir güvenlik özelliklerinden yoksundur. Bu çalışma, mikrodenetleyici ve Bluetooth iletişimi ile tasarlanmış, maliyet etkin, hibrit kontrollü bir akıllı ev otomasyon prototipini sunmaktadır. Sistem, sıcaklık, nem, ışık, hareket gibi çevresel algılamayı Sonlu Durum Makinesi (FSM) kontrol algoritmasıyla entegre eder. Kritik güvenlik olaylarını ve zaman damgalarını kaydetmek için EEPROM kullanılarak benzersiz bir "Kara Kutu" özelliği geliştirilmiştir. Sistem, mobil uygulama arayüzü (MIT App Inventor) ile geliştirilen özel bir Android uygulaması aracılığıyla kontrol edilir ve hem manuel geçersiz kılma hem de otonom çalışma imkânı sağlar. Deneysel sonuçlar, üstel düzeltme filtresi kullanan uyarlanabilir aydınlatma algoritmasının titremeyi önlediğini ve güvenlik kayıt sisteminin tüm yetkisiz girişimleri başarıyla kaydettiğini göstermiştir. Önerilen sistem, Nesnelerin İnterneti (IoT) ve gömülü sistem için sağlam bir uygulama modeli sunmaktadır.

Anahtar Kelimeler: *Akıllı ev, IoT, Bluetooth, FSM, EEPROM kaydı, Uyarlanabilir aydınlatma*

Introduction

Rapid urbanization and the demand for energy efficiency have accelerated the need to transform traditional living spaces into intelligent spaces (Shahidehpour and Ganji, 2018; Oluwafemi et al., 2023). Smart home automation systems have become critical solutions to satisfy these requirements, providing improved user comfort, security, and substantial energy savings through the intelligent control of lighting, heating, and ventilation systems (Gunge and Yalagi, 2016; Mehmood et al., 2019; Visan and Diaconu, 2022). Advances in Internet of Things (IoT) technology have enabled the integration of wireless protocols and microcontrollers (MCUs), leading to cost-effective and scalable cyber-physical systems (He et al., 2016; Algani et al., 2021). The main goal of these systems is not only to provide remote control functionality but also to establish an autonomous environment that can sense environmental changes and respond accordingly to optimize resource utilization (Giannaros et al., 2023; Popescu et al., 2024).

Fossil fuels have been the main source of energy for residential heating and electricity generation; however, their exhaustion and the resulting greenhouse gas emissions have led to the quest for sustainable alternatives (Meng et al., 2024; Chinnathambi et al., 2022). In this regard, the smart home system has become a critical solution that curtails unnecessary energy consumption through automation based on sensor inputs (Babangida et al., 2022; Naved et al., 2022). The latest research in the academic community has comprehensively investigated different architectures for home automation. For example, Candan and Erdem (2020) illustrated the viability of low-cost smart home systems using mobile control interfaces, emphasizing the feasibility of MCU-based solutions. Similarly, Gupta and Singh (2019) analyzed the automation of IoT through Bluetooth sensor networks and demonstrated that short-range wireless communication can be used for reliable connectivity in home automation. Moreover, the works of Kim and Park (2018) have highlighted the need for accurate environmental sensing through DHT22 sensors for maintaining indoor air quality and temperature, while İlhan et al. (2021) explored energy-adaptive lighting systems that employ light-dependent resistor (LDR) sensors to control artificial lighting based on available natural daylight.

Despite these major breakthroughs, there is a critical need in low-cost embedded automation (Yavuz, 2024; Chakraborty et al., 2023). Currently, most existing systems and “Do-It-Yourself” (DIY) solutions available are nothing more than basic remote-control interfaces and lack the reliability, fault tolerance, and traceability of data necessary for successful implementation (Pascoal et al., 2025; Dobrojevic and Bacanin, 2022). Unlike industrial automation, which uses Programmable Logic Controllers (PLCs) and Supervisory Control and Data Acquisition (SCADA) systems to ensure safety and data logging, most DIY smart home solutions lack the ability to retain critical data even after a power outage (Khairullah and Sharkawy, 2022; He et al., 2024). This is a major security concern, as unauthorized access attempts or alarms may go unlogged. Additionally, most systems lack the ability to strike a balance between autonomous and manual control, often leading to conflicting commands in which manual overrides interfere with automated sensor loops (Rahmanovic et al., 2023; Naved et al., 2022).

The research work is an attempt to meet these engineering requirements by developing and testing a resilient, hybrid-controlled smart home automation system that models the complexity of contemporary industrial networks. Unlike traditional systems, the developed system is based on a Finite State Machine (FSM) architecture, which is strongly recommended in current literature because of its determinism and stability. The system can thus smoothly switch between autonomous sensor-controlled states and user-controlled states without any logical inconsistencies. One of the innovative aspects of this project is the integration of a “Black Box” security logging system based on industrial traceability best practices. Using the MCU’s built-in Electrically Erasable Programmable Read-Only Memory (EEPROM), the system is designed to provide a non-volatile logging mechanism for critical events such as fire alarms, door presses, and failed password attempts, all of which are timestamped. Unlike most low-cost do-it-yourself (DIY) smart home systems that rely on simple procedural loops and lack persistent event logging, our approach explicitly combines an FSM with EEPROM-based security recording. The FSM ensures deterministic state transitions and prevents conflicting manual/ autonomous commands, while the EEPROM ‘Black Box’ provides non-volatile traceability of security events-a feature typically absent in MCU-based home automation projects.

Materials and methods

This section describes the hardware components, circuit topology, and software development tools that were used to implement the embedded smart home automation system. The system integrates analog sensor interfaces and digital communication protocols on a single hardware platform. The choice of materials was guided by the requirements for low power consumption, cost-effectiveness, and industrial-grade reliability. The system architecture is given by Figure 1.

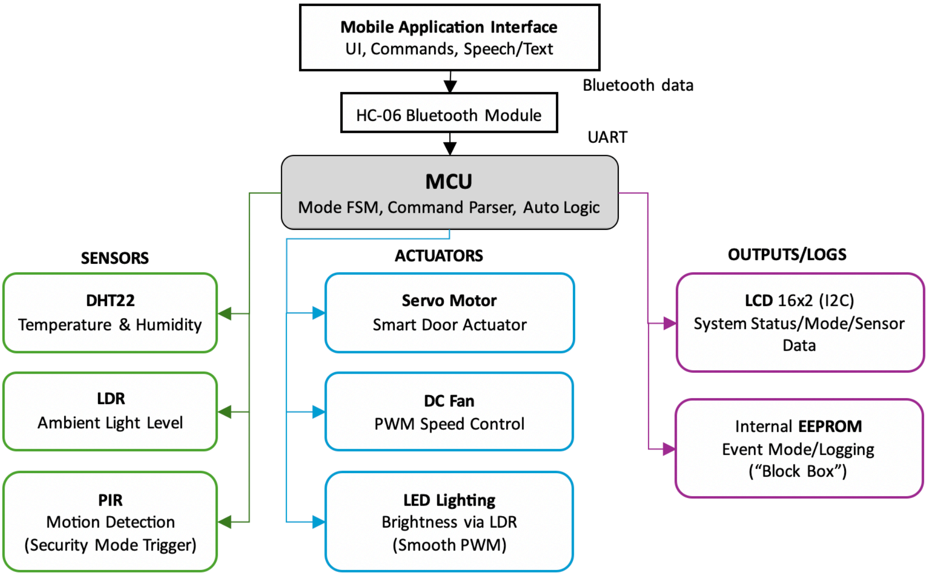


Figure 1. System architecture diagram

Embedded control unit and system architecture

The system operates through the Arduino UNO MCU board which functions as its main processing unit while running on the ATmega328P architecture. The system architecture consists of four separate modules which include the sensing layer and the actuation layer and the communication layer and the user interface layer. The designers separated high-current devices which included the DC fan and servo motor from the analog sensor lines to protect these lines from Electromagnetic Interference (EMI) and to stop voltage drops. The system required a common grounding method which used star-ground topology to eliminate ground loops that would otherwise interfere with analog digital conversion (ADC) reading accuracy.

Environmental sensing subsystem

The sensing layer comprises three heterogeneous sensors chosen to provide a comprehensive environmental profile. One of them is DHT22 temperature and humidity sensor. This sensor utilizes a capacitive humidity sensor and a thermistor to measure the surrounding air. It communicates via a proprietary single-wire protocol. The sensor provides digital output reflecting the thermal comfort levels necessary for the automated fan control logic. Secondly, it is the LDR. To enable adaptive lighting, an LDR module was employed. Since the LDR operates in the analog domain, it was configured in a voltage divider circuit with a fixed resistor. This configuration allows the MCU's 10-bit ADC to sample a voltage proportional to ambient luminosity. Lastly, it is Passive Infrared (PIR) sensor motion sensor, integrated for security monitoring. The sensor operates as a digital trigger, outputting a high logic signal when motion is detected (Salah and Jamal, 2017). To mitigate false triggers caused by thermal noise or power fluctuations, the PIR sensor was powered by a stable 5 V rail and placed physically distant from heat-generating components like the voltage regulator.

The actuation layer handles the physical manipulation of the environment through three primary mechanisms: ventilation, access control, and illumination. An N-channel MOSFET was selected to drive the DC fan. This component acts as a low-side switch, allowing the MCU to regulate the fan speed via Pulse Width Modulation (PWM) without bearing the high current load directly (Alim et al., 2024). During the initial design phase, a "Floating Gate" phenomenon was observed, where the fan would activate spuriously during system resets due to electrostatic coupling. To resolve this, a 10 kΩ pull-down resistor was installed between the Gate and Source pins, ensuring the MOSFET remains in a strictly "OFF" state when the MCU signal is high impedance. Another actuator is a servo motor mechanism. A standard micro servo motor is used to simulate the smart door lock. Since servo motors can draw high instantaneous currents during movement, potentially causing voltage dips that reset the MCU, the servo was supplied with a robust power line separate from the sensor array. Lastly, a PWM-controlled LED was used for lighting. The light emitting diode (LED) lighting system is driven directly by the MCU's PWM output (Trivedi, 2016). This setup allows for linear brightness dimming control rather than simple on/off switching, enabling the implementation of the smooth lighting algorithm.

The communication backbone of the system is the HC-06 Bluetooth Serial Module (Kumar and Verma, 2020). This module utilizes the Universal Asynchronous Receiver-Transmitter (UART) interface to establish a bidirectional data link between the embedded system and the mobile application (Xu et al., 2022). Since the HC-06 operates at 3.3V logic while the MCU operates at 5 V, a voltage divider circuit consisting of resistors was implemented on the Receiver (RX) line of the module. The module was physically positioned away from inductive loads, such as the fan motor, to reduce susceptibility to Radio Frequency (RF) noise (Dwiansyah and Prasetya, 2022). For the user interface, custom Android application was developed using Mobile Application Interface (MIT App Inventor) (Sharto, 2025). This platform was chosen for its ability to handle event-driven programming, which is essential for processing such as temperature, humidity, security status real-time telemetry data received from the MCU. The application serves as a Human-Machine Interface (HMI), replacing traditional physical control panels. To implement the "Black Box" security logging feature, the system utilizes the MCU's internal EEPROM. Unlike volatile RAM, EEPROM retains data after power loss (Das et al., 2023).

The ATmega328P's EEPROM has a specified endurance of approximately 100,000 write cycles per cell. To manage this limitation, the system implements a conditional logging strategy: only critical events (mode changes, failed password attempts, security triggers) are recorded. Furthermore, the same event type is not written more than once per second, even if continuously triggered. Under typical daily usage (e.g., 10–20 security events per day), this yields an estimated EEPROM lifetime of over 10 years. Wear-leveling was not implemented due to the small log size (32 bytes) and the low write frequency, but it remains a recommended improvement for high-frequency logging scenarios.

Finite state machine control architecture

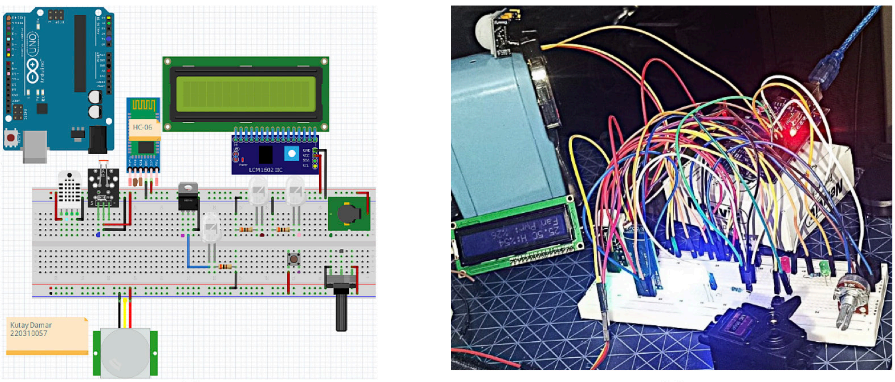
To manage the complex and often asynchronous interactions between autonomous sensor inputs and manual user overrides, FSM model was implemented (Merlin Sajini et al., 2023). In standard procedural programming, utilizing functions like delay() can block the processor, rendering the system unresponsive to urgent interrupts such as security alarms (Vandana et al., 2022). To overcome this, the FSM approach divides the system's operation into discrete, mutually exclusive states, where transitions are governed by specific trigger events. The FSM consists of some primary operational states, each with a defined behavioral profile. One of them is initialization state. Upon system power-up, the MCU executes a setup routine. This includes initializing the UART communication at 9600 baud rate, configuring the I/O pins for the sensors and actuators, and retrieving the last known state from the EEPROM to ensure continuity after a power loss. Later, automatic mode is the default operating state where the system functions as a closed-loop controller. The MCU continuously polls the DHT22 and LDR sensors. A proportional control algorithm maps the ambient temperature to the fan's PWM duty cycle; for instance, as the temperature rises from 25-40°C, the fan speed increases linearly. Simultaneously, the LED brightness is adjusted inversely to the ambient light level. To prevent rapid switching around threshold values, a hysteresis band is applied to the sensor readings. Third is manual mode. This state mimics the "Manual Override" function found in industrial SCADA systems. It grants the user full authority over the actuators via the Bluetooth interface. However, safety interrupts retain higher priority and can override user commands. Another is security mode. Designed for perimeter monitoring, this state prioritizes the PIR sensor interrupt. This sequence involves activating the buzzer, locking the servo-driven door mechanism, and logging the event timestamp to the non-volatile memory. Lastly, it is parent mode. This state implements an access control layer. When a user attempts to access restricted features, the system demands a password. The input is hashed and compared against the stored credential in the firmware. FSM state transition table is given in Table 1.

Table 1. FSM state transition.

Current State	Trigger Event	Next State	System Action
Init	Setup completed	Auto	Initialize UART (9600 baud), configure I/O pins, retrieve last state from EEPROM
Auto	"Manual" command received via Bluetooth	Manual	Disable sensor-based automatic control, enable user control over fan/LED/servo
Auto	PIR motion detected + Parent mode active	Security	Activate buzzer, lock servo (door), log event with timestamp to EEPROM
Auto	Temperature > critical threshold (e.g., 40°C)	Auto (fail-safe)	Force fan to 100% PWM, open servo (emergency egress)
Manual	No user input for 5 minutes (timeout)	Auto	Return to closed-loop sensor-based control
Manual	"Auto" command received via Bluetooth	Auto	Switch back to automatic mode
Security	"Alarm reset" command received	Auto	Deactivate buzzer, unlock servo, resume normal operation
Any state	"Parent" command received via Bluetooth	Parent	Request password via Bluetooth, disable other commands

Mobile interface and event-driven logic

The user interface was developed using MIT App Inventor, functioning as an HMI. The application logic is event-driven, meaning it reacts to asynchronous user inputs and incoming data packets. To maintain synchronization between the physical system and the digital twin, the app utilizes a polling timer. Every few seconds, it sends a status request command to the MCU. The MCU responds with a concatenated string containing current temperature, humidity, and mode status, which the app parses to update the on-screen labels. Voice commands are converted to text and matched against a predefined dictionary. The hardware and circuit design and setup of the system is given Figure 2.



(a) (b)

Figure 2. System (a) hardware wiring circuit design and (b) setup

In case of Bluetooth disconnection, the mobile application detects the loss of serial communication using a timeout mechanism in 2 seconds without response. The app then displays a “Device Disconnected” warning message and disables all manual control buttons to prevent user confusion. Meanwhile, the MCU system independently continues to operate in the last stable mode “Automatic or Security mode”, ensuring that safety functions remain active. When the connection is re-established, the app automatically re-synchronizes the status by requesting a full system update.

Electrical noise suppression and power management strategy

One of the primary challenges in mixed-signal embedded designs is the interference caused by high-current electromechanical actuators on sensitive analog signal lines. To mitigate this, a strict star-ground topology was implemented in circuit design. Instead of daisy-chaining ground connections, the ground return paths of the Servo Motor and DC Fan high-power components and the DHT22 and LDR sensitive sensor array were routed separately to the main power supply terminal. This methodology prevents ground loops and minimizes the ground bounce effect, which can introduce significant jitter into ADC readings.

Safety logic and fail-safe mechanisms

Beyond standard operation, the system incorporates dedicated fail-safe logic to handle abnormal environmental conditions or hardware failures. These safety protocols are hard-coded into the FSM with the highest execution priority. Firstly, if the DHT22 sensor reports a temperature exceeding a critical safety threshold, the system automatically overrides any user command. It forces the fan to 100% duty cycle and actuates the servo to the Open position to maximize ventilation and facilitate emergency egress. Secondly, to prevent brute-force attacks on the Parent Mode, the system implements a penalty mechanism. Thirdly, the firmware includes a sanity check for sensor data. If the DHT22 returns a NaN (Not a Number) or checksum error, the system discards the reading and maintains the previous valid state rather than acting on erroneous data, ensuring operational stability.

Results and discussion

This section presents the experimental results obtained from the implementation of the smart home automation system and provides an in-depth discussion of its operational behavior, reliability, and performance across different modes. The objective of this section is to demonstrate that the system fulfills the intended functional specifications, responds accurately to environmental changes, handles user commands consistently, and exhibits stability under long-term operation. The analysis includes prototype verification, sensor response profiling, actuator behavior evaluation, Bluetooth communication performance, FSM-driven mode transitions, and the interpretation of recorded data from the EEPROM logging subsystem.

Prototype verification and physical system evaluation

The physical assembly of the smart home system was built on a breadboard platform, which integrated all sensors, actuators, and communication modules into one cohesive embedded architecture. During the verification stage, the prototyped circuit was compared to the theoretical wiring diagram point by point to ensure that all power lines, signal paths, and grounding points were correctly placed as intended. The spatial arrangement of components minimized electrical noise, avoided unnecessary cross-interference between analog and high-current signals, and ensured that the system operated under stable voltage conditions. Once the correctness of hardware connections was assured, every subsystem was powered on separately to check for electrical stability. The system initialization sequence was observed on the LCD screen, confirming that the MCU successfully booted and loaded the firmware. The PIR module displayed stable idle output without false activations, and the HC-06 module entered pairing mode immediately, indicating stable UART communication. The servo motor maintained positional accuracy during calibration tests, and the MOSFET successfully drove the fan under multiple PWM duty cycles.

Table 2 presents the bill of materials and estimated cost of the prototype. The total cost is approximately 42 USD, which is significantly lower than commercial smart home systems (typically 100–300 USD for basic kits).

Table 2. Bill of materials and approximate cost of the prototype.

Component	Unit Cost (USD)	Quantity	Total Cost (USD)
Arduino UNO (clone)	5	1	5
HC-06 Bluetooth module	4	1	4
DHT22 temperature/humidity sensor	10	1	10
PIR motion sensor	2	1	2
LDR + 10kΩ resistor	1	1	1
Servo motor (SG90)	3	1	3
MOSFET (IRFZ44N) + DC fan	5	1	5
LCD 16x2 with I2C interface	6	1	6
Breadboard + jumper wires	5	1	5
Resistors (220Ω, 10kΩ, etc.)	1	1	1
		Total:	~42 USD

The prototype consumes approximately 250 mA at 5 V (1.25 W) when all sensors are active and the fan is running at 50% PWM duty cycle. In standby mode (no fan, no servo movement), the current drops to about 80 mA (0.4 W).

For comparison, typical Wi-Fi-based smart plugs (e.g., ESP8266) consume 1–2 W in active mode and have a standby current of ~70 mA. Zigbee nodes often operate below 0.5 W in active mode and offer lower power consumption due to their mesh network architecture. Our system's power draw is reasonable for a prototype but can be further reduced by implementing sleep modes on the Arduino and using a more energy-efficient MCU such as the ESP32 with deep-sleep capability.

Sensor behavior and environmental response analysis

Much of the system's performance depends upon the precision and responsiveness of the environmental sensors. The temperature and humidity readings from the DHT22 sensor were analyzed over a continuous period of operation. Figure 3 (a) presents the time-series data of the sensor readings. It was observed that the temperature reading increases steadily from a heat source and stabilizes without oscillation, demonstrating the effectiveness of the sampling filter implemented in the data-processing pipeline.

The humidity reading variations are slow, as expected for a capacitive humidity sensor, and are compliant with the datasheet expectations. These results validate that the DHT22 sensor is appropriate for driving the temperature-dependent fan control logic during Automatic Mode. To prevent rapid on-off switching around the threshold values, a hysteresis band of ±0.5 °C for temperature and ±10 lux for light intensity was applied. For example, the fan activates when the temperature rises above 28 °C but only deactivates when it drops below 27.5 °C.

Actuator performance and Bluetooth communication analysis

Bluetooth communication forms the backbone of the user-system interaction. Through performance testing, it was determined that the HC-06 module is stable in typical household distances. A critical aspect of the system was the real-time feedback loop between the LDR sensor and the mobile interface. Figure 3 (b) illustrates the continuous transmission of LDR sensor readings over the Bluetooth link. The rapid fluctuations correspond directly to ambient light changes, demonstrating low-latency communication and reliable real-time user-interface feedback.

The exponential smoothing algorithm applied to these readings ensured that the LED brightness modulation remained smooth, preventing the flickering effect often seen in raw sensor data applications. The latency measures were less than 150 ms for an average command response time, proving that the communication protocol is acceptable for real-time user interaction. The HC-06 Bluetooth module is typically specified as having an effective communication range of up to 10 meters in open space (line-of-sight). However, when a single interior wall (approximately 15 cm brick) is introduced, the reliable range is generally reduced to about 5 meters. Through two walls, communication often becomes intermittent. For comparison, Bluetooth Low Energy (BLE) modules (e.g., HM-10) typically offer ranges of 20–30 m line-of-sight and better wall penetration at similar power consumption. Wi-Fi-based solutions (e.g., ESP8266) provide virtually unlimited range via internet access but consume significantly more power (standby current of ~70 mA vs. HC-06's ~30 mA). For a low-cost, short-range indoor prototype, HC-06 remains acceptable, but future versions should consider BLE for improved real-world scalability.

Security subsystem and EEPROM logging validation

A critical element of the system is its ability to preserve key events through EEPROM logging. During validation, repeated mode transitions, incorrect password attempts, and security activations were performed to populate the log. Retrieval through the serial monitor confirmed that entries were written sequentially, without corruption, duplication, or memory pointer misalignment. Figure 3 (c) shows a segment of the recorded history containing mode changes, failed login attempts, and security alerts derived from the logged data. These results validate the selective-write strategy used to preserve memory longevity and prove that the system can serve as a “Black Box” for post-incident security auditing.

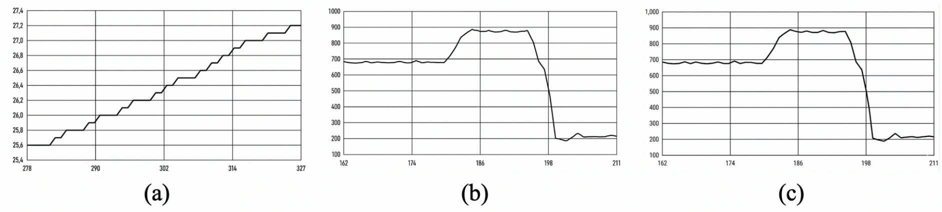


Figure 3. System (a) sensor data stability, (b) real-time LDR data and (c) event logs from EEPROM

The complete system was subjected to extended runtime tests to evaluate long-term performance. For several hours, the system was operated in Automatic Mode while environmental conditions were varied gradually. Throughout testing, the LDR responded accurately to both natural and artificial light shifts, the fan operated continuously without PWM collapse, and the servo executed emergency responses cleanly. Crucially, no unintended resets or freezes were observed. This confirmed the robustness of the electrical design, the reliability of the FSM structure, and the efficiency of the data processing pipeline. Overall, the results demonstrate that the smart home prototype meets all project objectives. Sensor behavior is accurate and stable, actuator responses are smooth and reliable, operational modes function without interference, communication latency is minimal, and EEPROM logging provides durable traceability. The results confirm that the system is both technically sound and suitable for real-world deployment as a compact home automation controller.

Conclusions

The smart home automation system prototype developed in this study was comprehensively tested in terms of hardware integrity, sensor/actuator performance, Bluetooth communication, and EEPROM-based event logging, and it has been experimentally verified that all fundamental requirements are met. The physical assembly exactly matched the theoretical circuit schematic; the DHT22 sensor produced stable temperature and humidity readings without oscillation, while the PIR sensor exhibited no false activations. The average command response time of under 150 ms via Bluetooth demonstrated sufficient performance for real-time user interaction, and the exponential smoothing algorithm applied to LDR data eliminated flickering effects in LED brightness, providing precision and comfort in actuator behavior. Critical events such as unauthorized access attempts, mode changes, and fire alarms recorded in EEPROM maintained data integrity even after repeated tests, establishing the system as a reliable black box for security auditing. During long-term operation tests, no freezing, resets, or PWM degradation were observed; responses to varying environmental conditions in Automatic Mode were accurate and delay-free. In conclusion, the developed prototype meets all design objectives in terms of measurement accuracy, response speed, communication stability, and logging reliability, offering a compact, scalable, and deployment-ready smart home controller as an original application for real-world conditions. Despite its successful operation, the current prototype has several limitations. First, the HC-06 Bluetooth module offers limited range (approximately 10 m line-of-sight) and poor wall penetration, which restricts deployment in larger or multi-room homes. Second, the internal EEPROM has a finite write endurance of about 100,000 cycles. Although conditional logging minimizes writes, heavy security event logging could eventually wear out the memory. Third, the system was tested only in a laboratory environment, not in a real household over extended periods. Future work will address these limitations by migrating to BLE, e.g., HM-10 module, for improved range and lower power consumption. A wear-leveling algorithm will be implemented to extend EEPROM lifetime, or external flash memory will be used instead. Additionally, real-home long-term deployment and user experience studies are planned to validate the system's practical reliability.

Funding

This research received no funding.

Conflict of interest statement

Authors declare no competing interests.

Ethics Committee Statement

This study was carried out solely within the scope of prototype production and technical evaluation, without involving any human or animal participants. Therefore, ethics committee approval was not required.

References

[1] Algani, Y. M., Balaji, S., AlbertRaj, A., Elangovan, G., PJ, S. K., Agordzo, G. K., & Pentang, J. T. (2021). Integration of internet protocol and embedded system on IoT device automation.

[2] Alim, S. A., Shobowale, K. O., David, C. N., Mahmood, M. N., &

Ismail, N. (2024). Development of smart home automation system for light, sockets and fan control with mobile application. *Journal of Engineering for Development*, 16(4), 40-54.

[3] Babangida, L., Perumal, T., Mustapha, N., & Yaakob, R. (2022). Internet of Things (IoT) based activity recognition strategies in smart homes: A review. *IEEE Sensors Journal*, 22(9), 8327-8336.

[4] Candan, A., & Erdem, M. (2020). Design and implementation of a low-cost smart home automation system using Arduino and mobile control. Journal of Electronic Systems, 12(3), 145–152.

[5] Chakraborty, A., Islam, M., Shahriyar, F., Islam, S., Zaman, H. U., & Hasan, M. (2023). Smart home system: a comprehensive review. *Journal of Electrical and Computer Engineering*, 2023(1), 7616683.

[6] Chinnathambi, N. D., Nagappan, K., Samuel, C. R., & Tamilarasu, K. (2022). Internet of things-based smart residential building energy management system for a grid-connected solar photovoltaic-powered DC residential building. *International Journal of Energy Research*, 46(2), 1497-1517.

[7] Das, D., Nishimura, Y., Vivek, R. P., Takeda, N., Fish, S. T., Ploetz, T., & Chernova, S. (2023). Explainable activity recognition for smart home systems. *ACM Transactions on Interactive Intelligent Systems*, 13(2), 1-39.

[8] Dobrojevic, M., & Bacanin, N. (2022). IoT as a backbone of intelligent homestead automation. *Electronics*, 11(7), 1004.

[9] Dwiansyah, R. O., & Prasetya, D. A. (2022). Smart home system with radio frequency using blynk application. Emitor: *Jurnal Teknik Elektro*, 22(2), 177-184.

[10] Giannaros, A., Karras, A., Theodorakopoulos, L., Karras, C., Kranias, P., Schizas, N., ... & Tsolis, D. (2023). Autonomous vehicles: Sophisticated attacks, safety issues, challenges, open topics, blockchain, and future directions. *Journal of Cybersecurity and Privacy*, 3(3), 493-543.

[11] Gunge, V. S., & Yalagi, P. S. (2016). Smart home automation: a literature review. *International Journal of Computer Applications*, 975(8887-8891), 24679-2568.

[12] Gupta, N., & Singh, R. (2019). IoT-based home automation using Bluetooth and sensor networks. International Conference on Smart Technologies (pp. 88–93). IEEE.

[13] He, J., Lo, D. C. T., Xie, Y., & Lartigue, J. (2016, October). *Integrating Internet of Things (IoT) into STEM undergraduate education: Case study of a modern technology infused courseware for embedded system course*. In 2016 IEEE frontiers in education conference (FIE) (pp. 1-9). IEEE.

[14] He, W., Baig, M. J. A., & Iqbal, M. T. (2024). An Internet of things—supervisory control and data acquisition (IoT-SCADA) architecture for photovoltaic system monitoring, control, and inspection in real time. *Electronics*, 14(1), 42.

[15] İlhan, B., Karakaya, S., & Doğan, F. (2021). Development of an energy-adaptive lighting system using LDR-based feedback control. International Journal of Embedded Systems, 9(2), 101–110.

[16] Khairullah, S. S., & Sharkawy, A. N. (2022). Design and implementation of a Reliable and Secure Controller for Smart Home Applications based on PLC. *Journal of Robotics and Control* (JRC), 3(5), 614-621.

[17] Kim, J., & Park, H. (2018). Real-time environmental monitoring using DHT22 temperature-humidity sensors for low-power automation systems. Sensors and Actuators A, 275, 72–80.

[18] Kumar, V., & Verma, A. (2020). Performance evaluation of HC-05/HC-06 Bluetooth modules in embedded control applications. International Journal of Wireless Technologies, 4(1), 33–39.

[19] Mehmood, M. U., Chun, D., Han, H., Jeon, G., & Chen, K. (2019). A review of the applications of artificial intelligence and big data to buildings for energy-efficiency and a comfortable indoor living environment. *Energy and Buildings*, 202, 109383.

[20] Meng, H., Feng, S., & Li, C. (2024). An integrated system of energy generation, storages, and appliances consumption based on machine learning techniques and internet of things. *Journal of Energy Storage*, 87, 111380.

[21] Merlin Sajini, M. L., Suja, S., Merlin Gilbert Raj, S., Kowsalyadevi, S., & Maria, C. (2023). Finite state machine-based load scheduling

algorithm for smart home energy management. *IETE Journal of Research*, 69(10), 7460-7475.

[22] Naved, H., Mustafa, R., & Khan, A. (2022). State-machine-based control architecture for hybrid manual/automatic smart systems. Proceedings of the International Conference on Computing and Automation (pp. 211–218). IEEE.

[23] Oluwafemi, I. B., Bello, O. O., & Obasanya, T. (2023). Design and implementation of a smart home automation system. *Innovare Journal of Engineering and Technology*, 16(4), 3-7.

[24] Popescu, S. M., Mansoor, S., Wani, O. A., Kumar, S. S., Sharma, V., Sharma, A., ... & Chung, Y. S. (2024). Artificial intelligence and IoT driven technologies for environmental pollution monitoring and management. *Frontiers in Environmental Science*, 12, 1336088.

[25] Rahmanović, A., Hakić, E., Saračević, M., & Međedović, E. (2023). Application and Development of Embedded Systems with IoT Components: Aspect of Safety and Reliability. *SAR Journal* (2619-9955), 6(4), 236-246.

[26] Salah, W. A., & Jamal, K. (2017). PIR-based motion detection systems: Optimization of sensing angle and delay characteristics. Electrical and Electronic Engineering Research, 5(3), 121–128.

[27] Shahidehpour, M., Li, Z., & Ganji, M. (2018). Smart cities for a sustainable urbanization: Illuminating the need for establishing smart urban infrastructures. *IEEE Electrification Magazine*, 6(2), 16-33.

[28] Suharto, A. (2025). Design and Build IoT Smart Home System with Blocks Based App Inventor Programming. *Jurnal Penelitian Pendidikan* IPA, 11(7), 1016-1026.

[29] Trivedi, P. (2016). PWM-controlled dimming algorithms for MOSFET-based LED drivers. International Journal of Power Electronics, 14(2), 92–101.

[30] Xu, B., Hussain, B., Wang, Y., Cheng, H. C., & Yue, C. P. (2022). Smart home control system using vlc and bluetooth enabled ac light bulb for 3d indoor localization with centimeter-level precision. *Sensors*, 22(21), 8181.

[31] Vandana, S. S., Abinandhan, K., & Ramesh, S. R. (2022, June).

Implementation of smart home using finite state machine model. In 2022 7th International Conference on Communication and Electronics Systems (ICCES) (pp. 45-49). IEEE.

[32] Visan, I., & Diaconu, E. M. (2022). Home automation system using ESP8266 microcontroller and Blynk application. *The Scientific Bulletin of Electrical Engineering Faculty*, 21(2), 59-62.

[33] Yavuz, E. (2024). Low-Cost Embedded System Design for Smart Home Automation Using RF Technology. *Applied Computer Systems*, 29(2), 1-13.