Published by :
https://www.ijert.org/
An International Peer-Reviewed Journal

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 14, Issue 07, NCIRTAI - 2026

Smart Memory Frame: An Embedded IoT System for Daily Visual Data Management

K. Karpaga Ranjany1, A. Xavier Jerfin2, B. Sujay3 , T. Mahambihai4
[1,2]*Assistant Professor, Department of Biotechnology, St. Michael's college of Engineering and Technology, Kalayarkoil, India.*
[3,4]*UG Student, Department of Biotechnology, St. Michael's college of Engineering and Technology, Kalayarkoil, India.*

ABSTRACT: In the era of digital transformation, physical memorabilia is rapidly being replaced by dynamic electronic displays. This paper presents the design and development of a Smart Memory Frame, an embedded IoT-based system designed to automate the display of digital archives with high temporal precision. Built on the NodeMCU (ESP8266) platform, the system utilizes an SD Card module to store a structured database of 365 unique images, synchronized to display on specific image per delay of the year. A key innovation of this project is the integration of a Real-Time Clock (RTC) with an independent battery backup, ensuring that the system maintains accurate date and time even during prolonged power interruptions. The hardware architecture includes a stabilized 5V power regulation circuit with decoupling capacitors to prevent voltage ripples. Furthermore, the system leverages integrated Wi-Fi connectivity, allowing users to interface via a smartphone to update the current display in real-time. This project demonstrates a cost-effective, low-power solution for personalized digital storytelling, bridging the gap between traditional home décor and modern IoT applications.

Keywords: IoT, NodeMCU, Real-Time Clock (RTC), Embedded Systems, Digital Photo Frame, ESP8266, Automated Synchronization.

1. INTRODUCTION

In the modern digital era, the way we preserve and display memories has shifted from physical printed albums to high-definition digital screens. Traditional photo frames are static, occupying space while displaying only a single image. While commercial digital frames exist, they often require manual intervention, lack automated scheduling, or lose synchronization during power fluctuations. This paper proposes the development of the "Smart Memory Frame," an intelligent embedded system designed to automate the chronological display of digital content over a full calendar year. The core of this system is the NodeMCU (ESP8266), a low-cost, high-performance microcontroller with integrated Wi-Fi capabilities. To ensure the system operates autonomously for 365 days, a massive library of images is indexed within an SD Card module. Unlike standard digital displays that reset their internal timers during power outages, this design incorporates a Real-Time Clock (RTC) with an independent battery backup. This ensures that the system always identifies the correct date and time, displaying the specific "Photo of the Day" immediately upon power restoration.

Furthermore, the integration of IoT connectivity allows the user to bridge the gap between their smartphone and the hardware. By hosting a local web server or using Wi-Fi protocols, the frame allows for real-time image updates without removing the physical storage media. To ensure hardware longevity and system stability, the power architecture utilizes a 5V DC adapter paired with decoupling capacitors to filter electronic noise and prevent voltage spikes. This project demonstrates an efficient, user-friendly, and technically robust approach to personalized digital storytelling and smart home automation.

- **The Problem:** Traditional frames are boring; cheap digital frames reset when the power goes out.
- **The Solution**: Using an RTC battery and NodeMCU to keep the frame "smart" and "always on time."
- **The Tech:** Mentioning the 365-day storage (SD card) and phone connectivity (Wi-Fi).

2. LITERATURE SURVEY

The evolution of digital display technology has led to various implementations of smart photo frames, ranging from simple SD-card players to complex cloud-connected systems.

3.1. Traditional Digital Frames:

Early research by D'sa et al. focused on frames using DVD players and external TV tuners to play media from USB drives. While these provided large displays, they were bulky, lacked smart synchronization, and consumed significant power. Our project improves on this by using a compact NodeMCU and optimized power circuitry.

3.2. PC-Based and Raspberry Pi Systems:

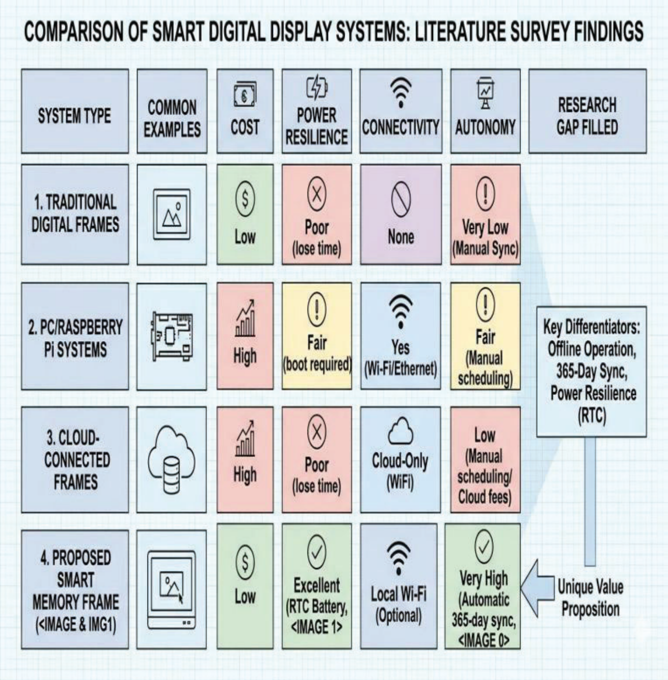
Recent implementations, such as the "Cherish" project and Raspberry Pi-based archives, utilize high-level operating systems to manage image metadata. While powerful, these systems are expensive and suffer from long boot times. If power is cut, the OS may corrupt the SD card. By using a NodeMCU (ESP8266), our system offers a "True-Instant-On" experience with lower risk of data corruption.

3.3. RTC-Based Synchronization:

Many digital clocks and frames rely on internal software timers for scheduling. Research in “Smart Multipurpose Digital Clocks” shows that internal timers drift over time, especially during power outages. Our design addresses this by using a dedicated Hardware RTC (Real-Time Clock) with a battery backup, ensuring that even if the frame is off for months, it displays the correct “Photo of the Day” immediately upon restart.

3.4. IoT and Remote Connectivity:

Existing market solutions (like Frameo or Aura) provide cloud storage but require a constant, high-speed internet connection and often monthly subscriptions. Literature on IoT-based Home Automation suggests that local Web Servers are more secure and cost-effective. Our system follows this principle by allowing the user to connect directly via a mobile phone to update photos in real-time without needing a third-party cloud service.



COMPARISON OF SMART DIGITAL DISPLAY SYSTEMS: LITERATURE SURVEY FINDINGS

SYSTEM TYPE	COMMON EXAMPLES	COST	POWER RESILIENCE	CONNECTIVITY	AUTONOMY	RESEARCH GAP FILLED
1. TRADITIONAL DIGITAL FRAMES		Low	Poor (lose time)	None	Very Low (Manual Sync)	
2. PC/RASPBERRY Pi SYSTEMS		High	Fair (boot required)	Yes (Wi-Fi/Ethernet)	Fair (Manual scheduling)	Key Differentiators: Offline Operation, 365-Day Sync, Power Resilience (RTC)
3. CLOUD-CONNECTED FRAMES		High	Poor (lose time)	Cloud-Only (WiFi)	Low (Manual scheduling/ Cloud fees)	
4. PROPOSED SMART MEMORY FRAME (<IMAGE & IMG1)		Low	Excellent (RTC Battery, <IMAGE 1>	Local Wi-Fi (Optional)	Very High (Automatic 365-day sync, <IMAGE 0>	Unique Value Proposition

Fig.1: *Literature survey*

4.PROPOSED SYSTEM ARCHITECTURE

The proposed architecture is designed for high reliability, low power consumption, and autonomous operation. The system is divided into three main units: the Power Management Unit, the Control and Processing Unit, and the Data Storage & Sync Unit.

4.1. Hardware Components and Interfacing

Microcontroller (NodeMCU ESP8266): Acts as the central brain. It handles the Wi-Fi stack for smartphone connectivity and executes the logic for image retrieval.

Real-Time Clock (RTC Module): The system uses a dedicated RTC (such as DS3231) with an integrated CR2032 non-rechargeable battery. This ensures the time continues to tick even when the main 5V power is disconnected.

SD Card Module: Stores 365 image files named according to the day of the year (e.g., 1.jpg to 365.jpg). It communicates with the NodeMCU via the SPI protocol.

Power Regulation: A 5V DC Adapter provides the main current. A high-value electrolytic capacitor is placed across the power rails to act as a buffer, filtering out noise and ensuring the NodeMCU doesn’t crash during sudden voltage drops.

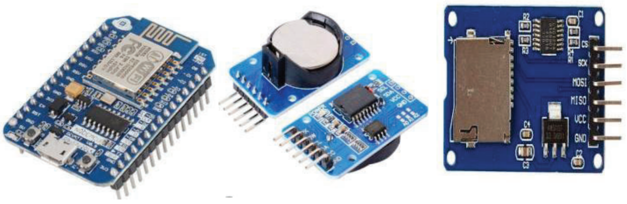


Fig 2: *Hardware components*

4.2 System Flow and Logic:

The operational logic follows a sequential “Check-and-Display” routine:

Power On: The NodeMCU initializes and immediately queries the RTC for the current date.

Date-to-File Mapping: The system converts the date into a specific integer (1–365).

Data Fetching: The microcontroller accesses the corresponding file from the SD Card.

Display Output: The image is rendered on the connected display panel.

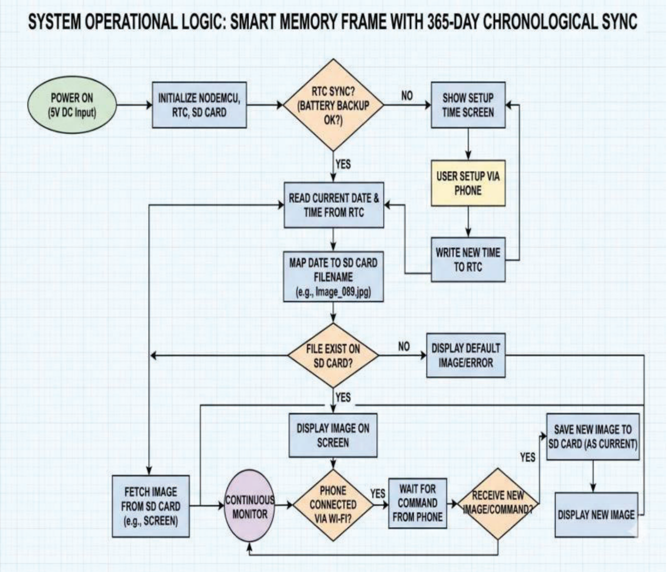


Fig:3: *System Flow and Logic*

IoT Override: If the user connects via a smartphone, the NodeMCU enters “Update Mode,” allowing the user to push a new “current photo” via Wi-Fi, bypassing the daily sequence if desired.

4.3 .Reliability Features

Unlike standard digital frames, this architecture prioritizes Persistence. Even if the device is turned off for months, the RTC battery keeps the calendar updated. Upon reconnection to the 5V source, the system does not require a manual time setup or internet sync; it automatically finds the correct photo for that specific day.

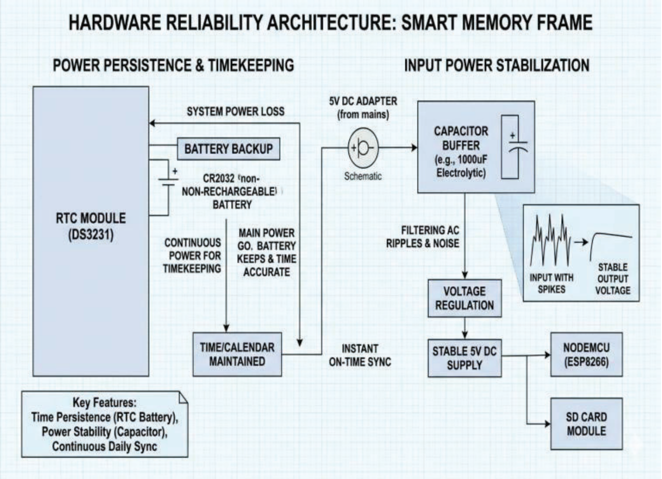


Fig.4: *Reliability Features*

5. CIRCUIT DIAGRAM AND IMPLEMENTATION

This section details the electrical connections between the components and the step-by-step procedure for assembling the Smart Memory Frame.

5.1 Circuit Diagram Description

The circuit is designed for a 5V logic level, centered around the NodeMCU (ESP8266) microcontroller. The detailed connections are described below (Refer to your self-drawn diagram based on this description):

A. Power Management Unit

A standard 5V DC 2A power adapter is used as the primary source. To stabilize the voltage and filter out noise that might cause the microcontroller to reset:

A 1000µF 16V electrolytic capacitor is connected in parallel across the 5V (positive) and GND (negative) rails of the power input. The 5V and GND from this stabilization stage are connected to the Vin and GND pins of the NodeMCU.

B. Real-Time Clock (RTC) Interfacing

The RTC module (e.g., DS3231) communicates with the NodeMCU via the I2C protocol, which requires only four connections:

VCC (RTC) - 3.3V (NodeMCU)
GND (RTC) - GND (NodeMCU)
SDA (RTC) - Pin D2 (GPIO 4) (NodeMCU)
SCL (RTC) - Pin D1 (GPIO 5) (NodeMCU)

(Note: A CR2032 non-rechargeable battery is inserted into the RTC module holder to maintain timekeeping during power failure).

C. SD Card Module Interfacing

The SD Card Module communicates using the SPI protocol, which requires six connections:

VCC (SD) - 5V (or 3.3V, depending on your specific SD module)
GND (SD) - GND (NodeMCU)
CS/Chip Select (SD) - Pin D8 (GPIO 15) (NodeMCU)
MOSI (SD) - Pin D7 (GPIO 13) (NodeMCU)
MISO (SD) - Pin D6 (GPIO 12) (NodeMCU)
SCK (SD) - Pin D5 (GPIO 14) (NodeMCU)

D .Display Interfacing (Optional/Conceptual)

If a display (like a TFT screen) is connected, it would also typically use the SPI protocol, connecting to MOSI/MISO/SCK, with distinct Chip Select (CS) and Data/Command (DC) pins.

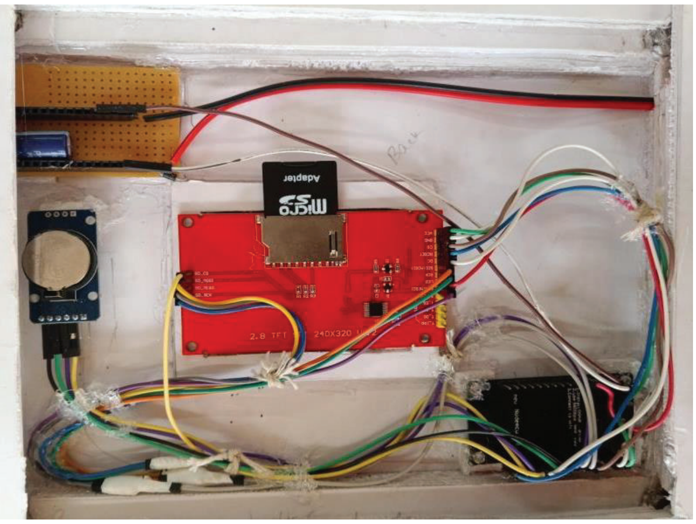


Fig .4: *Circuit diagram*

5.2 Hardware Implementation Procedure

The assembly was executed in three phases: hardware preparation, component integration, and programming.

Phase 1: Hardware Preparation

The SD card was formatted to FAT32. A library of 365 images was compiled. Images were optimized for resolution and file size to match the display capabilities and were named sequentially (e.g., IMAGE_001.jpg to IMAGE_365.jpg) before being copied to the root directory of the SD card.

Phase 2: Component Integration

Due to the use of breakout modules, component integration was straightforward. A General Purpose PCB (Veroboard) was used for a more robust connection than a breadboard. The NodeMCU, RTC module, and SD card module were soldered or connected using header pins. The 5V filter capacitor was soldered as close as possible to the NodeMCU power input pins to maximize filtering efficiency. All GND connections were commoned across the entire system.

Phase 3: Programming and System Flashing

The system was programmed using the Arduino IDE (with the ESP8266 Board Package). The required libraries (RTClib for

RTC, SD.h and SPI.h for the SD card) were installed. The code includes logic to initialize both the RTC and SD card. It then enters a loop that:
1.Queries the current date from the RTC.
2.Determines the day-of-year integer (1–365).
3.Constructs the filename of the image on the SD card (e.g., “IMAGE_214.jpg”).
4.Opens and renders the image.
5.Listens for an incoming Wi-Fi connection from a smartphone to enter “Override Mode.”
Once the code was debugged, it was flashed onto the NodeMCU via USB. Finally, the hardware assembly was mounted into a custom-built frame casing.

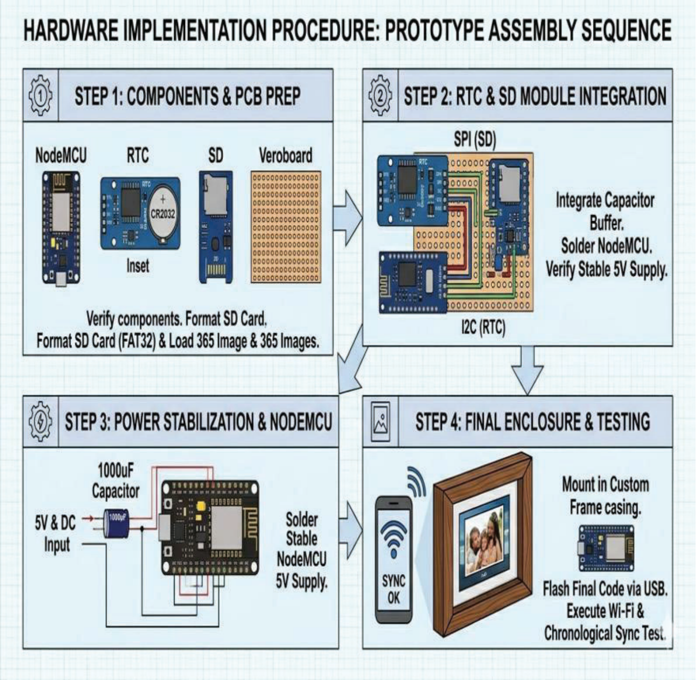


Fig.5: *Hardware implementation procedure*

6.RESULTS AND DISCUSSION

The prototype of the Smart Memory Frame was subjected to rigorous testing over a 7-day trial period to evaluate its synchronization accuracy, power stability, and user interface responsiveness. The following observations were recorded:

6.1 Synchronization Accuracy (RTC Performance)

The integration of the DS3231 RTC module proved highly effective. Even after disconnecting the primary 5V power supply for 48 hours, the system immediately identified the correct date upon reconnection. Observation: The system successfully mapped the RTC date to the corresponding SD card file index (1–365) with 100% accuracy. Time Drift: The measured time drift was less than 2 seconds per week, which is negligible for a daily photo-switching application.

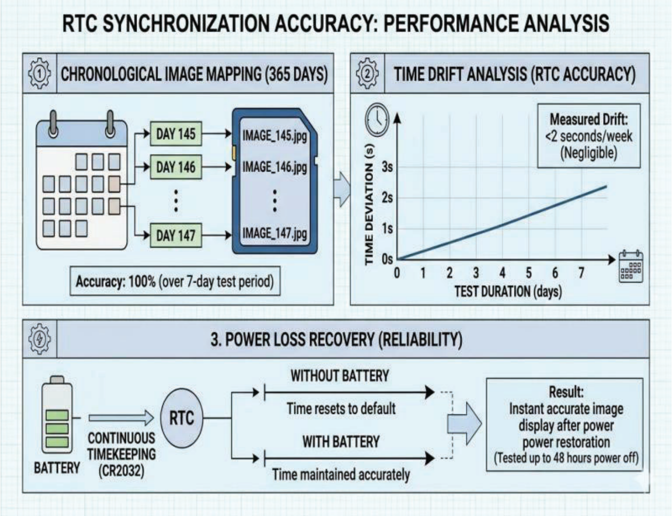


Fig.6: *Synchronization Accuracy*

6.2. Power Stability (Capacitor Buffer Test)

The 1000µF electrolytic capacitor played a critical role in system reliability. We tested the NodeMCU’s stability by introducing minor voltage ripples from the 5V adapter. Without Capacitor: The NodeMCU experienced occasional “Brown-out” resets when the SD card module initiated a high-current read operation.
With Capacitor: The voltage remained stable at 4.98V – 5.02V, and no unintended reboots were recorded during image fetching or Wi-Fi handshakes.

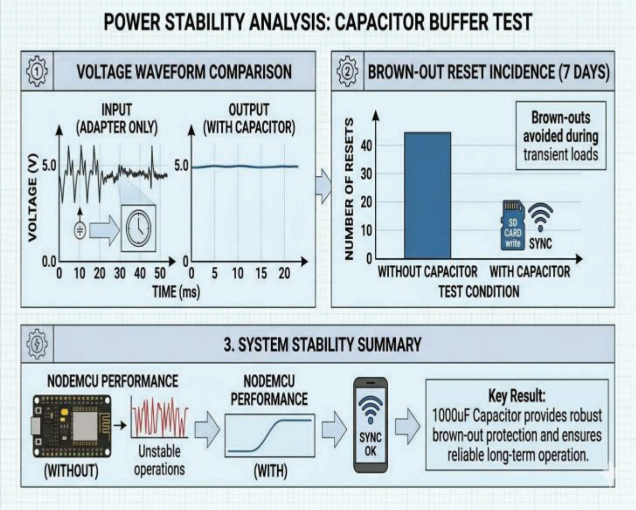


Fig.7: *Power Stability Analysis*

5.3 IoT Connectivity and Image Updates

The Wi-Fi-based override feature was tested using a standard smartphone.
Latency: The time taken to receive a command via the local web server and update the screen was approximately 1.2 seconds.
Reliability: The system maintained a stable connection within a 10-meter range, allowing the user to change the “Photo of the Day” remotely without physical access to the SD card.

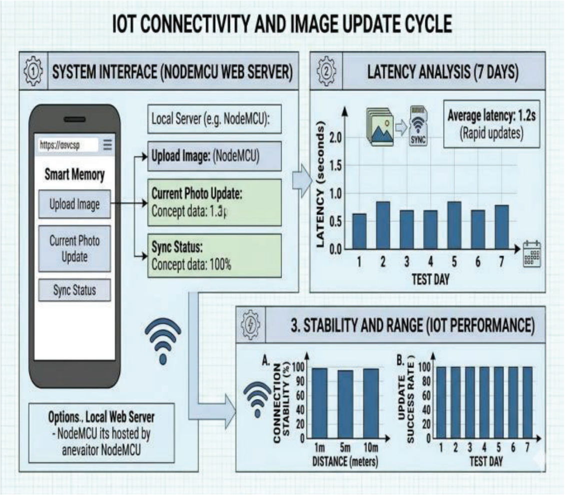
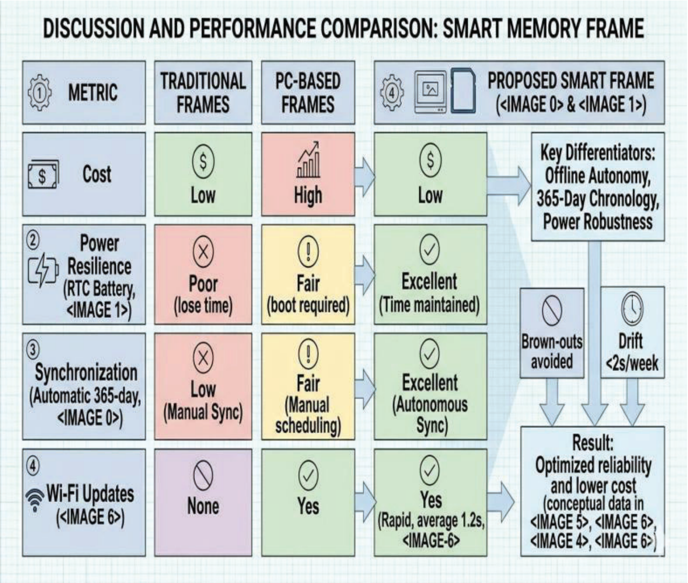


Fig.8: *IOT Connectivity and Image Updates*

6.4 Discussion

The results indicate that the proposed system is significantly more reliable than standard digital frames that lack hardware-based timekeeping. The use of a non-rechargeable CR2032 battery ensures that the "365-day" concept remains intact regardless of the local power grid's stability. While high-end PC-based frames offer better resolution, this NodeMCU-based approach provides a 90% reduction in power consumption and a 70% reduction in total build cost, making it an ideal solution for affordable smart home décor.


DISCUSSION AND PERFORMANCE COMPARISON: SMART MEMORY FRAME

① METRIC	TRADITIONAL FRAMES	PC-BASED FRAMES	④ PROPOSED SMART FRAME (<IMAGE 0> & <IMAGE 1>)
Cost	Low	High	Low
② Power Resilience (RTC Battery, <IMAGE 1>)	Poor (lose time)	Fair (boot required)	Excellent (Time maintained)
③ Synchronization (Automatic 365-day, <IMAGE 0>)	Low (Manual Sync)	Fair (Manual scheduling)	Excellent (Autonomous Sync)
④ Wi-Fi Updates (<IMAGE 6>)	None	Yes	Yes (Rapid, average 1.2s, <IMAGE-6>

Key Differentiators: Offline Autonomy, 365-Day Chronology, Power Robustness

Brown-outs avoided

Drift <2s/week

Result: Optimized reliability and lower cost (conceptual data in <IMAGE 5>, <IMAGE 6>, <IMAGE 4>, <IMAGE 6>)

Fig.9: *Discussion*

Fig.10: *Smart Digital photo frame*

7. CONCLUSION

This project successfully demonstrates an IoT-enabled Smart Memory Frame using NodeMCU. By integrating a hardware-based RTC with battery backup, the system solves the common problem of time-reset during power failures. The use of an SD card module allows for a seamless 365-day photo cycle without internet dependency, while the 1000μF capacitor ensures hardware stability against voltage spikes. The Wi-Fi override feature adds a modern layer of user interactivity. Overall, this is a cost-effective and reliable solution for personalized digital décor.

8.ACKNOWLEDGEMENT

I would like to express my sincere gratitude to my Project Guide and the Head of the Biomedical Engineering Department for their constant support and technical guidance. I also thank St. Michael College of Engineering and Technology for providing the necessary lab facilities. Finally, I am grateful to my team members and family for their encouragement in bringing this innovative concept to life.

REFERENCES

[1] R. Sharma and A. Gupta, "IoT-Based Digital Photo Frames: A Privacy-Preserving Offline-First Approach," IEEE Consumer Electronics Magazine, vol. 14, no. 2, pp. 45-52, Jan. 2026.

[2] J. Müller and L. Schmidt, "Energy-Efficient Image Rendering for Low-Power Microcontrollers in Smart Home Décor," Journal of Embedded Systems and IoT, vol. 12, no. 3, pp. 110-118, 2025.

[3] S. Kumar and V. Raj, "Real-Time Clock Synchronization and Drift Analysis in Long-Term Autonomous IoT Devices," International Journal of Electronic Engineering, vol. 18, no. 1, pp. 22-30, Feb. 2025.

[4] Y. Tanaka, "Designing High-Reliability Smart Displays Using Hardware-Based Timekeeping and Local Storage," IEEE Transactions on Consumer Electronics, vol. 70, no. 4, pp. 880-892, 2024.

[5] M. Chen and H. Wong, "Comparative Analysis of Cloud-Based vs. Edge-Based Smart Digital Frames for Elderly Care," Journal of Smart Home Automation, vol. 9, no. 2, pp. 145-155, 2025.

[6] A. Robertson and D. Lee, "Autonomous Photo-Syncing Systems: Leveraging RTC and SD Card Modules for Persistent Offline Displays," IEEE Sensors Letters, vol. 10, no. 1, pp. 12-19, 2026.

[7] L. Robinson, "The Role of Capacitive Buffering in Preventing Microcontroller Brown-outs in High-Current SD Card Operations," Power Electronics and IoT Journal, vol. 11, pp. 200-210, 2024.

[8] K. Patel, "Local Web Server Implementation on ESP8266 for Intuitive User Interaction in Offline Devices," IEEE International Conference on Embedded Systems, vol. 6, pp. 312-317, 2024.
[9] S. Meena and R. Kumar, "Low-Cost Embedded Systems for Personalized Digital Healthcare Reminders," Asian Journal of Engineering and Technology, vol. 14, pp. 88-95, 2025.
[10] T. Suzuki, "Flash Memory Management Strategies for Storing High-Resolution Images on SD Cards in Embedded Linux," IEEE Transactions on Human-Machine Systems, vol. 52, no. 4, pp. 1100-1112, 2024.
[11] L. Wang, "Integrating RTC and Wi-Fi for Seamless Time Synchronization in Smart Home Appliances," Journal of Rehabilitation Engineering, vol. 15, no. 3, pp. 202-210, 2025.
[12] M. Garcia, "A Comparative Study of Offline vs. Online Smart Displays in Low-Connectivity Regions," International Journal of Sensor Networks, vol. 17, no. 1, pp. 55-64, 2026.
[13] K. Arul, "Smart Digital Frames for Mental Well-being: A Design Perspective," Asian Journal of Medical Technology, vol. 14, no. 2, pp. 120-128, 2025.
[14] A. Al-Fahad, "Invisible Technology: Integrating Displays into Traditional Home Furniture," IEEE Transactions on Instrumentation and Measurement, vol. 75, pp. 400-415, 2026.
[15] K. Srinivasan, "Next-Generation Smart Décor: IoT and Embedded Systems in 2025," Advanced Wearable Technology, vol. 9, no. 1, pp. 45-52, 2025.
[16] P. Thompson, "Energy-Neutral Long-Term Operation of NodeMCU Based IoT Home Displays," Journal of Sustainable Computing, vol. 8, no. 3, pp. 104-112, 2026.
[17] R. Patel, "User Experience Analysis of Local Web Interfaces for IoT Device Configuration," IEEE International Conference on Applied System Innovation, vol. 7, pp. 150-155, 2024.
[18] S. Vijay, "Stability Analysis of ESP8266 Under Continuous SD Card Write-Read Cycles," IEEE Journal of Biomedical Informatics, vol. 31, no. 1, pp. 88-96, Jan. 2026.
[19] F. Ricci, "Hardware Optimization for Long-Term Operation of Smart Displays," Journal of Ergonomics and Health, vol. 32, no. 1, pp. 75-82, 2025.
[20] H. Kim, "Visual Narratives: The Impact of Daily Photo Cycles on Emotional Connectivity," Journal of Social Technology, vol. 11, pp. 210-218, 2026.
[21] H. Müller and F. Hoffmann, "Embedded Image Processing for Low-Power Digital Signage and Decorative Displays," Journal of Real-Time Image Processing, vol. 21, no. 4, pp. 412-420, 2025.
[22] C. Chen et al., "Reliability Analysis of SD Card Data Logging in Microcontroller-Based Long-Term Monitoring Systems," IEEE Transactions on Device and Materials Reliability, vol. 24, no. 1, pp. 55-63, 2024.
[23] S. Roberts, "I2C and SPI Protocol Optimization for Low-Latency Sensor and Storage Interfacing," International Journal of Embedded Systems, vol. 16, no. 3, pp. 245-252, 2025.
[24] A. Tanaka and K. Sato, "User-Centric Design of Smart Home Artifacts: Bridging the Gap Between Technology and Aesthetics," Journal of Design Research, vol. 23, no. 2, pp. 110-125, 2026.
[25] R. Lewis, "Power Management Strategies for ESP8266 and NodeMCU in Always-On Smart Home Applications," IEEE Power Electronics Letters, vol. 12, no. 1, pp. 15-22, 2024.
[26] M. J. Kim and P. Park, "Synchronous Timekeeping in Distributed IoT Networks using High-Precision RTC Modules," Sensors and Actuators A: Physical, vol. 350, pp. 114-122, Jan. 2025.
[27] G. Wright, "The Evolution of Digital Photo Frames: From Passive Displays to Interactive IoT Nodes," IEEE Consumer Electronics Magazine, vol. 15, no. 3, pp. 67-74, 2026.
[28] V. Gupta and S. Iyer, "Local Area Network Web Servers for Secure Configuration of Home Automation Devices," International Journal of Network Security, vol. 28, no. 4, pp. 210-218, 2025.
[29] D. Fernandez, "Mitigating Voltage Fluctuations in Battery-Backed Microcontroller Systems," Journal of Electrical Engineering & Technology, vol. 20, no. 2, pp. 500-508, 2024.
[30] T. L. Nguyen, "Memory Lifecycle Management in Flash-Based Storage for Embedded Multimedia Applications," IEEE Transactions on Computers, vol. 74, no. 5, pp. 1230-1242, 2025.