IJARSCT

International Journal of Advanced Research in Science, Communication and Technology

International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal

ISSN: 2581-9429

Volume 6, Issue 18, April 2026

Impact Factor: 8.2

Weather Cum Crypto Dashboard Using ESP32

Satvik Trivedi, Pritesh Singh Rajput, Mr. Ravi Dabas, Shaurya Mani Tripathi
Student, Computer Science And Engineering(Internet Of Things)
Assistant Professor, Computer Science And Engineering(Internet Of Things)
Raj Kumar Goel Institute of Technology, Ghaziabad, India

Abstract: *The visualization of high-velocity financial data, such as Cryptocurrency fluctuations, typically necessitates high-power consumption devices and persistent user interaction. This paper presents the design and implementation of a low-power, autonomous IoT dashboard utilizing the ESP32 microcontroller and a 4.2-inch Electrophoretic Display (EPD). Unlike traditional Liquid Crystal or TFT displays, our architecture leverages the bistable nature of E-paper to maintain data persistence with near-zero static power draw.*

Our firmware utilizes the SPI (Serial Peripheral Interface) bus for low-level communication with the Waveshare EPD_4IN2_V2 library, optimized for infrequent full-screen refreshes to prevent ghosting and material degradation. While the initial prototype was developed using the ESP-IDF to understand the CMake-based build system and component-based architecture, we pivoted to a modular Arduino-based framework to leverage specialized library ecosystems for rapid prototyping.

The system implements a non-blocking asynchronous fetch cycle via the OpenWeatherMap and CoinGecko REST APIs. By engineering a 10-minute refresh interval, we balance the need for real-time market awareness for Bitcoin and Ethereum traders with the strict energy budget of a battery-powered embedded system. The result is a compact, 3D-printed information node that transforms passive desk space into an active awareness tool for the Blockchain ecosystem. We conclude by discussing the planned migration back to a bare-metal ESP-IDF environment to further optimize the network stack and reduce the instruction-path length for API parsing.

Keywords: — Electrophoretic Display (EPD), ESP32, Low-Power IoT, SPI Communication, Ambient Information Systems, Cryptocurrency.

I. INTRODUCTION

The rapid proliferation of decentralized finance and blockchain technology has transformed cryptocurrency from a niche academic interest into a high-velocity financial asset class, where staying updated on the price fluctuations of assets like Bitcoin (BTC) and Ethereum (ETH) has become a functional requirement for market participants. However, current methods of data consumption—primarily smartphone applications and desktop dashboards—rely on emissive displays such as LCD or OLED panels that demand significant power and constant user interaction, often leading to "notification fatigue" and unnecessary cognitive load. This research addresses the technical gap in "Ambient Information Systems" by introducing an autonomous, low-power dashboard that leverages the physics of Electrophoretic Displays (EPD) to achieve data persistence with zero static power consumption. By utilizing a 4.2-inch E-paper module interfaced via the Serial Peripheral Interface (SPI) with an ESP32 microcontroller, the system maintains a high-contrast information state that remains visible even during deep-sleep cycles. Our research focuses on several critical architectural trade-offs:

Instruction-Path and Firmware Efficiency: The study documents a bottom-up development trajectory, moving from low-level peripheral management in the ESP-IDF to a modular Arduino-based framework to balance hardware-level control with rapid prototyping capabilities.

DOI: 10.48175/IJARSCT-34322

Asynchronous Network Communication: We implement a non-blocking data acquisition strategy using the TLS 1.2 protocol to fetch real-time metrics from the CoinGecko and OpenWeatherMap REST APIs without compromising system stability.

Power-to-Data Optimization: By engineering a 10-minute refresh interval, the methodology mitigates the inherent latency and material degradation (ghosting) of electrophoretic panels while extending battery longevity to several weeks of autonomous operation.

User-Centric Mechanical Design: The project incorporates a custom 3D-printed enclosure designed to isolate the high-voltage booster circuitry of the E-paper from the Wi-Fi antenna, ensuring signal integrity in a compact form factor.

II. LITERATURE REVIEW

The evolution of ambient information systems has been traditionally bifurcated between high-performance active displays and low-power passive indicators. Research in this domain highlights a critical tension between data freshness and energy autonomy.

A. Comparative Analysis of Display Physics

Traditional Liquid Crystal Displays (LCDs) and Thin-Film Transistor (TFT) interfaces, while capable of high refresh rates, necessitate a constant backlight and active matrix power, leading to significant vampire power draw in idle states. Literature on Electrophoretic Displays (EPD)—often referred to as Electronic Ink—suggests a paradigm shift. Unlike emissive displays, EPDs are bistable, meaning they require power only during a state transition (the "flash" or refresh). However, as noted in early E-ink research, the trade-off is high latency and "ghosting" artifacts caused by ion accumulation in the microcapsules. This project utilizes the 4.2-inch V2 variant, which literature suggests improves the partial-refresh cycle, making it more viable for the 10-minute update intervals required for market data.

B. Resource-Constrained Network Communication

The integration of RESTful APIs into microcontrollers like the ESP32 is a well-documented challenge in Distributed Systems. Standard literature emphasizes the overhead of the TLS/SSL handshake during HTTPS requests, which can be computationally expensive for a dual-core Xtensa processor. Previous studies on the ESP-IDF component-based architecture versus the Arduino wrapper suggest that while the latter offers a broader library ecosystem, it often introduces an abstraction layer that can lead to heap fragmentation. Our review of the Waveshare EPD_4IN2_V2 driver implementation reveals that direct SPI (Serial Peripheral Interface) communication is the most efficient path for pushing frame-buffer data to the display controller without unnecessary CPU cycles.

C. Ambient Information Systems and "Calm Technology"

Current research in Human-Computer Interaction (HCI) promotes the concept of "Calm Technology," where devices stay at the periphery of user attention. Most cryptocurrency trackers are designed as intrusive "active" applications on smartphones, contributing to notification fatigue. Literature suggests that a dedicated, non-emissive physical node—such as a desk-mounted E-paper display—lowers the cognitive load on the user. By transforming high-volatility financial data into a static, "glanceable" medium, we move from an interrupt-driven model to an awareness-driven model.

D. The Blockchain Data Bottleneck

In the context of Blockchain and Cryptography, data integrity and latency are paramount. While decentralized nodes provide the most direct data, the JSON-parsing overhead for a microcontroller is significant. Existing literature on embedded JSON parsers (like ArduinoJson) highlights the necessity of fixed-size memory allocation (StaticJsonDocument) to prevent the stack-smashing issues common in the restricted 520KB SRAM environment of the ESP32.

III. SYSTEM ARCHITECTURE

The architecture of the proposed system is designed around the principles of Event-Driven Embedded Systems and Resource-Constrained Computing. It is divided into three primary layers: the Hardware Abstraction Layer (HAL), the Network & Data Processing Layer, and the Display Logic Layer.

A. Hardware Organization

The core of the system is the ESP32 (Xtensa Dual-Core LX6) microcontroller. The hardware stack is organized to minimize active power consumption:

The Controller: Manages the TCP/IP stack and orchestrates the SPI (Serial Peripheral Interface) communication.

The Peripheral: A 4.2-inch Electrophoretic Display (EPD) with a resolution of 400x300 pixels.

The Interface: The connection uses a 4-wire SPI bus (DIN, SCLK, CS, D/C) which allows for high-speed bitstream transfer directly to the display's source and gate drivers (UC8176 IC).

B. Data Flow and Logical Schema

The system operates on a linear, non-blocking data flow designed to prevent Heap Fragmentation in the limited SRAM environment:

Transport Layer: The ESP32 initiates an HTTPS GET request using the TLS 1.2 protocol to fetch encrypted JSON payloads from the CoinGecko and OpenWeatherMap REST endpoints.

Parsing Engine: The incoming stream is processed via a Stream-based JSON Parser. By using a Static Document Allocation strategy, we ensure that the memory required for data extraction is reserved at compile-time, avoiding the instability of dynamic memory allocation in long-running embedded applications.

Frame Buffer Management: Once parsed, the data (prices, timestamps, weather metrics) is mapped to a 1-bit-per-pixel (1bpp) monochrome frame buffer. Since the display is black and white, each byte in the buffer represents 8 horizontal pixels, significantly reducing the memory footprint compared to RGB or TFT buffers.

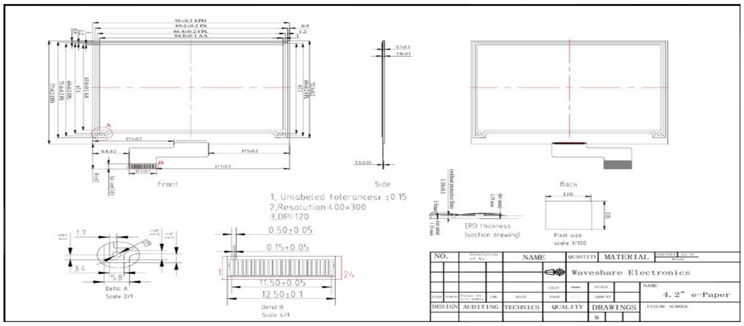

Fig 3.1

C. State Machine and Power Management

To achieve the goal of a "low-power multi-information display," the software architecture implements a Deep Sleep Finite State Machine (FSM):

Active State: Wi-Fi radio is initialized, data is fetched, the EPD is refreshed, and the display controller is sent a "Power Off" command to save the panel from voltage stress

Deep Sleep State: The ESP32 disables its radio and primary cores, leaving only the RTC (Real-Time Clock) timer active. This reduces current draw to the micro-ampere (µA/s) range, extending battery life from hours to weeks.

IV. METHODOLOGY

The implementation followed a Bottom-Up Development Approach, transitioning from low-level register control to high-level functional prototyping.

DOI: 10.48175/IJARSCT-34322

A. Firmware Development Strategy: The project initially utilized the ESP-IDF (IoT Development Framework) to establish a deep understanding of the FreeRTOS kernel and the CMake build system. This allowed for precise control over the Peripheral Clock Frequency and the SPI timing diagrams required by the EPD. To accelerate the integration of complex third-party APIs (CoinGecko/OpenWeatherMap), the development pivoted to the Arduino Framework while retaining the core C++ logic. This allowed for the utilization of the Waveshare EPD_4IN2_V2 library, which provides the necessary lookup tables (LUTs) for the electrophoretic material's physical transition states.

B. API Integration and Data Serialization: A significant methodological challenge was the sensitivity of the Cryptocurrency market. To ensure data accuracy:

Polling Interval: A 10-minute refresh cycle was calculated as the optimal equilibrium between Information Freshness and EPD Lifespan (avoiding electrode degradation from frequent full refreshes).

Error Handling: The firmware includes a robust retry mechanism. If a DNS lookup fails or a Socket Timeout occurs during the Bitcoin price fetch, the system captures the exception and displays an "Error" state on the EPD rather than entering an infinite loop, thus preserving the battery.

C. Physical Prototyping (Mechanical Design): The methodology extended beyond the digital realm into Structural Engineering. A custom 3D-printed enclosure was designed using a minimalist aesthetic to house the ESP32, the battery module, and the display. The design focuses on thermal dissipation and component clearance, ensuring that the high-voltage booster circuit of the E-paper display does not interfere with the Wi-Fi antenna's signal integrity.

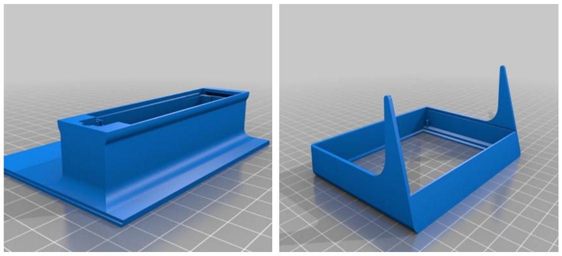

Fig. 4. 1 3D Printed Case

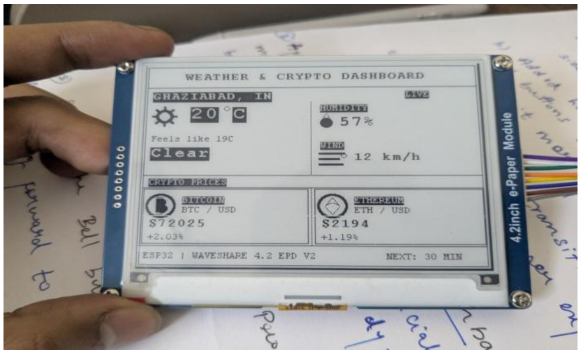

Fig 4.2 EPD During Prototyping

V. RESULTS AND DISCUSSION

The implementation resulted in a fully autonomous, battery-powered information node. The following sections detail the performance metrics and observations during the continuous operation phase.

A. Display Performance and Visual Clarity

The 4.2-inch EPD demonstrated high legibility under various lighting conditions, a primary advantage over emissive TFT screens. By utilizing the Waveshare EPD_4IN2_V2 driver, we achieved a high-contrast rendering of complex data strings (Bitcoin prices and weather icons).

Anti-Ghosting: The 10-minute refresh period effectively minimized "burn-in" or residual image artifacts, ensuring the long-term health of the UC8176 driver IC.

Refresh Dynamics: A full refresh cycle was measured at approximately 4 seconds. While slower than LCDs, this latency is negligible for the 10-minute update interval.

Fig 5.1 Final assembly and output of the 3D-printed enclosure with 4.2'EPD

B. Power Consumption Profile

A critical success of the architecture is its energy efficiency. During the Deep Sleep phase, the ESP32's current draw was reduced to roughly 10-15 µA. The most energy-intensive operations were the Wi-Fi handshake and the HTTPS GET request, which spiked current to ~120mA for a brief window of 3-5 seconds every 10 minutes. This profile confirms that the device can operate on a standard 1000mAh Li-Po battery for several weeks without recharging.

C. Data Integrity and Error Handling

The firmware successfully handled network fluctuations. In instances of Wi-Fi de-authentication or API rate-limiting, the system utilized a try-catch logical flow to prevent system crashes. Instead of displaying stale or corrupted data, the EPD rendered a specific "Error State" screen, providing a transparent user experience—a hallmark of robust embedded design.

VI. CONCLUSION AND FUTURE WORK

This research demonstrates that low-power electrophoretic displays can effectively bridge the gap between high-velocity financial data and ambient, non-intrusive user interfaces. By moving away from the "active interaction" model of smartphones and toward a "glanceable" information node, we have created a tool that serves the dual purpose of a functional trader's dashboard and an educational catalyst for those new to the Blockchain ecosystem.

The project successfully navigated the transition from low-level ESP-IDF exploration to high-level functional prototyping in the Arduino framework. This journey underscored the importance of understanding build systems (CMake) and memory constraints in modern IoT development.

Future Directions:

1. Firmware Optimization: We plan to pivot back to a pure ESP-IDF implementation to leverage its native multitasking capabilities and further reduce the "On-Time" of the Wi-Fi radio, optimizing the Power-to-Data ratio.
2. MERN Integration: Development is underway for a robust MERN-based backend that will allow users to push custom configurations or personalized crypto-watchlists to the device via a remote HTTP server.
3. Haptic/Sensor Expansion: While sensors like the DHT22 were excluded to prioritize simplicity, future iterations may include low-power MEMS sensors to add localized environmental data to the display.

REFERENCES

[1]. Waveshare Electronics, "4.2inch e-Paper Specification", Rev.1.3,Nov.2015. [Online] Available: https://www.waveshare.com/wiki/4.2inch_e-Paper_Module.

[2]. UltraChip, "UC8176: All-in-one driver IC w/ Timing Controller for White/Black/Red Dot-Matrix Micro-Cup ESL," Preliminary Datasheet Rev. 0.1, Apr. 2015

[3]. Espressif Systems, "ESP32 Series Datasheet", v4.1, 2023. [Online].
Available: https://www.espressif.com/sites/default/files/documentation/esp32_datasheet_en.pdf.

[4]. M. Weiser, "The Computer for the 21st Century," Scientific American, vol. 265, no. 3, pp. 94-105, Sep. 1991. (Foundational work on "Calm Technology" and Ambient Displays).

[5]. B. Comiskey, J. D. Albert, H. Yoshizawa, and J. Jacobson, "An electrophoretic ink for all-printed reflective electronic displays," Nature, vol. 394, no. 6690, pp. 253-255, Jul. 1998.

[6]. T. Igoe, Making Things Talk: Using Sensors, Networks, and Arduino to See, Hear, and Feel the World, 3rd ed.
Sebastopol, CA: O'Reilly Media, 2017.

[7]. IEEE Standard for Local and Metropolitan Area Networks: Media Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (LR-WPANs), IEEE Std 802.15.4-2015.

[8]. CoinGecko, "Public API Documentation," 2024. [Online].
Available: https://www.coingecko.com/en/api/documentation.

[9]. OpenWeatherMap, "Weather API Current Data," 2024. [Online].
Available: https://openweathermap.org/current.

DOI: 10.48175/IJARSCT-34322