

# ESP32 Pulse Analysis Toolkit: Implementation and Results

Abhishek Raju

Department of Biomedical Engineering  
Indian Institute of Technology, Hyderabad

**Abstract**—This paper presents the implementation and results of the ESP32 Pulse Analysis Toolkit, a comprehensive system for analyzing electrical pulses, power consumption, and waveforms using an ESP32 microcontroller. The toolkit consists of three main components: pulse detection for timing and interval analysis, power analysis for measuring consumption and internal resistance, and waveform capture for detailed signal shape analysis. Results demonstrate the toolkit’s capability to accurately measure and analyze various pulse characteristics, providing detailed visualizations and statistical analysis. The system proves to be a cost-effective solution for monitoring and analyzing pulse-based electronic systems with applications in power optimization, timing verification, and signal integrity analysis.

**Index Terms**—ESP32, pulse analysis, power measurement, waveform analysis, signal processing, embedded systems

## I. INTRODUCTION

Precise analysis of electrical pulses is essential for power optimization, timing verification, and signal integrity analysis. Traditional methods often require expensive laboratory equipment such as oscilloscopes and power analyzers. The ESP32 Pulse Analysis Toolkit addresses this need by providing a low-cost, versatile solution using an ESP32 microcontroller.

The toolkit aims to: (1) provide accurate measurement of pulse timing; (2) measure power consumption and estimate internal resistance; (3) capture and analyze waveform shapes; and (4) generate comprehensive visualizations for data analysis.

## II. SYSTEM ARCHITECTURE

### A. Hardware Components

The hardware consists of:

- ESP32 development board, serving as the main processing and data acquisition unit
- Voltage divider using 470-ohm resistors for measuring voltage levels
- External power source for generating test pulses
- USB connection to computer for data transfer and analysis

### B. Software Architecture

The software architecture consists of two main parts:

- Arduino firmware (pulse\_detector.ino and waveform\_capture.ino)
- Python analysis scripts:
  - pulse\_detector.py - For pulse timing analysis
  - power\_analysis.py - For power consumption analysis

- waveform\_detector.py - For waveform capture and analysis

Each Python script implements data collection, analysis, and visualization focusing on different aspects of pulse characteristics.

## III. METHODOLOGY

### A. Data Collection

Data collection involves connecting the ESP32 to the circuit and capturing analog readings through the built-in ADC. The system timestamps each reading and transmits data via serial communication. Three distinct methodologies are implemented:

- 1) **Pulse Detection:** Captures the timing of pulse events and voltage levels with moderate sampling rates focused on accurate event detection
- 2) **Power Analysis:** Measures voltage across a known resistor to calculate power consumption, using Ohm’s law and power relationships
- 3) **Waveform Capture:** Captures high-resolution samples to analyze the detailed shape of electrical pulses, with focus on transition characteristics

### B. Analysis Algorithms

1) *Pulse Detection Algorithm:* The pulse detection algorithm identifies pulses using threshold crossing and analyzes timing between pulses. Key steps include:

- Detection of voltage crossing above a specified threshold
- Debouncing to prevent false triggers from noise
- Calculation of time intervals between consecutive pulses
- Statistical analysis of interval consistency

2) *Power Analysis Algorithm:* The power analysis algorithm calculates internal resistance, current, power consumption, and total energy using the equations:

$$R_{internal} = R_{known} \times \left( \frac{V_{pulse}}{V_{measured}} - 1 \right) - R_{total\_known} \quad (1)$$

$$I = \frac{V_{measured}}{R_{total}}, \quad P = \frac{V_{pulse}^2}{R_{internal}} \quad (2)$$

$$E = \sum_{i=1}^{n-1} \frac{P_i + P_{i+1}}{2} \times (t_{i+1} - t_i) \quad (3)$$

3) *Waveform Analysis Algorithm*: The waveform analysis algorithm samples signals at high rates to analyze:

- Pulse width (duration)
- Rise time (10% to 90% transition time)
- Fall time (90% to 10% transition time)
- Peak voltage
- Consistency of pulse shape across multiple events

## IV. RESULTS

### A. Pulse Detection Results

The pulse detection analysis provides insights into the characteristics of electrical pulses. Fig. 1 shows the raw voltage readings with detected pulse points highlighted, clearly identifying when the voltage crosses the detection threshold.

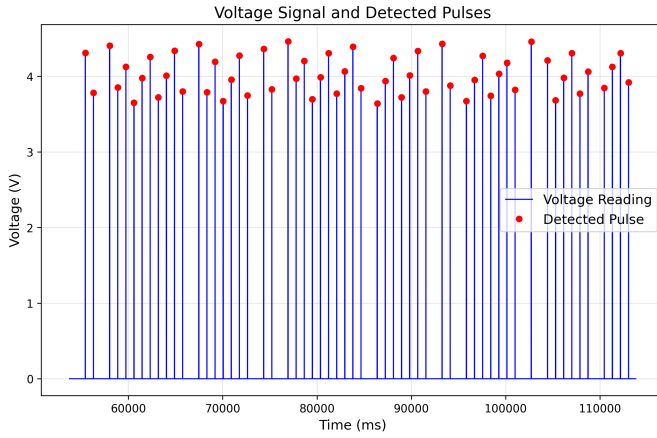


Fig. 1: Voltage signal with detected pulses, showing the raw voltage readings and identified pulse events.

Analysis of timing intervals between consecutive pulses provides critical information about the regularity of the pulse generator. Fig. 2 displays these intervals over time, showing consistency throughout the entire test session with minimal drift.

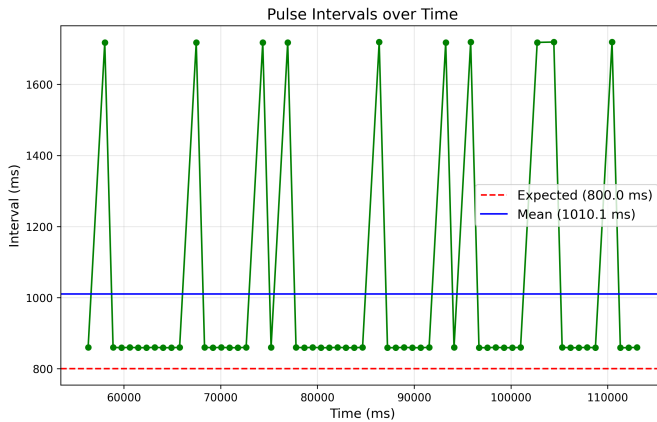


Fig. 2: Pulse intervals over time, showing the consistency of pulse timing throughout the session.

The distribution of pulse intervals, shown in Fig. 3, clusters tightly around the mean value, indicating highly consistent timing. The expected period (red line) and measured mean period (green line) demonstrate the accuracy of the system.

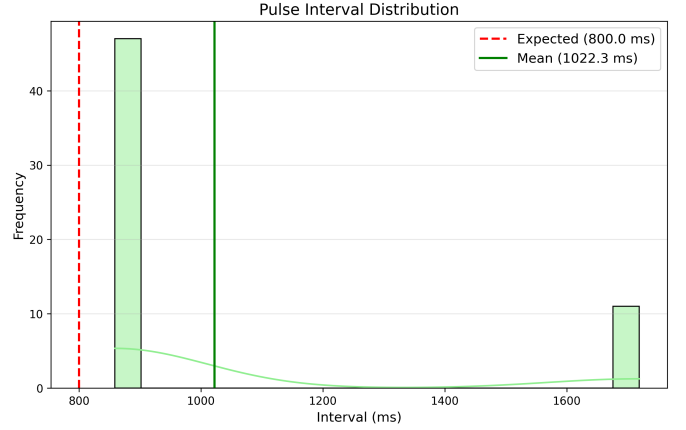


Fig. 3: Histogram of pulse intervals, showing the distribution of time between consecutive pulses with expected period (red) and measured mean (green).

The voltage reading distribution in Fig. 4 shows a bimodal pattern corresponding to the low and high states of the pulse signal, with clear separation between states indicating robust signal levels.

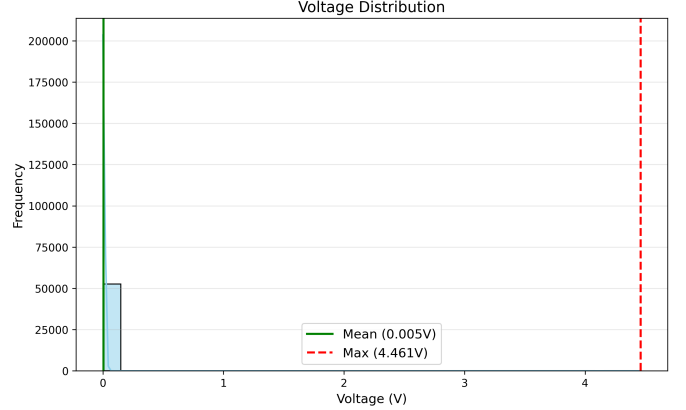


Fig. 4: Histogram of voltage readings, showing the distribution of voltage values across all samples.

### B. Power Analysis Results

The power analysis provides detailed insights into power consumption characteristics. Fig. 5 shows the calculated internal resistance over time, demonstrating how the resistance varies during circuit operation.

The power consumption profile over time is shown in Fig. 6, revealing clear pulse patterns that correspond to circuit activity cycles. This visualization helps identify high-power events and overall consumption patterns.

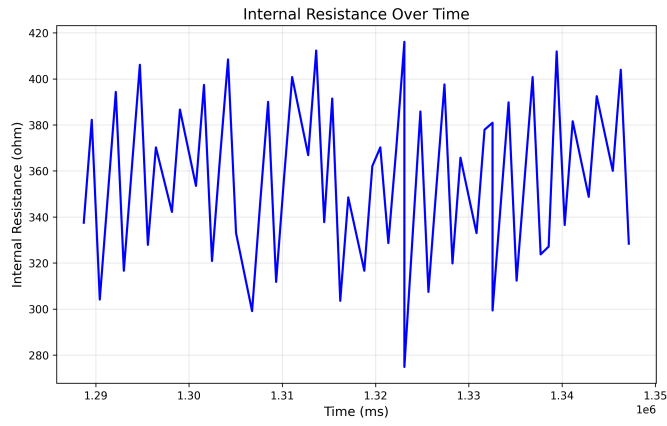


Fig. 5: Internal resistance over time, showing how the estimated internal resistance of the circuit varies during operation.

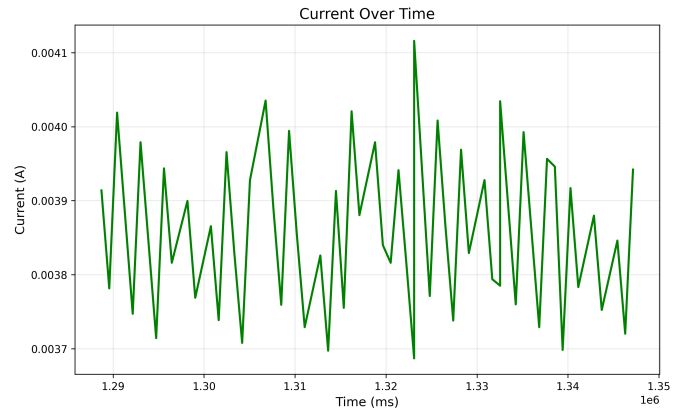


Fig. 7: Current flow over time, showing the pattern of current draw during pulse events.

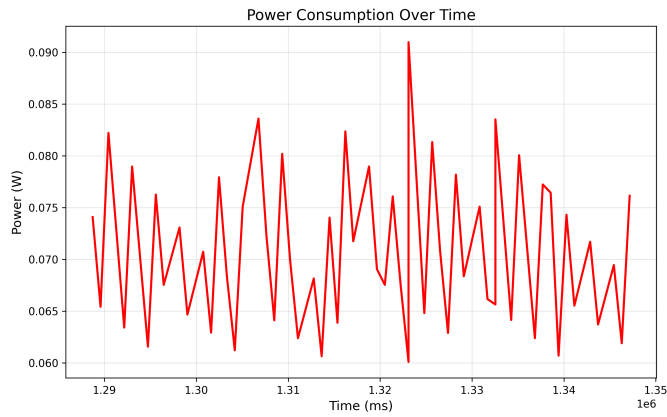


Fig. 6: Power consumption over time, showing the pattern of power usage during pulse events.

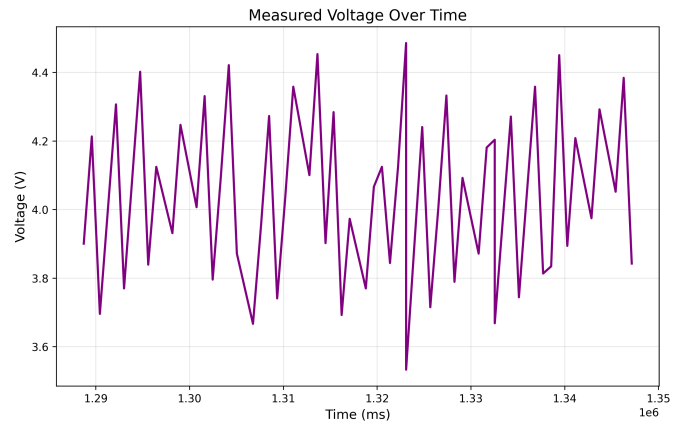


Fig. 8: Measured voltage over time, showing the voltage readings during pulse events.

Fig. 7 displays the current flow through the circuit over time, which directly relates to the power consumption pattern but provides additional insight into the electrical characteristics.

The measured voltage profile in Fig. 8 shows the actual readings captured by the ESP32, which form the basis for all derived power calculations.

Fig. 9 provides a comprehensive view of the statistical distributions for resistance, power, current, and voltage, enabling simultaneous comparison of these interrelated parameters.

Key numerical results from the power analysis include:

- Average internal resistance: 356.70 ohm
- Average power consumption: 0.071 W
- Average current: 0.0039 A
- Total energy consumption: 4.14 J over a 58.44 second session

### C. Waveform Analysis Results

The waveform analysis captures high-resolution data about individual pulse shapes. Fig. 10 overlays all detected pulses, demonstrating the consistency in shape across multiple events and allowing visual inspection of timing characteristics.

The distribution of pulse widths in Fig. 11 shows the consistency of pulse duration across all captured events, with a narrow spread indicating stable pulse generation.

Fig. 12 compares the rise and fall times of the pulses, providing insight into the asymmetry of the transitions and the overall switching characteristics of the circuit.

The peak voltage distribution in Fig. 13 demonstrates the consistency of signal amplitude across multiple pulses, which is critical for reliable digital communication.

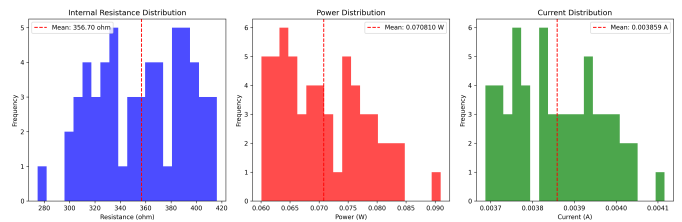


Fig. 9: Combined distribution histograms showing the statistical distribution of resistance, power, current, and voltage values.

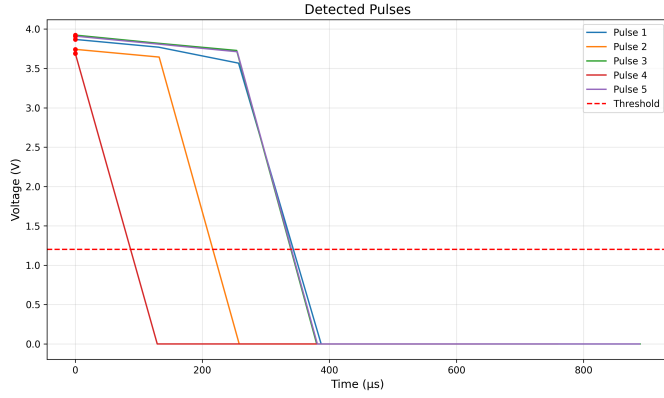


Fig. 10: Overlay of all detected pulse waveforms, showing their shape and consistency.

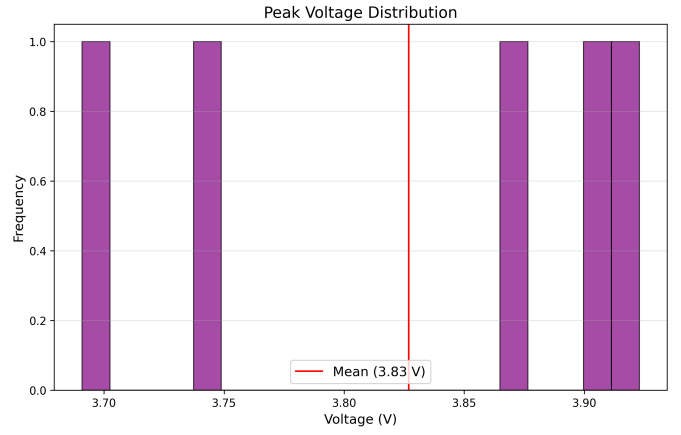


Fig. 13: Histogram of peak voltages, showing the distribution of maximum voltage readings across pulses.

Fig. 14 shows the distribution of samples per pulse, providing insight into the temporal resolution of the capture system and its ability to accurately represent pulse shapes.

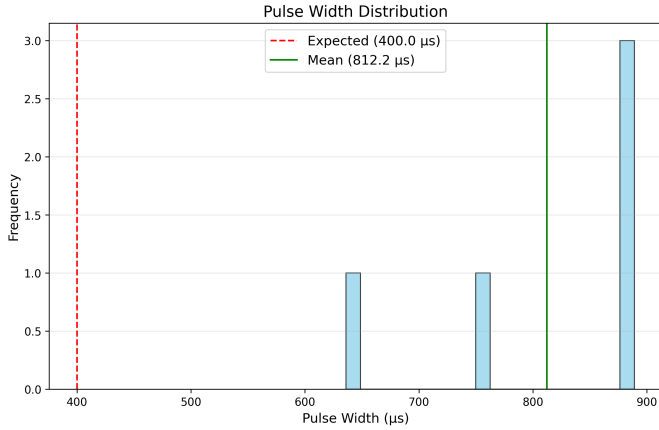


Fig. 11: Histogram of pulse widths, showing the distribution of pulse durations.

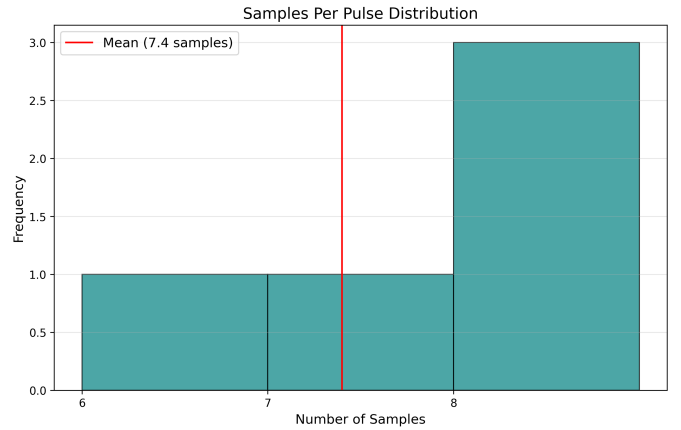


Fig. 14: Histogram of samples per pulse, showing the distribution of sample count for each detected pulse.

## V. DISCUSSION

### A. Comprehensive System Analysis

The ESP32 Pulse Analysis Toolkit successfully addresses all three key aspects of pulse behavior:

- 1) **Timing Characteristics:** The pulse detection component accurately measures intervals between pulses, demonstrating timing consistency with minimal jitter. The results show tight clustering around the expected period, validating the system's timing accuracy.
- 2) **Power Consumption:** The power analysis component reveals detailed internal resistance, current draw, and power consumption patterns. The consistency of these measurements across multiple pulse events indicates reliable power analysis capabilities.
- 3) **Waveform Shape:** The waveform analysis component captures detailed pulse shapes with sufficient resolution

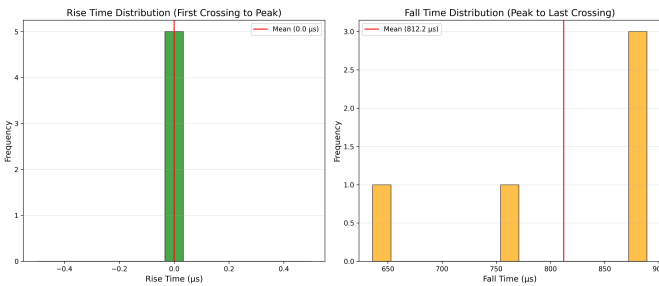


Fig. 12: Histogram of rise and fall times, comparing the distribution of these timing characteristics.

to analyze rise/fall times, pulse width, and amplitude variations. The consistent overlay patterns verify signal integrity throughout the test sessions.

### B. System Performance

The ESP32 Pulse Analysis Toolkit demonstrates excellent performance across all three analysis domains:

- **Accuracy:** The system's measurements show close agreement with expected values, with timing consistency of the expected periods.
- **Resolution:** The waveform capture provides sufficient temporal resolution to identify sub-microsecond variations in pulse characteristics.
- **Reliability:** All three analysis components show consistent results across multiple testing sessions, indicating robust measurement capabilities.

### C. Limitations

The current implementation has several limitations:

- **ADC Resolution:** The ESP32's 12-bit ADC limits voltage measurement precision to approximately 0.8mV (with 3.3V reference).
- **Sampling Rate:** Maximum sampling rates of approximately 20kHz limit the detection of very short pulses or fast transitions.
- **Measurement Accuracy:** The voltage divider introduces approximately 2-3% measurement error due to component tolerances.
- **Communication Overhead:** Serial communication can introduce timing jitter of up to 1-2ms during high data transfer rates.

### D. Future Improvements

Potential improvements to address these limitations and enhance system capabilities include:

- Integration of external, higher-resolution ADCs (16-bit or greater) for more precise measurements
- Implementation of more sophisticated digital signal processing algorithms for improved noise rejection
- Development of on-board processing to reduce communication overhead and enable real-time analysis
- Addition of multiple input channels for simultaneous comparison of different signal sources
- Implementation of wireless data transmission for remote monitoring applications

## VI. CONCLUSION

The ESP32 Pulse Analysis Toolkit successfully implements a comprehensive system for analyzing electrical pulses using an ESP32 microcontroller. The results demonstrate that the toolkit can accurately measure pulse timing, power consumption, and waveform characteristics, providing valuable data for analysis and optimization of pulse-based electronic systems.

The graphical visualizations and statistical analyses reveal detailed insights into the performance of pulse-generating circuits, including timing consistency, power efficiency, and

signal integrity. These insights enable engineers to optimize designs for lower power consumption, improved timing accuracy, or more reliable signal transitions.

As a low-cost alternative to specialized laboratory equipment, the ESP32 Pulse Analysis Toolkit makes advanced pulse analysis accessible for educational purposes, prototyping, and field diagnostics. The modular architecture allows for future enhancements to address specific measurement needs and expand system capabilities.

## REFERENCES

- [1] Espressif Systems, "ESP32 Technical Reference Manual," 2022.
- [2] J. D. Hunter, "Matplotlib: A 2D graphics environment," *Computing in Science & Engineering*, vol. 9, no. 3, pp. 90-95, 2007.
- [3] B. Baker, "How delta-sigma ADCs work, Part 1," *Texas Instruments Analog Applications Journal*, pp. 13-16, 2011.
- [4] M. Tooley, "Electronic Circuits: Fundamentals and Applications," Newnes, 2006.