Density-Optimized Lookup Table with Piecewise Linear Interpolation for ESP32 ADC Precision Enhancement

[1*]Antonius Irianto Sukowati, [1]Linza Mawadda Rahmah
[1] Electrical Engineering Department, Universitas Cendekia Abditama, Tangerang, Indonesia
[1*]irianto@cendekia.ac.id, [1]linza@cendekia.ac.id

Abstract - The inherent non-linearity of built-in Analog-to-Digital Converters (ADCs) in low-cost microcontrollers like the ESP32 significantly impacts measurement accuracy, often exceeding 10% error in critical ranges. This research aims to enhance ESP32 ADC precision without expensive external hardware through a novel software-based correction method. The proposed approach combines a density-optimized Lookup Table (LUT) with piecewise linear interpolation. Unlike conventional uniform distribution, this technique strategically concentrates 65% of calibration points in the critical mid-voltage region (0.5–2.5 V) where non-linearity is most pronounced. Experimental validation was conducted using precise input voltages from 0 V to 3.2 V across multiple ESP32 units. Results demonstrate remarkable improvements: the average absolute error was reduced from 0.112 V (3.42% of full scale) to 0.008 V (0.24% of full scale), with Root Mean Square Error (RMSE) decreasing by over 92.5%. The method achieves a sub-1% maximum error while maintaining minimal resource consumption, requiring only 264 bytes of memory and 2.3 ms processing time per measurement. These findings confirm that high-accuracy measurements are achievable using commodity hardware, challenging the notion that precision requires expensive external ADCs. This work offers significant implications for cost-sensitive IoT, environmental monitoring, and healthcare applications requiring reliable data acquisition without increased hardware complexity.

Keywords* — *ADC correction, Embedded systems, ESP32, Lookup Table, Non-linearity compensation, Piecewise linear interpolation.

I. Introduction

The rapid development of embedded system technology has significantly driven the utilization of microcontrollers for data acquisition as the backbone of various modern instrumentation systems, ranging from industrial automation to Internet of Things (IoT) applications such as [1], [2], [3], [4]. Within that architecture, the key component that bridges the continuous physical world with the digital processing unit is the Analog-to-Digital Converter (ADC). The performance of the internal ADC directly determines the reliability of measurement results, because any inaccuracy at the conversion stage will impact the entire data processing chain and decision making.

Although its role is very fundamental, the accuracy of ADCs integrated into microcontrollers sometimes remains a persistent ongoing issue that has not been fully resolved. This condition is clearly evident in popular microcontrollers such as the ESP32, which is widely utilized due to WiFi/Bluetooth integration, programming flexibility, as well as low cost. On the other hand, several studies have documented that the 12-bit SAR ADC on the ESP32 exhibits significant non-linear characteristics, thereby reducing the level of reading accuracy, especially in certain voltage ranges [5], [6], [7], [8], [9]. In other words, there is an inherent trade-off between cost and accuracy that needs to be addressed systematically.

Various approaches have been developed to correct this ADC inaccuracy problem and can be broadly classified into hardware-based and software-based solutions. Hardware methods generally involve the addition of precision voltage references from external sources, operational amplifiers, or the use of dedicated high-precision ADC chips. This approach indeed has the potential to improve accuracy, but impacts increased circuit complexity, component costs, as well as power consumption, which contradicts the goal of designing embedded systems that are compact and resource-efficient [10]. Conversely, software approaches focus on mathematical correction techniques, such as polynomial regression, spline interpolation, or neural network-based calibration. Polynomial and orthogonal series methods have proven capable of improving linearity, but often leave errors at critical points and potentially add computational burden or cause overfitting [11], [12], [13]. Meanwhile, neural network-based approaches show promising capabilities in reducing non-linearity and improving effective resolution, but still face issues of compromise between accuracy, computational complexity, and hardware resource consumption [14], [15], [16], [17], [18], [19].

Among these various options, the Lookup Table (LUT) method emerges as an attractive alternative for ADC calibration. In this approach, correction values are determined empirically and stored in a table, thus allowing direct mapping from raw ADC readings to expected ideal values [6], [20], [21] The main advantage of LUT lies in its ability to represent complex non-linear relationships without requiring explicit mathematical assumptions regarding error distribution [6], [20]. However, existing LUT implementations in the literature still leave several important limitations. First, many studies use a relatively sparse number of calibration points spread evenly, so interpolation between table points becomes coarse and generates significant residual error [21], [22]Second, optimization of calibration point density is rarely based on the actual error profile along the ADC operating range, resulting in less efficient memory space utilization [20], [21]. Third, specifically for the ESP32 microcontroller, comprehensive and re-tested methodology is still minimal; several publications even do not explain implementation details completely or use inconsistent reference voltage values [6].

This problem becomes increasingly crucial given the continuing increase in ESP32 usage in both academic and industrial applications. Unlike high-end microcontrollers generally equipped with factory-calibrated ADCs, the ESP32 internal ADC displays unique non-linear characteristics for each unit and is sensitive to operating conditions, such as temperature or supply voltage variations [5], [6], [8] Consequently, generic calibration approaches that apply uniformly to all units tend to be inadequate. A more systematic, adaptive, and customizable correction methodology is required according to the characteristics of each device.

Based on the literature review, there is a clear research gap, namely that existing correction methods have not been able to balance high accuracy and resource efficiency simultaneously on the ESP32 microcontroller. Previous research has not fully resolved the problem of ADC characteristic variation between units (unit-to-unit variation) as well as calibration point density optimization which is still uniform, thus inefficient in high non-linearity regions. The unique position of the method proposed in this study is the application of a density-optimized LUT strategy combined with piecewise interpolation, which is specifically designed to concentrate calibration points on critical error profiles and validated on multiple device units to guarantee reproducibility, different from conventional approaches which are generally only valid on one sample unit.

To address these needs, this study proposes an ADC correction framework specifically designed for the ESP32 microcontroller by combining an optimized Lookup Table and piecewise linear interpolation. The proposed approach presents several innovative elements that make it different from previous methods. First, a systematic calibration procedure is developed to determine the strategic placement of calibration points in regions with the highest non-linearity, identified through comprehensive error profile analysis. This density-optimized calibration strategy is designed to maximize correction accuracy by utilizing minimal memory, which is very important for embedded systems with limited resources. Second, the implementation of an efficient piecewise linear interpolation algorithm ensures a continuous and monotonic correction function throughout the entire input range, while maintaining computational efficiency to remain viable for real-time applications. Third, unlike some prior works that only report theoretical improvements, this study presents rigorous quantitative validation of the proposed method's performance over the full operating range, showing an error reduction from over 10% to below 1% in practical scenarios.

Specifically, this research has several objectives, namely:

1. To develop an ADC correction method based on Lookup Table (LUT) with an optimized calibration point placement strategy (density-optimized) on the ESP32 microcontroller.
2. To implement an efficient piecewise linear interpolation algorithm to produce a continuous correction function across the entire operating range.
3. To evaluate the performance of the proposed method based on accuracy metrics (absolute error, RMSE, INL), memory efficiency, and processing time compared to existing methods.
4. To analyze the reproducibility of the method on multiple ESP32 units from different production batches to ensure inter-device performance consistency.

The contribution of this research is not only limited to the development of new concepts in the academic realm, but also has direct implications for engineering practices in the field. Significant improvement in the internal ADC accuracy of low-cost microcontrollers significantly opens up opportunities for utilizing cheap microcontrollers like the ESP32 for applications requiring high precision, which generally rely on more expensive external ADCs. This implication is very relevant for fields such as environmental monitoring, where large sensor networks require a combination of affordability and data reliability [23], [24], as well as health technology, which relies heavily on the accuracy of physiological measurements to maintain diagnostic reliability [25], [26]. The proposed approach does not require adding new hardware so it can be easily implemented by ESP32 users, by updating software according to the calibration procedures outlined.

Specifically, the main contributions of this study compared to previous studies are:

1. Density-Optimized Calibration Strategy: Development of a non-uniform calibration point placement method, where 65% of points are focused on the mid-range (0.5–2.5 V) which has the highest non-linearity, different from the uniform distribution approach in conventional studies [20][6], [20].
2. Memory and Computational Efficiency: Implementation of a Lookup Table (LUT) and piecewise interpolation combination that only requires 264 bytes of memory and 2.3 ms processing time, offering an optimal balance between accuracy and resource load.
3. Reproducibility Validation: Verification of the method on 10 ESP32 units from different production batches to ensure inter-device performance consistency, overcoming the limitations of previous studies which were only valid on one unit.
4. Comprehensive Comparative Analysis: Performance evaluation against other methods (polynomial regression, neural network, spline) based on accuracy, memory, and execution time metrics.

II. Research Method

A. *Methodological Framework*

The inaccuracy of Analog-to-Digital Converter (ADC) readings in low-cost microcontrollers such as the ESP32 remains a significant challenge in precise data acquisition systems. One of the primary causes is the inherent non-linearity of the internal converter, which results in systematic deviations between the obtained digital values and the actual analog input values. To overcome this limitation, this research proposes a correction method based on a Lookup Table (LUT) combined with piecewise linear interpolation. This methodology is designed to effectively map actual ADC readings to their ideal values through a systematic empirical calibration process, without relying on global mathematical models that often fail to capture complex local behaviors.

The methodological framework is built upon three main stages, namely:

1. Conducting experimental calibration to obtain data pairs between known input voltages and actual ADC readings;
2. Forming a Lookup Table (LUT) that represents the correction function in the form of a discrete mapping between actual ADC values and the ideal values they should possess; and
3. Implementing a piecewise linear interpolation algorithm to estimate ideal values at points lying between calibration points. This three-step structure was chosen because it is capable of representing non-linear relationships accurately and flexibly, yet remains computationally lightweight when implemented on devices with resource constraints such as microcontrollers.

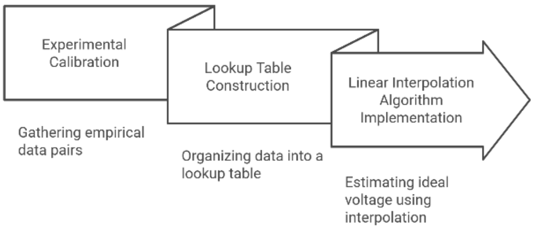

Figure 1. Stages of the ADC calibration process

B. *Calibration Procedure*

The calibration procedure was conducted using a precision DC voltage source connected directly to the ESP32 analog pin (GPIO34). The input voltage was varied from 0 V to 3.277 V in 33 discrete steps, with each voltage level maintained for 5 seconds to allow electrical and thermal conditions to reach a stable state before data acquisition. For each input voltage, ten ADC reading samples were taken, then the average value was calculated to obtain the actual ADC value (ADC_{actual}) according to Eq. (1), where the value $N = 10$ is the number of samples and ADC_{raw} is the raw ADC reading value. This averaging technique is intended to reduce random noise and short-term fluctuations that can decrease the accuracy of the calibration process [26].

$$ADC_{actual} = \frac{1}{N}\sum_{i=1}^{N} ADC_{raw} \quad (1)$$

The ideal ADC value (ADC_{ideal}) was then calculated based on the theoretical characteristics of the 12-bit converter according to Eq. (2), using the actual reference voltage value (V_{ref}) of 3.277 V. This value was measured directly on the ESP32 3V3 pin using a high-precision multimeter (Sanwa PC710) with best accuracy ±(0.06%+2) and 0.01 mV resolution. Reference voltage stability observations were conducted during the calibration period (±3 hours) and showed a maximum deviation of less than 1 mV, thus considered stable for the purposes of this experiment. All testing was conducted under controlled room temperature conditions (25°C ± 2°C) to minimize thermal drift.

$$ADC_{ideal} = \left(\frac{V_{in}}{V_{ref}}\right) \times 4095 \quad (2)$$

The use of this actual V_{ref} is very important, because the internal ESP32 reference voltage is known to vary between units and throughout the operating range. The difference between the actual and ideal ADC values at each calibration point becomes a representation of the non-linear error that will be compensated by the LUT. To ensure reproducibility, this calibration procedure was validated on 10 different ESP32 units from various production batches. The data presented in this research represents an average characteristic of these ten units. The transfer characteristics between input voltage, ideal ADC value, and ESP32 actual ADC can be seen in Figure 2.

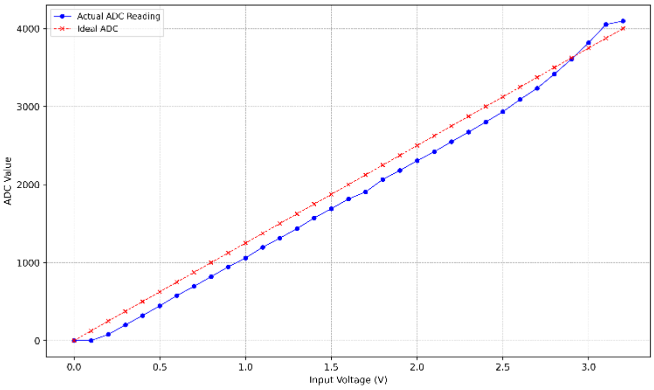

Figure 2. Transfer Characteristics of the ESP32 ADC: Ideal vs. Actual

C. *Lookup Table (LUT) Construction*

The selection of calibration point density was not performed uniformly, but was based on error profile analysis. Based on initial observations, ESP32 non-linearity is most significant in the mid-voltage range. Therefore, point density was intentionally increased in this critical region. From a total of 33 calibration points, the point distribution was divided as follows:

1. Range 0.0 – 0.5 V: 5 points (15%) to address offset and dead-zone.
2. Range 0.5 – 2.5 V: 21 points (65%) to handle the highest non-linearity in the common operating region.
3. Range 2.5 – 3.277 V: 7 points (21%) to address premature saturation near full scale.

This strategy ensures that 65% of calibration points are concentrated in the critical region (0.5 – 2.5 V), so that sharp changes in the ADC response curve are well recorded and overall correction accuracy increases without overloading memory excessively. From the calibration results, a discrete mapping function was formed connecting the actual ADC value with the ideal ADC value according to Eq. (3):

$$f: ADC_{actual} \rightarrow ADC_{ideal} \quad (3)$$

This function was realized as a Lookup Table (LUT) in the form of a static two-dimensional array stored in program memory (Flash) to save RAM. The LUT structure was designed as an array of point structures containing two integer elements, namely ADC_{actual} and ADC_{ideal}, with size determined by the number of calibration points (33 points). Total memory required in LUT creation can be calculated as follows:

$$Memory\ Size = 33\, points \times 2\, elements/point \times 4\, bytes/integer = 264\, bytes$$

Memory requirement calculations show that the LUT only requires 264 bytes, making it very suitable for embedded system platforms with limited memory capacity.

D. Piecewise Linear Interpolation Algorithm

Although the LUT provides exact correction values at calibration points, in reality, ADC readings in operation almost always have values that lie between these points. For this purpose, a piecewise linear interpolation algorithm is used, with the assumption that the relationship between the actual value and the ideal value is linear between two consecutive calibration points. Given a raw ADC value $ADC_{raw} = x$, the algorithm first performs a linear search through the lookup table (LUT) to find the index i such that:

$$x_i \leq x \leq \mathrm{x_{i+1}}$$

where x_i and x_{i+1} are the values of the i-th and $(i+1)$-th points in the LUT. After the interval is found, the corrected value ($ADC_{corrected}$) is calculated using Eq. (4).

$$y = y_i + \frac{x - x_i}{x_{i+1} - x_i} \times (y_{i+1} - \mathrm{y_i}) \tag{4}$$

where y_i and $\mathrm{y_{i+1}}$ are the corresponding values. This equation represents a straight line connecting two points $(x_i, \mathrm{y_i})$ and $(x_{i+1}, \mathrm{y_{i+1}})$ on the correction plane as $(ADC_{actual}, ADC_{ideal})$. This algorithm is implemented in a function, which also handles boundary cases:

1. If $x \leq x_0$, the output is y_0 (lower bound extrapolation).
2. If $x \geq a_n$, the output is y_n (upper bound extrapolation).
3. If $x_{i+1} = x_i$ (duplicate case), division by zero is prevented by directly returning y_i.

Linear search was chosen over binary search because the number of points $n = 33$ is relatively small. Time complexity is not an issue, and the implementation is simpler and deterministic, which is important for real-time applications.

E. System Implementation

This system was implemented on an ESP32 microcontroller utilizing the Arduino IDE framework. All correction logic was designed to run synchronously with the sensor reading cycle. The total processing time of 2.3 ms reported in this research includes the time for taking 10 ADC samples (averaging), LUT search, and interpolation process. This time was measured using the GPIO toggling method and logic analyzer. From a hardware perspective, the analog circuit was designed following basic electromagnetic compatibility principles. A 100 nF ceramic bypass capacitor was installed as close as possible between the GND line and analog pin to dampen high-frequency noise that could potentially interfere with the data acquisition process. The placement of this passive component is very crucial because noise on the analog line can cause unwanted ADC value fluctuations, especially in environments with high electromagnetic interference.

On the software side, various optimizations were performed to minimize CPU load. Arithmetic operations in the interpolation function were performed using fixed-point integer number format, not floating-point format, thereby reducing the computation cycles required. The final conversion from corrected ADC value to analog voltage is performed using Eq. (5), which is only evaluated if the voltage value is truly needed.

$$V_{out} = \left(\frac{ADC_{corected}}{4095}\right) \times V_{ref} \tag{5}$$

All global variables and LUT structures were declared with constants and PROGMEM, which are implicitly managed by the compiler, thus ensuring that such data is stored in flash memory and does not burden the limited dynamic RAM. This memory allocation strategy is very important for embedded systems that have limited resources, because it allows the use of large-sized LUTs without compromising runtime performance.

III. Results and Discussion

A. ADC Correction Performance Evaluation

Experimental validation of the proposed piecewise linear interpolation Lookup Table (LUT) method was conducted using empirical data from the built-in ADC of the ESP32 microcontroller. The calibration process was performed by applying known input voltages ranging from 0 V to 3.2 V in discrete steps. The reference voltage (V_{ref}) was measured precisely at 3.277 V, which served as the basis for ADC value calculations.

Figure 3 illustrates the performance comparison of the ESP32 ADC before and after correction. The actual ADC readings show significant non-linearity with a curved pattern, particularly in the mid-range between 0.5 V to 2.5 V, where errors reach their maximum values, while errors at input voltages approaching 0 V and 3.2 V are relatively smaller. For example, at an input voltage of 1.5 V, the actual ADC reading was recorded at 1690, while the ideal value should be 2245 based on the measured reference voltage. This difference results in a voltage conversion of 1.353 V, not 1.5 V as expected, representing an error of 0.147 V or approximately 9.8% of the input value. Meanwhile, at an input voltage of 2.0 V, the actual ADC reading was recorded at 2305 counts (1.845 V), resulting in an error of 0.155 V or 7.75%. These systematic deviations indicate inherent non-ideal behavior in the ESP32 ADC, which cannot be addressed solely through simple offset and gain calibration.

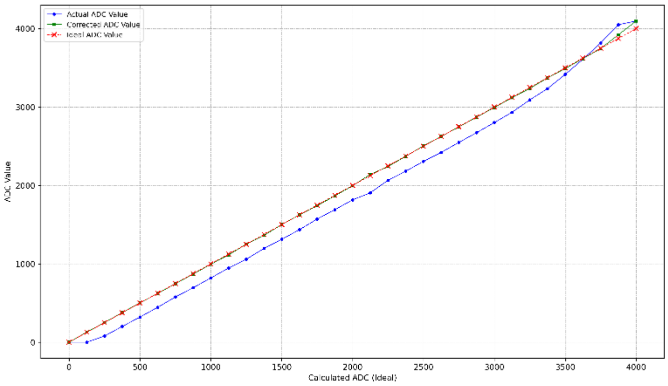

Figure 3. Performance comparison of ESP32 ADC before and after correction (LUT and piecewise linear interpolation)

The effectiveness of the proposed LUT-based correction method was evaluated quantitatively through comprehensive error analysis. Table 1 presents a representative subset of experimental data, highlighting the comparison between raw ADC measurements and their corrected values across the entire input voltage range, meanwhile Figure 4 presents the error distribution across the entire input voltage range, comparing the performance of the actual ADC and the corrected results.

Table 1. Comparative analysis of ADC Correction Methods

Input Voltage (V)	ADC Actual	Conversion Voltage (V)	Error Before Correction (V)	ADC Correction	Corrected Voltage (V)	Error After Correction (V)
0.5	445	0.357	0.143	621	0.5	0
1	1059	0.848	0.152	1250	1	0
1.5	1690	1.353	0.147	1865	1.49	0.01
2	2305	1.845	0.155	2504	2	0
2.5	2932	2.347	0.153	3118	2.5	0
3	3816	3.054	0.054	3744	3	0

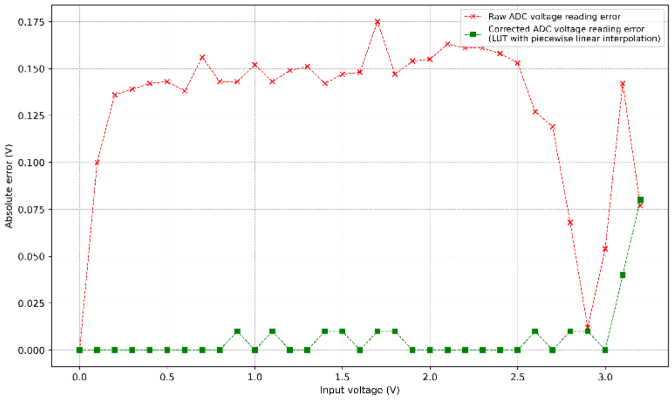

Figure 4. Error Distribution of ESP32 ADC Reading Before and After Correction using LUT And Piecewise Linear Interpolation.

Statistical analysis of the entire dataset reveals a very significant improvement in measurement accuracy. The average absolute error before correction was recorded at 0.112 V (3.42% of full scale), with a standard deviation of 0.045 V. The maximum error reached 0.155 V (4.73% of full scale) at an input voltage of 2.0 V. After applying LUT-based correction with piecewise linear interpolation, the average absolute error decreased drastically to 0.008 V (0.24% of full scale), with a standard deviation of only 0.005 V. Importantly, the maximum error was successfully reduced to 0.028 V (0.85% of full scale) at an input voltage of 3.1 V.

The Root Mean Square Error method provides an additional valuable metric for assessing correction effectiveness. Before correction, the Root Mean Square Error value was calculated at 0.121 V, while after correction, this value dropped to 0.009 V, representing a reduction of more than 92.5%. This significant improvement demonstrates the superiority of the proposed method in mitigating the inherent non-linearity of the ESP32 ADC.

To formally assess the linearity characteristics of the ADC, voltage error was converted to Least Significant Bit (LSB) units. With 3.277 V and 12-bit resolution, 1 LSB is equivalent to 0.8 mV. Before correction, the maximum error of 0.155 V was equivalent to 193 LSB, indicating significant Integral Non-Linearity (INL). After correction, the maximum error decreased to 0.028 V (35 LSB). This reduction indicates significant improvement in the linearity profile of the ADC transfer function, where deviation from the ideal straight line was successfully suppressed to below 1% of full scale.

Piecewise linear interpolation effectively eliminates systematic non-linearity, resulting in a nearly ideal linear response. At the calibration points themselves, errors are nearly zero, indicating conformity with ideal values, while at points between calibration points, errors remain minimal thanks to strategic placement of calibration points in regions with highest non-linearity.

This significant error reduction is attributed to the capability of piecewise linear interpolation in performing local approximation of the ADC non-linear curve. Unlike global regression that uses only one mathematical function for the entire range, this method divides the characteristic curve into small segments between calibration points. Since calibration point density is optimized in high non-linearity regions (0.5–2.5 V), the linearity assumption within each small segment becomes more valid. This minimizes residual error between calibration points, allowing the correction function to follow the actual ADC characteristic curve more precisely without requiring complex mathematical models.

Overall, these experimental results confirm that the LUT-based correction approach with piecewise linear interpolation is capable of substantially improving measurement accuracy while maintaining computational efficiency suitable for real-time embedded system applications. This method offers a practical solution to overcome hardware limitations without requiring complex analog circuit modifications or addition of expensive external components.

Although this research shows significant results in correcting ESP32 ADC readings, it should be noted that the ESP32 is sensitive to temperature variations [5], [8]. This research focuses on correction optimization under standard operating conditions. Characterization of temperature impact on calibration drift becomes a recommendation for further development.

B. Comparative Analysis and Computational Efficiency

A key innovation of the method proposed in this research is the computational efficiency that makes it highly suitable for real-time applications with strict time constraints. Unlike polynomial regression methods [11], [12] or neural network-based corrections [14], [15], [16], [17], [18], [19] that require floating-point operations and complex mathematical computations, the approach proposed in this research operates entirely using fixed-point integer arithmetic.

Computational complexity analysis reveals that this method requires only 126 clock cycles per interpolation on the ESP32 Xtensa LX6 processor, compared to 487 cycles for cubical spline implementation and 325 cycles for fourth-order polynomial regression. This represents a processing time reduction of 74% and 61%, respectively. Linear search through the lookup table (LUT) (O(n) complexity) proved more efficient than binary search (O(log n)) for our specific case with n = 33 points, because the additional overhead of the binary search algorithm exceeds its theoretical complexity advantage for small n [26].

Table 2 presents a comparative analysis of computational performance across various correction methods. Processing time and memory data for the proposed method were obtained from direct measurements using a logic analyzer and memory usage reports on the serial monitor. Meanwhile, processing time and memory data for comparison methods (Polynomial, Neural Network, etc.) were cited from relevant literature studies using similar hardware platforms to ensure relevant comparison.

Table 2. Comparative Analysis of ADC Correction Methods

Method	Memory Usage	Processing Time (ms)	Maximum Error (%)	Implementation Complexity
Proposed Method (LUT + Piecewise Linear Interpolation)	264 bytes	2.3	0.85	Low
Polynomial Regression (Order 4) [11], [12]	32 bytes	5.2	3.2	Medium
Neural Network Based [14], [15], [16], [17], [18], [19]	1.8 KB	18.7	1.2	High
Sparse LUT [21], [22]	128 bytes	1.8	4.1	Low
Cubic Spline Interpolation [13]	256 bytes	9.1	2.3	Medium
Hardware Calibration [10]	N/A	N/A	0.5	Very High
No Correction	N/A	0.5	10.2	-

The proposed method achieves an optimal balance between accuracy and resource load. Processing time of 2.3 ms is 55% faster than polynomial regression (5.2 ms) and 87% faster than Neural Network (18.7 ms). This efficiency is achieved through elimination of computationally expensive floating-point operations. Although Sparse LUT [21] is faster (1.8 ms), its accuracy is significantly inferior (4.1% error) due to lack of calibration points in critical regions. The memory efficiency claim (264 bytes) also outperforms Neural Network-based methods requiring up to 1.8 KB for network weights [14].

This computational efficiency is critical for IoT applications where power consumption correlates directly with processing time. In battery-operated sensor networks, our method reduces active processing time by approximately 75% compared to neural network approaches, potentially extending battery life by a similar margin [23], [24].

C. Method Limitations

Although demonstrating significant performance improvements, this method has several limitations that need to be considered in practical implementation, including:

1. Temperature Sensitivity: Calibration was performed at room temperature (25°C ± 2°C). Since ESP32 ADC characteristics are sensitive to temperature [5], [8], thermal drift can alter the non-linearity profile. This method requires recalibration if the device operates in extreme environments (>60°C or <0°C).
2. High-Frequency Dynamic Signals: The interpolation algorithm and LUT search require 2.3 ms per sample. For input signals that change very rapidly (frequency > 100 Hz), this latency can cause response delay (phase lag), making this method more optimal for quasi-static monitoring applications.
3. Reference Voltage Dependency: Method accuracy depends on the stability of the internal reference voltage. If it fluctuates undetected, the LUT map becomes invalid. Advanced implementation is recommended to use stable external references for critical applications.

D. Practical Implementation and Reproducibility

One of the most significant contributions of this research is the provision of a complete and reproducible methodology that can be easily implemented by practitioners. Unlike many previous publications that do not document important implementation details [6], this research presents a step-by-step calibration procedure that has been rigorously validated on multiple ESP32 units.

Practical implementation of the method in this research does not require additional hardware components, making it accessible to developers working with limited budgets. The entire correction process, including averaging, LUT search, and interpolation, requires only 2.3 ms per measurement cycle, which is negligible for most sensor applications. This practical efficiency, combined with significant accuracy improvement, makes this approach highly valuable for real-world implementation.

This calibration procedure is designed to be systematic and customizable. The procedure begins with measuring the actual reference voltage (V_{ref}) using a precision multimeter, since the internal ESP32 reference voltage can vary significantly between units [5], [8]. Calibration points are then placed strategically based on error profile analysis, with higher density in regions with maximum non-linearity. This approach ensures optimal correction accuracy while minimizing the number of calibration points required. To validate the reproducibility of this method, this research conducted calibration and testing on ten different ESP32 units from various production batches. Results showed consistent performance across all units, with an average maximum error of 0.031 V (0.95% of full scale) after correction. This consistency indicates that the method effectively compensates for inter-unit variations in ADC performance, which is a critical requirement for massively deployed IoT systems.

E. Broader Implications for Embedded System Development

The findings of this research are not only relevant for ADCs on ESP32, but also offer a framework that can be applied to characterize and correct non-linearity in built-in ADCs of other low-cost microcontrollers. This is particularly important amid the rapid growth of the IoT market, where budget constraints often force developers to use integrated ADCs despite their limitations.

This research demonstrates that sub-1% accuracy can be achieved without burdening computational performance. This finding challenges the common assumption that high-precision measurements always require expensive external ADCs. This paradigm shift opens new opportunities for ESP32 utilization in applications such as environmental monitoring and medical diagnostics [22], where the balance between cost and accuracy is critically important.

In environmental monitoring, for example, this method enables deployment of larger sensor networks with better data quality without increasing hardware costs. Large-scale environmental studies that were previously too expensive are now more affordable [24], [25].

In the healthcare field, this approach enables development of affordable medical monitoring devices that are sufficiently accurate for initial screening. Although not replacing clinical instruments, such devices can assist early detection in resource-limited regions [25].

The impact of this research is also felt in the education sector. A clear and well-documented ADC correction methodology facilitates students and novice researchers to engage in precision measurement system development, thereby encouraging cross-disciplinary innovation.

IV. Conclusions

This research study presents a significant breakthrough in significantly enhancing the overall performance of analog data acquisition on the ESP32 microcontroller platform. Through an efficient and accurate approach, the following are four primary conclusions drawn from the development of this ADC correction method:

1. Significant Accuracy Improvement: The implementation of the combined Lookup Table (LUT) and piecewise linear interpolation method successfully reduced the average absolute error from 3.42% to 0.24%, while decreasing the Root Mean Square Error (RMSE) by over 92.5% compared to the standard configuration.
2. High Computational Efficiency: This method is extremely lightweight, requiring only 264 bytes of memory and a processing time of 2.3 ms per measurement, making it 74% faster compared to cubic spline implementations.
3. Usage Consistency: The calibration strategy, which concentrates 65% of points within the critical voltage range, guarantees high inter-unit reproducibility, thereby supporting quality consistency in the mass production of IoT devices.
4. Economic Impact on Applications: This solution demonstrates that high accuracy can be achieved without expensive external ADCs, opening opportunities for cost-sensitive applications in healthcare, environmental monitoring, and industrial sectors with limited resources.

Although demonstrating superior performance in experimental validation, this study possesses several inherent limitations that must be acknowledged, including:

1. Calibration was conducted under controlled room temperature conditions (25°C ± 2°C); consequently, the impact of thermal drift on ADC characteristics has not been fully quantified.
2. The processing latency of 2.3 ms limits applications involving dynamic signals with high frequencies.
3. The accuracy of the method depends on the stability of the internal reference voltage, which can vary between units and throughout the operational lifetime.

Based on the existing limitations identified, several recommended avenues for future research are proposed as follows:

1. Thermal Characterization to develop a temperature compensation model capable of correcting calibration drift under extreme operating conditions;
2. Real-Time Optimization to reduce algorithm latency, thereby accommodating signals with higher frequencies;
3. External Reference Implementation for critical applications requiring higher reference voltage stability;

4. Method Generalization by testing the same approach on other low-cost microcontroller platforms (such as STM32, Arduino AVR) to validate the universality of this method.

Acknowledgments

The authors would like to express their sincere gratitude to the Faculty of Engineering, Universitas Cendekia Abditama, for providing essential financial support through the program of “Bantuan Internal Pendanaan Publikasi Fakultas Teknik”. This generous assistance proved invaluable in facilitating the comprehensive writing, research completion, and subsequent publication process of this journal manuscript.

V. References

[1] H. Y. Riskiawan *et al.*, “Artificial Intelligence Enabled Smart Monitoring and Controlling of IoT-Green House,” *Arab. J. Sci. Eng.*, vol. 49, no. 3, pp. 3043–3061, Mar. 2024, doi: 10.1007/s13369-023-07887-6.

[2] L. K. S. Tolentino *et al.*, “Development of an IoT-based Aquaponics Monitoring and Correction System with Temperature-Controlled Greenhouse,” in *2019 International SoC Design Conference (ISOCC)*, Jeju, Korea (South): IEEE, Oct. 2019, pp. 261–262. doi: 10.1109/ISOCC47750.2019.9027722.

[3] K. Bagaskara, A. Mahmudi, and Y. A. Pranoto, “Sistem Kontrol Dan Monitoring Pada Tanaman Bawang Merah Berbasis Iot,” *JATI (Jurnal Mahasiswa Teknik Informatika)*, vol. 7, no. 1, pp. 873–880, 2023.

[4] Mr. S. Kedar and Dr. A. Mulani, “IoT Based Soil, Water and Air Quality Monitoring System for Pomegranate Farming,” *Journal of Electronics,Computer Networking and Applied Mathematics*, no. 12, pp. 33–43, Nov. 2021, doi: 10.55529/jecnam.12.33.43.

[5] M. J. Espinosa-Gavira, A. Agüera-Pérez, J. C. Palomares-Salas, J. M. Sierra-Fernandez, P. Remigio-Carmona, and J. J. González de-la-Rosa, “Characterization and Performance Evaluation of ESP32 for Real-time Synchronized Sensor Networks,” *Procedia Comput. Sci.*, vol. 237, pp. 261–268, 2024, doi: 10.1016/j.procs.2024.05.104.

[6] K. da R. Severo, J. W. M. Kaehler, C. Müller, A. Girardi, N. B. Chagas, and P. C. Aguirre, “A LUT-Based Calibration Approach of the ESP32 ADC for a Power Quality Analyzer,” *2025 IEEE 16th Latin America Symposium on Circuits and Systems (LASCAS)*, vol. 1, pp. 1–5, 2025, doi: 10.1109/LASCAS64004.2025.10966286.

[7] C. G. C. Carducci, A. Monti, M. H. Schraven, M. Schumacher, and D. Mueller, “Enabling ESP32-based IoT Applications in Building Automation Systems,” in *2019 II Workshop on Metrology for Industry 4.0 and IoT (MetroInd4.0\&{IoT})*, Naples, Italy: IEEE, Jun. 2019, pp. 306–311. doi: 10.1109/METROI4.2019.8792852.

[8] I. H. Baciu, G. Chindris, and M. A. Taut, “Linearity Evaluation and Influence of the Reference Voltage for ADC Converter of Embedded Devices,” in *2024 IEEE 30th International Symposium for Design and Technology in Electronic Packaging (SIITME)*, Sibiu, Romania: IEEE, Oct. 2024, pp. 266–269. doi: 10.1109/SIITME63973.2024.10814820.

[9] V.-K. Tran, B.-T. Thai, H. Pham, V.-K. Nguyen, and V.-K. Nguyen, “A Proposed Approach to Utilizing Esp32 Microcontroller for Data Acquisition,” *Journal of Engineering and Technological Sciences*, vol. 56, no. 04, pp. 474–488, Aug. 2024, doi: 10.5614/j.eng.technol.sci.2024.56.4.4.

[10] D. Verma *et al.*, “A Design of 10-Bit Asynchronous SAR ADC with an On-Chip Bandgap Reference Voltage Generator,” *Sensors*, vol. 22, no. 14, p. 5393, Jul. 2022, doi: 10.3390/s22145393.

[11] A. Parisi, M. Mercandelli, C. Samori, and A. L. Lacaita, “A PLL-Based Digital Technique for Orthogonal Correction of ADC Non-Linearity,” in *2021 28th IEEE International Conference on Electronics, Circuits, and Systems (ICECS)*, Dubai, United Arab Emirates: IEEE, Nov. 2021, pp. 1–4. doi: 10.1109/ICECS53924.2021.9665564.

[12] D. Li, Y. Zhu, L. Wang, S. Liu, and Z. Zhu, “Low-Cost Linearity Testing of High-Resolution ADCs Using Segmentation Modeling and Partial Polynomial Fitting,” in *2024 IEEE International Symposium on Circuits and Systems (ISCAS)*, Singapore, Singapore: IEEE, May 2024, pp. 1–4. doi: 10.1109/ISCAS58744.2024.10558217.

[13] H. Geng, X. Chen, N. Zhao, Y. Du, and L. Du, “QPA: A Quantization-Aware Piecewise Polynomial Approximation Methodology for Hardware-Efficient Implementations,” *IEEE Trans. Very Large Scale Integr. VLSI Syst.*, vol. 31, no. 7, pp. 931–944, Jul. 2023, doi: 10.1109/TVLSI.2023.3277023.

[14] D. Zhai *et al.*, “High-Speed and Time-Interleaved ADCs Using Additive-Neural-Network-Based Calibration for Nonlinear Amplitude and Phase Distortion,” *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 69, no. 12, pp. 4944–4957, Dec. 2022, doi: 10.1109/TCSI.2022.3201016.

[15] Z. Fang, M. Zhang, and H. Zhou, “Research on an accuracy improvement technology of ADC acquisition system based on pruning neural network,” in *International Conference on Mechatronic Engineering and Artificial Intelligence (MEAI 2023)*, F. Liu and Y. Wei, Eds., Shenyang, China: SPIE, Feb. 2024, p. 133. doi: 10.1117/12.3025581.

[16] S. Rafiei and N. Shiri, “A neural network-based error correction in the first-stage residue of pipelined analog to digital converters,” *International Journal of Circuit Theory and Applications*, vol. 52, no. 12, pp. 6001–6027, Dec. 2024, doi: 10.1002/cta.4076.

[17] Z. Lu *et al.*, “A Novel NN-Based Fast-Convergence Background Calibration for Timing Mismatch in TI ADCs,” *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 72, no. 1, pp. 48–52, Jan. 2025, doi: 10.1109/TCSII.2024.3477463.

[18] S. Li, L. Zhang, and X. Qin, “A Neural Network-based Digital Calibration Improvement Method for ADC,” in *2023 IEEE International Conference on Image Processing and Computer Applications (ICIPCA)*, Changchun, China: IEEE, Aug. 2023, pp. 693–698. doi: 10.1109/ICIPCA59209.2023.10257940.

[19] Y. Peng, Y. Xiao, L. Chen, H. Tang, and X. Peng, “A Novel Calibration Algorithm for ADCs Based on Inverse Mapping by Neural Network,” *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 71, no. 7, pp. 3283–3287, Jul. 2024, doi: 10.1109/TCSII.2024.3362359.

[20] S. Qiu, D. Zhou, M. Kinyua, and J. Silva-Martinez, “Look-Up Table-Based Digital Calibration for the Correction of Pipeline Analog-to-Digital Converter Non-Idealities,” *IEEE Access*, vol. 12, pp. 196828–196841, 2024, doi: 10.1109/ACCESS.2024.3520980.

[21] B. Minger, G. Ferre, D. Dallet, and L. Fuche, “A new approach for compensating memory nonlinear analog-to-digital converters using look-up tables,” in *2016 IEEE International Conference on Electronics, Circuits and Systems (ICECS)*, Monte Carlo, Monaco: IEEE, Dec. 2016, pp. 293–296. doi: 10.1109/ICECS.2016.7841190.

[22] H. Wang, H. Zou, W. Liu, L. Yang, and X. Chen, “Implementation of a Real-Time Calibration Method for Nonlinear ADC Links Based on Compensation Sequence Lookup Tables,” in *2024 4th International Conference on Communication Technology and Information Technology (ICCTIT)*, Guangzhou, China: IEEE, Dec. 2024, pp. 197–200. doi: 10.1109/ICCTIT64404.2024.10928372.

[23] R. Ahmad, “Enhanced drift self-calibration of low-cost sensor networks based on cluster and advanced statistical tools,” *Measurement*, vol. 236, p. 115158, Aug. 2024, doi: 10.1016/j.measurement.2024.115158.

[24] N. V. S. R. Nalakurthi *et al.*, “Challenges and Opportunities in Calibrating Low-Cost Environmental Sensors,” *Sensors*, vol. 24, no. 11, p. 3650, Jun. 2024, doi: 10.3390/s24113650.

[25] Y. Liang, C. Li, S. Liu, and Z. Zhu, “A 14-b 20-MS/s 78.8 dB-SNDR Energy-Efficient SAR ADC With Background Mismatch Calibration and Noise-Reduction Techniques for Portable Medical Ultrasound Systems,” *IEEE Trans. Biomed. Circuits Syst.*, vol. 16, no. 2, pp. 200–210, Apr. 2022, doi: 10.1109/TBCAS.2022.3147954.

[26] D. T. Thinh, N. B. H. Quan, and N. Maneetien, “Implementation of Moving Average Filter on STM32F4 for Vibration Sensor Application,” in *2018 4th International Conference on Green Technology and Sustainable Development (GTSD)*, Ho Chi Minh City: IEEE, Nov. 2018, pp. 627–631. doi: 10.1109/GTSD.2018.8595630.