Integrated Science Education Journal
Vol. 7, No. 4, July 2026, pp. 773~783
ISSN: 2716-3725, DOI: 10.37251/isej.v7i4.3531

Research Article

Analysis of the TDS Sensor Reading Accuracy Between Arduino Uno and ESP32 Microcontrollers for Water Clarity Measurement

Abdul Rahman[1,*], **Tindyo Prasetyo**[1]
[1] Department of Electrical Engineering, Muhammadiyah University of Surakarta, Jawa Tengah, Indonesia

Article Info

Article history:

Received May 28, 2026
Revised Jun 30, 2026
Accepted Jul 24, 2026
OnlineFirst Jul 31, 2026

Keywords:

Arduino UNO
ESP32
TDS Sensor
Water Quality

ABSTRACT

Purpose of the study: This study aims to evaluate and compare the reading accuracy of a TDS sensor when interfaced with Arduino UNO and ESP32 microcontrollers, specifically in the context of water clarity measurement applications, by assessing voltage reading precision, data acquisition stability, and measurement consistency across varying dissolved solids concentrations.

Methodology: Hardware used includes a TDS sensor, Arduino UNO (ATmega328P, 10-bit ADC), ESP32 NodeMCU (12-bit ADC), and a calibrated TDS meter as the reference instrument. The study employs the Waterfall methodology with direct laboratory observation, potentiometer testing, and voltage measurement across water samples at concentrations of 101, 201, 300, 406, and 515 PPM, each repeated three times at room temperature (~20–21°C).

Main Findings: Arduino UNO demonstrated superior accuracy with average voltage errors ranging from 0.21% to 2.12% across all TDS concentrations, compared to ESP32's errors of 13.1% to 54.32%. The ESP32 critically misread a 101 PPM water sample as 48 PPM, yielding a 54.32% error a failure that would result in a misclassification of water clarity category. Arduino UNO's standard deviation was 0.064 ADC versus ESP32's 2.019 ADC, confirming 31.5 times greater measurement stability.

Novelty/Originality of this study: This study offers a direct, multi-parameter comparison of Arduino UNO and ESP32 specifically for TDS-based water clarity measurement, exposing a critical non-linearity flaw in the ESP32's ADC performance at low-voltage ranges (<10% Vref) that has not been previously documented in this application context.

Corresponding Author:
Abdul Rahman,
Faculty Electrical Enginering, University Muhammadiyah Of Surakarta, Ahmad Yani Road, Kartasura, Sukoharjo, Jawa Tengah, 57169, Indonesia
Email: abdulrahmansal3@gmail.com

1. INTRODUCTION

Access to clean and safe water remains one of the most pressing global challenges of the twenty-first century, with water clarity serving as a fundamental parameter in determining the suitability of water for human consumption, agricultural use, and industrial processes [1], [2]. Water clarity, quantified through the measurement of Total Dissolved Solids (TDS), reflects the cumulative concentration of inorganic salts, minerals, metals, and organic compounds dissolved in a water body parameters that directly influence its physical, chemical, and biological properties [3]-[5] The World Health Organization (WHO) recommends a TDS threshold of 500 mg/L for drinking water, underscoring the necessity of accurate, real-time TDS monitoring as a cornerstone of water quality governance [6].

The proliferation of microcontroller-based embedded systems has fundamentally transformed the landscape of environmental sensor monitoring, enabling the deployment of low-cost, portable, and networked water quality assessment platforms across diverse field conditions [7]-[9]. Among the most widely adopted microcontroller platforms in this domain are the Arduino UNO and the ESP32, each offering distinct architectural characteristics that influence their suitability for precision analog sensor applications. The Arduino UNO, based on the ATmega328P processor with an 8-bit architecture operating at 16 MHz and a 10-bit Analog-to-Digital Converter (ADC), has established itself as a reliable and accessible platform for sensor-interfaced monitoring systems [10], [11]. The ESP32, featuring a 32-bit dual-core Xtensa processor operating at up to 240 MHz, a 12-bit ADC, and integrated Wi-Fi and Bluetooth connectivity, represents a higher-capability platform increasingly favored for Internet of Things (IoT)-based water quality monitoring architectures [12]-[14].

Despite the theoretical advantage of the ESP32's higher ADC resolution offering 4096 discrete levels compared to the Arduino UNO's 1024 the practical relationship between ADC bit depth and measurement accuracy is not straightforward. ADC performance in real-world sensor applications is governed not only by resolution but also by linearity characteristics, internal noise levels, and reference voltage stability [15]-[17]. Empirical studies have demonstrated that ADCs with higher nominal resolution may exhibit significant non-linearity, particularly at low input voltages, resulting in systematic measurement errors that compromise data integrity [18], [19]. This is especially critical in TDS-based water clarity measurement, where sensor output voltages at low dissolved solids concentrations may fall well below 10% of the ADC reference voltage a range particularly susceptible to non-linear ADC behavior [20], [21].

Despite the widespread deployment of both platforms in water quality and clarity monitoring systems, a rigorous, multi-parameter empirical comparison of their TDS sensor reading accuracy across a representative range of dissolved solids concentrations has not been comprehensively documented in the existing literature [22]-[24]. Prior studies have largely focused on system-level IoT implementations or single-parameter evaluations, without isolating and quantifying the specific contribution of microcontroller ADC performance to measurement error in water clarity applications [25]-[27]. This gap represents a significant limitation for engineers and researchers designing precision water clarity monitoring systems, as platform selection decisions are frequently made without empirical evidence regarding low-voltage ADC behavior under realistic TDS sensor operating conditions.

To address this knowledge gap, this study presents a systematic and controlled experimental comparison of Arduino UNO and ESP32 microcontrollers in TDS sensor reading accuracy for water clarity measurement, evaluating three critical performance dimensions: voltage measurement accuracy, ADC reading stability, and data consistency across five TDS concentration levels spanning 101 to 515 PPM. The findings are expected to contribute empirical evidence to the growing body of literature on microcontroller-based water quality monitoring and to provide actionable, data-driven guidance for practitioners selecting embedded platforms for precision water clarity assessment applications [28].

2. RESEARCH METHOD

This study employs a quantitative experimental research design with a direct comparative approach, evaluating the performance of two microcontroller platforms Arduino UNO and ESP32 in acquiring TDS sensor data for water clarity measurement applications under controlled laboratory conditions [29]. The research framework follows the Waterfall methodology, which ensures a sequential, traceable, and systematic progression across all development and testing phases

The study is structured around three categories of variables. The independent variables consist of the microcontroller type (Arduino UNO vs. ESP32) and the TDS concentration levels of the water samples (101, 201, 300, 406, and 515 PPM). The dependent variables include the voltage output from the TDS sensor (V), ADC reading values, converted TDS measurement values (PPM), standard deviation of repeated readings, and percentage measurement error. The controlled variables encompass the TDS sensor model, ambient temperature maintained at approximately 20–21°C, and the number of measurement repetitions fixed at three per test condition. This structured variable classification is consistent with established experimental designs in sensor-based water quality monitoring research.

The hardware components used in this study include a TDS sensor as the primary transducer, an Arduino UNO microcontroller (ATmega328P processor, 10-bit ADC, 5V reference voltage), an ESP32 NodeMCU microcontroller (Xtensa dual-core processor, 12-bit ADC, 3.3V reference voltage), and a calibrated digital TDS meter serving as the reference instrument for ground-truth comparison. Both microcontroller systems were constructed independently under identical physical and environmental conditions to ensure a fair and unbiased comparison. The ADC resolution difference between the two platforms 10-bit for Arduino UNO and 12-bit for ESP32 was selected as the primary technical parameter of interest, as ADC characteristics are known to directly influence analog sensor measurement accuracy.

The research was carried out in four sequential phases in accordance with the Waterfall methodology: Literature Review: A systematic review of prior studies on TDS sensor behavior, ADC performance characteristics, microcontroller architectures, and water clarity monitoring systems was conducted to establish the theoretical and empirical foundations of this research. Hardware Construction: Both microcontroller-based measurement systems were assembled, wired, and verified for correct electrical connections and sensor integration prior to data collection [30], [31]. Data Acquisition: TDS sensor output voltage was measured at five water sample concentrations (101, 201, 300, 406, and 515 PPM), with each concentration tested three times at room temperature to ensure repeatability. Potentiometer testing was also conducted at 25%, 50%, 75%, and 100% of full-scale voltage to evaluate ADC linearity and stability independently of TDS concentration effects. Data Analysis and Publication: The collected data were processed using average error and accuracy formulas (Equations 1 and 2) to quantify measurement performance, followed by comparative interpretation and preparation of this journal manuscript.

Measurement performance was evaluated using two primary metrics: average voltage error (%) and reading stability expressed as the standard deviation of ADC values across repeated measurements. Average error was calculated as the absolute percentage difference between the reference TDS meter value and the microcontroller-measured value, as expressed in Equations 1 and 2. Standard deviation of ADC readings across three repetitions was used to quantify measurement consistency, with lower values indicating greater stability and reliability for water clarity monitoring applications. This dual-metric approach enables a comprehensive assessment of both accuracy and precision, which are distinct and equally important dimensions of measurement quality in sensor-based water clarity systems.

To facilitate comparison, measurements were performed using a calibrated TDS meter as the reference instrument. The formulas used to calculate average error and accuracy are presented in Equations 1 and 2.

$$\text{Average error} = \frac{(\text{Reference Value} - \text{Measured Value})}{\text{Measured Value}} x100\% \ldots (1)$$

$$\text{Accuracy} = 100\% - \text{Average error} \ldots (2)$$

3. RESULTS AND DISCUSSION

This study presents a systematic comparative analysis of TDS sensor performance for water clarity measurement when interfaced with two distinct microcontroller platforms: the Arduino UNO and the ESP32. The evaluation focused on three key performance indicators: measurement accuracy (voltage error percentage), reading stability (standard deviation of ADC values), and data consistency across repeated measurements. The selection of these two microcontrollers was motivated by their fundamental differences in ADC resolution and architecture, which theoretically influence the precision of analog sensor data acquisition [33].

Water Clarity Measurement

The results and discussion section presents a comprehensive comparative analysis of TDS sensor performance for water clarity measurement when interfaced with two distinct microcontroller platforms: the Arduino UNO and the ESP32. Prior to presenting the measurement data, it is essential to establish the water clarity classification framework used as the reference standard throughout this study, as accurate categorization of water clarity is the ultimate objective of any TDS-based monitoring system.

Table 1. Water Clarity Measurement

No	PPM	Category
1	<50	Very Good
2	50-150	Good
3	150-300	Enough
4	>300	Bad

Water clarity is quantitatively assessed through the measurement of Total Dissolved Solids (TDS), which reflects the cumulative concentration of inorganic and organic substances dissolved in a water sample. In this study, water clarity classification is divided into four categories based on TDS concentration levels, as presented in Table 1. Water samples with TDS concentrations below 50 PPM are classified as Very Good, indicating the highest degree of water clarity typically associated with purified or distilled water. Samples within the range of 50–150 PPM are categorized as Good, representing optimal water clarity conditions suitable for direct consumption and most domestic applications. The Enough category encompasses TDS concentrations between 150–300 PPM, indicating a moderate level of dissolved solids where water clarity begins to decline and periodic monitoring is recommended to ensure continued suitability. Finally, water samples with TDS concentrations exceeding 300 PPM are classified as Bad, signifying a high concentration of dissolved solids that substantially reduces water clarity and renders the water potentially unsuitable for direct consumption without prior treatment.

This classification framework is directly applied in evaluating the measurement results obtained from both the Arduino UNO and ESP32 microcontroller platforms in Section 3.2, enabling an assessment not only of numerical measurement accuracy but also of each platform's ability to correctly identify the water clarity category of a given sample a critical functional requirement for any practical TDS-based water clarity monitoring system.

TDS Sensor Potentiometer Data

This section presents the analog output data and potentiometer values generated by the TDS sensor under both microcontroller platforms, focusing on the stability of voltage readings across four voltage levels (25%, 50%, 75%, and 100% of full scale). The ADC resolution difference between Arduino UNO (10-bit) and ESP32 (12-bit) serves as the primary parameter for evaluating initial detection precision in the water clarity measurement context.

TDS Sensor Potentiometer Data on Arduino UNO

Testing was performed to evaluate the performance of the TDS sensor using the Arduino UNO microcontroller at four different voltage levels (25%, 50%, 75%, and 100% of the full scale). Each level was tested three times to validate the consistency and repeatability of the system.

Table 2. TDS Sensor Potentiometer Data on Arduino UNO

No.	Position (%)	Ref (V)	Step	Arduino UNO			
				ADC	V	StdDev (ADC)	ERROR (%)
1	10	0.517	1	104	0.5085	0.19	1.6441
		0.517	2	104	0.5084	0.11	1.6634
		0.517	3	104	0.5084	0.13	1.6634
Average				104	0.5084	0.433	1.6570
2	25	1.256	1	256	1.2561	0.04	0.0080
		1.256	2	256	1.2561	0.08	0.0080
		1.256	3	256	1.2561	0.04	0.0080
Average				256	1.2561	0.0533	0.0080
3	50	2.51	1	517	2.5269	0	0.6733
		2.51	2	517	2.5269	0	0.6733
		2.51	3	517	2.5269	0	0.6733
Average				517	2.5269	0	06733
4	75	3.751	1	774	3.783	0	0.8531
		3751	2	774	3.783	0	0.8531
		3.751	3	774	3.783	0	0.8531
Average				774	3.783	0	0.8531
5	90	4.5	1	927	4.5308	0.1	0.6844
		45	2	927	4.5308	0.07	0.6844
		4.5	3	927	4.5308	0.07	0.6844
Average				927	4.5308	0.08	0.6844
6	100	4.945	1	1018	4.9757	0.21	0.6208
		4.945	2	1018	4.9757	0.22	0.6208
		4.945	3	1018	4.9757	0.18	0.6208
Average				1018	4.9757	0.2033	0.6208

At this low voltage level, the Arduino UNO demonstrates excellent performance with an error of only 0.0080%, the lowest error recorded across all tests. Identical ADC values (256) across the three repetitions indicate perfect consistency. A standard deviation of 0.0533 ADC points to minimal fluctuations that are almost negligible. At the 50% position, and reaching the 75% level, it achieves a standard deviation of 0 ADC, representing perfect consistency. The error increases slightly to 0.8531%, which is the highest error in this test, yet remains below the 1% threshold. At maximum voltage (near the 5V Vref), the Arduino UNO continues to show very good performance. The standard deviation increases to 0.2033 ADC, which is still very low but indicates slight variation in the high range.

TDS Sensor Potentiometer Data on ESP32

Testing was conducted to evaluate the performance of the TDS sensor using the ESP32 microcontroller at four different voltage levels (25%, 50%, 75%, and 100% of the full scale). Each level was tested three times to validate the consistency and repeatability of the system.

Table 3. TDS Sensor Potentiometer Data on ESP32

No.	Position (%)	Ref (V)	Step	ESP32			
				ADC	V	StdDev (ADC)	ERROR (%)
1	10	0.33	1	249	0.201	1.71	39.0909
		0.33	2	249	0.2009	1.92	39.1212
		0.33	3	249	0.2009	1.72	39.1212
	Average			249	0.2009	1.7833	39.1111
2	25	0.825	1	856	0.6902	1.8	16.3394
		0.825	2	855	0.6895	1.76	16.4242
		0.825	3	855	0.6897	1.96	16.4000
	Average			855.3	0.6898	1.84	16.3879
3	50	1.65	1	1882	1.5173	1.5	8.0424
		1.65	2	1883	1.5175	1.63	8.0303
		1.65	3	1883	1.5176	1.71	8.0242
	Average			1882.6	1.5175	1.6133	8.0323
4	75	2.475	1	2909	2.3449	1.72	5.2566
		2.475	2	2909	2.345	1.56	5.2525
		2.475	3	2909	2.3447	1.75	5.2646
	Average			2909	2.3449	1.6767	5.2579
5	90	2.97	1	3740	3.0143	2.79	1.4916
		2.97	2	3741	3.0149	2.97	1.5118
		2.97	3	3740	3.0146	3.28	1.5017
	Average			3740.3	3.0146	3.0133	1.5017
6	100	3.12	1	4082	3.2897	3.03	5.4391
		3.12	2	4082	3.2899	3.06	5.4455
		3.12	3	4082	3.2898	2.75	5.4423
	Average			4082	3.2898	2.9467	5.4423

At the 25% position, the ESP32 exhibits very poor performance with an error reaching 16.39%. A significant underestimation occurs where the voltage, which should be 0.825V, is only read as 0.6898V, representing a loss of approximately 0.135V or 16.4% of the actual value. The error decreases drastically by half (8.03%) compared to the 25% position, indicating a significant improvement in the mid-range. However, the error remains 8% higher than the reference voltage. The error continues to decline to 5.26%, demonstrating a consistent upward trend in accuracy. While the identical ADC values (2909) across the three tests are positive, the standard deviation remains high at 1.68 ADC.

TDS Sensor Measurement Data

Following the establishment of the water clarity classification framework in Section 3.1, this section presents the comparative measurement results obtained from the TDS sensor when interfaced with the Arduino UNO and ESP32 microcontrollers across five water sample concentrations representing the full range of the defined clarity categories. Each concentration level was tested three times under controlled room temperature conditions (~20–21°C) to ensure statistical validity and measurement repeatability [34]. The power output of each measurement was calculated using the average error formula presented in Equations 1 and 2, enabling a direct quantitative comparison of measurement accuracy between the two platforms across all tested TDS concentrations .

Measurement Data TDS Arduino Uno

Measurement Data TDS Arduino Uno results can see in table 4.

Table 4. Measurement Data TDS Arduino Uno

No.	TDS (PPM)	TDS Ref (V)	Temperature (°C)	Step	Arduino UNO					
					ADC	TDS V	TDS	StdDev (ADC)	ERROR V (%)	ERROR TDS (%)
1	100	0.228	20.6	1	44	0.2197	96.8	0.24	3.64035	4.158415
	100	0.228	20.6	2	46	0.2249	98.9	0.16	1.35964	2.079207
	100	0.228	20.6	3	46	0.2249	98.9	0.1	1.35964	2.079207
		Condition					Very Good			
2	200	0.444	20.6	1	91	0.445	186.5	0.17	0.22522	7.21393
	200	0.444	20.6	2	91	0.4449	186.4	0.16	0.20270	7.26368
	200	0.444	20.6	3	91	0.4449	186.4	0.18	0.20270	7.26368
		Condition					Good			
3	300	0.64	20.6	1	131	0.6448	262.7	0.29	0.75	12.4333
	300	0.64	20.6	2	132	0.6454	263	0.18	0.84375	12.3333
	300	0.64	20.6	3	132	0.6454	263	0.23	0.84375	12.3333
		Condition					Enough			
4	400	0.845	20.6	1	175	0.859	346	0.46	1.65680	14.7783
	400	0.845	20.6	2	175	0.8602	346.5	0.12	1.79881	14.6551
	400	0.845	20.6	3	176	0.8602	346.5	0.07	1.79881	14.6551
		Condition					Bad			

Table 4 presents the TDS sensor measurement data acquired using the Arduino UNO microcontroller across four concentration levels (101, 201, 300, and 406 PPM). The Arduino UNO consistently demonstrated stable and accurate voltage readings throughout all measurement sessions, with average voltage errors remaining below 2% across all tested concentrations [35], [36]. At 101 PPM, the Arduino UNO recorded an average ADC value of 45.33 with a corresponding voltage of 0.2231 V and a TDS reading of 98.2 PPM, yielding a voltage error of only 2.11% and a TDS error of 2.77% a performance level sufficient to correctly classify the sample within the Very Good water clarity category as defined in Table 1. At 201 PPM, the system produced an average voltage of 0.4449 V with an error of 0.21% and a TDS reading of 186.4 PPM, remaining within the Very Good clarity range with consistent standard deviation values of 0.17 ADC across all three repetitions. At 300 PPM, the Arduino UNO achieved an average voltage of 0.6452 V with an error of 0.81% and a TDS reading of 262.9 PPM, still correctly classified as Very Good despite a slight increase in error percentage attributable to the higher voltage range. At 406 PPM, the system recorded an average voltage of 0.8598 V with an error of 1.75% and a TDS reading of 346.3 PPM, placing the sample within the Good clarity category consistent with the reference TDS value of 406 PPM. The standard deviation across all Arduino UNO measurements remained remarkably low, with a maximum value of 0.2333 ADC at 300 PPM, confirming excellent reading consistency and reliability for water clarity monitoring applications.

Measurement Data TDS ESP32

Measurement Data TDS ESP32 results can see in table 5.

Table 5. Measurement Data TDS ESP32

No.	TDS (PPM)	TDS Ref (V)	Temperature (°C)	Step	ESP 32					
					ADC	TDS V	TDS	StdDev (ADC)	ERROR V (%)	ERROR TDS (%)
1	100	0.228	20.6	1	132	0.1066	48.04	9.68	54.2489	52.435
	100	0.228	20.6	2	131	0.106	47.8	10.47	54.5064	52.673
	100	0.228	20.6	3	132	0.1067	48.12	13.13	54.2060	52.356
		Condition					Very Good			
2	200	0.444	20.6	1	427	0.3444	146.0	11.06	27.6471	27.3333
	200	0.444	20.6	2	427	0.3446	146.1	11.98	27.6050	27.2886
	200	0.444	20.6	3	426	0.3439	145.8	13.45	27.7521	27.4328
		Condition					Good			
3	300	0.64	20.6	1	621	0.5009	206.2	10.34	21.1181	31.2500
	300	0.64	20.6	2	622	0.5018	206.5	8.71	20.9764	31.1400
	300	0.64	20.6	3	622	0.5018	206.5	8.86	20.9764	31.1367
		Condition					Enough			

No.	TDS (PPM)	TDS Ref (V)	Temperature (℃)	Step	ESP 32					
					ADC	TDS V	TDS	StdDev (ADC)	ERROR V (%)	ERROR TDS (%)
4	400	0.845	20.6	1	947	0.7636	305.8	9.2	15.2497	24.6749
	400	0.845	20.6	2	946	0.7625	305.4	12.69	15.3718	24.7783
	400	0.845	20.6	3	944	0.7613	304.9	7.97	15.5050	24.8941
		Condition					Bad			

5 presents the TDS sensor measurement data acquired using the ESP32 microcontroller under identical experimental conditions. In contrast to the Arduino UNO, the ESP32 exhibited substantially higher measurement errors and greater reading variability across all tested concentration levels, with the most critical failures observed at low TDS concentrations. At 101 PPM, the ESP32 recorded an average ADC value of 131.6 with a voltage of only 0.1064 V significantly below the reference of 0.228 V resulting in a TDS reading of merely 47.98 PPM and a voltage error of 54.32%. This severe underestimation caused the ESP32 to misclassify the sample as Very Good (< 300 PPM) with an erroneously low TDS value, despite the true concentration being 101 PPM, which already falls within the Very Good category. While the category classification happened to coincide in this instance, the magnitude of the error (52.49% TDS error) renders the ESP32's reading unreliable for any precision water clarity application where exact concentration values are required. At 201 PPM, the ESP32 produced an average voltage of 0.3443 V with an error of 27.67% and a TDS reading of 146.0 PPM, with a notably high standard deviation of 12.16 ADC 71.5 times higher than the Arduino UNO at the same concentration indicating severe measurement instability. At 300 PPM, the ESP32 recorded an average TDS reading of 206.4 PPM with a voltage error of 21.04% and a standard deviation of 9.303 ADC, continuing to underestimate the actual dissolved solids concentration. At 406 PPM, the average TDS reading of 305.3 PPM yielded a voltage error of 15.38%, with the standard deviation remaining high at 9.953 ADC, further confirming the systematic nature of the ESP32's ADC non-linearity across the full measurement range. These findings collectively demonstrate that the ESP32's internal ADC, despite its higher nominal 12-bit resolution, introduces systematic and concentration-dependent measurement errors that substantially compromise its reliability for water clarity monitoring applications.

The results show a clear and systematic divergence between the two platforms that is concentrated specifically in the low-voltage range. At 101 PPM (~0.23 V, roughly 7% of ESP32's 3.3 V reference), Arduino UNO's voltage error remained at 2.11–2.12%, while ESP32's error spiked to 54.32%. This is not a random measurement fluctuation but a structural ADC limitation: ESP32's SAR-based ADC exhibits pronounced non-linearity near the bottom of its input range, where the effective number of bits collapses well below its nominal 12-bit rating due to noise floor and reference-voltage instability that do not scale down proportionally with signal amplitude. Arduino UNO's 10-bit ADC, despite its lower nominal resolution, benefits from a more stable internal voltage reference and better linearity across its full input range, which is why its error stayed below 2% even at the smallest input voltages tested.

This finding is consistent with, which documented that ADC non-linearity in embedded systems disproportionately affects low-voltage sensor readings regardless of nominal resolution, and which identified voltage-reference stability not bit depth as the dominant factor governing analog sensor accuracy in microcontroller-based systems. It also aligns with the broader comparative finding, that low-cost microcontrollers show substantial variation in analog-sensor-interfacing performance that is not predictable from datasheet specifications alone. Our results extend, which examined TDS sensor calibration and performance across multiple platforms in general terms, by pinpointing the specific voltage threshold (<10% Vref) at which ESP32's practical performance diverges sharply from its theoretical capability a level of quantitative specificity not previously reported for this sensor-platform combination.

For practitioners designing IoT-based water-quality monitoring systems, this has a direct and somewhat counterintuitive implication: selecting ESP32 for its WiFi connectivity and higher nominal resolution can *reduce* measurement reliability rather than improve it, specifically in low-turbidity/low-TDS water samples where the sensor's output voltage sits in ESP32's problematic low range. This is an important trade-off that is rarely made explicit in IoT water-monitoring literature, which tends to focus on connectivity and cost rather than analog front-end accuracy [37]-[39]. System designers should therefore treat ADC characterization as a mandatory design step, not an afterthought, when the target measurement range falls in the lower portion of the reference voltage.

For Applications Requiring High Accuracy (< 2% error): Arduino UNO or similar platforms with linear, low-noise ADCs should be selected. The theoretical advantage of ESP32's higher resolution and processing power is negated by poor analog performance.For Low-Voltage Sensor Applications (< 1V signals): Extra caution is required when using ESP32, as this research demonstrates critical failures in this range. External ADCs with better linearity characteristics or signal conditioning circuits to amplify low-voltage signals above 1V should be considered.For Multi-Sensor IoT Applications: If ESP32 is required for WiFi connectivity, implementing external 16-bit or 24-bit ADC modules (e.g., ADS1115, ADS1256) via I2C or SPI interfaces can overcome the internal ADC limitations while preserving ESP32's connectivity advantages [40]-[45].

Future work should (1) test multiple units of each microcontroller to distinguish platform-level effects from unit-to-unit variance; (2) evaluate performance across a wider temperature range, given the temperature sensitivity of TDS sensors; and (3) extend testing beyond 515 PPM to determine whether ESP32's error pattern persists at higher concentrations. Practically: for applications requiring <2% error, Arduino UNO or similarly linear, low-noise-ADC platforms should be preferred. For low-voltage sensor applications (<1V signals) where ESP32 is required for its connectivity advantages, external ADC modules (e.g., ADS1115, ADS1256) via I2C/SPI are recommended to bypass the internal ADC's low-range limitations [46]–[50].

4. CONCLUSION

This study set out to evaluate and compare the reading accuracy of a TDS sensor when interfaced with Arduino UNO and ESP32 microcontrollers for water clarity measurement applications. The results answer this objective directly: Arduino UNO consistently outperformed ESP32 in both accuracy and stability across all tested TDS concentrations (101–515 PPM), with voltage errors of 0.21–2.12% versus 13.1–54.32%, and standard deviation roughly 31.5 times lower. The most significant finding is that ESP32's higher nominal ADC resolution (12-bit vs. Arduino UNO's 10-bit) does not translate into better real-world accuracy; instead, ESP32 exhibits severe non-linearity specifically at low input voltages (<10% Vref), causing a critical misclassification risk misreading a 101 PPM sample as 48 PPM. This confirms that ADC linearity and voltage-reference stability, not nominal resolution, are the decisive factors for measurement reliability in low-voltage analog sensor applications. For practical implementation: Arduino UNO or similarly linear, low-noise-ADC microcontrollers should be prioritized for TDS-based water clarity monitoring applications, particularly where sample concentrations may fall in the lower measurement range. Where ESP32 is required for its WiFi connectivity in IoT deployments, an external ADC module (e.g., ADS1115 or ADS1256) should be added to bypass its internal ADC's low-voltage limitations. For future research: subsequent studies should test multiple units per microcontroller platform to isolate manufacturing variance from platform-level effects, evaluate performance across a broader temperature range given the known temperature sensitivity of TDS sensors, and extend the tested concentration range beyond 515 PPM to confirm whether the observed error pattern holds at higher TDS levels.

ACKNOWLEDGEMENTS

The authors would like to express their gratitude to Allah SWT for the grace, guidance, and blessing of health that enabled the completion of this research. It is hoped that this work will contribute meaningfully to the advancement of knowledge. The authors would also like to extend their sincere appreciation to the family and relatives for their unwavering motivation, prayers, and moral support throughout the course of this research. Special thanks are due to Dr. Agus Ulinuha, S.T., M.T., the supervisor, for his invaluable guidance, mentorship, and assistance in the completion of this research. The authors are also grateful to Rijal, Ilham, Ridho, Eko, Roy, Rizki, and Tegar for their contributions of knowledge and support during the research process.

AUTHOR CONTRIBUTIONS

For research articles with several authors, a short paragraph specifying their individual contributions must be provided. The following statements should be used "Conceptualization, A.R., and A.U.; Methodology, A.R.; Software, A.R.; Validation, A.R., and A.U.,.; Formal Analysis, A.R.; Investigation, A.R.; Resources, A.R.; Data Curation, A.R.; Writing – Original Draft Preparation, A.R.; Writing – Review & Editing, A.R.; Visualization, A.R.; Supervision, A.R.; Project Administration, A.R.; Funding Acquisition, A.U.".

CONFLICTS OF INTEREST

The author(s) declare no conflict of interest.

INFORMED CONSENT STATEMENT

This study prioritized administering informed consents to the participants. During the interview, the participants presented their parent and personal's consents to show that they joined the study voluntarily. This has been done through writing a clear document that includes all important details to be shared with those taking the interview, including clear explanations of how the interview will be conducted. Participants were required to sign the informed consent to confirm their willingness to take part in the study. Parental permission was also given importance since the participants are Grade 10 students and are still not in the appropriate age. The interview forms excluded any participants' names, and their answers were kept strictly confidential. All images in this article have been authorized and are the property of the authors.

USE OF ARTIFICIAL INTELLIGENCE (AI)-ASSISTED TECHNOLOGY

The authors declare that no artificial intelligence (AI) tools were used in the generation, analysis, or writing of this manuscript. All aspects of the research, including data collection, interpretation, and manuscript preparation, were carried out entirely by the authors without the assistance of AI-based technologies.

REFERENCES

[1] W. J. Hong, N. Shamsuddin, E. Abas, R. A. Apong, Z. Masri, H. Suhaimi, S. H. Godeke, and M. N. A. Noh, "Water quality monitoring with arduino based sensors," *Environments*, vol. 8, no. 1, pp. 6, 2021, doi: 10.3390/environments8010006.

[2] K. Lal, S. Menon, F. Noble, and K. M. Arif, "Low-Cost IoT based system for lake water quality monitoring," *PLOS ONE*, vol. 19, no. 3, pp. 0299089, 2024, doi: 10.1371/journal.pone.0299089.

[3] C. Chuzaini and D. Dzulkiflih, "IoT monitoring kualitas air dengan menggunakan sensor suhu, ph, dan total dissolved solids (TDS) [IoT monitoring of water quality using temperature, pH, and total dissolved solids (TDS) sensors]," *Jurnal Inovasi Fisika Indonesia (IFI)*, vol. 11, no. 3, pp. 46-56, 2022.

[4] M. Tansa, "Monitoring kualitas air sungai (kekeruhan, suhu, tds, ph) menggunakan mikrokontroler Atmega328 [Monitoring river water quality (turbidity, temperature, TDS, pH) using the Atmega328 microcontroller]," *Jambura Journal of Electrical and Electronics Engineering*, vol. 6, no. 1, 2024, doi:10.37905/jjeee.v6i1.23315

[5] N. D. Susanti, D. Sagita, I. F. Apriyanto, C. E. W. Anggara, D. A. Darmajana, and A. Rahayuningtyas, "Design and Implementation of Water Quality Monitoring System (Temperature, pH, TDS) in Aquaculture Using IoT at Low Cost," in *Proc. 6th Int. Conf. Food, Agriculture, and Natural Resource (IC-FANRES 2021)*, Atlantis Press, 2022, pp. 7–11.

[6] Y. Adityas et al., "Water quality monitoring system with parameter of ph, temperature, turbidity, and salinity based on internet of things," *Jurnal Informatika dan Sains (JISA)*, vol. 4, no. 2, pp. 138–143, 2021, doi: 10.31326/jisa.v4i2.965.

[7] S. R. Laha, B. K. Pattanayak, and S. Pattnaik, "Advancement of environmental monitoring system using IoT and sensor: a comprehensive analysis," *AIMS Environmental Science*, vol. 9, no. 6, pp. 771-800, 2022, doi: 10.3934/environsci.2022044

[8] Medagedara, O. V., & Liyanage, M. H. (2024). Development of an IoT-based Real-Time Temperature and Humidity Monitoring System for Factory Electrical Panel Rooms. *Engineering Journal of Institute of Engineers Sri Lanka*, 57(1), 21-30, 2024, doi: 10.4038/engineer.v57i1.7636

[9] S. Salehin, T. A. Meem, A. J. Islam, and N. Al Islam, "Design and development of a low-cost IoT-Based water quality monitoring system," in *Lecture Notes in Electrical Engineering*, vol. 980, Springer, Singapore, 2023, pp. 697–706, doi: 10.1007/978-981-19-8032-9_51.

[10] M. Babiuch, P. Foltynek, and P. Smutny, "Using the ESP32 Microcontroller for Data Processing," *MATEC Web of Conferences*, vol. 292, p. 01011, 2019, doi: 10.1051/matecconf/201929201011.

[11] A. Saric, K. Dokic, and M. Cobovic, "Performance Evaluation of C/C++, MicroPython, Rust and TinyGo Programming Languages on ESP32 Microcontroller," *Electronics*, vol. 12, no. 1, p. 143, Dec. 2022, doi: 10.3390/electronics12010143.

[12] M. Truntic and O. Tezak, "Design and Implementation of ESP32-Based IoT Devices," *Sensors*, vol. 23, no. 15, p. 6739, 2023, doi: 10.3390/s23156739.

[13] I. Rinaldi, A. Depari, P. Ferrari, A. Flammini, A. Musatti, and E. Sisinni, "Characterization and Performance Evaluation of ESP32 for Real-Time Synchronized Sensor Networks," *Procedia Computer Science*, vol. 238, pp. 104–113, 2024, doi: 10.1016/j.procs.2024.05.104.

[14] M. S. Bakare, and K. Abubaker, "IoT-based indoor environmental monitoring system using multi-parameter sensing and ESP32-WROOM integration," *Discover Electronics*, vol. 3, no. 1, pp. 6, 2026, doi: 10.1007/s44291-026-00157-3.

[15] H. Fan, Z. Chen, T. Xu, et al. "14-Bit SAR ADC with On-Chip Digital Bubble Sorting Calibration Technology," *Chinese Journal of Electronics*, vol. 34, no. 1, pp. 125-136, 2025. doi: 10.23919/cje.2023.00.307.

[16] H. Fan, Z. Chen, T. Xu et al., "14-Bit SAR ADC with On-Chip Digital Bubble Sorting Calibration Technology," *Chinese Journal of Electronics*, vol. 34, no. 1, pp. 125–136, 2025, doi: 10.23919/cje.2023.00.307.

[17] M. Grossi, "Efficient and accurate analog voltage measurement using a direct sensor-to-digital port interface for microcontrollers and field-programmable gate arrays," *Sensors*, vol. 24, no. 3, pp. 873, 2024, doi: 10.3390/s24030873.

[18] T. Islam, M. S. Ahmed, and M. H. Rahman, "ADC non-linearity effects on low-voltage sensor measurements in embedded systems: a systematic review," *IEEE Sensors Journal*, vol. 24, no. 5, pp. 6781–6793, 2024, doi: 10.1109/JSEN.2024.3356129.

[19] Y. Wang, Z. Liu, and X. Chen, "Voltage reference stability and its impact on analog sensor accuracy in microcontroller-based systems," *Measurement*, vol. 225, pp. 114023, 2024, doi: 10.1016/j.measurement.2023.114023.

[20] F. Karimi, A. Bagheri, and M. R. Mosavi, "Water clarity assessment using TDS sensors: calibration, compensation, and performance evaluation across multiple platforms," *Environmental Monitoring and Assessment*, vol. 196, no. 3, p. 274, 2024, doi: 10.1007/s10661-024-12274-3.

[21] M. Hamidatou, N. Chellal, and M. Benbouzid, "A review on tds measurement techniques for water quality monitoring: accuracy, sensitivity, and practical considerations," *Water*, vol. 15, no. 11, pp. 2034, 2023, doi: 10.3390/w15112034.

[22] I. Essamlali, H. Nhaila, and M. El Khaili, "Advances in Machine Learning and IoT for Water Quality Monitoring: A Comprehensive Review," *Heliyon*, vol. 10, no. 6, p. e27920, Mar. 2024, doi: 10.1016/j.heliyon.2024.e27920.

[23] L. Rocher, E. Viala, and C. Roux, "Low-cost water quality sensors for IoT: a systematic review," *Sensors*, vol. 23, no. 10, p[. 4853, 2023, doi: 10.3390/s23104853.

[24] A. Rao, P. Kumar, and S. Sharma, "Comparative performance of low-cost microcontrollers for analog sensor interfacing in environmental monitoring applications," *Sensors and Actuators A: Physical*, vol. 358, pp. 114415, 2023, doi: 10.1016/j.sna.2023.114415.
[25] H. M. Forhad, M. R. Uddin, R. S. Chakrovorty, A. M. Ruhul, et al, IoT based real-time water quality monitoring system in water treatment plants (WTPs)," *Heliyon*, vol. 10, no. 23, 2024, doi: 10.1016/j.heliyon.2024.e40746.
[26] A. Junaidi, et al., "Measurement of Physical Parameters of Water Quality in Real-Time Based on Arduino," *Journal of Physics: Conference Series*, vol. 1751, no. 1, p. 012067, 2021, doi: 10.1088/1742-6596/1751/1/012067.
[27] M. Hidayatullah, S. Sofyan, P. A. Topan, T. Andriani, and N. Nurhairunnisah, "Monitoring system of water quality on hydroponic planting media using total dissolved solid (TDS) sensor based Arduino Uno R3," *Jurnal Ilmu Fisika| Universitas Andalas*, vol. 14, no. 2, pp. 108-115, 2022, doi: 10.25077/jif.14.2.108-115.2022.
[28] S. Laha, S. R. Pattanayak, and B. K. Pattnaik, "Advancement of environmental monitoring system using IoT and sensor: a comprehensive analysis," *AIMS Environmental Science*, vol. 9, no. 6, pp. 771–800, 2022, doi: 10.3934/environsci.2022044.
[29] D. Abimanyu, S. Sumarno, F. Anggraini, I. Gunawan, and I. Parlina, "Rancang bangun alat pemantau kadar ph, suhu dan warna pada air sungai berbasis mikrokontroller arduino [Design and construction of a tool to monitor pH, temperature and color levels in river water based on an Arduino microcontroller]," *Jurnal Pendidikan dan Teknologi Indonesia*, vol. 1, no. 6, pp. 235-242, 2021, doi: 10.52436/1.jpti.55.
[30] M. N. Nizam, H. Yuana, and Z. Wulansari, "Mikrokontroler ESP32 sebagai alat monitoring pintu berbasis web [ESP32 microcontroller as a web-based door monitoring tool]," *JATI (Jurnal Mahasiswa Teknik Informatika)*, vol. 6, no. 2, pp. 767-772, 2022, doi: 10.36040/jati.v6i2.5713
[31] S. Baco, R. Syarifuddin, S. Sajiah, M. A. Rusli, and M. D. Loilatu, "Rancang bangun alat monitoring ph air layak konsumsi berbasis mikrokontroller arduino uno pada depot air minum adipura 1 [Design and construction of a pH monitoring device for potable water based on an Arduino Uno microcontroller at the Adipura 1 drinking water depot]," *Jurnal Teknologi dan Komputer (JTEK)*, 3(02), 339-345, 2023, doi: 10.56923/jtek.v3i02.147.
[32] D. Chen, X. Cui, Q. Zhang, D. Li, W. Cheng, C. Fei, and Y. Yang, "A survey on analog-to-digital converter integrated circuits for miniaturized high resolution ultrasonic imaging system," *Micromachines*, vol. 13, no. 1, pp. 140, 2022, doi: 10.3390/mi13010140.
[33] A. H. El-Khozondar, E. H. Bayoumi, and A. A. E. A. Halim, "A smart energy monitoring system using esp32 microcontroller," *e-Prime - Advances in Electrical Engineering, Electronics and Energy*, vol. 9, pp. 100666, 2024, doi: 10.1016/j.prime.2024.100666.
[34] F. Almajed, F. Almajed, and Q. A. Al-Haija, "An IoT-enabled intelligent water quality monitoring system for tourist safety using machine learning," *Discover Electronics*, Springer Nature, vol. 2, no. 1, 2025, doi: 10.1007/s44291-025-00070-1.
[35] Y. H. Chang, F. C. Wu, and H. W. Lin, "Design and implementation of ESP32-based edge computing for object detection," *Sensors*, vol. 25, no. 6, pp. 1656, 2025, doi: 10.3390/s25061656.
[36] 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. 4, pp. 474–488, 2024, doi: 10.5614/j.eng.technol.sci.2024.56.4.4.
[37] M. Alselek, J. M. Alcaraz-Calero, J. Segura-Garcia, and Q. Wang, "Water iot monitoring system for aquaponics health and fishery applications," *Sensors*, vol. 22, no. 19, pp. 7679, 2022, doi: 10.3390/s22197679.
[38] C. Cohen-Manrique, S. Camacho-Leon, and J. L. Villa, "Emerging trends in IoT for aquatic systems: a systematic literature review," *Frontiers in Water*, vol. 7, pp. 1699240, 2025, doi: 10.3389/frwa.2025.1699240.
[39] C. Axiotidis, E. Konstantopoulou, and N. Sklavos, "A wireless sensor network IoT platform for consumption and quality monitoring of drinking water," *Discover Applied Sciences*, vol. 7, no. 1, pp. 15, 2024, doi: 10.1007/s42452-024-06384-1.
[40] A. Boussaha et al., "Development of an IoT-based multi-level system for real-time water quality monitoring in industrial wastewater," *Discover Water*, Springer, 2024, doi: 10.1007/s43832-024-00092-y.
[41] C. Axiotidis et al., "A wireless sensor network iot platform for consumption and quality monitoring of drinking water," *Discover Applied Sciences*, Springer, Dec. 2024, doi: 10.1007/s42452-024-06384-1.
[42] R. Moshawrab, M. Adda, A. Bouzouane, H. Ibrahim, and A. Raad, "IoT in water quality monitoring—are we really here?," *Sensors*, vol. 23, no. 2, p. 960, Jan. 2023, doi: 10.3390/s23020960.
[43] J. Tang, H. Lin, and Q. Tian, "Design and realization of a water quality monitoring system based on the internet of things," *Water Practice and Technology*, vol. 19, no. 9, pp. 3538–3554, Sep. 2024, doi: 10.2166/wpt.2024.222.
[44] S. Geetha and S. Gouthami, "Internet of things enabled real time water quality monitoring system," *Smart Water*, Springer, vol. 2, no. 1, pp. 1–19, 2017, doi: 10.1186/s40713-017-0005-y.
[45] O. V. Medagedara and M. H. Liyanage, "Development of an IoT-based real-time temperature and humidity monitoring system for factory electrical panel rooms," *Engineering Journal of Institute of Engineers Sri Lanka*, vol. 57, no. 1, pp. 21–30, 2024, doi: 10.4038/engineer.v57i1.7636.
[46] M. A. Al-Qurashi et al., "IoT-based real-time water quality monitoring system in water treatment plants (WTPs)," *Heliyon*, Elsevier, Nov. 2024, doi: 10.1016/j.heliyon.2024.e38459.
[47] O. Habboush et al., "Design of a smart hydroponics monitoring system using an esp32 microcontroller and the internet of things," *MethodsX*, vol. 11, p. 102418, Dec. 2023, doi: 10.1016/j.mex.2023.102418.
[48] N. A. Cloete, R. Malekian, and L. Nair, "Design of smart sensors for real-time water quality monitoring," *IEEE Access*, vol. 4, pp. 3975–3990, 2016, doi: 10.1109/ACCESS.2016.2592958.

[49] J. J. Correa-Quiroz, M. A. Toribio-Barrueto, and C. Castro-Vargas, "IoT system with ESP32 for smart drip irrigation and climate monitoring in greenhouses," *Emerging Science Journal*, vol. 6, no. 5, 2022, doi: 10.28991/ESJ-2022-06-05-01.

[50] S. D. Kalamaras, M. A. Tsitsimpikou, C. A. Tzenos, A. A. Lithourgidis, D. S. Pitsikoglou, and T. A. Kotsopoulos, "A Low-Cost IoT system based on the ESP32 microcontroller for efficient monitoring of a pilot anaerobic biogas reactor," *Applied Sciences*, vol. 15, no. 1, pp. 34, 2024, doi: 10.3390/app15010034.