Simulation of an IoT-Based Automated Fertilization and Irrigation System Using ESP32 on Wokwi

Septia Harliansyah, Eko Hariyanto

Abstract
Proper management of water and nutrients is essential for efficient plant cultivation, yet schedule-based irrigation and unmeasured fertilization may lead to waste. Control logic should therefore be tested before it is implemented on physical hardware. This study designs and evaluates irrigation and fertilization control rules for an IoT-based system through simulation on Wokwi using an ESP32 microcontroller. Soil moisture and pH are represented by virtual analog inputs, while the water pump and nutrient pump are represented by LED indicators. The water pump is controlled with hysteresis, turning on when moisture falls below 40% and off when it reaches or exceeds 60%. The nutrient pump is triggered when pH falls outside the 6.0–7.0 range as a preliminary control rule. Five input scenarios were tested, and output states were observed through the Serial Monitor. The results show that the system responded according to the control rules in all scenarios: the water pump turned on at 30.0% moisture, retained its state at 50.4%, and turned off at 65.6%, while the nutrient pump activated at pH 5.50 and above 7.0. These findings are limited to the consistency of control logic in simulation and do not represent the performance of physical sensors and actuators, IoT communication, or the effect on plant growth.
Keywords: *automated irrigation; fertilization; Internet of Things; ESP32; Wokwi; hysteresis; smart farming*

Septia Harliansyah
Master of Information Technology, Universitas Pembangunan Panca Budi, Indonesia
e-mail : septiaharliansyah@unsam.ac.id

Eko Hariyanto, Zulham Sitorus
e-mail : eko.hariyanto@dosen .pancabudi.ac.id ,
3rd International Conference Epicentrum of Economic Global Framework (ICEEGLOF)
Theme: Cross-Sector and Cross-Disciplinary Strategies for Achieving the SDGS
https://proceeding.pancabudi.ac.id/index.php/ICEEGLOF

Introduction

Proper plant management requires regulation of water and nutrients that matches the condition of the growing medium. Irrigation based solely on a schedule can cause water to be applied when the soil is still sufficiently moist, while unmeasured fertilization risks inefficient fertilizer use. The development of smart farming offers opportunities to support data-driven decision making through sensors, microcontrollers, and digital monitoring systems [1],[2]. However, the deployment of wireless sensor devices in the field often faces various practical constraints, ranging from communication reliability to the need for system validation before physical deployment. Before a system is deployed on physical devices, the control logic needs to be tested so that the responses produced match the input conditions. Therefore, the design of an integrated automated irrigation and fertilization system is an important solution for improving the efficiency of plant nutrient management [3].

The Internet of Things (IoT) enables devices to exchange information and supports remote monitoring. Wireless-capable microcontrollers can be used as processing units to read inputs and control outputs [4], [5], [6]. Previous studies have discussed automatic irrigation based on moisture as well as the application of control logic in smart farming [7], [8]. In addition, IoT applications have also been developed for monitoring environmental conditions, such as monitoring systems and early detection of air pollution [9]. Testing with physical devices requires components, assembly time, and calibration procedures. Electronic simulation can be used as an initial stage to check the consistency of control logic before a prototype is built.

This study uses Wokwi to test the irrigation and fertilization automation logic in a virtual environment. The official Wokwi hardware list includes support for the ESP32, whereas the ESP8266 is not listed as an officially supported microcontroller [10]. Therefore, the simulation in this manuscript uses the ESP32 as the virtual controller. Moisture and pH values are represented through analog inputs that can be varied to test the control rules; these values are not the result of physical sensor measurements. The simulation is not used to draw conclusions about plant growth, water flow efficiency, or actual fertilizer dosage.

The objective of this study is to design and evaluate irrigation and fertilization control rules through measurable input scenarios. The contribution of this study lies in the preparation of test scenarios that combine two virtual outputs—a water pump and a nutrient pump—and the application of moisture hysteresis to reduce excessively frequent pump state changes. The pH-based trigger is treated as a preliminary control rule, not as a direct measurement of nutrient content. The simulation results need to be verified through screenshots and Wokwi output logs before the manuscript is submitted to the proceedings.

Literature Review

1. Concepts of Smart Farming and the Internet of Things

Smart farming uses digital technology to help monitor cultivation conditions and manage resources precisely. In an IoT architecture, sensors or analog inputs provide environmental condition values, a microcontroller executes processing rules, and actuators take action based on those decisions[11]. In simulation-based research, virtual inputs can be used to test the software decision flow, but they do not replace field measurements or agronomic validation [1], [2], [9].

2. NodeMCU ESP8266 Microcontroller and Sensor Devices

The ESP32 is a microcontroller with wireless connectivity that is supported by the Wokwi simulator. In this study, two analog inputs are used to represent soil moisture and pH. The control outputs are represented by LEDs or digital indicators that show pump status. This representation is sufficient for testing program branching and ON/OFF state changes, but it does not model relay electrical characteristics, pump flow rate, physical sensor response, or soil chemical changes. The components used must conform to the official simulator list [10].

3. State of the Art

A summary of previous research on agricultural automation, microcontrollers, and IoT is presented in Table 1. This comparison is used to position the contribution of this study as a preliminary test of control logic through simulation.

Table 1. State of the Art

No	Authors & Year	Research Focus/Method	Key Findings/Results	Comparison with This Study
1	Herdianto & Fachri (2021)	Design of an automatic spinach watering system based on water content and nutrients using an ATmega16 microcontroller.	The watering system worked with a 100% success rate over 15 trials. Nutrient sensor testing has not been carried out	This study develops that system using an IoT-enabled NodeMCU ESP8266 and integrates simultaneous pH/nutrient sensor testing.
2	Adli et al. (2023)	A comprehensive review of the application of *Artificial Intelligence of Things* (AIoT) in the *smart agriculture* sector.	The convergence of AI and IoT (*AIoT*) improves the efficiency of environmental parameter monitoring and precision irrigation/fertilization automation.	Serves as the theoretical basis for developing *cloud*-based IoT architecture and remote monitoring.
3	Satria (2022)	Monitoring of air temperature and humidity parameters based on NodeMCU ESP8266.	NodeMCU ESP8266 proved to be stable and economical for acquiring environmental sensor data in *real-time*.	Used as the reference for selecting the NodeMCU ESP8266 microcontroller as the main processing unit.
4	Syahputra (2022)	Implementation of the *Fuzzy Sugeno* method in *smart farming* based on measurements of temperature and soil water content.	The intelligent algorithm successfully optimized plant watering duration based on environmental dynamics.	Illustrates the importance of precisely reading soil physical conditions for watering decisions.
5	Ananda et al. (2025)	Implementation of a *real-time* power status detection system based on ESP8266 as a *web server*.	The ESP8266 module was able to transmit device condition data stably through a web/IoT interface.	Serves as a reference for designing a *real-time* dashboard and IoT platform connectivity.
6	Ojha et al. (2021)	Simulation and validation of microcontroller-based irrigation controller logic before physical implementation.	Simulation speeds up the identification of *hardware-in-the-loop* logic errors by up to 80%.	Serves as the basis for validating the use of the Wokwi simulator for control function testing.
7	Pezol et al. (2020)	Design of an IoT-based smart irrigation and fertilization system using Fuzzy Logic for chili plants.	Water and nutrient efficiency increased significantly with adaptive control.	This study complements it with a simple hysteresis test in a simulation environment

4. Research Gap

Previous studies have demonstrated the use of microcontrollers, sensors, and control logic to support agricultural automation [5], [7], [8]. This study does not claim that simulation-based automation is superior to physical systems. The gap addressed is more limited, namely the need for documentation of repeatable test scenarios to verify the consistency of two-output control logic before hardware implementation.

The contribution of this study is the design and functional testing of irrigation and fertilization rules on the Wokwi simulator. The system uses moisture thresholds of 40% and 60% to control the water pump, and a pH range of 6.0–7.0 as a provisional trigger rule for the nutrient pump indicator. These thresholds are design parameters that must be adjusted to the

plant type and agronomic recommendations; pH cannot be used as a direct measurement of N, P, or K nutrient content.

Research Methodology

1. Research Flow

This study uses a simulation-based experimental approach. Testing was performed on a virtual circuit and program in Wokwi, not through physical hardware assembly. The research stages include control rule design, analog input modeling, execution of test scenarios, output recording, and analysis of response conformity. The research flow is shown in Figure 1.

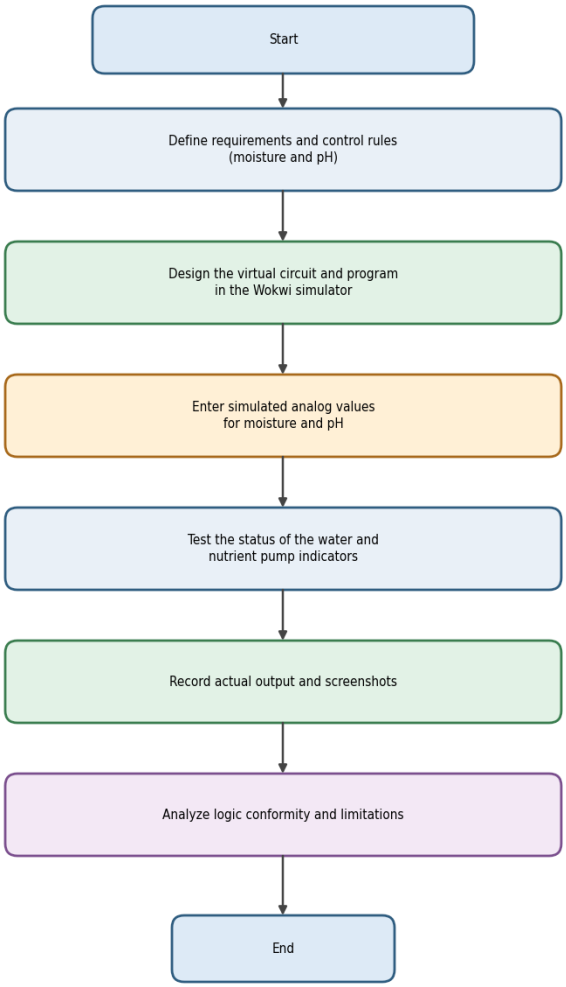


Figure 1. Flowchart of the Research Method

The first stage is to determine the system requirements and the threshold values to be tested. The second stage is to design the virtual circuit using an ESP32, two analog inputs, and output indicators for the water pump and nutrient pump. The third stage is to write the program and run it in Wokwi. The fourth stage is to vary the input values to run each scenario and to record the input values and output states from the Serial Monitor or the circuit indicators. The final stage is to compare the actual simulation output with the expected output. The testing does not include physical sensor calibration, measurement of water flow rate, fertilizer dose, or cloud latency unless those communication features are actually implemented and measured.

2. Simulation System Design

The simulation system design consists of virtual analog inputs, an ESP32 processing unit, virtual outputs representing pump status, and a Serial Monitor to record values and program decisions. This architecture is shown in Figure 2.

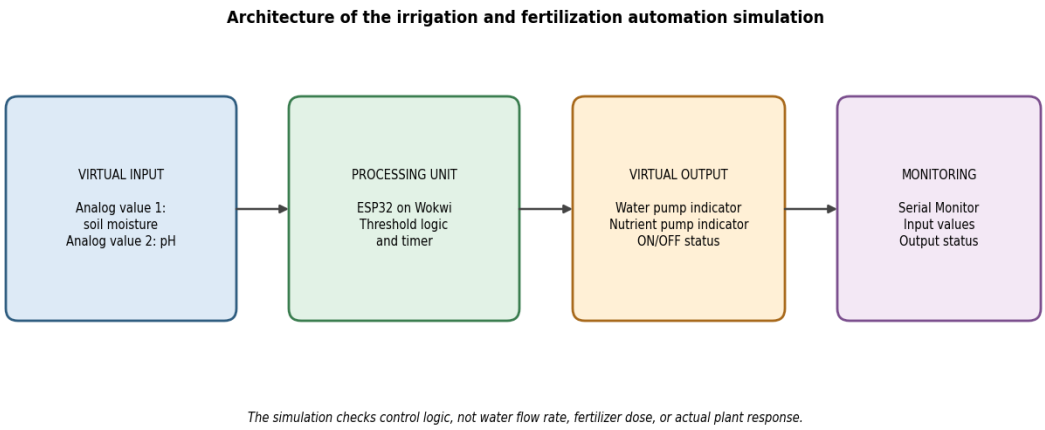


Figure 2. Hardware Block Diagram

In the virtual circuit, the first analog input represents soil moisture on a 0–100% scale, while the second input represents pH within the range defined in the program. The ESP32 runs the control logic and changes the state of the output indicators. The first indicator represents the water pump and the second indicator represents the nutrient pump. Because the sensor values are generated virtually, the results obtained only demonstrate the consistency of the software with respect to the given input values, not sensor accuracy or real pump performance.

3. Software Design

The program was developed using the Arduino IDE in C/C++ and run on the Wokwi simulator. Analog values are converted to the defined simulation range. For irrigation, the water pump is turned on when moisture is below 40%, turned off when moisture reaches or exceeds 60%, and keeps its previous state in the range from 40% to less than 60%. This hysteresis rule avoids state changes caused merely by an input value hovering around a single threshold. For fertilization, the nutrient pump indicator is triggered if pH is below 6.0 or above 7.0 for a duration defined by the program. This pH rule is used only as prototype logic; its application to real plants requires an agronomic basis and additional nutrient measurements.

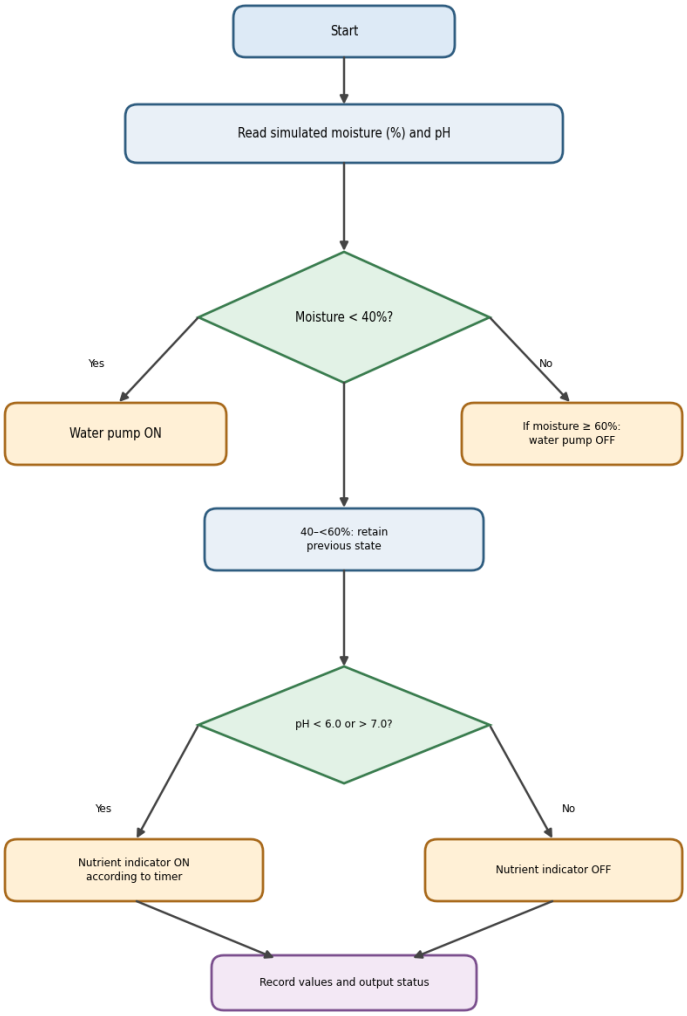


Figure 3. Program Flow Diagram (*Flowchart*)

Each test scenario is run by entering specific moisture and pH values. At a moisture of 30%, the water pump is expected to be ON; at 50%, the water pump state is retained; and at 65%, the water pump is expected to be OFF. For the pH rule, values of 5.5 and 7.5 trigger the nutrient pump indicator, whereas a value of 6.5 does not. These outputs are logical expectations based on the program rules and must be confirmed against the actual results from the simulator.

4. Testing and Data Collection Methods

Testing was designed using input scenarios that represent dry, transitional, and moist conditions, as well as pH below, within, and above the control range. Each scenario is recorded in a table containing the input values, the expected water pump state, the expected nutrient pump state, and the actual state after the simulation is run. Functional consistency is calculated by comparing the actual output with the expected output in each scenario. Once all scenarios have been run, the conformity percentage is calculated with the following formula: the number of matching outputs divided by the total number of scenarios, multiplied by 100%. This testing is not described as sensor calibration, physical actuator testing, or measurement of water/fertilizer efficiency.

Result

1. System Test Results

Testing was carried out to evaluate the response of the automated plant irrigation and fertilization system to changes in simulated soil moisture and acidity level (pH) values using Wokwi. The system uses two virtual actuators, namely a water pump and a nutrient pump, represented by LED indicators. Input values and actuator states were observed through the Serial Monitor.

The irrigation system is designed using the hysteresis method, in which the water pump is active when soil moisture is below 40% and turns off when moisture reaches or exceeds 60%. When moisture is between the two thresholds, the water pump state is retained. Meanwhile, the nutrient pump is triggered when the pH value is outside the 6.0–7.0 range.

Based on the five test scenarios carried out, the results are shown in Table 2.

Table 2. Test Results

No	Soil Moisture	pH	Water Pump	Nutrient Pump
1	30.0	6.53	ON	OFF
2	50.4	6.53	ON	OFF
3	65.6	6.53	OFF	OFF
4	50.7	5.5	OFF	ON
5	50.7	7.36	OFF	ON

Source: Wokwi simulation data, 2026.

Note: In the fifth test, the pH range of 7.83–8.20 comes from two Serial Monitor output lines that recorded the nutrient pump as ON. These data do not mean that both pH values were measured simultaneously. In the fourth test, the nutrient pump was also recorded as ON on two output lines. The pump activation duration has not been verified using dedicated timing records.

Discussion of Test Results

First test: low moisture

In the first test, a soil moisture of 30.0% and a pH of 6.53 resulted in the water pump being ON and the nutrient pump OFF. The moisture of 30.0% is below the 40% activation threshold, so the system activates the water pump. On the other hand, the pH is still within the defined range, so the nutrient pump is not activated. This result shows that the system responds to low-moisture conditions in accordance with the defined control rules.

Second test: moisture in the middle range

The second test produced a moisture of 50.4% and a pH of 6.53. The water pump remained ON, while the nutrient pump remained OFF. Although moisture had risen past the 40% activation threshold, it had not yet reached the 60% stop threshold. Therefore, the system retained the water pump state from the previous test. This response shows the application of the hysteresis mechanism, which can reduce repeated state changes when the moisture value is around the control threshold.

Third test: high moisture

In the third test, moisture reached 65.6% with a pH of 6.53. The system produced a water pump OFF and nutrient pump OFF state. Because moisture had exceeded the 60% stop threshold, the water pump was deactivated. The nutrient pump also remained off because the pH was still within the control range. This result shows that the system can stop irrigation based on the defined moisture threshold.

Fourth test: low pH

The fourth test used a moisture of 50.4% and a pH of 5.50. The observed results show the water pump OFF and the nutrient pump ON. A pH of 5.50 is below the lower limit of 6.0 and therefore meets the nutrient pump trigger condition. Meanwhile, the water pump remained OFF because moisture was between the 40% and 60% thresholds, with the previous state being OFF.

1. **Fifth test: high pH**

In the fifth test, the recorded pH value rose from 7.83 to 8.20, while moisture remained at 50.4%. The water pump state remained OFF and the nutrient pump was ON. This pH value is above the upper limit of 7.0 and therefore meets the nutrient pump trigger condition in the program used. This result shows that the system can respond to high-pH conditions. However, further testing is still needed to confirm the pump activation duration and to prevent repeated activation while the pH remains outside the range.

2. System Performance Evaluation

Based on the five scenarios tested, the system showed responses consistent with the control rules in all observed scenarios. In the first to third tests, the water pump state changed or was retained in accordance with the hysteresis mechanism. In the fourth and fifth tests, the nutrient pump was observed to be ON when the pH was below 6.0 or above 7.0.

Nevertheless, these results are still a preliminary evaluation of control logic function. The testing has not covered measurement of physical sensor accuracy, power consumption, IoT network response time, or the effectiveness of fertilization on plants. Therefore, the simulation results cannot be directly used to conclude improvements in plant growth or in the efficiency of water and fertilizer use.

There is also a limitation in the nutrient pump trigger mechanism. The code used in the refinement stage allows the pump to be triggered again after the previous activation duration ends if the pH is still outside the control range. This behavior needs to be evaluated so that the system does not supply nutrients repeatedly without adequate pauses or dose limits.

In addition, pH indicates the acidity or alkalinity of a medium, not the concentration of nutrients directly. Therefore, the use of pH as the sole basis for activating the nutrient pump in this study must be understood as a preliminary control rule for simulation, not as a validated agronomic fertilization recommendation.

3. Summary of Results

The simulation results show that the system is able to execute the basic irrigation rules and the nutrient pump trigger based on the given input values. The hysteresis mechanism retains the water pump state at intermediate moisture, while pH conditions outside the 6.0–7.0 range can trigger the nutrient pump. Thus, this simulation can be used to evaluate control logic before it is applied to physical devices.

However, additional testing is needed to validate the nutrient pump activation duration, repeated-trigger behavior, and IoT communication integration. The conclusions of the study are limited to the system's behavior in simulation and do not yet reflect physical device performance or the impact on plant growth.

Conclusion

Based on the results of the simulation of the automated plant irrigation and fertilization system using Wokwi, it can be concluded that the system is able to execute control logic based on soil moisture and acidity level (pH) parameters. In the irrigation tests, the water pump was active when moisture reached 30.0% and remained active when moisture rose to 50.4%. When moisture reached 65.6%, the water pump stopped operating. These results show that the hysteresis mechanism can retain the pump state based on the predefined activation and stop thresholds.

In the fertilization tests, the nutrient pump was observed to be active when the pH was below 6.0, namely 5.50, and when the pH was above 7.0, namely 7.83–8.20. These results show that the nutrient pump trigger logic can respond to pH conditions outside the defined range. However, the pump activation duration and repeated-trigger behavior still require further testing.

Overall, this simulation shows that the basic logic of irrigation automation and nutrient pump triggering can be applied to a microcontroller-based control system. This study is still limited to logic testing through simulation and does not yet cover validation of physical sensors, full IoT communication, or the effect of the system on plant growth. Future research is advised to test the system on physical devices, add online data monitoring, and develop a nutrient control mechanism that takes plant nutrient requirements into account so that the system can be applied more reliably in a smart farming environment.

References

[1] K. Jha, A. Doshi, P. Patel, and M. Shah, “A comprehensive review on automation in

agriculture using artificial intelligence," *Artif. Intell. Agric.*, vol. 2, pp. 1–12, 2019, doi: 10.1016/j.aiia.2019.05.004.

[2] T. Ojha, S. Misra, and N. Singh, "Wireless sensor networks for agriculture : The state-of-the-art in practice and future challenges," *Comput. Electron. Agric.*, vol. 118, pp. 66–84, 2015, doi: 10.1016/j.compag.2015.08.011.

[3] N. S. Pezol, R. Adnan, and M. Tajjudin, "Design of an Internet of Things (Iot) Based Smart Irrigation and Fertilization System Using Fuzzy Logic for Chili Plant," *IEEE Int. Conf. Autom. Control Intell. Syst.*, no. June, pp. 69–73, 2020.

[4] O. Friha, M. A. Ferrag, L. Shu, L. Maglaras, and X. Wang, "Internet of Things for the Future of Smart Agriculture: A Comprehensive Survey of Emerging Technologies," *IEEE/CAA J. Autom. Sin.*, vol. 8, no. 4, pp. 718–752, 2021, doi: 10.1109/JAS.2021.1003925.

[5] B. Satria, "IoT Monitoring Suhu dan Kelembaban Udara dengan Node MCU ESP8266," *sudo J. Tek. Inform.*, vol. 1, no. 3, pp. 136–144, 2022, doi: 10.56211/sudo.v1i3.95.

[6] S. Ananda, A. Dani, and B. Satria, "Implementasi Sistem Deteksi Kegagalan Daya Secara Real-Time Berbasis Internet of Things Menggunakan ESP8266 sebagai Web Server," *J. Komput. Teknol. Inf. Sist. Komput.*, vol. 4, no. 3, pp. 2309–2316, 2026.

[7] Herdianto, "Perancangan Sistem Penyiraman Tanaman Otomatis Berdasarkan Kadar Air dan Nutrisi Menggunakan Mikrokontroler," *METHOMIKA J. Manaj. Inform. Komputerisasi Akunt.*, vol. 6, no. Seminar of Social Sciences Engineering & Humaniora, pp. 174–186, 2021.

[8] Z. Syahputra, M. Iqbal, and M. I. Syarif, "Penerapan Metode Fuzzy Sugeno Pada Smart Farming Berdasarkan Pengukuran Suhu Dan Kadar Air," *Semin. Nas. Halal*, pp. 11–18, 2021.

[9] M. Hasanuddin and H. Herdianto, "Sistem Monitoring dan Deteksi Dini Pencemaran Udara Berbasis Internet Of Things (IOT)," *J. Comput. Syst. Informatics*, vol. 4, no. 4, pp. 976–984, 2023, doi: 10.47065/josyc.v4i4.4034.

[10] Wokwi, "Supported Hardware," Wokwi Documentation. [Online]. Available: https://docs.wokwi.com/getting-started/supported-hardware. Accessed: October 19, 2026.

[11] H. K. Adli *et al.*, "Recent Advancements and Challenges of AIoT Application in Smart Agriculture: A Review," *Sensors*, vol. 23, no. 7, pp. 1–22, 2023, doi: 10.3390/s23073752.