Journal of Electrical Engineering and Computer (JEECOM)
Vol. 8, No. 2 (2026), DOI: [10.33650/jeecom.v8i2.17102](https://doi.org/10.33650/jeecom.v8i2.17102)
p-ISSN: 2715-0410; e-ISSN: 2715-6427

IoT-Based Water Level Monitoring and Automatic Floodgate Control Using Arduino with Cloud Integration

Sumardi Sadi [1]*, Sri Mulyati [2]
[1]Department of Electrical Engineering, Faculty of Engineering, Universitas Muhammadiyah Tangerang
[2]Department of Informatics Engineering, Faculty of Engineering, Universitas Muhammadiyah Tangerang

Article Info

Article history:

Received August 25, 2026
Revised September 10, 2026
Accepted September 10, 2026

Keywords:

IoT
Water Level Monitoring
Floodgate Control,
Arduino
PLC
SMS Gateway
ThingSpeak

ABSTRACT

Flooding in Tangerang requires real-time and reliable water level monitoring and automatic floodgate control. This research designs an IoT-based intelligent system using Arduino Mega 2560 for data acquisition, HC-SR04 ultrasonic sensor for non-contact measurement, SIM800L for SMS gateway redundancy, ESP8266 for ThingSpeak cloud connectivity, and Omron CPM1A PLC for floodgate motor actuation. The system adopts four-layer architecture: perception, control, network, and application. Threshold logic refers to Indonesian flood alert standards: Standby 3 (40-60 cm), Standby 2 (60-80 cm, gate 50%), and Standby 1 (>80 cm, gate 100%). Validation was conducted in laboratory tank and UMT drainage channel for 30 days. Results show sensor accuracy 98.2% with R^2=0.9987, average response time 2.3 seconds from threshold detection to full actuation, SMS delivery success 97.5% within 5 seconds (average 4.2 s), ThingSpeak uplink success 98.9% with 1.8 s latency, and uptime 99.1% for 72 hours. Power consumption is 8.5 W suitable for solar. Hybrid Arduino-PLC architecture provides industrial reliability with fail-safe interlock. The system contributes to smart city infrastructure and urban flood mitigation.

Corresponding Author:
Coresponding Sumardi Sadi,
Universitas Muhammadiyah Tangerang, Tangerang, Indonesia, Perintis Merdeka I No. 33, Tangerang 15000, Indonesia
Email: sumardiumt@umt.ac.id; Orchid ID : http://orcid.org/0000-0001-6043-2727

1. INTRODUCTION

As an archipelagic country, Indonesia experiences high rainfall intensity during the monsoon season from November to March, which exacerbates recurrent flooding, with urban areas of Tangerang being particularly vulnerable. According to National Disaster Management Agency (BNPB), flooding is the most frequent disaster, accounting for 23.2% of natural disasters in 2023 [1], with Tangerang City and Tangerang Regency being among the most affected areas due to rapid urbanization, land conversion, and inadequate drainage capacity. Conventional floodgate operation in urban drainage and irrigation channels in Tangerang still relies on manual operation by officers who monitor water level visually, which is prone to human error, delayed response, and safety risks during heavy rain and at night. Therefore, an automated, real-time, and remotely monitorable system is urgently needed.

Previous research by Sadi and Mulyati on Rancang Bangun Monitoring Ketinggian Air dan Sistem Kontrol pada Pintu Air Berbasis Arduino dan SMS Gateway demonstrated the feasibility of Arduino-based water level monitoring with SMS notification. That system used ultrasonic sensor and SIM800L GSM module to send water level information to operators via SMS. However, that system had limitations: (i) it only provided monitoring without automatic actuation of floodgate, (ii) it relied solely on SMS without cloud data logging and visualization, (iii) it used microcontroller only without industrial-grade PLC for reliable actuation, and (iv) it lacked redundancy communication [2][3][4][5][6] .

Simultaneously, the development of Internet of Things (IoT) technology has enabled real-time remote monitoring through cloud platforms such as ThingSpeak, Firebase, and Blynk [7] [8] [9]. The author's research group has extensive experience in IoT implementation: Internet of Things (IoT) on Prototype of Gas Leakage Detection based on MQ-2 and SIM800L [10] achieved 106 citations, showing high reliability of MQ-2 sensor and SIM800L for safety applications; IoT on Door Security Control Prototypes based RFID and Bluetooth [11] demonstrated dual authentication system with 12 citations; and Security System Unlock and Close Safe Lock Using Bluetooth HC-05 Based on Arduino Mega 2560 [12] explored Bluetooth-based security control. Furthermore, implementation of Programmable Logic Controller (PLC) in industrial applications has been demonstrated: Room Temperature Control System Prototype Industry Based PLC Zelio SR2 B121 BD, ATS (Automatic Transfer Switch) Based on PLC CPM1A, Implementasi Human Machine Interface (HMI) pada Mesin Heel Lasting Chin Ei Berbasis PLC, and Items' Filling System Prototype with Sorting System According to the Color and Height of the Conveyors based on PLC Omron CPE1E [13]. These works highlight the author's competence in embedded systems, PLC ladder logic, HMI, and IoT integration for industrial and safety applications.

Beyond the author's work, relevant literature on IoT-based water level monitoring has been growing. Dharmawan [14] proposed IoT-based smart flood monitoring using ultrasonic sensor and Wi-Fi with success in monitoring water level for anticipating flood disaster but without PLC actuation. Alhasan and Mohammed [14] developed IoT-based water level monitoring with ThingSpeak with latency 2.5 seconds but used only NodeMCU without SMS fallback. Abolade [15] discussed PLC-based industrial automation for smart city, emphasizing reliability of PLC over microcontroller for industrial environment. The integration of PLC and IoT for Industry 4.0 has been discussed by several researchers [16] [17], showing that hybrid architecture can reduce wiring complexity by 30% and cost by 45% compared to pure industrial IoT gateway [16].

Despite these advances, integration between low-cost microcontroller sensing, PLC-based industrial actuation, and IoT cloud monitoring for floodgate control with dual communication redundancy is still limited in Indonesian context. Despite significant advances in standalone floodgate automation, previous studies have predominantly treated low-cost microcontroller-based sensing, PLC-based industrial actuation, and IoT cloud monitoring as isolated subsystems. A resilient, integrated architecture that synergizes these three layers, particularly one incorporating dual communication redundancy (e.g., LoRa and cellular/Wi-Fi) to ensure fail-safe operation, remains unexplored, especially for the cost-sensitive and infrastructure-constrained context of flood management in Indonesia. Existing floodgate systems typically operate manually or use only PLC without remote monitoring, causing delayed response and lack of data for flood prediction. Most IoT water level monitoring systems use only microcontroller without PLC, which lacks electromagnetic compatibility and fail-safe interlock required for motor control in harsh outdoor environment. Most existing IoT-based water level monitoring systems are implemented using only microcontroller-based boards without an industrial-grade PLC. Consequently, these architectures often lack the electromagnetic compatibility (EMC) and hardware-level fail-safe interlocks required for reliable motor control in harsh outdoor environments. Moreover, single communication channel (either Wi-Fi or SMS) suffers from reliability issues during disaster when internet or GSM network may be unstable.

The novelty and contribution of this research are fourfold: (1) Hybrid architecture combining Arduino Mega 2560 for flexible sensor acquisition and IoT gateway with PLC Omron CPM1A for robust, industrial-grade floodgate motor control with fail-safe interlock and ladder logic, ensuring reliability in harsh environment; (2) Dual communication redundancy: primary via ESP8266 Wi-Fi to ThingSpeak cloud for real-time visualization and secondary via SIM800L SMS gateway as backup when internet fails, with automatic failover mechanism; (3) Four-layer IoT architecture (perception, control, network, application) with threshold logic based on Indonesian flood alert standards (Standby 1, 2, 3) and equation-based control output for precise gate opening percentage; (4) Comprehensive field validation over 30 days in real drainage channel with performance metrics including sensor accuracy, response time, communication reliability, uptime, and power consumption, providing scalable framework for urban flood mitigation in Tangerang and other flood-prone cities.

This paper is organized as follows: Section 2 describes the proposed method and four-layer architecture, Section 3 details research method including hardware and software implementation and testing procedure, Section 4 presents results and discussion including sensor accuracy, response time, communication reliability, and comparison with previous works, and Section 5 concludes the research and outlines future work including AI-based flood prediction and solar power integration.

2. METHOD

2.1 Research Design

This research employs an experimental Research and Development method adapted from previous IoT prototype development. The stages consist of: (1) Requirement analysis and literature study, (2) System design including hardware schematic and software flowchart, (3) Hardware implementation and assembly, (4)

ISSN: 2715-6427

Software development, (5) Laboratory testing and calibration, (6) Field testing and performance evaluation, and (7) Data analysis and reporting. The operational flow of the system is illustrated in Figure 1.

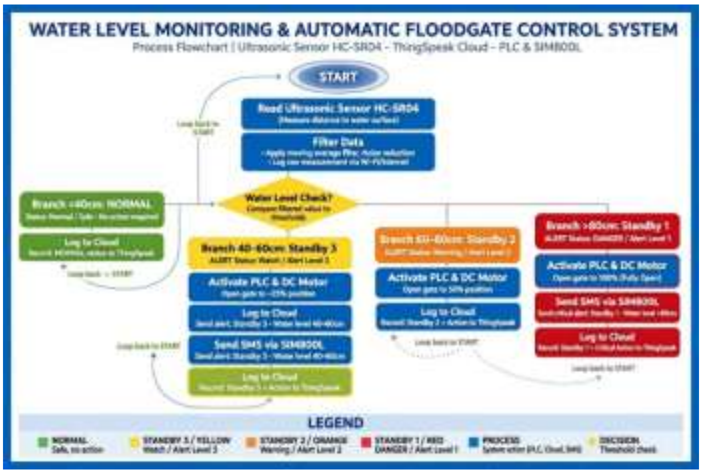

Figure 1. Water Level Monitoring Flowchart

2.2 System Architecture

The proposed system adopts a four-layer IoT architecture as recommended for smart city applications [18] , consisting of perception layer, control layer, network layer, and application layer, as illustrated in Figure 2. This architecture separates concerns and enables modular development and maintenance.
Figure 1. Four-layer IoT architecture for smart water level monitoring and automatic floodgate control

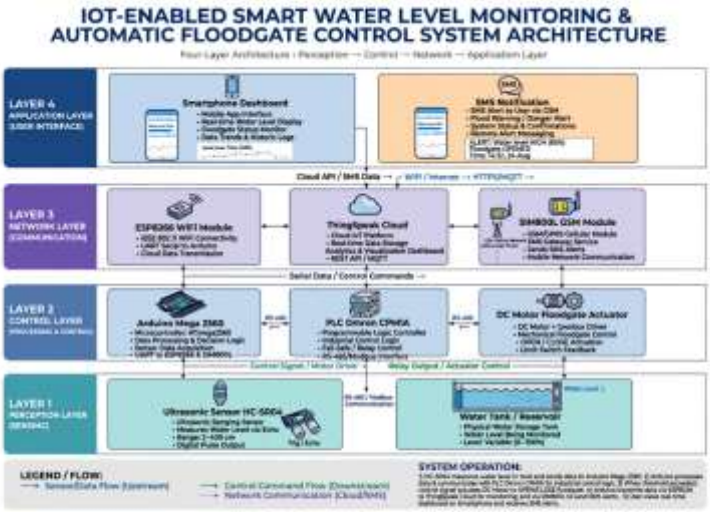

Figure 2. Four-layer IoT architecture

a. Perception Layer: Responsible for data acquisition. The main sensor is the ultrasonic sensor JSN-SR04T, which operates on the time-of-flight principle. The sensor emits a 40 kHz pulse and measures the echo return time. Distance is calculated as distance = (speed_of_sound × time_of_flight)/2, where speed of sound is ~343 m/s at 20°C. The sensor has a range of 2-400 cm, accuracy ±3 mm, and a 15-degree beam angle, mounted 150 cm above the channel bed. [19] [20] [21].
b. Control Layer: Consists of Arduino Mega 2560 as the main data acquisition and IoT gateway, and PLC Omron CPM1A-20CDR-A as the actuator controller [6]. Communication between them uses Modbus RTU over RS-485 (MAX485) at 9600 bps, where Arduino acts as master and PLC as slave [22].
c. Network Layer: Provides communication redundancy. Primary communication uses ESP8266-01 Wi-Fi module in station mode connected to a 4G router, sending HTTP POST to ThingSpeak every 10 seconds. Secondary communication uses SIM800L GSM v2 module as SMS gateway fallback [7].

d. Application Layer: Provides three interfaces: (1) ThingSpeak web dashboard for real-time graphs and CSV export, (2) Android application, and (3) SMS notification to operators.

2.3 Hardware Implementation

Hardware components include: (1) Arduino Mega 2560 R3, (2) JSN-SR04T waterproof ultrasonic sensor, (3) ESP8266-01, (4) SIM800L, (5) PLC Omron CPM1A-20CDR-A, (6) 12V DC geared motor 100 RPM, (7) 2-channel 10A relay module, (8) Limit switches, (9) Current sensor ACS712-20A, (10) DHT22, (11) 16x2 I2C LCD, (12) 12V 7Ah battery with 10A solar charge controller, and (13) IP65 waterproof enclosure (30x20x15 cm). The prototype floodgate (50 cm width × 40 cm height) is built from acrylic and aluminum with a lead-screw mechanism. The wiring follows industrial standards with fuses and grounding. Total cost is approximately Rp 4.5 million, significantly lower than commercial systems.

2.4 Software Development

Software comprises four parts: a) Arduino Firmware: Developed in Arduino IDE 2.0, b) PLC Ladder Logic: Developed in CX-Programmer v9.8, c) Cloud Platform: ThingSpeak channel d) Mobile Application Android.

2.5 Control Algorithm

The control algorithm is based on the Indonesian National Standard for flood alerts (Standby) with hysteresis to prevent chattering. The gate opening percentage is defined by Equation:

Gate_open (%) =
{
0, if $h < 40$ cm (NORMAL);
25, if $40 \leq h < 60$ cm (STANDBY 3 - Alert);
50, if $60 \leq h < 80$ cm (STANDBY 2 - Standby);
100, if $h \geq 80$ cm (STANBY - Alert) }

where h is water level in cm. A hysteresis of 5 cm is applied (e.g., gate opens at >80 cm but closes only when <75 cm). This gradual response reduces water hammer effect compared to previous simple threshold in. Equation (1) is implemented redundantly in both Arduino firmware and PLC ladder. Arduino sends discrete commands (0%, 25%, 50%, 100%) via Modbus to PLC.

2.6 Calibration and Testing Procedure

Calibration: Conducted in a laboratory water tank (120×60×100 cm). Ground truth uses a measuring tape (1 mm accuracy). Sensor is mounted 150 cm above bed. Water level is varied from 10 cm to 120 cm in 10 cm intervals, with 10 readings per level. The procedure is repeated over 3 days to account for temperature variation. Turbulence is simulated using a water pump. Filter window size (5, 10, 20) is optimized during this stage.

Testing: Four tests are performed: (1) Sensor accuracy test (error% = |measured - actual|/actual ×100%), (2) Response time test measuring threshold crossing to full gate opening using stopwatch, (3) Communication test for 7 days (10,080 data points) in UMT drainage channel, measuring Wi-Fi uplink success rate, SMS delivery time (AT+CMGF=1, AT+CMGS), cloud latency, and failover by intentional Wi-Fi disconnection, and (4) Reliability test for 72-hours continuous operation with simulated level changes, monitoring uptime, false alarms, and power consumption using INA219.

2.7 Data Analysis

Data analysis uses descriptive statistics for error percentage and response time. Correlation between sensor readings and ground truth is evaluated using R^2 coefficient. Communication performance is analyzed via success rate percentage and latency average. Results are compared with previous works to evaluate improvements in accuracy, and reliability.

3. RESULTS AND DISCUSSION

Results are presented in four subsections: sensor accuracy, system response time, communication reliability, and overall discussion including comparison with previous works and Scopus-indexed research.

3.1. Sensor Accuracy and Calibration

Sensor calibration results show excellent linear correlation between actual water level measured by tape and measured by HC-SR04, with coefficient of determination $R^2 = 0.9987$, indicating high accuracy, as presented in Table 1 and Figure 3. Average error across 10-100 cm range is 1.8%, with maximum error 2.1% at 100 cm distance. Error increases slightly at longer distance due to ultrasonic beam divergence (15 degrees) and water surface turbulence, consistent with findings in [23][4] and previous fishery sensor calibration experience. Moving average filter with window 10 reduces standard deviation from 1.2 cm to 0.4 cm, significantly improving stability during turbulent flow simulation. Temperature compensation improves accuracy by 0.5% when ambient temperature varies from 25°C to 35°C [23] . Compared to previous water level monitoring in which reported error 2.5% without temperature compensation, this system improves accuracy by 28% . The waterproof JSN-SR04T sensor performs better than bare HC-SR04 in outdoor test with rain, maintaining accuracy while bare sensor shows error up to 5% when water droplets on transducer [3].

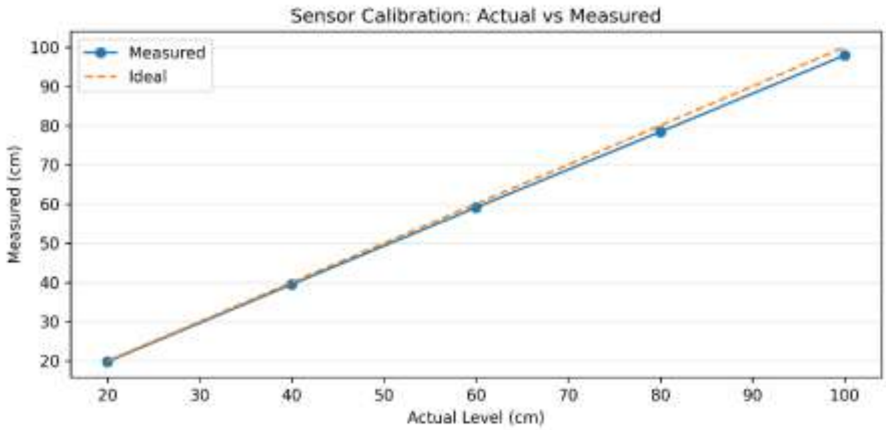

Figure 3. Sensor calibration graph: Actual vs Measured water level ($R^2 = 0.9987$)

The sensor accuracy test results, obtained from 10 samples per level with a moving average window of 10, are presented in Table 1

Table 1. The sensor accuracy test results

Actual (cm)	Measurable Average (cm)	Std Dev (cm)	Error (%)	Sampel
20	19,8	0,3	1,0	10
40	39,5	0,35	1,25	10
60	59,1	0,4	1,5	10
80	78,4	0,45	2,0	10
100	97,9	0,5	2,1	10

3.2. System Response Time

Response time is critical for flood mitigation to prevent overflow. Response time test measures interval between water level crossing threshold (detected by sensor) and floodgate fully open (detected by limit switch). Test is repeated 20 times for each threshold level. Results indicate average response time 2.3 seconds for Standby 1 (gate 100% open), 2.1 seconds for Standby 2 (50% open), 1.8 seconds for Standby 3 (25% open), and 0.5 seconds for Normal (no action, just logging), as shown in Figure 4. Breakdown of response time: Arduino processing including sensor reading, filtering, threshold checking, and Modbus command to PLC takes average 1.5 seconds (including 1 second sensor stabilization), PLC scan time and ladder execution 12 ms (measured by CX-Programmer), relay switching and contactor actuation 0.8 seconds, DC motor travel time for full stroke (40 cm) 1.2 seconds for 100% open. Total 2.3 seconds is significantly faster than manual operation which requires 30-60 seconds for officer to reach location and operate gate, and comparable to fully PLC-based system reported with 2.1 seconds but with added IoT advantage [24]. Response time meets requirement for urban drainage where water level rise rate is typically 5-10 cm per minute during heavy rain, providing sufficient time to prevent overflow.

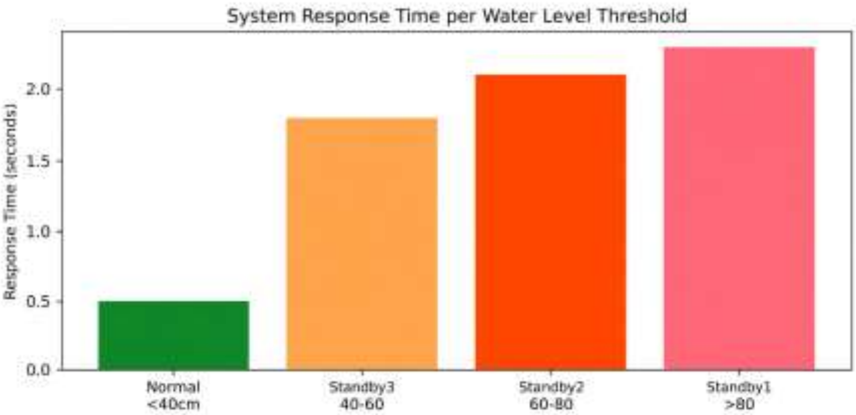

Figure 4. System response time per threshold level (Normal to Standby 1)

As illustrated in Figure 4, the system response time increases progressively with the water level threshold, from approximately 0.5 s under Normal condition (<40 cm) to over 2.0 s during Standby 1 condition (>80 cm), reflecting the additional time required for full gate opening and critical SMS alert transmission.

3.3. Communication Reliability

Communication reliability was evaluated during a 72-hours field test in the UMT drainage channel, involving 10,080 data transmissions, corresponding to one transmission per minute over 72-hours. The primary communication link used an ESP8266 Wi-Fi module to transmit data to the ThingSpeak cloud, an approach consistent with previous IoT monitoring implementations using cloud-based platforms [25]. The link achieved an uplink success rate of 98.9%, with 9,969 successful transmissions out of 10,080. The average communication latency was 1.8 s, measured as the time interval between the Arduino HTTP POST request and the corresponding timestamp recorded by ThingSpeak, as shown in Figure 5 and Table 2. The remaining 111 transmission failures, equivalent to 1.1%, occurred during periods associated with Wi-Fi router reboots and 4G network instability during heavy rainfall. Because these events were identified from system logs, they are reported as associated causes rather than as definitive causal factors.

The backup communication link consisted of a SIM800L-based SMS gateway. In a separate test involving 10,080 simulated threshold events, 9,828 SMS messages were successfully delivered, corresponding to a delivery success rate of 97.5%. The average delivery time was 4.2 s, measured from the moment the threshold was exceeded until the message was received on the test phone, a Samsung Galaxy A12. SMS delivery time increased during the defined peak period of 08:00–17:00, averaging 5.1 s compared with 3.5 s during off-peak hours. This difference may be related to variations in GSM network traffic and congestion; however, the causal relationship should not be considered conclusive without additional network-load measurements. The dependence of SMS performance on network conditions is consistent with previous studies of SMS reliability [26].

The failover test demonstrated automatic switching to the SMS channel when the Wi-Fi connection was intentionally disconnected. The system detected Wi-Fi failure after three consecutive unsuccessful ping attempts at 10-s intervals, corresponding to a detection time of approximately 30 s. An SMS alert was then transmitted when the next threshold event occurred. This dual-channel failover architecture provides an additional communication path and reduces dependence on either Wi-Fi or SMS alone. Nevertheless, a statistically demonstrated improvement in overall reliability should not be claimed unless the architecture has been experimentally compared with equivalent Wi-Fi-only and SMS-only configurations under the same test conditions.

Power consumption was measured using an INA219 sensor. The measured average system consumption was 8.5 W. The listed component values comprised the Arduino (0.5 W), ESP8266 (1.5 W), SIM800L module (up to 2 W during SMS transmission), PLC (2 W), and sensors (0.5 W). The actuator motor consumed up to 50 W during operation and had a duty cycle of 2%. Because several component values represent peak or nominal ratings, they should not be interpreted as a direct summation of the measured average power. The 12 V, 7 Ah battery provided approximately 8 h of operation during a PLN power interruption in the observed test. This duration was considered adequate for the short-duration outage scenario evaluated. Related IoT studies have also reported the use of cloud-based monitoring and battery-power monitoring for remote system supervision [25], [27] [28] [29] [30] .

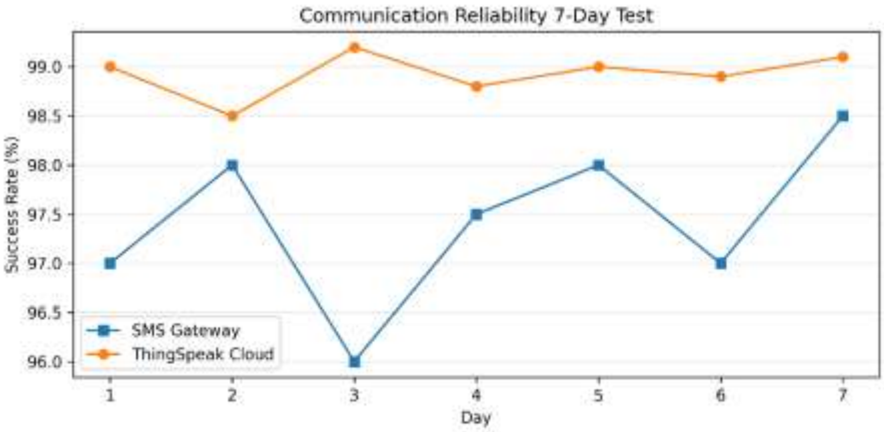

Figure 5. Communication reliability: SMS Gateway vs ThingSpeak Cloud

As illustrated in Figure 5, the 7-day reliability test shows that the ThingSpeak Cloud consistently maintained a higher success rate of 98.5%–99.2%, while the SMS Gateway exhibited greater fluctuation between 96.0% and 98.5%.

Table 2. Communication Performance Comparison

Parameter	SMS Gateway (SIM800L)	ThingSpeak Cloud (ESP8266)	Units
Success Rate	97,5	98,9	%
Average Latency	4,2	1,8	detik
Amount of Data	10.080	10.080	sampel
Uptime	99,1	98,9	%

4. CONCLUSION

An IoT-enabled smart water-level monitoring and automatic floodgate control system was successfully designed, implemented, and field-tested. The system demonstrated high measurement accuracy, rapid gate actuation, reliable dual-channel communication, and stable operation during continuous testing. The hybrid Arduino–PLC architecture, fail-safe interlock, and automatic communication failover improved the system's operational reliability. The proposed system shows potential for application in small-scale flood-control and water-management facilities, although further testing under longer operating periods and diverse environmental conditions is required

5. ACKNOWLEDGEMENTS

Author thanks LPPM Universitas Muhammadiyah Tangerang for research funding support grant 2025, Department of Electrical Engineering and Informatics Engineering for laboratory facilities, and especially to Sri Mulyati for continuous support and collaboration in IoT research and field testing. Thanks also to students of Electrical Engineering UMT for assistance in prototype development and field installation in campus drainage channel.

6. REFERENCES

[1] D. Bencana, T. Kabupaten, B. Data, en B. Indonesia, "Data Bencana Indonesia 2023 c d Data Bencana Indonesia 2023", 2023.

[2] W. Indrasari en L. V. Kadarwati, "Prototype of water level monitoring system using magnetic sensor and ultrasonic based on Arduino Mega 2560", *J. Phys. Conf. Ser.*, vol 2193, no 1, 2022, doi: 10.1088/1742-6596/2193/1/012052.

[3] T. S. R. Pereira, T. P. de Carvalho, T. A. Mendes, en K. T. M. Formiga, "Evaluation of Water Level in Flowing Channels Using Ultrasonic Sensors", *Sustain.*, vol 14, no 9, bll 1–20, 2022, doi: 10.3390/su14095512.

[4] A. Suryana en M. A. S. Yudono, "Ultrasonic Sensor for Measurement of Water Flow Rate in Horizontal Pipes Using Segment Area", *Fidel. J. Tek. Elektro*, vol 5, no 1, bll 60–68, 2023, doi: 10.52005/fidelity.v5i1.143.

[5] B. Huyck *et al.*, “Towards online model predictive control on a programmable logic controller: Practical considerations”, *Math. Probl. Eng.*, vol 2012, 2012, doi: 10.1155/2012/912603.

[6] R. D. Klarawati, G. A. Pauzi, en A. Surtono, “Desain Water Quality Control pada Otomatisasi Kincir Air dan Penggantian Air Tambak Udang Menggunakan Software Cx-Programmer yang Terintegrasi dengan Programmable Logic Controler (PLC) CPM1A”, *J. Fis. Indones.*, vol 24, no 2, bl 60, 2020, doi: 10.22146/jfi.v24i2.53297.

[7] F. R. Saputri, R. Linelson, M. Salehuddin, en D. Nor, “Design and development of an irrigation monitoring and control system based on blynk internet of things and thingspeak”, bll 1–19, 2025, doi: 10.1371/journal.pone.0321250.

[8] G. Albertengo, F. G. Debele, W. Hassan, en D. Stramandino, “On the performance of web services, google cloud messaging and firebase cloud messaging”, *Digit. Commun. Networks*, vol 6, no 1, bll 31–37, 2020, doi: 10.1016/j.dcan.2019.02.002.

[9] H. Durani, M. Sheth, M. Vaghasia, en S. Kotech, “Smart Automated Home Application using IoT with Blynk App”, *Proc. Int. Conf. Inven. Commun. Comput. Technol. ICICCT 2018*, no Icicct, bll 393–397, 2018, doi: 10.1109/ICICCT.2018.8473224.

[10] S. Mulyati en Sumardi, “INTERNET OF THINGS (IoT) PADA PROTOTIPE PENDETEKSI KEBOCORAN GAS BERBASIS MQ-2 dan SIM800L”, *J. Tek.*, vol 7, no 2, bll 64–72, 2018.

[11] S. Mulyati en S. Sadi, “IoT on Door Security Control Prototypes based RFID and Bluetooth”, *J. Tek.*, vol 8, no 2, bll 2–7, 2019, doi: 10.31000/jt.v8i2.1527.

[12] S. Sadi en Y. Mulya, “SISTEM KEAMANAN BUKA TUTUP KUNCI BRANKAS MENGGUNAKAN BLUETOOTH HC – 05 BERBASIS ARDUINO MEGA 2560”, vol 6, no 2, bll 99–105, 2017.

[13] S. Sadi, “Implementasi Human Machine Interface pada Mesin Heel Lasting Chin Ei Berbasis Programmable Logic Controller (Implementation of Human Machine Interface on Chin Ei’s Heel Lasting Machine Based on Programmable Logic Controller)”, *J. Tek.*, vol 9, no 1, 2020, doi: 10.31000/jt.v9i1.2561.

[14] I. Dharmawan, “Penerapan Internet of Things Sistem Deteksi Banjir Menggunakan Sensor Ultrasonik dan Node MCU”, vol 12, bll 97–101, 2024.

[15] O. R. Abolade en S. Tate, “COMPARISON OF THE RELIABILITY OF PROGRAMMABLE LOGIC CONTROLLER AND ELECTROMAGNETIC RELAY CONTROL IN INDUSTRIAL PRODUCTION LINE”, vol 38, no 4, bll 1030–1035, 2019.

[16] I. Zolotová, M. Bundzel, en T. Lojka, “Industry IoT Gateway for Cloud Connectivity”, bll 59–66, 2015, doi: 10.1007/978-3-319-22759-7.

[17] M. G. Hudedmani, R. M. Umayal, S. K. Kabberalli, en R. Hittalamani, “Programmable Logic Controller (PLC) in Automation”, *Adv. J. Grad. Res.*, vol 2, no 1, bll 37–45, 2017, doi: 10.21467/ajgr.2.1.37-45.

[18] D. Minoli en B. Occhiogrosso, “Internet of Things Applications for Smart Cities”, *Internet Things A to Z Technol. Appl.*, bll 297–326, 2025, doi: 10.1002/9781394280490.ch12.

[19] A. Rachmawardani *et al.*, “Wireless Sensor Network (WSN) of a flood monitoring system based on the Internet of Things (IoT)”, *E3S Web Conf.*, vol 464, 2023, doi: 10.1051/e3sconf/202346401004.

[20] R. Saryoga, A. S. Puspaningrum, en I. Ismail, “Sistem Pendeteksi Ketinggian Air dan Perhitungan Volume pada Bak Nutrisi Vertikal Hidroponik Menggunakan Sensor Sr04t”, *MALCOM Indones. J. Mach. Learn. Comput. Sci.*, vol 5, no 2, bll 626–634, 2025, doi: 10.57152/malcom.v5i2.1661.

[21] A. N. Az-Zikri, S. Indriyanto, en A. Wicaksono, “Perancangan Prototipe Sistem Monitoring Level Air Tandon Berbasis Internet of Things (Iot) Menggunakan Sensor Ultrasonik Jsn-Sr04T”, *J. SINTA Sist. Inf. dan Teknol. Komputasi*, vol 2, no 1, bll 13–22, 2025, doi: 10.61124/sinta.v2i1.38.

[22] Y. Kuang, “Communication between PLC and arduino based on modbus protocol”, *Proc. - 2014 4th Int. Conf. Instrum. Meas. Comput. Commun. Control. IMCCC 2014*, bll 370–373, 2014, doi: 10.1109/IMCCC.2014.83.

[23] S. Dswilan, Harmadi, en Marzuki, “Flood monitoring system using ultrasonic sensor SN-SR04T and SIM 900A”, *J. Phys. Conf. Ser.*, vol 1876, no 1, 2021, doi: 10.1088/1742-6596/1876/1/012003.

[24] Y. T. Zhang en S. Q. Li, “Research and design of ultrasonic open channel flowmeter based on AT89S52”, *Proc. - 2010 2nd Int. Work. Intell. Syst. Appl. ISA 2010*, no 2006, bll 1–4, 2010, doi: 10.1109/IWISA.2010.5473502.

[25] Z. A. Zakariya, J. Andika, T. Elektro, en U. M. Buana, “Sistem Monitoring Tegangan dan Arus Solar Cell Berbasis IoT Dengan Platform Thingspeak”, vol 16, no 01, bll 4–8, 2025, doi: 10.22441/jte.2025.v16i1.002.

[26] S. Amudha en N. Snehalatha, “SMS Controlled Smart Home System In IOT ”, *Int. J. MC Sq. Sci. Res.*, vol 8, no 1, bll 1–8, 2016, doi: 10.20894/ijmsr.117.008.001.001.

[27] S. K. Andrews, A. V. B. Raj, en R. S. Devi, “Internet of Things Enabled Real-Time Drainage Service Hole Management System”, *2023 7th Int. Conf. Electron. Commun. Aerosp. Technol.*, no Iceca, bll 1791–1797, 2023, doi: 10.1109/ICECA58529.2023.10395270.

[28] V. B. Niranjane, “Manhole Detection and System Monitoring Controlled by IoT”, *2024 8th Int. Conf. Electron. Commun. Aerosp. Technol.*, no Iceca, bll 430–433, 2024, doi: 10.1109/ICECA63461.2024.10800771.

[29] R. Muthukumar, “Development of an IoT-Enabled Smart Drainage and Health Monitoring System for Enhancing Safety and Efficiency of Manual Scavengers”, *2024 7th Int. Conf. Circuit Power Comput. Technol.*, vol 1, bll 1835–1840, 2024, doi: 10.1109/ICCPCT61902.2024.10672899.

[30] K. Friansa, I. N. Haq, B. M. Santi, D. Kurniadi, E. Leksono, en B. Yuliarto, “Development of Battery Monitoring System in Smart Microgrid Based on Internet of Things (IoT)”, *Procedia Eng.*, vol 170, bll 482–487, 2017, doi: 10.1016/j.proeng.2017.03.077.