

Design and Performance Evaluation of an ESP32-Based IoT Trainer Kit with Telegram Integration for Vocational Learning

Widya Juliani Putri¹ , Ryan Fikri^{1*} , Zulwisli¹ , Sartika Anori¹ 

¹ Universitas Negeri Padang, Padang, Indonesia

*Corresponding Author: ryanfikri@ft.unp.ac.id

Abstract - The rapid development of Internet of Things (IoT) technology requires practical learning media that can help vocational students understand the integration of sensors, actuators, microcontrollers, and remote communication systems. This study aims to design and evaluate an ESP32-based IoT trainer kit integrated with Telegram as a remote monitoring and control platform for microcontroller and IoT learning. The development process followed a waterfall model consisting of requirement analysis, system design, implementation, and functional testing. The trainer kit uses an ESP32 microcontroller as the central controller and integrates several input modules, including an HC-SR04 ultrasonic sensor, DHT22 temperature and humidity sensor, passive infrared sensor, light-dependent resistor, MQ-2 gas sensor, and RC522 RFID module. The output modules consist of a 16x2 LCD, LED indicators, buzzer, servo motor, relay, and solenoid door lock. Functional testing showed that all modules operated according to the designed logic. The RFID module successfully read cards at a distance of 0–3 cm, the ultrasonic sensor triggered servo movement when the object distance was less than 6 cm, the passive infrared sensor produced an output change from approximately 0 V to 3.1 V when motion was detected, and the MQ-2 sensor activated the buzzer when the gas reading exceeded the threshold value of 800. The system also displayed sensor data on the LCD and delivered event notifications through Telegram. These results indicate that the proposed trainer kit is feasible as an interactive learning medium for ESP32-based IoT practice.

Keywords - ESP32, Internet of Things, Telegram, Trainer Kit, Vocational Learning.

I. INTRODUCTION

The rapid development of Internet of Things (IoT) technology has transformed the way physical devices are monitored, controlled, and integrated through internet-based communication. IoT enables sensors, actuators, embedded controllers, and cloud or messaging platforms to exchange data in real time, making it widely applicable in smart homes, industrial monitoring, environmental sensing, and educational laboratories [1]. In the context of technical and vocational education, this development requires learning media that can help students understand not only microcontroller programming, but also the integration of hardware, software, networking, and remote control mechanisms [2].

Vocational learning in electronics and informatics is strongly associated with practical activities. Students are expected to understand abstract concepts such as sensor acquisition, actuator control, digital communication, and embedded programming through direct experimentation. However, conventional learning media often separate these topics into independent modules, causing students to learn sensors, actuators, and communication systems without seeing how they operate as a complete IoT system [3]. Therefore, an integrated trainer kit is needed to provide a more realistic learning environment where students can observe input processing, decision making, output control, and remote notification in one compact platform [4].

Several studies have reported the development of IoT-based learning media and trainer kits to support practical learning in control systems, embedded systems, and microcontroller courses [5]. These learning tools are generally designed to improve student understanding through project-based and hands-on activities. Nevertheless, many existing trainer kits still emphasize local control or single-sensor demonstrations and do not fully expose students to remote monitoring and control using internet-based platforms [6]. This limitation becomes important because modern IoT competence requires students to understand how devices communicate with users through online services, application interfaces, or messaging platforms [7].

DOI: <https://doi.org/10.24036/voteteknika.v%vi%i.138186>

Received : 2025-05-02
Revised : 2026-05-21
Accepted : 2026-06-18
Published : 2026-06-26



For all articles published in VOTETEKNIKA
<http://ejournal.unp.ac.id/index.php/voteknika>, © copyright is
retained by the authors. This is an open-access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.

ESP32 is one of the most suitable microcontrollers for educational IoT development because it combines processing capability, programmable input/output pins, and built-in Wi-Fi connectivity in a low-cost development board [8]. Compared with conventional microcontroller boards that require additional communication modules, ESP32 allows students to develop internet-connected prototypes more efficiently. Its compatibility with various sensors and actuators also makes it appropriate for a multi-module trainer kit that includes environmental sensing, motion detection, gas detection, access control, and output actuation [9].

In addition to the microcontroller platform, the communication interface is an important part of an IoT trainer kit. Telegram can be used as a practical medium for IoT learning because it supports bot-based interaction, remote notifications, and command-based control through an application that is already familiar to many users [10]. Through Telegram integration, students can observe how sensor data and system events are transmitted from the physical trainer kit to a remote user interface. This approach strengthens the learning process because students can connect embedded programming with real-world remote monitoring and control scenarios [11].

A multi-sensor trainer kit also provides broader learning opportunities than a single-function prototype. The HC-SR04 ultrasonic sensor can be used to introduce distance measurement and threshold-based control, while the DHT22 sensor can demonstrate temperature and humidity monitoring [12]. The passive infrared sensor supports motion detection, the light-dependent resistor introduces automatic light response, and the MQ-2 gas sensor demonstrates safety monitoring based on gas or smoke detection [13]. Furthermore, the RC522 RFID (Radio Frequency Identification) module can be used to introduce access-control concepts, while output devices such as LEDs (Light Emitting Diodes), buzzers, servo motors, relays, solenoid locks, and LCDs allow students to observe how sensor inputs are converted into physical responses [14].

Based on these considerations, this study develops an ESP32-based IoT trainer kit integrated with Telegram for vocational learning. The proposed trainer kit combines multiple sensor inputs, actuator outputs, local display, and remote notification in one acrylic-based learning platform. Unlike simple IoT demonstrations that focus on only one sensor or actuator, the proposed system is intended to function as an integrated instructional medium for learning microcontroller programming, sensor interfacing, actuator control, and Telegram-based remote communication [15].

The main contributions of this study are as follows:

1. Designing and implementing an ESP32-based IoT trainer kit that integrates multiple input modules, including HC-SR04, DHT22, PIR, LDR, MQ-2, and RC522 RFID;
2. Developing an output and response system consisting of LCD (Liquid Crystal Display), LED indicators, buzzer, servo motor, relay, and solenoid door lock;
3. Integrating the trainer kit with Telegram to support remote notification and control; and
4. Evaluating the functional performance of the trainer kit through sensor, actuator, voltage, and communication testing.

II. RESEARCH METHOD

A. Research Design

This study adopted a waterfall-based development model to design and implement an ESP32-based Internet of Things (IoT) trainer kit integrated with Telegram. The waterfall model was selected because the development process required sequential stages, beginning with requirement analysis and followed by system design, implementation, and testing. This model is suitable for hardware-oriented educational media development because each design stage can be verified before the next stage is executed [16].

The development process consisted of four main stages. The first stage was requirement analysis, which identified learning needs, hardware modules, software functions, and expected system responses. The second stage was system design, which defined the input, processing, output, and communication architecture of the trainer kit. The third stage was implementation, in which the circuit, mechanical layout, firmware, and Telegram bot interaction were developed. The final stage was testing, which evaluated the operation of each sensor, actuator, display, and Telegram-based notification and control function.

B. Requirement Analysis

Requirement analysis was conducted to define the functional characteristics of the trainer kit as a practical learning medium for microcontroller and IoT courses. Based on the identified learning needs, the trainer kit was required to help students understand the relationship between sensor input, microcontroller processing, actuator output, and remote communication in real time. Practical learning media of this type are important because IoT competence requires students to connect embedded programming with physical sensing, output control, and internet-based interaction [17].

The main requirements of the trainer kit were formulated as follows. First, the system had to use an ESP32 microcontroller as the central controller for reading sensor data and controlling output devices. Second, the trainer kit had to support remote monitoring and control through the internet using Telegram. Third, the input modules had to include common IoT sensors, namely an HC-SR04 ultrasonic sensor, DHT22 temperature and humidity sensor, PIR (Passive Infrared) motion sensor, LDR light sensor, MQ-2 gas sensor, and RFID RC522 module. Fourth, the system had to display sensor data locally through an LCD and send selected events as Telegram notifications. Fifth, the output modules had to include LED indicators, a buzzer, relay, solenoid door lock, and servo motor to represent different actuator responses. Finally, the system had to be arranged on an acrylic-based trainer panel with a clear component layout to support classroom demonstration and practical use.

C. System Architecture

The trainer kit was designed using an input-process-output architecture. This architecture is commonly used in embedded and IoT learning systems because it clearly separates data acquisition, microcontroller processing, actuator response, and user communication [18]. The input block consists of sensors and identification modules that collect environmental and user-related data. The processing block is handled by the ESP32

microcontroller, which reads input signals, applies programmed decision logic, controls the actuators, updates the local display, and communicates with Telegram through a Wi-Fi connection. The output block consists of devices that provide visual, audio, mechanical, and notification responses.

The input block includes the HC-SR04 ultrasonic sensor for distance measurement, DHT22 for temperature and humidity monitoring, PIR sensor for motion detection, LDR sensor for light intensity detection, MQ-2 sensor for gas or smoke detection, and RFID RC522 for card-based access identification. Data from these modules are transmitted to the ESP32 for further processing. The output block includes a 16x2 LCD for displaying information, LED indicators for visual status, a buzzer for alarm signals, a servo motor for movement demonstration, and a relay-connected solenoid door lock for access-control simulation. Telegram functions as the remote communication interface, allowing users to receive notifications and send commands to the trainer kit [19].

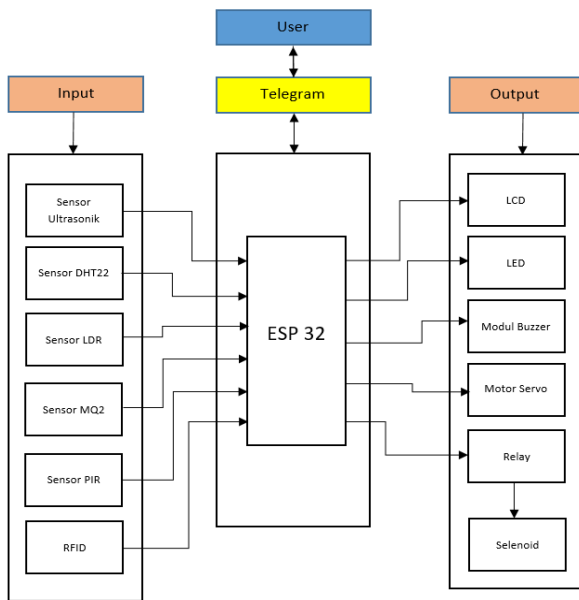


Fig. 1. Block Diagrams Block diagram of the ESP32-based IoT trainer kit with Telegram integration.

D. Hardware Design

The hardware design was developed by integrating the input, processing, output, power, and communication modules into a single trainer kit. Table 1 summarizes the main hardware components used in the prototype.

Table 1
Main hardware components of the ESP32-based IoT trainer kit

Module	Component	Function
Controller	ESP32 development board	Main controller for sensor reading, actuator control, and Wi-Fi communication
Distance sensing	HC-SR04 ultrasonic sensor	Measures object distance and triggers servo response
Environmental sensing	DHT22	Measures temperature and humidity
Motion sensing	PIR sensor	Detects movement in the surrounding area

Light sensing	LDR module	Detects bright or dark conditions
Gas sensing	MQ-2 sensor	Detects gas or smoke levels
Access identification	RFID RC522	Reads RFID cards for access-control simulation
Local display	16x2 LCD	Displays sensor readings and system status
Visual indicator	LED	Indicates system output or light-control condition
Alarm output	Buzzer	Provides warning signal for gas or event detection
Mechanical actuator	Servo motor	Demonstrates movement based on sensor input
Switching actuator	Relay module	Controls the solenoid door lock
Door lock simulator	Solenoid door lock	Simulates access opening and locking
Mechanical frame	Acrylic panel	Supports organized component placement for learning practice

The ESP32 was selected because it provides built-in Wi-Fi connectivity and sufficient input/output pins for connecting multiple sensors and actuators. These characteristics make ESP32 suitable for compact IoT prototypes and educational trainer kits that require wireless communication and multi-device interfacing [20]. The HC-SR04 ultrasonic sensor was connected to digital pins for trigger and echo signals, while the DHT22, PIR, LDR, MQ-2, and RFID RC522 were connected according to their respective signal requirements. The RFID RC522 communicated with the ESP32 through the SPI interface. Output devices were connected to GPIO pins, where the servo motor was driven using a PWM (Pulse Width Modulation) signal, the buzzer and LED were controlled digitally, and the relay module was used to switch the solenoid door lock.

The hardware was arranged on an acrylic panel to make the circuit easy to observe during practical activities. The panel layout was designed so that each component could be identified clearly by students, which is important for trainer kits because visible wiring and modular placement support step-by-step learning and troubleshooting [21]. The rear side of the panel contained the wiring connections between the ESP32, sensors, actuators, power supply, and display module.

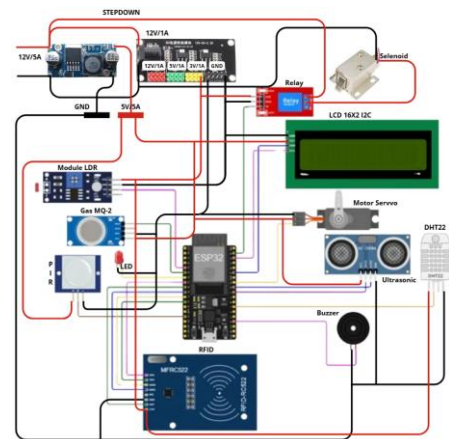


Fig. 2. Overall circuit schematic of the ESP32-based IoT trainer kit.

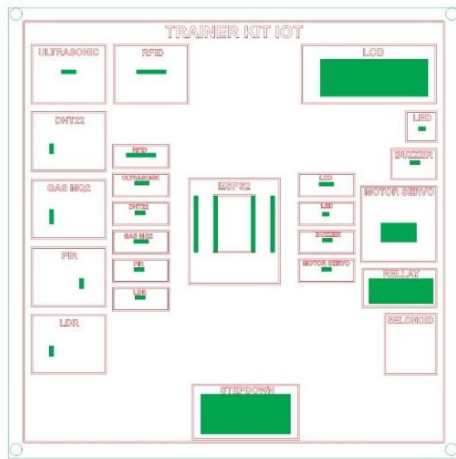


Fig. 3. Mechanical layout design of the IoT trainer kit panel.

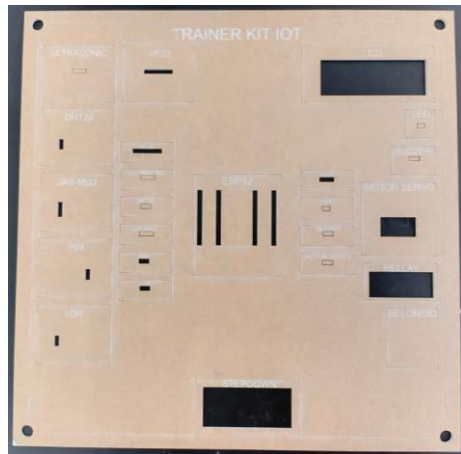


Fig. 4. Acrylic cutting result for the IoT trainer kit panel.



Fig. 5. Component wiring on the rear side of the trainer kit.

E. Software and Telegram Bot Design

The software was developed using Arduino IDE with C/C++ programming. The firmware was responsible for initializing the ESP32, connecting to the Wi-Fi network, configuring the Telegram bot, reading all sensor inputs, processing decision logic, activating output devices, updating the LCD, and sending notifications through Telegram. Arduino-based development was used because it provides broad library support for ESP32, sensor interfacing, actuator control, and messaging-based IoT applications [22].

The Telegram bot was used as the remote communication interface between the trainer kit and the user. When certain conditions were detected, such as motion, gas, RFID access, or a sensor threshold event, the ESP32 sent a notification to the user through Telegram. The bot also allowed command-based interaction, enabling users to request information or control selected outputs remotely. Messaging-based IoT interfaces such as Telegram bots are useful for educational prototypes because they enable students to observe event-driven communication without developing a separate mobile application [23].

The main firmware workflow can be described as follows. First, the ESP32 connects to the configured Wi-Fi network and Telegram bot service. Second, the system continuously reads input data from the ultrasonic sensor, DHT22, PIR, LDR, MQ-2, and RFID RC522. Third, the firmware compares the sensor values with predefined conditions or thresholds. Fourth, the ESP32 activates the corresponding output, such as LED, buzzer, servo motor, relay, or LCD message. Fifth, if the event is configured as a remote notification, the system sends a message to Telegram.

F. Testing Procedure and Evaluation Metrics

Testing was conducted to verify whether each module operated according to the designed function. The evaluation focused on functional response, voltage compatibility, sensor detection behavior, actuator response, local display output, and Telegram notification. Functional and electrical testing are necessary in IoT trainer kit development because the device must be safe, stable, and understandable for repeated use in practical learning environments [24].

First, input module testing was conducted for each sensor and identification module. The RFID RC522 was tested by placing a card near the reader at different distances to determine the readable range. The ultrasonic sensor was tested by placing an object at selected distances and observing the servo response when the measured distance crossed the programmed threshold. The DHT22 sensor was tested by observing temperature and humidity readings on the LCD and through Telegram requests. The PIR sensor was tested by comparing output voltage and system response under motion and no-motion conditions. The LDR sensor was tested under bright and dark conditions to verify automatic LED control. The MQ-2 sensor was tested by observing the buzzer response when the gas reading exceeded the programmed threshold.

Second, output module testing was conducted to verify the response of the LED, buzzer, servo motor, relay, solenoid door lock, and LCD. The servo motor was evaluated based on its ability to move according to the ultrasonic threshold condition. The relay and solenoid were tested to verify the access-control simulation triggered by the RFID module. The LCD was evaluated based on its ability to display sensor readings and system messages.

Third, Telegram communication testing was conducted to verify the ability of the trainer kit to send notifications and respond to user commands. The test included motion notification, gas warning, RFID access notification, and sensor information delivery. A successful Telegram communication test was defined as an event in which the correct notification was received by the user after the corresponding sensor condition occurred. This criterion was used to evaluate whether

the IoT communication layer could support real-time learning activities and remote observation [25].

Fourth, voltage testing was conducted to compare measured operating voltages with the nominal voltage requirements of each component. This test ensured that the ESP32, sensors, display, actuators, and communication modules operated within acceptable voltage ranges. The voltage test also served as a basic verification that the trainer kit wiring and power distribution were suitable for stable operation. The evaluation metrics used in this study are summarized in Table 2.

Table 2
Testing parameters and evaluation metrics

Test category	Tested component	Evaluation parameter	Success criterion
RFID access test	RFID RC522	Readable distance and access response	Valid card is detected and access output responds correctly
Distance test	HC-SR04 and servo	Object distance and servo movement	Servo moves when the distance is below the programmed threshold
Environment test	DHT22	Temperature and humidity reading	Data appear on LCD and can be requested through Telegram
Motion test	PIR sensor	Output voltage and notification	Motion is detected and Telegram notification is sent
Light test	LDR and LED	Bright/dark response	LED changes state according to light condition
Gas test	MQ-2 and buzzer	Gas threshold and alarm response	Buzzer activates when reading exceeds the threshold
Output test	LCD, LED, buzzer, servo, relay, solenoid	Functional response	Output device responds according to programmed logic
Communication test	Telegram bot	Notification and command response	Correct message is delivered to Telegram
Voltage test	All modules	Measured operating voltage	Voltage remains within safe operating range

This testing structure was used to determine whether the trainer kit met the functional requirements as a learning medium for ESP32-based IoT practice. The detailed results of the sensor tests, output responses, Telegram notifications, and voltage verification are presented in the next section.

III. RESULTS AND DISCUSSION

A. Prototype Implementation

The ESP32-based IoT trainer kit was implemented as an integrated learning platform that combines sensor input, microcontroller processing, actuator output, local display, and Telegram-based remote communication. The final prototype was arranged on an acrylic panel so that each component could be observed clearly during practical learning activities. The input modules consisted of an HC-SR04 ultrasonic sensor, DHT22 temperature and humidity sensor, PIR motion sensor, LDR light sensor, MQ-2 gas sensor, and RC522 RFID module. The output modules consisted of a 16x2 LCD, LED indicator, buzzer, servo motor, relay, and solenoid door lock.

The ESP32 acted as the main controller for acquiring sensor data, applying decision logic, controlling output devices, updating the LCD, and sending messages through Telegram. This configuration demonstrates the basic structure of an IoT system, where physical events are captured by sensors, processed by embedded firmware, and converted into actuator responses or remote notifications [26]. For learning purposes, the trainer kit provides an observable platform where students can study digital input, analog input, actuator control, access simulation, and messaging-based communication in one device [27].



Fig. 6. Final appearance of the ESP32-based IoT trainer kit.

B. RFID Access Testing

RFID testing was conducted to verify the ability of the RC522 module to read RFID cards and trigger the access-control output. The card was placed near the reader at several distances. The test showed that the RFID reader could detect the card at a distance of 0 cm to approximately 3 cm. When the distance exceeded 3 cm, the card was no longer detected reliably. The module operated at approximately 3.2 VDC and communicated with the ESP32 through the SPI interface.

When a valid RFID card was detected, the system displayed an access message on the LCD and activated the relay or solenoid output according to the programmed logic. When the card was not detected or the distance was beyond the readable range, the access output was not activated. This result indicates that the RFID module can be used to demonstrate short-range identification and access-control logic in an IoT trainer kit. Although the readable range is limited, this condition is

acceptable for an educational access-control simulation because the card must be intentionally brought near the reader [28].

Table 3
RFID access test result

Test parameter	Observed result
RFID module	RC522
Communication interface	SPI
Operating voltage	3.2 VDC
Readable distance	0–3 cm
Response for valid card	Access accepted, relay or solenoid activated
Response for unreadable card	Access not activated
Learning function	RFID-based access-control simulation

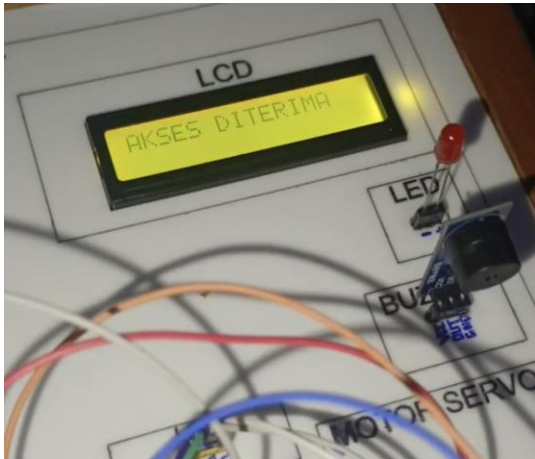


Fig. 7. LCD display when the RFID card is accepted.

C. Ultrasonic Sensor and Servo Response Testing

The ultrasonic sensor was tested to verify distance-based detection and servo response. The HC-SR04 sensor was connected to the ESP32 through GPIO (General Purpose Input/Output) 5 as the trigger pin and GPIO 18 as the echo pin. The measured operating voltage of the ultrasonic module was approximately 4.6 VDC. The programmed threshold was set at 6 cm. When an object was detected at a distance of less than 6 cm, the servo motor moved to approximately 30°. When the object distance was greater than 6 cm, the servo returned to its initial position at 0°.

This behavior confirms that the trainer kit can demonstrate threshold-based control using distance measurement. In an educational context, this module helps students understand how a sensor value can be converted into an actuator response through conditional programming. The use of a distance threshold also introduces the concept of decision logic in embedded systems, where output behavior depends on measured input values [29].

Table 4

Ultrasonic sensor and servo response test	
Test condition	System response
Object distance < 6 cm	Servo moves to approximately 30°
Object distance > 6 cm	Servo returns to 0°
Sensor operating voltage	4.6 VDC
Trigger pin	GPIO 5
Echo pin	GPIO 18
Learning function	Distance-based actuator control



Fig. 8. LCD display of DHT22 and ultrasonic sensor readings.

D. Temperature and Humidity Monitoring Using DHT22

The DHT22 sensor was tested to verify environmental monitoring on the trainer kit. The sensor was connected to the ESP32 and operated at approximately 3.2 VDC. The temperature and humidity readings were displayed on the LCD and could also be delivered to the user through Telegram commands. The test showed that the DHT22 module produced stable readings during operation and could be used to demonstrate real-time environmental data monitoring.

Although the test did not compare the DHT22 reading with a calibrated thermohygrometer, the observed behavior was sufficient for functional verification of the trainer kit. In this study, the main objective of the DHT22 module was to demonstrate digital sensor acquisition, local display, and remote data reporting through an IoT communication interface [30]. For stronger measurement validation, future testing should compare DHT22 readings with a calibrated reference instrument to calculate measurement error.

Table 5
DHT22 functional test result

Test parameter	Observed result
Sensor module	DHT22
Measured operating voltage	3.2 VDC
Output data	Temperature and humidity
Local display	Data shown on LCD
Remote communication	Data can be sent through Telegram
Learning function	Environmental monitoring and data reporting

E. PIR Motion Detection Test

The PIR sensor was evaluated to determine its response to motion. The sensor output was connected to GPIO 13 of the ESP32, and the measured supply voltage was approximately 4.6 VDC. During the no-motion condition, the output voltage was close to 0 V. When motion was detected, the output voltage increased to approximately 3.1 VDC. The system then displayed a motion message on the LCD and sent a notification to the user through Telegram.

The clear voltage difference between no-motion and motion conditions shows that the PIR sensor can function as a digital event detector. This module is useful for introducing motion-based IoT applications because students can observe how movement in the physical environment changes the GPIO input state and triggers both local and remote responses [31]. The Telegram notification also demonstrates event-driven communication, in which a sensor condition automatically generates a message to the user.

Table 6
PIR motion detection test result

Test condition	Output voltage	System response
No motion detected	Approximately 0 V	No motion notification
Motion detected	Approximately 3.1 VDC	LCD message and Telegram notification
Supply voltage	Approximately 4.6 VDC	Sensor active
ESP32 pin	GPIO 13	Digital input reading



Fig. 9. LCD display when motion is detected by the PIR sensor.

F. LDR-Based Light Control Test

The LDR sensor was tested to verify automatic light-control behavior under bright and dark conditions. The sensor was connected to GPIO 34, and the measured operating voltage was approximately 3.1 VDC. When the environment was dark, the measured output voltage was approximately 0.4 VDC and the system activated the LED. Under bright conditions, the sensor output increased and the LED was turned off according to the programmed logic.

This result shows that the LDR module can be used to demonstrate analog sensing and threshold-based output control. Students can observe how changes in light intensity affect the sensor output and how the ESP32 uses this signal to control an actuator. Therefore, this module supports learning activities related to analog input, environmental sensing, and automatic lighting control [32].

Table 7
LDR light-control test result

Test condition	Observed sensor behavior	Output response
Dark condition	Output voltage approximately 0.4 VDC	LED turns on
Bright condition	Output voltage increases	LED turns off
ESP32 pin	GPIO 34	Analog input reading
Learning function	Automatic light-control simulation	

G. MQ-2 Gas Detection and Buzzer Response Testing

The MQ-2 sensor was tested to evaluate gas or smoke detection and alarm activation. The sensor was supplied with approximately 4.5 VDC, and its analog output was connected to GPIO 35 of the ESP32. Under normal conditions, the sensor output remained in the normal range and the buzzer was inactive. When gas or smoke was detected, the sensor output increased to approximately 3.5 VDC. In the programmed logic, the buzzer was activated when the sensor reading exceeded the threshold value of 800.

This result confirms that the MQ-2 module can demonstrate safety monitoring in an IoT learning context. The combination of sensor reading, threshold comparison, buzzer activation, LCD display, and Telegram notification helps students understand how an IoT system can provide an early warning for hazardous conditions [33]. However, MQ-2 readings are affected by calibration, gas concentration, and environmental conditions; therefore, future tests should use controlled gas concentration levels to obtain more quantitative performance data.

Table 8
MQ-2 gas detection test result

Test condition	Observed value	System response
Normal condition	Sensor output in normal range	Buzzer inactive
Gas or smoke detected	Output voltage approximately 3.5 VDC	Buzzer active
Programmed threshold	≥ 800	Alarm triggered
Operating voltage	Approximately 4.5 VDC	Sensor active
ESP32 pin	GPIO 35	Analog input reading

H. Telegram Notification and Remote Monitoring Test

Telegram communication was tested to verify that the trainer kit could deliver event-based notifications and support remote monitoring. The system was configured to send messages when selected events occurred, such as motion detection, gas detection, RFID access, and sensor information requests. The test showed that Telegram could receive notifications from the ESP32 after the corresponding sensor condition was detected.

This result indicates that the communication layer worked according to the intended function. Telegram integration allows the trainer kit to demonstrate a practical IoT scenario without requiring students to develop a separate mobile application. In vocational learning, this approach is useful because students can directly observe the relationship between sensor events, embedded processing, internet communication, and user notification [34].

Table 9
Telegram communication test result

Event or command	Expected Telegram response	Test result
Motion detected by PIR	Motion notification received	Successful
Gas detected by MQ-2	Gas warning notification received	Successful
Valid RFID access	Access notification received	Successful
Sensor data request	Sensor information sent to user	Successful
Learning function	Remote monitoring and notification	Functional



Fig. 10. Telegram notification display from the ESP32-based IoT trainer kit.

I. Voltage Verification of Trainer Kit Components

Voltage testing was conducted to verify that each module received an operating voltage close to its required range. The measured voltages showed that the ESP32, sensors, LCD, LED, buzzer, servo motor, relay, and RFID module operated within acceptable voltage ranges for functional testing. This verification is important because unstable supply voltage can affect sensor readings, actuator responses, and communication reliability [35].

Table 10
Voltage test result of trainer kit components

No.	Component	Nominal voltage based on datasheet	Measured voltage during testing
1	ESP32	3.3 V and 5 V	3.3 V and 5 V
2	HC-SR04 ultrasonic sensor	5 V	4.6 V
3	DHT22	3.3–5 V	3.2 V
4	PIR sensor	5 V	4.6 V
5	LDR module	3.3–5 V	3.1 V
6	MQ-2 gas sensor	5 V	4.5 V
7	16x2 LCD with I2C	5 V	4.6 V
8	3 mm LED	2–3 V	2.6 V
9	Buzzer	3.3–5 V	3.3 V
10	Servo motor	4.8–5 V	4.7 V
11	3 V relay module	3 V	2.9 V
12	RFID RC522	2.5–3.3 V	3.2 V

Most modules showed measured voltages slightly below their nominal supply values, such as 4.6 V for the HC-SR04 and PIR, 4.5 V for the MQ-2, and 4.7 V for the servo motor. These values are still close to their operating requirements and did not prevent the modules from functioning during the test. The relay module operated at approximately 2.9 V and successfully switched the solenoid door lock based on RFID conditions. This confirms that the wiring and power

distribution of the trainer kit were sufficient for prototype-level operation.

J. Overall System Discussion

The overall test results show that the trainer kit successfully integrates multiple sensor inputs, ESP32 processing, actuator outputs, local LCD display, and Telegram-based remote notification. The RFID module demonstrated access-control simulation within a 0–3 cm reading range, the ultrasonic sensor triggered servo movement below the 6 cm threshold, the PIR sensor generated a clear voltage transition during motion detection, the LDR controlled the LED according to light conditions, and the MQ-2 sensor activated the buzzer when the programmed threshold was exceeded. The voltage test also confirmed that all modules received suitable operating voltages.

From an educational perspective, the trainer kit provides a complete learning platform for introducing IoT concepts through practical experimentation. Students can learn digital input, analog input, sensor thresholding, actuator control, display output, access control, and Telegram-based communication in one system. Compared with single-function prototypes, this trainer kit offers broader learning coverage because several IoT application scenarios are integrated into one physical platform.

Nevertheless, several limitations remain. First, the testing was mainly functional and did not include repeated-cycle reliability testing or statistical analysis of sensor accuracy. Second, Telegram communication was verified based on successful notification delivery, but communication latency was not quantitatively measured. Third, the DHT22, MQ-2, and HC-SR04 readings were not compared with calibrated reference instruments, so measurement accuracy was not fully evaluated. Fourth, the trainer kit was developed as a prototype learning medium, and further validation involving students or teachers is needed to assess its effectiveness in improving learning outcomes.

IV. CONCLUSION

This study designed and evaluated an ESP32-based Internet of Things (IoT) trainer kit integrated with Telegram as a practical learning medium for vocational microcontroller and IoT courses. The trainer kit was developed using a waterfall-based approach and integrated several input modules, including the HC-SR04 ultrasonic sensor, DHT22 temperature and humidity sensor, PIR motion sensor, LDR light sensor, MQ-2 gas sensor, and RC522 RFID module. The output modules consisted of a 16x2 LCD, LED indicator, buzzer, servo motor, relay, and solenoid door lock. The system was implemented on an acrylic-based trainer panel to make the component layout visible and suitable for practical classroom activities.

The functional testing results showed that all input, processing, output, and communication modules operated according to the designed logic. The RFID module detected cards within a reading distance of approximately 0–3 cm, the ultrasonic sensor triggered servo movement when the object distance was below 6 cm, the PIR sensor produced a voltage transition from approximately 0 V to 3.1 V when motion was detected, and the MQ-2 sensor activated the buzzer when the programmed threshold was exceeded. The LDR module successfully controlled the LED based on light conditions,

while the DHT22 sensor displayed environmental data on the LCD and supported remote information delivery through Telegram. Voltage verification also showed that the main components operated within acceptable voltage ranges for prototype-level testing. These findings indicate that the proposed trainer kit is feasible as an interactive learning medium for introducing sensor interfacing, actuator control, access simulation, and Telegram-based IoT communication.

Despite these results, the study still has several limitations. The evaluation was mainly based on functional testing and voltage verification, without repeated-cycle reliability testing or statistical analysis of sensor accuracy. Telegram communication was verified through successful message delivery, but notification latency was not measured quantitatively. In addition, several sensors, such as DHT22, HC-SR04, and MQ-2, were not compared against calibrated reference instruments. Therefore, future work should include repeated testing, latency measurement, comparison with reference instruments, and classroom validation involving students and teachers to assess the effectiveness of the trainer kit in improving IoT learning outcomes.

REFERENCES

- [1] P. Jacko, M. Bereš, I. Kováčová, J. Molnár, T. Vince, J. Dziak, B. Fecko, Š. Gans, and D. Kováč, "Remote IoT education laboratory for microcontrollers based on the STM32 chips," *Sensors*, vol. 22, no. 4, Art. no. 1440, 2022, doi: <https://doi.org/10.3390/s22041440>
- [2] A. Awouda, E. Traini, M. Asranov, and P. Chiabert, "Bloom's IoT taxonomy towards an effective Industry 4.0 education: Case study on open-source IoT laboratory," *Education and Information Technologies*, vol. 29, pp. 15043–15065, 2024, doi: <https://doi.org/10.1007/s10639-024-12468-7>
- [3] D. Sidik, A. Risal, and A. Hudiah, "IoT-based microcontroller trainer media: Innovation for vocational education essential programs," *International Journal of Latest Technology in Engineering, Management & Applied Science*, vol. 13, no. 11, 2024, doi: <https://doi.org/10.51583/ijltemas.2024.131102>
- [4] A. Mahmood, K. Ahmed, and H. Mahmood, "Design and implementation of a microcontroller training kit for blend learning," *Computer Applications in Engineering Education*, vol. 30, no. 4, pp. 1236–1247, 2022, doi: <https://doi.org/10.1002/cae.22517>
- [5] M. Sukardjo, V. Oktaviani, S. Tawari, I. Alfajar, and I. Ichsan, "Design of control system trainer based on IoT as electronic learning media for natural science course," *Jurnal Penelitian Pendidikan IPA*, vol. 9, no. 2, 2023, doi: <https://doi.org/10.29303/jppipa.v9i2.3097>
- [6] L. McLauchlan, D. Hicks, M. Mehrubeoglu, and H. Bhimavarapu, "Enabling remote student learning of IoT technologies," in *Proc. 2023 ASEE Annual Conference & Exposition*, Baltimore, MD, USA, 2023, doi: <https://doi.org/10.18260/1-2--43273>
- [7] A. Saxena, A. Agarwal, B. Nagrath, C. Jayavanth, S. Thulasidoss, S. Maheswari, and P. Sasikumar, "Deep learning-driven IoT solution for smart tomato farming," *Scientific Reports*, vol. 15, 2025, doi: <https://doi.org/10.1038/s41598-025-15615-3>
- [8] J. Joko, A. Putra, and B. Isnawan, "Implementation of IoT-based human machine interface-learning media and problem-based learning to increase students' abilities, skills, and innovative behaviors of Industry 4.0 and Society 5.0," *TEM Journal*, vol. 12, no. 1, 2023, doi: <https://doi.org/10.18421/TEM121-26>
- [9] D. Hercog, T. Lerher, M. Truntič, and O. Težak, "Design and implementation of ESP32-based IoT devices," *Sensors*, vol. 23, no. 15, Art. no. 6739, 2023, doi: <https://doi.org/10.3390/s23156739>
- [10] A. Pradeep, "Enabling IoTs with ESP32 for affordable education," in *Proc. 2023 5th International Conference on Inventive Research in Computing Applications (ICIRCA)*, Coimbatore, India, 2023, pp. 1368–1373, doi: <https://doi.org/10.1109/ICIRCA57980.2023.10220594>
- [11] M. Ficco, A. Guerriero, E. Milite, F. Palmieri, R. Pietrantuono, and S. Russo, "Federated learning for IoT devices: Enhancing TinyML with on-board training," *Information Fusion*, vol. 104, Art. no. 102189, 2023, doi: <https://doi.org/10.1016/j.inffus.2023.102189>
- [12] J. Capcha-Ochoa, J. Chahua-Benito, M. Serafin-Cayllahua, S. Mamani-Martinez, J. Mendivil-Imbertis, J. Mendoza-Fernandez, R. Casas-Miranda, M. Cabana-Cáceres, and C. Castro-Vargas, "Smart irrigation system with IoT, machine learning, and solar power for efficient plant care," *Emerging Science Journal*, 2025, doi: <https://doi.org/10.28991/esj-2025-09-03-012>
- [13] M. Yusro, M. Ma'sum, M. Muhamad, and A. Jaenul, "Pengembangan trainer aplikasi multi-sensors (TAMS) berbasis Arduino dan Raspberry Pi," *Risenologi*, vol. 6, no. 1, pp. 77–85, 2021, doi: <https://doi.org/10.47028/j.risenologi.2021.61.150>
- [14] H.-Y. Lee, P. Zhou, A. Duan, J. Wang, V. Wu, and D. Navarro-Alarcon, "A multisensor interface to improve the learning experience in arc welding training tasks," *IEEE Transactions on Human-Machine Systems*, vol. 53, pp. 619–628, 2023, doi: <https://doi.org/10.1109/THMS.2023.3251955>
- [15] A. Almufarreh and M. Arshad, "Promising emerging technologies for teaching and learning: Recent developments and future challenges," *Sustainability*, vol. 15, no. 8, Art. no. 6917, 2023, doi: <https://doi.org/10.3390/su15086917>
- [16] M. Fikri, A. Rossydi, A. Risal, P. Makassar, and U. Makassar, "Design of a security cameras for smart rooms based on Internet of Things (IoT) as an automation learning media at Makassar Aviation Polytechnic," *Airman: Jurnal Teknik dan Keselamatan Transportasi*, vol. 7, no. 2, 2024, doi: <https://doi.org/10.46509/ajtk.v7i2.510>
- [17] M. W. Habibi and I. G. P. A. Buditjahjanto, "Impact of training kit-based Internet of Things to learn microcontroller viewed in cognitive domain," *TEM Journal*, vol. 13, no. 2, 2024, doi: <https://doi.org/10.18421/TEM132-30>
- [18] F. Oliveira, D. Costa, F. Assis, and I. Silva, "Internet of Intelligent Things: A convergence of embedded systems, edge computing and machine learning," *Internet of Things*, vol. 26, Art. no. 101153, 2024, doi: <https://doi.org/10.1016/j.iot.2024.101153>

- [19] O. Khattach, O. Moussaoui, and M. Hassine, "End-to-end architecture for real-time IoT analytics and predictive maintenance using stream processing and ML pipelines," *Sensors*, vol. 25, no. 9, Art. no. 2945, 2025, doi: <https://doi.org/10.3390/s25092945>
- [20] A. Pradeep, "Developing an IoT enabled smart classroom S3TH system," in *Proc. 2023 5th International Conference on Inventive Research in Computing Applications (ICIRCA)*, Coimbatore, India, 2023, pp. 1374–1379, doi: <https://doi.org/10.1109/ICIRCA57980.2023.10220862>
- [21] K. Umam, A. F. Ibadillah, A. Ubaidillah, H. Sukri, D. Rahmawati, and R. Alfita, "Pengembangan trainer Internet of Things (IoT) sebagai media pembelajaran dengan menggunakan NodeMCU ESP32CAM," *Energy: Jurnal Ilmiah Ilmu-Ilmu Teknik*, vol. 14, no. 1, 2024, doi: <https://doi.org/10.51747/energy.v14i1.1937>
- [22] R. Fikri, A. D. Samala, D. Faiza, and Almasri, "Performance analysis of an ESP32-based integrated smart home access system: A laboratory scale study," *Jurnal Vocational Teknik Elektronika dan Informatika*, vol. 13, no. 4, 2025, <https://doi.org/10.24036/voteteknika.v13i4.136492>
- [23] K. Okafor and O. Longe, "Smart deployment of IoT-TelosB service care StreamRobot using software-defined reliability optimisation design," *Heliyon*, vol. 8, Art. no. e09634, 2022, doi: <https://doi.org/10.1016/j.heliyon.2022.e09634>
- [24] A. Apriliana, A. Pradana, and D. Hartanti, "Comparison of the use of Telegram and Blynk platforms in IoT-based gas leak detection," *Jurnal Riset Informatika*, vol. 7, no. 3, 2025, doi: <https://doi.org/10.34288/jri.v7i3.381>
- [25] D. Bestari and A. Wibowo, "IoT based real-time weather monitoring system using Telegram bot and ThingsBoard platform," *International Journal of Interactive Mobile Technologies*, vol. 17, no. 6, pp. 4–19, 2023, doi: <https://doi.org/10.3991/ijim.v17i06.34129>
- [26] G. Pereira, M. Chaari, and F. Daroge, "IoT-enabled smart drip irrigation system using ESP32," *IoT*, vol. 4, no. 3, pp. 221–243, 2023, doi: <https://doi.org/10.3390/iot4030012>
- [27] S. Chakraborty and P. S. Aithal, "Let us create multiple IoT device controller using AWS, ESP32 and C#," *International Journal of Applied Engineering and Management Letters*, vol. 7, no. 3, 2023, doi: <https://doi.org/10.47992/ijaeml.2581.7000.0172>
- [28] A. Borkowski, J. Brożyna, and J. Lesiuk, "Implementation of the telemetric integration of the BIM-RFID in context of access control," *Buildings*, vol. 14, no. 11, Art. no. 3356, 2024, doi: <https://doi.org/10.3390/buildings14113356>
- [29] N. Hu and Y. Zhang, "Dynamic data management in industry university integration platform: An embedded decision framework," *Journal of Computational Methods in Sciences and Engineering*, 2025, doi: <https://doi.org/10.1177/14727978251374329>
- [30] F. Puspasari, T. P. Satya, U. Oktiawati, I. Fahrurrozi, and H. Prisyanti, "Analisis akurasi sistem sensor DHT22 berbasis Arduino terhadap termohygrometer standar," *Jurnal Fisika dan Aplikasinya*, vol. 16, no. 1, pp. 40–45, 2020, doi: <https://doi.org/10.12962/j24604682.v16i1.5776>
- [31] D. Suthar and M. Patel, "IoT-based home security system using Telegram," in *Proc. 2025 12th International Conference on Computing for Sustainable Global Development (INDIACom)*, New Delhi, India, 2025, pp. 1–9, doi: <https://doi.org/10.23919/INDIACom66777.2025.11115737>
- [32] G. Kilari, R. Mohammed, and R. Jayaraman, "Automatic light intensity control using Arduino UNO and LDR," in *Proc. 2020 International Conference on Communication and Signal Processing (ICCSP)*, Chennai, India, 2020, pp. 862–866, doi: <https://doi.org/10.1109/ICCSP48568.2020.9182238>
- [33] M. Prananda, S. Syahputra, and M. Syari, "Design of an LPG leak detection system using IoT based MQ-2 sensor," *Journal of Artificial Intelligence and Engineering Applications*, vol. 3, no. 1, 2023, doi: <https://doi.org/10.59934/jaiea.v3i1.337>
- [34] B. Pratama, Z. Z., A. Hadi, and L. Mursyida, "Development of network infrastructure monitoring system at vocational high school using MikroTik and Telegram integration," *Journal of Hypermedia & Technology-Enhanced Learning*, vol. 2, no. 3, 2024, doi: <https://doi.org/10.58536/j-hytel.v2i3.133>
- [35] M. Gerten, S. Frei, M. Kiffmeier, and O. Bettgens, "Voltage stability of automotive power supplies during tripping events of melting and electronic fuses," in *Proc. 2022 IEEE 95th Vehicular Technology Conference (VTC2022-Spring)*, Helsinki, Finland, 2022, pp. 1–6, doi: <https://doi.org/10.1109/VTC2022-Spring54318.2022.9860939>