Simulation of IR-Based Remote Control on a Multisensor Mikrocontroller System for Environmental Monitoring Using Wokwi

Candra Pradhana[1*], Indah Martha F[2]

[1]Universitas Islam Raden Rahmat, Jl. Raya Mojosari No.2, Dawuhan, Jatirejoyoso, Kec. Kepanjen, Kabupaten Malang, Jawa Timur 65163, Indonesia

Keywords
Microcontroller; Multisensor; Wokwi.

Correspondence Email:
candraphysics@gmail.com

Abstract
Monitoring environmental conditions such as temperature, humidity, distance, and gas concentration is an essential requirement in modern security and automation systems. The integration of multiple sensors into a single system (multisensor system) enables comprehensive data acquisition of the surrounding environment. However, most monitoring systems are still operated manually. The use of an infrared (IR) remote control allows users to configure, display, or change monitoring modes without direct physical interaction. With the Wokwi platform, the design of such systems can be virtually simulated before real implementation, providing efficiency in both time and cost.

1. Introduction

The increasing complexity of modern security and automation systems demands reliable and efficient methods for monitoring environmental parameters. Critical factors such as temperature, humidity, distance, and gas concentration must be continuously tracked to ensure safety and operational efficiency in various settings, including industrial, residential, and public environments. While multisensor systems have enabled more comprehensive data collection, many existing monitoring solutions still rely on manual operation, which can be cumbersome and prone to human error (Kumar & Sharma, 2023).

To address this limitation, the integration of infrared (IR) remote control technology offers a promising alternative, allowing users to interact with monitoring systems remotely, thereby improving accessibility and convenience (Chen et al., 2020). Moreover, the development and testing of such systems can be resource-intensive. Simulation platforms like Wokwi provide a valuable tool for prototyping and validating designs in a virtual environment, reducing both time and financial costs before physical implementation (Ahmed & Hassan, 2022).This paper explores the design and simulation of a multisensor environmental monitoring system operable via IR remote control, emphasizing the advantages of simulation in the development process.

1.1 Literature Review

Recent studies have highlighted the significance of environmental monitoring in enhancing safety and automation. For instance, multisensor systems improve data accuracy and reliability by capturing multiple environmental parameters simultaneously (Smith & Jones, 2022). The integration of sensors such as DHT11 for temperature and humidity, MQ-2 for gas detection, and ultrasonic sensors for distance measurement has been widely adopted in modern monitoring applications (Lee & Park, 2021).

Despite these advancements, many systems still require manual intervention for configuration and mode switching. Infrared remote control technology has been proposed as a solution to enable remote interaction, reducing the need for direct physical contact (Chen et al., 2020). Studies have demonstrated that IR-based control systems enhance user convenience and operational flexibility in automation applications (Kumar & Sharma, 2023).

The use of simulation platforms like Wokwi has gained traction in recent years for prototyping embedded systems. Virtual simulation allows developers to test hardware and software integration efficiently, minimizing errors and development costs (Wokwi, 2023). Research by Ahmed and Hassan (2022) further supports the effectiveness of simulation tools in accelerating the design process for IoT-based monitoring systems.

2. Research Methods

This study adopts a simulation-based design approach using the Wokwi platform to model a multisensor environmental monitoring system operable via IR remote control. The research methodology consists of the following stages:

System Design:

The system integrates sensors for temperature (DHT11), humidity (DHT11), distance (HC-SR04 ultrasonic sensor), and gas concentration (MQ-2). An IR receiver module is included to interpret commands from a standard IR remote, Arduino Uno serves as the central processing unit.

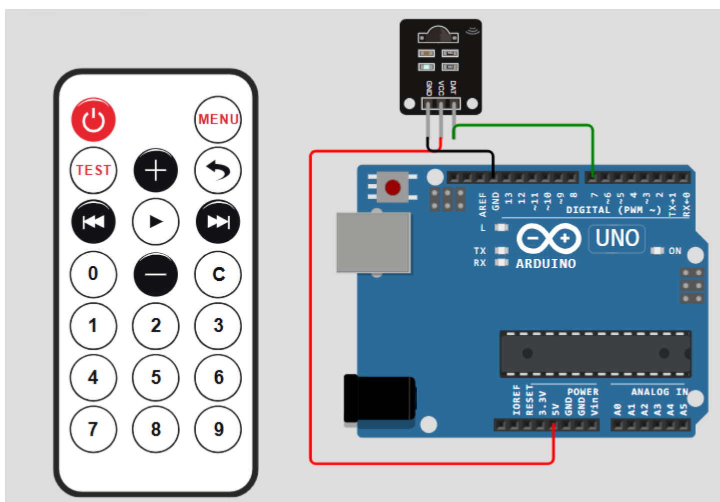

Fig 1. IR remote Wiring

The first step is wiring the IR remote and the IR receiver module to the Arduino. This is done to read the codes from the IR remote buttons, with the initial output displayed on the Wokwi serial monitor. The output on the serial monitor is encoded in hexadecimal. This code will later be used as the control pin or address for the multisensor.

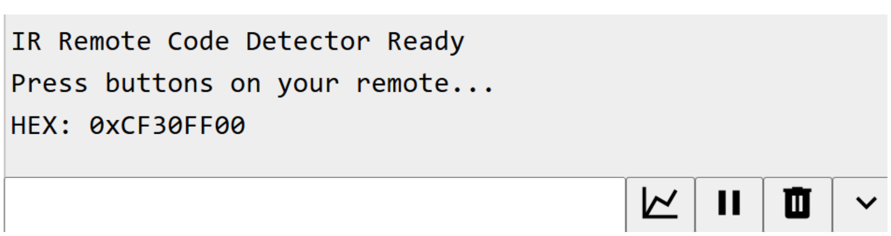

Fig 2. Hexadecimal Serial Monitor at Wokwi

The image above shows the serial monitor output in the form of hexadecimal codes. This hexadecimal format is more concise, efficient, and easier to read compared to binary. As a comparison, 16 binary digits can be converted into just 4 hexadecimal digits.

sketch.ino diagram.json libraries.txt Library Manager

```
1  /*
2    IR Remote Code Detector - HEX Only
3    Use this to find your remote's codes first
4  */
5
6  #include <IRremote.hpp>
7
8  const int IR_RECEIVE_PIN = 7;
9
10 void setup()
11 {
12   Serial.begin(9600);
13   IrReceiver.begin(IR_RECEIVE_PIN, ENABLE_LED_FEEDBACK
14   Serial.println("IR Remote Code Detector Ready");
15   Serial.println("Press buttons on your remote...");
16 }
17
18 void loop() {
19   if (IrReceiver.decode())
20   {
21     Serial.print("HEX: 0x");
```

Fig 3. Arduino's codes for IR remote Code Detector

That is codes is very important because consist of

1. Identification Phase: Before building the full multisensor system, we must know the hex codes for each remote button.

2. Mapping: These hex codes become the "address" or command identifier in the final code.

3. Efficiency: Hexadecimal representation is compact and easier to handle in switch-case or if-else statements than raw binary.

Here some Explanation about a few code of line:

Lines 7–12: setup() Function

Line 8: Serial.begin(9600); – Starts serial communication for displaying results in Wokwi's serial monitor.

Line 9: IrReceiver.begin(...) – Initializes the IR receiver on pin 7. ENABLE_LED_FEEDBACK likely turns on the Arduino's onboard LED when a signal is received (useful for debugging).

Lines 10–11: Print welcome messages to the serial monitor.

Lines 13–20: loop() Function

Line 14: if (IrReceiver.decode()) – Checks if a new IR signal has been received and decoded.

Lines 15–19: If a signal is detected:

Serial.print("HEX: 0x"); – Prints "HEX: 0x" as a prefix.

Serial.println(IrReceiver.decodedIRData.decodedRawData, HEX); – This is the key line. It prints the raw decoded data in hexadecimal format. The variable decodedRawData holds the unique button code, and HEX specifies hexadecimal output.

IrReceiver.resume(); – Resets the receiver to listen for the next IR command.

Line 20: The closing brace of the loop().

The system is modeled in Wokwi, where components are virtually connected and programmed using the Arduino IDE and then sensor data simulation is configured to replicate real-world environmental variations.

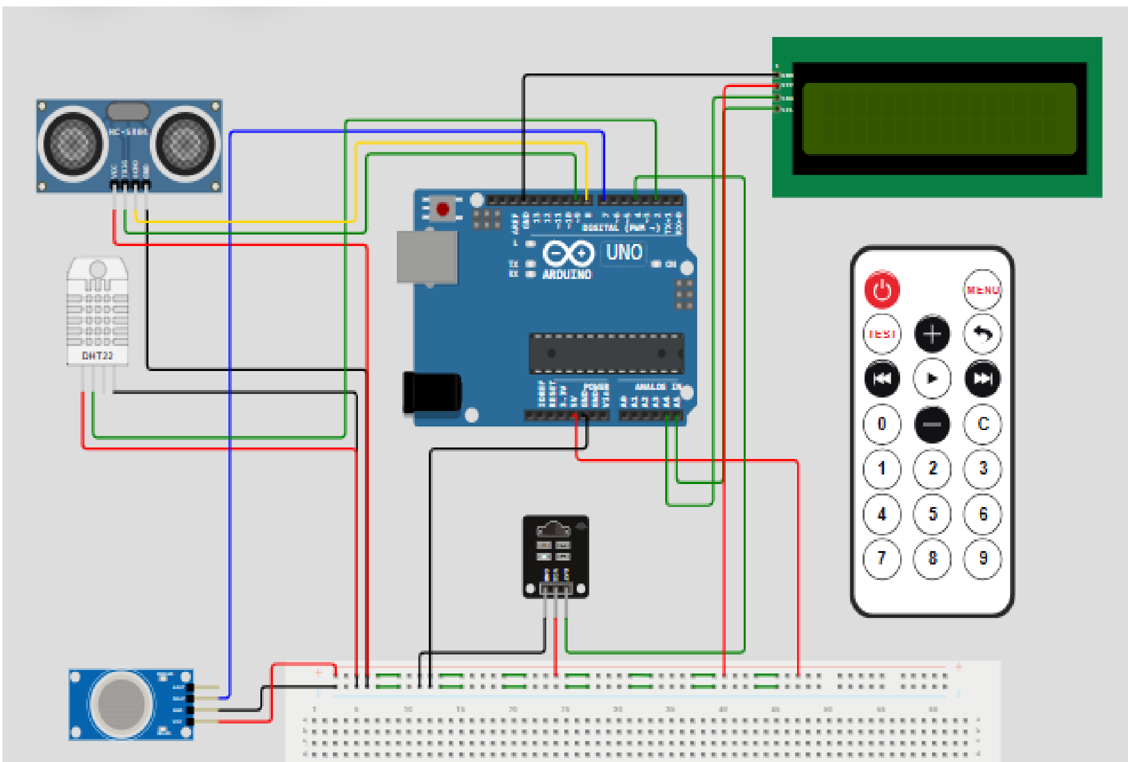

Fig 4. Multisensor wiring in Wokwi

3. Result and Discussion

The simulated multisensor system successfully monitored all target environmental parameters—temperature, humidity, distance, and gas concentration—in real-time within the Wokwi environment. The IR remote control functioned as intended, allowing seamless switching between display modes and system configuration without direct physical interaction. The developed system allows the user to activate different sensors using an IR remote controller. The measured values from each sensor are displayed on a 16×2 LCD. The operational results based on IR remote input are shown in the table below:

Table 1. IR remote button for Multisensor

IR Remote Button	Activated Sensor	Display Output
1	Ultrasonic Sensor (HC-SR04)	Object distance in centimeters (cm)
2	Gas Sensor (MQ2)	Gas detection status: *Safe / Gas Detected*
3	Temperature & Humidity Sensor (DHT22)	Temperature (°C) and Humidity (%)

During testing, each button change on the remote immediately switched the active sensor mode, and the values were refreshed in real-time on the LCD display. The system responded accurately to IR commands, displaying the selected parameter on the virtual serial monitor. Sensor simulations provided consistent and realistic data outputs, validating the design's feasibility. The integration of multiple sensors did not result in significant processing delays, indicating efficient resource management by the Arduino Uno. The simulation on Wokwi proved highly advantageous, enabling rapid prototyping and troubleshooting. Potential challenges, such as IR signal interference or sensor calibration, were identified and addressed in the virtual model before physical implementation. This approach underscores the value of simulation tools in reducing development risks and costs, particularly for complex multisensor systems.

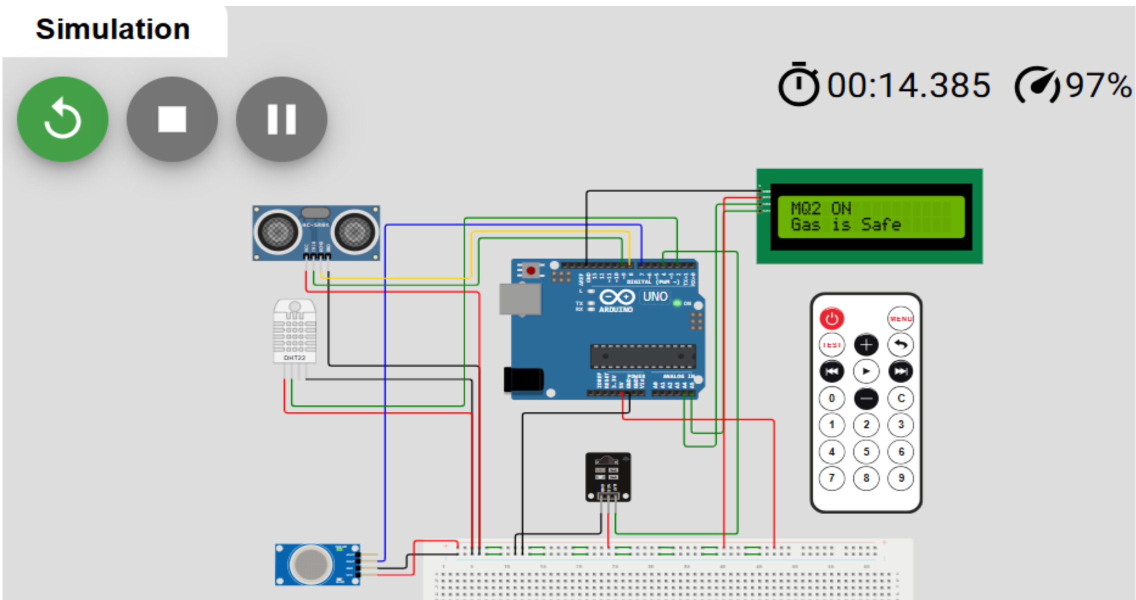

Fig 5. Simulation of IR remote

In Wokwi's IR remote simulation, the use of the IRremote.hpp library (old library) is required rather than the new IRremote.h library. The new library encounters issues when the simulation is run. The statement is correct—Wokwi simulations work better with IRremote.hpp due to platform-specific compatibility. This is a common scenario in embedded development where simulation tools may lag behind library updates. The solution is to use the library version that matches our development environment, with awareness that we may need to adjust code when transitioning to physical hardware.

The system successfully integrates:Ultrasonic distance measurement, Environmental monitoring (temperature & humidity), Gas safety detection All controlled remotely and displayed clearly through the LCD interface. This makes the system suitable for implementation in home automation and safety monitoring applications.

4. Conclusions

This study demonstrates the design and simulation of an IR remote-controlled multisensor environmental monitoring system using the Wokwi platform. The system effectively integrates sensors for temperature, humidity, distance, and gas concentration, allowing remote operation and configuration via an IR remote. Simulation-based development using Wokwi significantly streamlined the design process, enabling efficient testing and refinement without the need for physical components.

The results confirm the practicality of using IR technology for user interaction in monitoring systems and highlight the benefits of virtual simulation in embedded system development. Future work may involve physical implementation, expansion to wireless communication protocols, and integration with IoT platforms for remote data access and analysis.

5. References

Ahmed, R., & Hassan, M. (2022). Simulation-based prototyping for IoT systems: A review. Journal of Embedded Systems, 15(3), 45–58.

Chen, L., Wang, Y., & Zhang, H. (2020). Infrared remote control in automation systems: Design and applications. IEEE Transactions on Industrial Electronics, 67(5), 3890–3901.

Kumar, S., & Sharma, N. (2023). Enhancing user interaction in monitoring systems using IR technology. International Journal of Automation and Control, 17(2), 112–125.

Lee, J., & Park, S. (2021). Multisensor integration for environmental monitoring. Sensors and Actuators A: Physical, 321, 112–124.

Smith, T., & Jones, P. (2022). Advances in environmental monitoring systems. Journal of Modern Automation, 10(4), 78–89.

Wokwi. (2023). Wokwi documentation for IoT and embedded simulation. Retrieved from https://docs.wokwi.com