# Integration of the XY-MD02 Module in an IoT-Based Humidity and Temperature Monitoring System with Graphic Display on Nextion LCD

Syafriyadi Nor
*Electrical Engineering*
*Banjarmasin State Polytechnic*
Banjarmasin, Indonesia
syafriyadi.nor@poliban.ac.id

Sarifudin
*Electrical Engineering*
*Banjarmasin State Polytechnic*
Banjarmasin, Indonesia
sarif@poliban.ac.id

***Abstract*—Temperature and humidity monitoring is crucial in various industrial sectors such as agriculture, manufacturing and health. Integrating the latest technology, such as IoT (Internet of Things), opens up new opportunities to increase the efficiency and accuracy of monitoring systems. This research focuses on integrating the XY-MD02 module into an IoT-based monitoring system to measure and monitor humidity and temperature levels consistently. The system uses the XY-MD02 module to collect data, which is transmitted wirelessly to the server using the MQTT protocol. The collected data is processed and displayed in real-time on the Nextion LCD, providing an intuitive graphical representation of environmental conditions. Integrating the XY-MD02 module and Nextion LCD in an IoT-based monitoring system demonstrates practical and reliable humidity and temperature measurements. The average value of the measured temperature is 31.58 with a standard deviation of 0.17, indicating high accuracy in temperature measurement with low variation. Meanwhile, for humidity, the average is 62.34, with a standard deviation of 1.01. The system's compatibility with the MQTT protocol ensures smooth communication and data exchange between devices. Integrating the XY-MD02 module into an IoT-based monitoring system has proven successful in providing consistent and real-time monitoring of humidity and temperature, offering an effective solution in environmental monitoring while keeping up with the latest technological developments.***

***Keywords— ESP32, internet of things, modbus, nextion, XY-MD02***

## I. INTRODUCTION

Ambient humidity and temperature play essential roles in various applications, including the industrial sector [1][2][3][4], information technology[5][6][7], agriculture [8][9][10] [11][12], household[13], and other sectors [14][15]. Humidity and temperature monitoring can help maintain product quality, increase user comfort, and detect unsafe or unsuitable conditions.

The ESP32-S is a highly versatile IoT platform that expands the well-known ESP8266 family. Compared to the ESP8266, the ESP32-S offers significantly more I/O pins, additional flexibility and Bluetooth support advantages. The SHT20 sensor on the XY-MD02 module is a popular humidity and temperature sensor. It is compatible with the industrial Modbus RTU protocol and is renowned for its high accuracy, fast response and low power consumption. This sensor will provide accurate data for monitoring environmental humidity and temperature.

The Internet of Things (IoT) allows devices to connect and access data remotely via the Internet [16][17][18][19]. With IoT integration, humidity and temperature monitoring can be done in real-time from anywhere, allowing users to respond quickly to changing environmental conditions.

Nextion LCD is a touch screen and an innovative Human Machine Interface (HMI) solution. It provides a control and visualisation interface between users and processes, machines, applications, or equipment [20]. This LCD can display information clearly and attractively. Nextion LCD integration in monitoring systems allows users to visualise humidity and temperature data in a graphical form that is easy to understand.

Previous studies have primarily focused on IoT-based monitoring systems for humidity and temperature. While these systems provide real-time data and remote monitoring capabilities, they often lack an integrated display solution for immediate, on-site feedback. Furthermore, the integration of Modbus RTU protocol for sensor communication and serial communication for display is often overlooked.

This research aims to create a compelling and easy-to-use humidity and temperature monitoring system by integrating the XY-MD02 sensor via the Modbus RTU protocol, and connecting the Nextion LCD with the ESP32-S through serial communication. By leveraging the ESP32-S IoT platform, accurate sensors like the XY-MD02 module, and the Nextion LCD touch screens, this research provides a comprehensive solution that enables users to monitor environmental conditions both remotely and locally in real-time. This integration not only presents new technological innovations but also offers practical and relevant solutions for improving environmental monitoring systems.


**How to cite**: Nor, Syafriyadi., Sarifudin (2025). Integration of the XY-MD02 Module in an IoT-Based Humidity and Temperature Monitoring System with Graphic Display on Nextion LCD. *JAREE (Journal on Advanced Research in Electrical Engineering), 9(1).*

## II. METHOD

### A. System Design

Design the overall system design, including the hardware and software used.

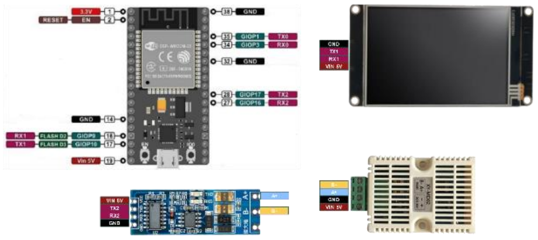


Fig. 1. Hardware Configuration

Figure 1 describes the hardware used. NodeMCU ESP-32S with Pinout 38 is a development board that uses the ESP-32S chip and has 38 pinouts that can be used for connections. The ESP-32S NodeMCU acts as the system's brain, controlling overall operations. The XY-MD02 module is used to measure temperature and humidity, and this sensor is connected to the NodeMCU via Modbus RTU protocol using an RS485 module to transfer data. The RS485 module is used for serial communication with other devices in the RS485 network, and in this setup, the RS485 module is connected to the NodeMCU via UART2. The Nextion NX4832K035 LCD is a TFT (Thin-Film Transistor) touchscreen with a resolution of 480x320 pixels, which connects to the NodeMCU via UART1. This screen is usually used to display system information such as temperature, humidity, and other status and receive input from the user via the touch screen. In this setup, the ESP-32S NodeMCU acts as the main controller, collecting data from the XY-MD02 module, sending it via RS485 to other devices in the network, and displaying the information received or collected from other devices on the Nextion LCD screen. The following is the serial configuration between devices:

TABLE 1. SERIAL CONFIGURATION

| NodeMCU ESP-32S | Modul XY-MD02 | Nextion NX4832K035 |
|---|---|---|
| Modbus RTU (Master) | Modbus RTU (Slave), ID : 1 | - |
| UART1 (9600, 8N1) | - | UART1 (9600, 8N1) |
| UART2 (9600, 8N1) | UART2 (9600, 8N1) | - |

In Table 1, the serial configuration between connected devices is shown. Communication between the ESP-32S NodeMCU and the XY-MD02 module via the Modbus RTU protocol uses UART2, while communication between the ESP-32S NodeMCU and the Nextion NX4832K035 LCD uses UART1. Parameter 9600 refers to the baud rate used for UART communication. In contrast, parameter 8N1 describes the serial setting with 8 data bits, N, which means no parity (None Parity), and 1, which is the number of stop bits.

### B. Sending Data Using the MQTT Protocol

To access the Internet, the NodeMCU ESP-32S must be connected to the same Wi-Fi network as the router. The NodeMCU ESP-32S will use the Wi-Fi protocol to communicate with the router. After a successful connection, the router will assign the ESP-32S NODEMCU a valid IP address, allowing internet access.

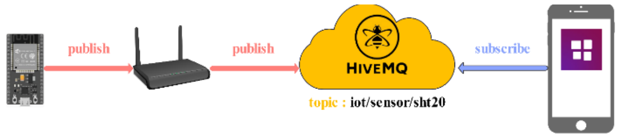



Fig. 2. Inter-Device Communication Using the MQTT Protocol

Following are the steps for the process of sending data from the NodeMCU ESP-32S to the client using the MQTT protocol:

1) NodeMCU ESP-32S will read data from the XY-MD02 module. The collected data will be packaged in JSON format.

2) NodeMCU ESP-32S will use the MQTT protocol to send data to the MQTT server.

3) An MQTT message with the topic "iot/sensor/sht20" will be published, and data in JSON format will be included as the payload.

4) Mobile Device Using the MQTT Dashboard Application

5) The mobile device user has installed the MQTT Dashboard application on the Android device.

6) In the application, the user will set up a connection to the same MQTT server used by the NodeMCU ESP-32S.

7) Users can choose the dashboard's display for temperature and humidity data according to the available widgets.

8) Users will choose to subscribe to the same topic, namely "iot/sensor/sht20".

9) The application will receive MQTT messages published by the NodeMCU ESP-32S and display the received data in JSON format.

With this setup, the NodeMCU ESP-32S will connect to the internet via a router, collect data from the XY-MD02 module, and send it to the MQTT server. On the other hand, mobile device users will use the MQTT Dashboard application to retrieve data from the MQTT server in the same way, namely by subscribing to the appropriate topic, so that they can view the sensor data sent by the NodeMCU ESP-32S in JSON format.

### C. Software Implementation

Create a program that connects the XY-MD02 module with the ESP-32S NodeMCU to obtain temperature and humidity data. This program refers to Table 2 in the XY-MD02 module datasheet, which shows which registers can be accessed.

TABLE 2. REGISTER CONFIGURATION ON XY-MD02 [21]

| Slave ID | Function Code | Register Address | Register Contents |
|---|---|---|---|
| 0x01 (1) | 0x04 (Read input register) | 0x0001 (2 Bytes) | Temperature |
| | | 0x0002 (2 Bytes) | Humidity |

- Slave ID: This is the identification address of the device that serves as the communication target. In this case, the slave address is 0x01 (or 1 in decimal).
- Function Code: The function code determines the type of operation requested by the controller device (for example, NodeMCU ESP-32S) of the XY-MD02 module device. Function code 0x04 indicates an input register read operation.
- Register Address: This is the address of the register in the XY-MD02 module that will be accessed. These registers are identified by the hexadecimal numbers 0x0001 and 0x0002. The first register (0x0001) contains temperature data, while the second register (0x0002) contains humidity data. Each register is 2 bytes long.
- Register Contents: These are the contents of the register indicated by the corresponding register address. Register 0x0001 (Temperature) contains temperature data, while register 0x0002 (Humidity) contains humidity data. The data in these registers is usually organised in a predefined format, such as in floating-point or integer format, according to the XY-MD02 module specifications.

The ModbusMaster library [22] is required to read data from the XY-MD02 module using the NodeMCU ESP-32S via the Modbus protocol. This library will help you access Modbus data more easily. The data obtained from the XY-MD02 module must be divided by 10 to get the actual temperature and humidity results. For example, if the temperature value read is 250, it represents a temperature of 25°C; if the humidity value read is 650, it means a humidity of 65%RH.

### D. *Setting up the display and communication between the NodeMCU ESP-32S and the Nextion NX4832K035 LCD*

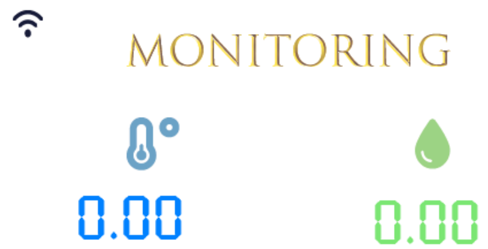



Fig. 3. Display on Nextion NX4832K035 LCD

Figure 3 is a graphic display on the Nextion LCD so that it can display humidity and temperature data visually. Nextion Editor is used to design user interfaces by adding particular objects and setting the behaviour of each element. A format that Nextion can understand needs to be provided to display temperature and humidity data. In this case, we add a text object to display that data. After the text object is added, the next step is to assign the value (val) of the object to store the data received via serial communication.

TABLE 3. OBJECT FORMAT IN NEXTION

| Measurement | Object Name | Format |
|---|---|---|
| Temperature | temp | temp.val= |
| Humidity | hum | hum.val= |

Table 3 shows the objects and format strings used to display temperature and humidity data on the Nextion LCD. After assigning the val function to each object that wants to display temperature and humidity data, the Nextion LCD will automatically update the display according to the data received via serial communication received from the Arduino.

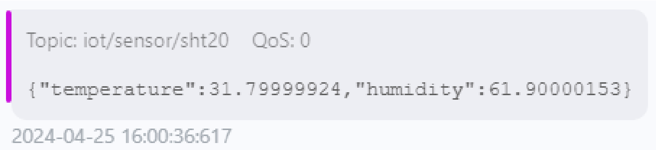
## III. RESULTS AND DISCUSSION

In this section, we evaluate the integration of the XY-MD02 Module into an IoT-based humidity and temperature monitoring system with a graphical display on a Nextion LCD touchscreen. The XY-MD02 module collects temperature and humidity data, which is then processed and displayed visually to the user via the Nextion LCD display. The results of this research discuss how to integrate the XY-MD02 Module into IoT and graphic displays on the Nextion LCD, data description and analysis, limitations and future development of this system.

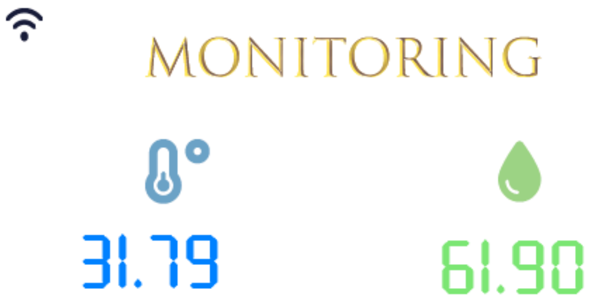
### A. *Integration of XY-MD02 Module into IoT and graphic display on Nextion LCD*

The XY-MD02 module is connected to the ESP-32S NodeMCU via serial communication. The data is accessed from the register address via the Modbus RTU protocol to obtain temperature and humidity data. The data that is successfully read is then formatted into JSON. Using the JSON format to send temperature and humidity data to MQTT, we can send both data via just one topic as a publisher, allowing for more efficient data delivery. In this context, we use the topic name "iot/sensor/sht20", and the data sending interval is every 30 seconds. In JSON format, we can organise temperature and humidity data in one object, which is then sent via a predefined MQTT topic. This reduces the number of messages that need to be sent and simplifies data processing on the receiving end because temperature and humidity data can be extracted from the same JSON object once received. Thus, using the JSON format to send data to MQTT with just one topic can increase communication and data processing efficiency. On the subscriber side, data is received according to the format sent.

In addition, the Graphic display on the Nextion LCD, connected to the ESP-32S NodeMCU via serial communication, also displays data corresponding to the data sent to MQTT. Sending data to the Nextion LCD refers to the "Xfloat" object with the "val" function and ws1 (point left) equals 2, this is done to display float data with two numbers after the comma. So, when sending data to the Nextion LCD, it must be multiplied by 100 to display the appropriate value. The object names are "temp.val" for temperature, and "hum.val" for humidity. Using the Nextion LCD touchscreen, the monitoring system can visually display temperature and humidity information to the user. This graphic display provides a more interactive and informative user experience, making it easier to understand the environmental conditions being monitored.




*a*




*b*

Fig. 4. Response from MQTT Client (a), Graphic Display on Nextion LCD (b)

Overall, the integration of the XY-MD02 module with IoT and the Nextion LCD, as shown in Figure 4, demonstrates that the module effectively transmits data, providing consistent collection of temperature and humidity data from the surrounding environment.. This integration allows the system to obtain the information needed to monitor environmental conditions promptly.

### *B. Data Description and Analysis*

The temperature and humidity data sent to MQTT are stored in the SQLite database as a data source for data analysis purposes. SQLite was chosen because of its small size and ease of use and implementation in IoT. It is a file-based database, so it does not require server configuration.

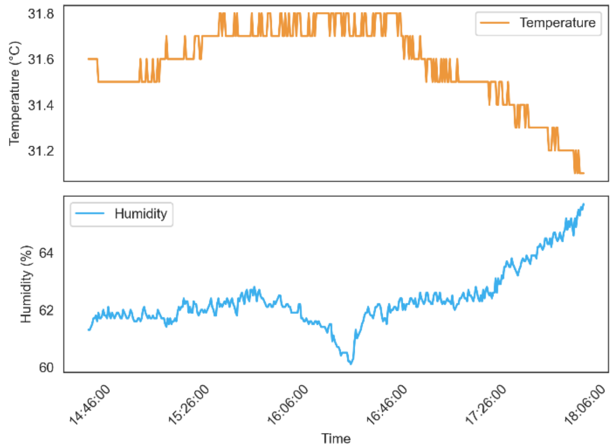



Fig. 5. Plotting Results of Temperature and Humidity Data

Figure 5 shows the plotting results from the SQLite data source which illustrates the relationship between temperature and humidity. To evaluate the relationship between temperature and humidity, Pearson's correlation coefficient was used, calculated using the following equation (1):

$$r = \frac{n(\sum xy) - (\sum x)(\sum y)}{\sqrt{[n\sum x^2 - (\sum x)^2][n\sum y^2 - (\sum y)^2]}} \quad (1)$$

Where:

- $r$ is the Pearson correlation coefficient.
- $n$ is the number of data pairs.
- $x$ and $y$ are the values of the two variables being analyzed.

The Pearson correlation coefficient was calculated as $r = -0.789$, indicating a strong negative correlation between temperature and relative humidity. This result shows that as temperature increases, relative humidity tends to decrease, reflecting the inverse relationship between these two variables.

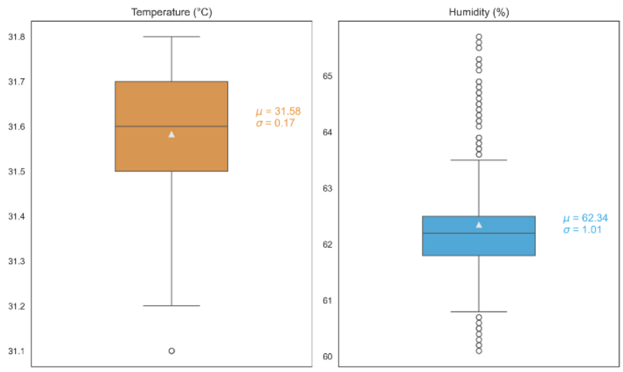



Fig. 6. Temperature and Humidity (Average and Standard Deviation

Figure 6 displays the distribution of temperature and humidity data, along with their respective statistical measures. This analysis focuses on the consistency of the readings as indicated by the average values and standard deviations.

For temperature data, the average value is 31.58°C, with a standard deviation of 0.17°C. This low standard deviation suggests that the temperature readings are consistently close to the average value, indicating minimal variability in the data. For humidity data, the average value is 62.34%, with a standard deviation of 1.01%. Although the standard deviation is slightly higher compared to temperature, the variability remains within an acceptable range, reflecting consistent humidity measurements.

The results demonstrate that the monitoring system effectively captures temperature and humidity data with a high degree of consistency. The data transmission to the IoT system operates smoothly, ensuring users receive up-to-date information about environmental conditions.

Although the integration of the XY-MD02 Module has been successful, some limitations still need to be considered, such as data collection via MQTT, which relies on an API intermediary. Additionally, SQLite is a lightweight and straightforward database suitable for small to medium-scale applications. However, it has limitations in terms of features and scalability when compared to more advanced database management systems (RDBMS) such as MySQL or PostgreSQL and the complexity of the system's initial configuration. For future development, improving system performance and simplifying configuration will be the main focus to increase the affordability and usability of this system for users.

Thus, integrating the XY-MD02 Module in an IoT-based humidity and temperature monitoring system with a graphic display on a Nextion LCD touchscreen promises an effective and easy-to-use environmental monitoring solution in various application contexts.

## IV. Conclusion

Integrating the XY-MD02 Module into an IoT-based humidity and temperature monitoring system with a graphic display on the Nextion LCD touchscreen has proven its effectiveness. This system can collect temperature and humidity data consistently and responsively, and visually display this information to users in a clear and accessible manner. The main contribution of this study is the integration of the Modbus RTU protocol for XY-MD02 sensor communication and serial communication for Nextion LCD, offering a comprehensive and user-friendly solution for real-time environmental monitoring. However, several limitations, such as dependence on API intermediaries and constraints in terms of features and scalability in the database, still need to be considered. Therefore, for further development, it is recommended that the system performance be improved and the initial configuration simplified to improve the affordability and overall performance of the system. Thus, this integration promises an effective solution in environmental monitoring relevant to the latest technologies, significantly contributing to better understanding and managing the environment in the future.

## Acknowledgment

The author would like to thank Pusat Penelitian dan Pengabdian kepada Masyarakat (P3M) Politeknik Negeri Banjarmasin for providing financial support through research grants for this research.

## References

[1] I. Riadi and R. Syaefudin, "Monitoring and Control Food Temperature and Humidity using Internet of Things Based-on Microcontroller," *J. Ilm. Tek. Elektro Komput. dan Inform.*, vol. 7, no. 1, p. 108, 2021, doi: 10.26555/jiteki.v7i1.20213.

[2] D. Tazakka Ma'arij, A. Yudhana, D. T. Ma'arij, and A. Yudhana, "Temperature and Humidity Monitoring System in Internet of Things-based Solar Dryer Dome," *Bul. Ilm. Sarj. Tek. Elektro*, vol. 5, no. 3, pp. 323–335, 2023, doi: 10.12928/biste.v5i3.8633.

[3] N. Juliana, A. Mallongi, and W. O. Megasari, "Analysis Of Humidity, Temperature, Working Period, And Personal Protective Equipment In Home Industry At Gold Craftsmen," *J. Ilmu Kesehat.*, vol. 9, no. 2, pp. 81–89, 2021, doi: 10.30650/jik.v9i2.2995.

[4] M. Awaludin, A. Y. Rangan, and A. Yusnita, "Internet of Things (IoT) Based Temperature and Humidity Monitoring System in the Chemical Laboratory of the Samarinda Industry Standardization and Research Center," *Tepian*, vol. 2, no. 3, pp. 85–93, 2021, doi: 10.51967/tepian.v2i3.344.

[5] S. Z. Effendi and U. Y. Oktiawati, "Implementation and Performance Analysis of Temperature and Humidity Monitoring System for Server Room Conditions on Lora-Based Networks," *J. Internet Softw. Eng.*, vol. 3, no. 1, pp. 20–25, 2022, doi: 10.22146/jise.v3i1.4834.

[6] T. H. Nasution, M. A. Muchtar, S. Seniman, and I. Siregar, "Monitoring temperature and humidity of server room using Lattepanda and ThingSpeak," *J. Phys. Conf. Ser.*, vol. 1235, no. 1, pp. 0–6, 2019, doi: 10.1088/1742-6596/1235/1/012068.

[7] D. E. Kurniawan, M. Iqbal, J. Friadi, R. I. Borman, and R. Rinaldi, "Smart Monitoring Temperature and Humidity of the Room Server Using Raspberry Pi and Whatsapp Notifications," *J. Phys. Conf. Ser.*, vol. 1351, no. 1, pp. 0–8, 2019, doi: 10.1088/1742-6596/1351/1/012006.

[8] R. N. Sonawane, A. S. Ghule, A. P. Bowlekar, and A. H. Zakane, "Design and Development of Temperature and Humidity Monitoring System," *Agric. Sci. Dig.*, vol. 39, no. 2, pp. 114–118, 2019, doi: 10.18805/ag.D-4893.

[9] D. D. Sanjaya, A. Fadlil, and A. Fadlil, "Monitoring Temperature and Humidity of Boiler Chicken Cages Based on Internet of Things ( IoT )," *Bul. Ilm. Sarj. Tek. Elektro*, vol. 5, no. 2, pp. 180–189, 2023, doi: 10.12928/biste.v5i2.4897.

[10] F. A. R. Mas, S. W. Suciyati, G. A. Pauzi, and J. Junaidi, "Smart Greenhouse Monitoring With Soil Temperature and Humidity Control on Internet of Things (IoT) Based Orchid Plants," *J. Energy, Mater. Instrum. Technol.*, vol. 3, no. 3, pp. 79–88, 2022, doi: 10.23960/jemit.v3i3.111.

[11] Ronny, Salmon, and A. R. Hakim, "Temperature and Humidity Monitoring System on Android Based Wallet House," *Tepian*, vol. 2, no. 3, pp. 94–100, 2021, doi: 10.51967/tepian.v2i3.349.

[12] S. H. Sasono, "Sensor Data Analysis On Monitoring And Control System Of Temperature And Humidity Based On Android In Soybean Seed Storage Room Using Nodemcu," *Jaict*, vol. 3, no. 1, p. 8, 2019, doi: 10.32497/jaict.v3i1.1290.

[13] W. Daniel, A. Pramono, J. F. Wijaya, and N. P. Wijaya, "Integrating IoT-Based Devices for Monitoring The Humidity and Temperature In the Boarding House Space," *Procedia Comput. Sci.*, vol. 227, no. C, pp. 204–213, Feb. 2024, doi: 10.1016/j.procs.2023.10.518.

[14] M. Widjaja, D. K. Halim, and R. Andarini, "The Development of an IoT-based Indoor Air Monitoring System Towards Smart Energy

Efficient Classroom," *Ultim. Comput. J. Sist. Komput.*, vol. 14, no. 1, pp. 28–35, 2022, doi: 10.31937/sk.v14i1.2565.

[15] M. A. Fauzi, R. Hidayat, and T. Hidayat, "Storage Room Temperature and Humidity Monitoring Iot- Based Medicine," *Teknokom*, vol. 6, no. 2, pp. 78–85, 2023, doi: 10.31943/teknokom.v6i2.135.

[16] J. E. Siegel, S. Kumar, and S. E. Sarma, "The future Internet of things: Secure, efficient, and model-based," *IEEE Internet Things J.*, vol. 5, no. 4, pp. 2386–2398, 2018, doi: 10.1109/JIOT.2017.2755620.

[17] S. Kumar, P. Tiwari, and M. Zymbler, "Internet of Things is a revolutionary approach for future technology enhancement: a review," *J. Big Data*, vol. 6, no. 1, 2019, doi: 10.1186/s40537-019-0268-2.

[18] A. S. Albahri *et al.*, "IoT-based telemedicine for disease prevention and health promotion: State-of-the-Art," *J. Netw. Comput. Appl.*, vol. 173, no. October 2020, p. 102873, 2021, doi: 10.1016/j.jnca.2020.102873.

[19] S. Nor, "Penerapan Internet Of Things (Iot) Sebagai Pengendali Peralatan Listrik Dan Pemantau Daya Listrik Berbasis Web," *J. EEICT (Electric, Electron. Instrumentation, Control. Telecommun.*, vol. 2, no. 2, pp. 22–28, 2019, doi: 10.31602/eeict.v2i2.4431.

[20] Nextion, "Datasheets-NX4832K035." Accessed: Oct. 03, 2024. [Online]. Available: https://nextion.tech/datasheets/nx4832k035/

[21] U. Steps, "XY-MD02." [Online]. Available: http://www.sah.rs/media/sah/techdocs/xy-md02-manual.pdf

[22] D. Walker, "ModbusMaster." GitHub, 2016. [Online]. Available: https://github.com/4-20ma/ModbusMaster