RCTA
Revista Colombiana de Tecnologías de Avanzada
UNIPAMPLONA

Digital Object Identifier: 10.24054/rcta.v1i43.2800

Development of an electronic system for remote communication in Modbus RTU industrial devices via MQTT

Desarrollo de un sistema electrónico para comunicación remota en dispositivos industriales Modbus RTU a través de MQTT

MSc(c). Guillermo Acevedo[1], PhD. Omar Pinzón[1]

[1] ***Universidad Pontificia Bolivariana seccional Bucaramanga,*** *Facultad de ingeniería eléctrica y electrónica, Floridablanca, Santander, Colombia.*

Correspondence: *omar.pinzon@upb.edu.co*

Received: *October 15, 2023.* ***Accepted:*** *December 17, 2023.* ***Published:*** *March 01, 2024.*

How to cite: *G. A. Acevedo Hernandez and O. Pinzón Ardila, "Development of an electronic system for remote communication in modbus RTU industrial devices via MQTT", RCTA, vol. 1, no. 43, pp. 44–50, Mar. 2024. Recuperado de https://ojs.unipamplona.edu.co/index.php/rcta/article/view/2800*

Abstract: This document presents the development process and initial testing phases of an electronic system designed to facilitate web-based communication for an industrial device. The industrial device operates using Modbus RTU communication, and the system utilizes the MQTT telemetry protocol, which is integrated into a 32-bit microcontroller. The microcontroller operates within a real-time operating system framework, ensuring efficient communication management. The system integration with a PLC has been established during the equipment development, and validation has been achieved through the utilization of a comprehensive database, Python scripts, and the incorporation of visualization capabilities provided by a web-based platform. The prototype is capable of remotely reading and modifying Modbus registers through a web interface, which not only allows parameter adjustments but also provides access to historical measurement records, enhancing the overall functionality of the industrial device.

Keywords: IIOT, MQTT, Modbus, Microcontroller, RTOS.

Resumen: Este documento muestra el proceso de desarrollo y las fases iniciales de prueba de un sistema electrónico diseñado para facilitar la comunicación a través de la web de un dispositivo industrial. El dispositivo industrial opera mediante comunicación Modbus RTU y el sistema emplea el protocolo de telemetría MQTT, el cual está integrado en un microcontrolador de 32 bits. El microcontrolador opera dentro de un marco de sistema operativo en tiempo real, asegurando una gestión eficiente de la comunicación. En el desarrollo del equipo se ha establecido la integración del sistema con un PLC y se ha logrado la validación mediante la utilización de una base de datos integral, scripts de Python y añadiendo capacidades de visualización proporcionadas por una plataforma basada en la web. El prototipo es capaz de leer y modificar de forma remota registros Modbus a través de una interfaz web que mediante una interfaz no sólo permite ajustes de parámetros, sino que también ofrece acceso a los registros históricos de mediciones, mejorando la funcionalidad general del dispositivo industrial.

Palabras clave: IIOT, MQTT, Modbus, Microcontroladores, RTOS.

1. INTRODUCTION

Since the Industrial Revolution, there has been a constant pursuit of continuous monitoring of equipment and machinery. In its early stages, constant presence of personnel was required to supervise the operation of each piece of equipment [1]. With the passage of years, primitive industrial sensors and communications were developed, with the fundamental purpose of ensuring worker safety, optimal equipment performance, and the implementation of preventive maintenance [2]. With time, various industrial communication protocols emerged, among them Modbus, which became one of the most widely used protocols worldwide [3]. However, Modbus presents limitations, such as the restriction of 255 connected clients per node and communication that only initiates when the server requests data, lacking the possibility of asynchronous access.

At present, in a world characterized by the presence of the internet and extended distances, it is imperative that industrial communications evolve, especially in industries such as Oil and Gas, as well as in the transportation of these resources, where real-time information on assets is crucial for decision-making [4]. Innovative companies have installed fiber optic cables alongside pipelines to achieve communication and motion detection [5]. However, the cost of implementing an extensive fiber optic infrastructure and vulnerability to damage, particularly in regions like Colombia, make it necessary to implement wireless communications to avoid loss of information.

In the literature, various authors have made advancements in wireless communications using different protocols over the web [6], [7], [8]. However, some of these solutions prove costly due to high data consumption. Additionally, fixed IP addresses in wireless communications incur additional expenses. To address these challenges, the Publisher-Subscriber (Pub-Sub) technology emerges. Unlike the widely used client-server model, Pub-Sub involves publishers placing messages in an intermediary (broker), which subscribers retrieve using their identifiers. Messages remain in the cloud, ensuring data availability even if the internet connection is intermittent [9].

Among Pub-Sub protocols, MQTT (MQ Telemetry Transport) stands out. MQTT is a lightweight communication protocol with minimal overhead, capable of connecting multiple devices simultaneously, making it a crucial component for Internet of Things (IoT) development. The MQTT protocol has been widely used in IoT, enabling continuous monitoring of equipment and PLCs via Modbus [10], [11], [12]. However, the literature points to the need for bidirectional communication to control native Modbus, as existing applications only allow reading variables and not writing. Therefore, this work proposes bidirectional communication for configuring Modbus registers via the web using MQTT.

Previous developments in the literature often focus on single-use devices or individual machines. This work aims to create a versatile device applicable to different industrial machines, PLCs, and remote monitoring equipment using the Modbus communication protocol [11], [10], [13], [14]. Centralizing multiple devices on a single platform, customized based on the knowledge of connected equipment, without the need for additional wiring, is feasible. It is possible to create a SCADA-like application via the web, leveraging MQTT's ability to integrate multiple subscribers with independent functionalities simultaneously [15].

The security of wireless communication remains a significant concern for the industry, with fears of data loss due to disconnections. To address this, the proposed solution includes an SD memory to ensure data retention. MQTT's Quality of Service (QoS) confirmation ensures data reception by the broker. The SD memory functions as a FIFO, storing and forwarding data during network recovery.

2. DEVELOPMENT

The initial process in the development of the device involves selecting the most appropriate microcontroller and real-time operating system (RTOS) for the project. To make this decision, various essential parameters were considered, such as memory capacity to store the program, microcontroller clock speed, availability of a development board with Ethernet port, use of open-source programming software, and integration with program debugger hardware.

Regarding the RTOS, a comprehensive analysis was conducted covering aspects such as the memory space required by the system, task scheduling capabilities, compatibility with the specific device, scalability, support offered, available documentation, performance obtained, and associated costs. Considering these criteria, the decision is made to opt for the MSP432E

LaunchPad development board from Texas Instruments, along with the RTOS provided by the same company, known as TI-RTOS.

This choice is based on a thorough evaluation that ensures alignment of the project needs with the capabilities and features offered by the MSP432E LaunchPad and TI-RTOS. Both elements complement each other to provide an efficient and reliable development environment.

Next, the Modbus Master library is developed, which acts as an intermediary between the device and the equipment with which communication is established. In this process, it is essential to consider the configurability of the Modbus parameters, ensuring seamless connection capability to a wide range of Modbus devices. Additionally, the aim is to allow simultaneous connection to multiple devices, each with its own distinctive ID.

Once the library creation is completed, comprehensive communication tests are conducted between the microcontroller and various Modbus devices. These tests involved modifying parameters such as serial port speed, parity, Modbus ID, RS232 and RS485 communication methods, number and type of registers, among others. These adjustments allowed for meticulous verification of the functionality of the library designed for this specific task.

The next step involves the development of the necessary code for communication via Ethernet. To achieve this, the configuration of the microcontroller ports is carried out using the relevant RTOS libraries. This configuration encompasses preparing the port and implementing the required steps to establish a TCP/IP connection between the equipment and the Internet infrastructure.

As part of the development and testing phase, connectivity and functionality verifications are carried out using Ping tests. These tests are essential to ensure that the equipment is properly connected to the Internet and that communication is effectively taking place.

Following this, the MQTT communication module is developed. For this purpose, functions focused on connection, disconnection, subscription, and message publishing are implemented. Additionally, the implementation of asynchronous tasks of the RTOS is considered, which are adjusted according to the specific operation mode of MQTT.

A key aspect is the implementation of multiple text formats for data transmission via MQTT. This flexibility allows the end-user to select the most suitable format for the target platform and the desired level of security in the system. Among the incorporated text formats, JSON, BSON, among others, stand out.

Among the notable features of this device, its data storage capacity and queue for transmissions are highlighted, which are useful in situations of intermittent communication. For this purpose, an SD memory is integrated, where data is stored in the selected text format. When the Internet connection is restored, the data is transmitted in the same order in which it was stored on the device. The microcontroller ports configuration was adjusted for SD handling, leveraging available libraries for this task.

Additionally, various customizable peripherals are implemented, such as battery level reading, analog measurements, digital inputs and outputs, and additional communication ports according to specific needs.

In the final coding phase of the device, the implementation of real-time tasks and interrupts for peripherals is addressed. These tasks are assigned to the different activities that the device must carry out periodically and asynchronously. The execution intervals of these tasks can be adjusted by the end-user, allowing for optimal adaptation of the device to the specific application in which it will be used.

After completing this stage, the implementation of the Mosquitto-based broker on a server is carried out. This broker operates continuously and is assigned a static IP address for easy location by connected devices. It is important to highlight that the only entity with a static IP is the broker, as one of the advantages of MQTT is its ability to discover connected devices without the need for a static IP to determine the message delivery address.

Subsequently, the implementation of a MySQL database is required. This database receives data from all devices connected through the device. It should be noted that the device has the ability to connect to any MQTT server configured by the user. However, for testing purposes, the decision was made to implement this database.

As the final stage of development, a web platform is created that allows for the user-friendly visualization of data sent by the different devices. Additionally, this platform allows for modifications to the Modbus registers of the connected devices. This web interface provides the end-user with an intuitive and convenient way to interact with the device and access data efficiently.

In summary, the development of the device involved a series of key stages, from the selection of the microcontroller and RTOS, through the creation of the Modbus Master library, the implementation of MQTT communication, and the configuration of the broker and database, to culminate in the creation of a web platform for data visualization and modification. Each of these stages is approached rigorously, ensuring a versatile and efficient device that can easily adapt to various industrial applications.

3. RESULTS

During the testing phase, all functionalities developed in the device are validated. These tests are conducted with the objective of ensuring that the device meets the design requirements.

In the validation stage, a connection test is performed with a Schneider Electric M221 PLC through the Modbus RS232 interface, using a program that utilizes integer and floating-point registers, as well as read and write registers. This configuration allows for a detailed evaluation of the interoperability and proper communication between the developed device and the PLC. Data transfer tests, Modbus register reading and writing, as well as the device's response to commands and requests sent to and from the PLC, are also conducted. The laboratory test setup can be observed in Fig. 1. The Modbus configuration of the PLC used for this test is shown in Fig. 2.

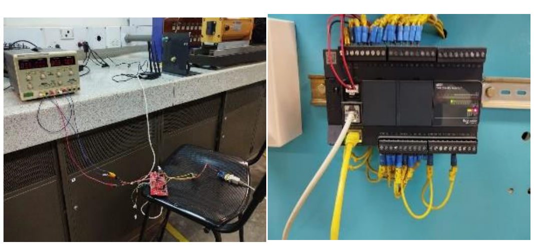

***Fig. 1**. Assembly of the test of the equipment connected to PLC M221.*
***Source:** Own elaboration.*

***Fig. 2**. Serial and Modbus configuration of the PLC M221.*
***Source:** Own elaboration.*

In addition to the communication tests with the PLC, the data storage functionalities and queue for transmissions in intermittent communication situations are evaluated. The device's ability to store data in the SD memory in the selected text format is verified, and this data is retransmitted in the same order after restoring the Internet connection. It is important to note that this data was sent via GPRS communication, using an RUT240 modem connected to the development board via Ethernet.

Below, some tests performed on the device are summarized.

3.1 MQTT Broker Communication Tests

Communication tests are conducted using both a wired connection via Ethernet and a wireless connection through GPRS using a modem. Both methods establish the connection satisfactorily. However, it is observed that the Ethernet connection exhibits higher speed in data transmission and reception compared to wireless communication, which was expected. Although the Ethernet connection is faster, it is important to note that wireless communication is more suitable for this application, as it allows connected devices to operate without depending on physical cable connections, facilitating communication in industrial environments.

The tests were conducted using port 1883 for the MQTT protocol. Data was sent with the topic "PLC/001/W," and subscription to the topic "PLC/001/R" was performed, through which the data used to configure the PLC registers was received.

Fig. 3 illustrates the connection to the MQTT broker, subscription to topics, and encrypted data transmission through the aforementioned topic. This activity is visualized using a serial monitor connected to the microcontroller.

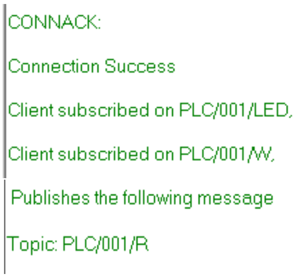

Fig. 3. *Connection to the Broker, Subscription, and Message Publishing.*
Source: *Own elaboration.*

3.2 Communication Tests with Python Script

Other tests of reading and writing data from the PLC are performed using a Python application that connects to the broker over the Internet. This application allows for data logging, reading by subscribing to the topic where the device sends information, and writing to the PLC registers through the topic to which the device is subscribed.

One of the most significant tests of this script involves turning on and off a PLC LED at 10-minute intervals, which activates a counter to record how many times the LED is turned on. The commands sent for this purpose through the Python script are shown in Fig. 4.

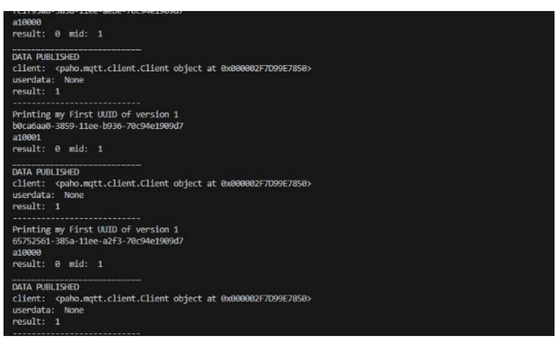

Fig. 4. *LED On and Off from a PC Connected to the Broker.*
Source: *Own elaboration.*

3.3 Communication Tests with Online Platform

The web platform developed for data visualization and Modbus register modification has also undergone a series of tests. User interface, data presentation accuracy, and efficiency in real-time modifications are verified.

The connected device is displayed on an interactive map where, upon clicking the corresponding icon, the current data of the equipment along with its latest information sent is displayed. The PLC location on the map is shown in Fig. 5.

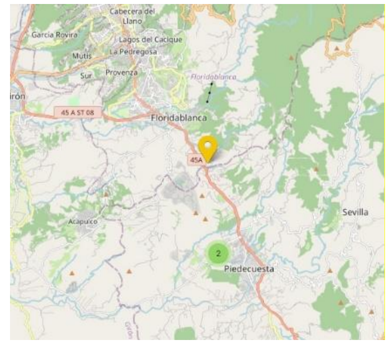

Fig. 5. *Position on Map of Connected PLC.*
Source: *Own elaboration.*

When zooming in for a detailed view, it is possible to visualize graphs representing some of the variables or registers sent by the PLC. An example of one of the analog channels is shown in Fig. 6, where a distance sensor has been connected over a period of one day.

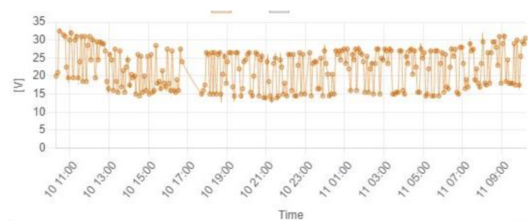

Fig. 6. *Graph of a Day from the Analog Sensor Connected to the PLC*
Source: *Own elaboration.*

In conclusion of the tests, data is sent through the web platform to modify a specific register. This change allows for controlling the on/off state of an LED in the PLC, as well as resetting the system counters.

The results obtained in these tests demonstrate that the device has successfully met the established performance and functionality criteria. Interaction with the Schneider Electric M221 PLC via Modbus RS232 is performed, validating the interoperability between both systems. The data storage functionalities, transmission queue, and web platform also performed as expected, ensuring data integrity and availability under various conditions.

4. CONCLUSIONS

This work involves the selection of components and tools in accordance with the specific needs of the application. As a result, a device capable of establishing bidirectional communication via the Modbus protocol is implemented, facilitating data transfer to an MQTT broker. This process is materialized through the interaction of specialized scripts and web platforms that enable reading and writing of information.

The underlying purpose of this approach is to promote significant advancement in the Colombian industry, steering it towards the prerogatives of Industry 4.0. What has been achieved is a low-cost device that crucially does not require alteration of the existing infrastructure. This innovative solution not only addresses current demand but also paves the way for a smooth and profitable transition to a more advanced and technologically connected industrial environment.

Together, the development and tests conducted confirm the robustness and functionality of the device in laboratory environments. The combination of Modbus communication, MQTT, and the web platform provides a comprehensive solution for efficient monitoring and control of devices, paving the way for successful implementation in various industrial applications that require reliable and efficient communication with devices like the Schneider Electric M221 PLC via the Modbus RS232 protocol.

REFERENCES

[1] E. Lalvay, Diseño y construcción de una interfaz hombre máquina para el monitoreo y supervisión del sobrellenado del tanque V4H en la empresa INTERQUIMEC., 2021.

[2] W. Olarte, M. Botero y B. Cañon, «Importancia del mantenimiento industrial dentro de los procesos de producción,» *Scientia et technica,* vol. 16, nº 44, pp. 354-356, 2010.

[3] Modbus Inc. , «Modbus org.,» [En línea]. Available: https://www.modbus.org. [Último acceso: 17 Agosto 2023].

[4] T. Wanasinghe, R. Gosine, L. James, G. Mann, O. De Silva y P. Warrian, «The internet of things in the oil and gas industry: a systematic review,» *IEEE Internet of Things Journal,* vol. 7, nº 9, pp. 8654-8673, 2020.

[5] P. Salgado Díaz, Estudio y desarrollo de sensores de fibra óptica para detección de vibraciones en ductos ascendentes submarinos, Telecomunicacion, 2013.

[6] Q. Bai, B. Jin, D. Wang, Y. Wang y X. Liu, «Compact Modbus TCP/IP protocol for data acquisition systems based on limited hardware resources,» *Journal of Instrumentation,* vol. 13, nº 4, p. T04004, 2018.

[7] J. Song, . S. Han, A. Mok, D. Chen, M. Lucas, M. Nixon y W. Pratt, «WirelessHART: Applying wireless technology in real-time industrial process control,» *2008 IEEE Real-Time and Embedded Technology and Applications Symposium,* pp. 377-386, 2008.

[8] A. Jestratjew y A. Kwiecien, «Performance of HTTP protocol in networked control systems,» *IEEE Transactions on Industrial Informatics,* vol. 9, nº 1, pp. 271-276, 2012.

[9] N. Defossé , R. López, M. Gómez, P. Konstantinoff, S. Whaler, L. Castro y G. Harris, «Implementación de middleware publicador/subscriptor para aplicaciones web de monitoreo,» de *XIX Workshop de Investigadores en Ciencias de la Computación (WICC 2017, ITBA, Buenos Aires)*, 2017.

[10] D. Torres, A. Soto, D. Coral, P. Parra, H. Muñoz, J. Rojas y N. Córdoba, «Sistema basado en internet de las cosas (IoT) para la monitorización en tiempo real de variables de temperatura y humedad en un equipo de refrigeración de área de farmacia de un hospital de cuarto nivel,» de *Encuentro Internacional de Educación en Ingeniería-EIEI-ACOFI 2019*, 2019, pp. 1-9.

[11] C. Silva y F. Silva, «An IoT gateway for Modbus and MQTT integration,» de *2019 SBMO/IEEE MTT-S International Microwave and Optoelectronics Conference (IMOC)*, IEEE, 2019, pp. 1-3.

[12] M. Tyrovolas y T. Hajnal, «Inter-communication between Programmable Logic Controllers using IoT technologies: A Modbus RTU/MQTT Approach,» *arXiv preprint arXiv:2102.05988,* 2021.

[13] M. Ali, A. Miry y T. Salman, «IoT based water tank level control system using PLC,» de *2020 International Conference on Computer Science and Software Engineering (CSASE)*, IEEE, 2020, pp. 7-12.

[14] S. Palacio, Diseño de un sistema de monitorización y control energético basado en IoT, Universitat Politécnica de Catalunya, 2021.

[15] MQTT org, «MQTT org,» [En línea]. Available: https://mqtt.org/ . [Último acceso: 14 agosto 2023].

[16] Modbus Inc., «Application protocol V1,» [En línea]. Available: https://www.modbus.org/docs/Modbus_Appli

cation_Protocol_V1_1b3.pdf . [Último acceso: 15 Agosto 2023].