Robot móvil con set de instrucciones de control de movimiento sobre protocolo Modbus RTU

José Maximiliano Mendoza Martínez[1,2], Irma Yolanda Sánchez Chávez[1,3], Fernando Martell Chávez[4], Víctor Arturo Maldonado Ruelas[1] y Raúl Arturo Ortiz Medina[1]

[1] Universidad Politécnica de Aguascalientes, Dirección de Posgrado e Investigación, Calle Paseo San Gerardo No. 207, Fracc. San Gerardo C.P.20342 Aguascalientes, Ags., México, mc240020@alumnos.upa.edu.mx, irma.sanchez@upa.edu.mx, victor.maldonado@upa.edu.mx, raul.ortiz@upa.edu.mx

[2] Universidad Tecnológica El Retoño, Departamento de Carreras Técnicas, Carr. San Luis Potosi - Aguascalientes, C.P. 20337 El Llano, Ags., México, jose.mendoza@utr.edu.mx

[3] Ingeniería Mecatrónica, S.A de C.V., Córdoba No. 204, Fracc. El Dorado C.P. 20235 Aguascalientes, Ags., México, isanchez@ingmt.com

[4] Centro de Investigaciones en Óptica, Prol. Constitución 607, Fracc. Loma Bonita C.P.20200 Aguascalientes, Ags., México, fmartell@cio.mx

Resumen

Los robots móviles en la automatización son un recurso para mejorar la eficiencia y la seguridad de diversas operaciones tanto en la industria como en la vida cotidiana. Su funcionamiento confiable depende de la comunicación y ejecución de instrucciones básicas de movimiento. Su interoperabilidad con otros sistemas es fundamental para su integración y coordinación para la realización de tareas más complejas. Respecto a los robots móviles, hay una alta dependencia tecnológica del extranjero en México. En esta contribución documentamos el desarrollo de un robot móvil con tracción diferencial para permitir la ejecución estable de instrucciones de movimiento mediante el protocolo abierto de comunicación industrial Modbus. El protocolo Modbus se integra así al lenguaje de operación del robot para homologarse con el lenguaje de comunicación de otros sistemas de automatización. Para el desarrollo de nuestro prototipo, se adaptó la metodología VDI 2206, la cual incluye las fases de especificación funcional y de componentes, diseño de conjunto de instrucciones para los modos de operación, implementación del control y pruebas integrales de comunicación y funcionalidad. Se definen distintos modos de operación en el sistema de control para proveer una funcionalidad básica y amigable del robot móvil. La arquitectura del sistema integra una interfaz de usuario en LabVIEW® para la especificación de instrucciones a un alto nivel y un microcontrolador Arduino para el control de bajo nivel de los motores a pasos, la adquisición de datos y cálculos de odometría. Para la comunicación de estos dos elementos se utiliza el protocolo Modbus RTU: se configuran la interfaz del usuario como maestro y el microcontrolador como esclavo. Los resultados demuestran la capacidad del sistema de para seguir trayectorias cartesianas.

***Palabras clave*— Control de movimiento, Modbus RTU, Odometría, Robot de tracción diferencial, Simulación hardware-in-the-loop.**

Movil robot with set of instructions for movement control on Modbus RTU protocol

Abstract

Mobile robots in automation are a resource to improve the efficiency and safety of various operations both in industry and in daily life. Their reliable operation depends on the communication and execution of basic movement instructions. Their interoperability with other systems is essential for their integration and coordination to carry out more complex tasks. Regarding mobile robots, Mexico has a high technological dependence on foreign countries. In this contribution we document the development of a mobile robot with differential traction to enable the stable execution of movement instructions using the open industrial communication protocol Modbus. The Modbus protocol is thus integrated into the robot's operating language to align with the communication language of other automation systems. For the development of our prototype, the VDI 2206 methodology was adapted, which includes the phases of function and component specification, design of instruction set for operating modes, control implementation, and comprehensive communication and functionality tests. Different modes of operation are defined in the control system to provide the mobile robot with basic and user-friendly functionality. The system architecture integrates a LabVIEW® user interface for high-level instruction specification and an Arduino microcontroller for low-level control of stepper motors, data acquisition, and odometry calculations. For the communication of these two elements, the Modbus RTU protocol is used: the user interface is configured as master and the microcontroller as slave. The results demonstrate the system's ability to follow Cartesian trajectories.

***Keywords*— Differential drive robot, Hardware-in-the-loop simulation, Modbus RTU, Movement control, Odometry.**

Recibido: 2 de diciembre de 2025
Aceptado: 27 de mayo de 2026
Publicado: 3 de junio de 2026

I. INTRODUCCIÓN

Los robots móviles tienen en la actualidad mucha relevancia en la automatización para la ejecución de tareas de manera remota y en forma guiada o autónoma. Sus aplicaciones abarcan servicios domésticos, el traslado de materiales en la industria de manufactura [1] y en el sector salud [2], el monitoreo de cultivos y la fumigación en la agricultura [3], y la exploración espacial [4], entre otras.

En México, la Secretaría de Economía reporta que el 90% de los robots móviles son importados de países como Japón, Alemania, China, Estados Unidos y Corea del Sur. En 2023, la importación de robots industriales fue de 239 millones de dólares, 7.8% mayor respecto a 2022. Aun cuando en el 2024, la importación bajó a 192 millones de dólares, México sigue siendo un mercado importante de robots industriales en América Latina [5] promovido por la automatización de las industrias automotriz y electrónica principalmente [6]. La disminución en importaciones propicia oportunidades para que universidades y empresas mexicanas proporcionen soluciones propias, y también para diversificar inversiones en otros sectores. De aquí el interés por desarrollar un robot móvil general que pudiera luego adaptarse para una actividad particular.

La integración de los robots móviles en sistemas automáticos necesita de protocolos de comunicación robustos. En la industria, el protocolo Modbus ha sido ampliamente utilizado bajo dicho requerimiento, por lo que constituye una opción competitiva con respecto a otros recursos más especializados, como el sistema operativo ROS (del inglés Robotic Operating System) [7], facilitando la interacción con controladores lógicos programables (PLC). Aunque el protocolo TCP (Transmission Control Protocol) ha permitido resolver la interoperabilidad en aplicaciones con equipos de múltiples marcas [8], en el prototipo aquí presentado se inicia con la versión RTU de Modbus para el desarrollo de funcionalidades básicas y en preparación para el posterior uso de la versión TCP.

Los robots móviles actualmente consideran la incorporación del Internet de las cosas (IoT, del término en inglés), lo cual implica ciertas funcionalidades y tecnologías. El uso de protocolos Wi-Fi es soportado por plataformas de programación, protocolos de comunicación y sistemas operativos especiales.

Equipos computacionales portables como Raspberry Pi [9] y Arduino [10] permiten el prototipado rápido de sistemas robóticos facilitando la programación y la conexión de instrumentación. Las capacidades de la microcomputadora Raspberry Pi facilitan el manejo de imágenes que es clave en diversas tareas de exploración robótica y la ejecución de algoritmos complejos [11]. El microcontrolador Arduino, generalmente a cargo del control directo de motores, puede ser eliminado ante la posibilidad de simplificar la arquitectura y programar todo el código necesario en una Raspberry Pi [12], o bien, bajo una funcionalidad mínima, puede optarse por utilizar sólo el Arduino.

Respecto a los protocolos de comunicación, Modbus es reconocido por su importancia en la industria, y también como una opción de comunicación de robots. Las versiones RTU y TCP pueden aprovecharse en distintos contextos de investigación y desarrollo, dada la disponibilidad del código abierto para hacer adecuaciones [13], [14]. Modbus RTU ha sido utilizado con éxito en proyectos de IoT aprovechando su seguridad atribuida a la conexión por cable RS-485 [15]. La variante Modbus-S, basado en Modbus TCP, ofrece resolver desventajas de seguridad o confidencialidad mediante funciones de autenticación, cifrado simétrico, firma digital y filtrado de códigos de función [16]. Otros protocolos de comunicación serial utilizados con robots son I2C (del inglés Inter-Integrated Circuit) [17], UART (Universal Asynchronous Receiver/Transmitter) y SPI (Serial Peripheral Interface) [18], pero tienen problemas de interferencias y están limitados a distancias cortas, lo cual reduce su aplicabilidad industrial.

La necesidad de operación a tiempo real es importante en la operación de los robots y depende de los sistemas operativos. Algunos investigadores han utilizado RTLinux para la ejecución de funciones de control y comunicación [19], caracterizadas por su intensivo procesamiento. Sistemas operativos de uso general como Ubuntu o Windows también soportan la comunicación y el control eficiente de robots móviles [15]. El sistema ROS consiste en una biblioteca de funciones que facilitan la programación de robots y se hacen cargo de ciertas funciones del sistema operativo para la abstracción de hardware [20]. Sin embargo, presenta desventajas para su integración con controladores lógicos programables (PLC del inglés) y coordinación con otras maquinarias.

Los avances en la navegación de los robots móviles han involucrado control óptimo [21], y técnicas de localización y mapeo simultáneo (SLAM del término en inglés) que involucran filtros de Kalman e inteligencia artificial [22] y logran identificar características del terreno, obstáculos y lugares distintivos al mismo tiempo que el robot se desplaza. Dichas técnicas pueden lograr un desempeño superior y autónomo, pero a mayor costo computacional.

De entre los recursos mencionados con relación a proyectos de investigación en el área, la propuesta de este trabajo considera apropiados el uso del microcontrolador Arduino y de una interfaz gráfica en una computadora personal comunicados con Modbus RTU para una etapa inicial y genérica de desarrollo de un robot móvil que facilite su posterior integración en sistemas del IoT bajo distintas aplicaciones. El trabajo aquí reportado contribuye con la definición de modos básicos de operación prescindiendo de mediciones físicas mediante el uso de funciones de odometría que aprovechan las características de los motores a pasos.

Aún y cuando el protocolo Modbus es bien conocido en procesos industriales, se ha utilizado poco en la robótica móvil moderna, por lo que existe la oportunidad de innovación en manejarlo para dar interoperabilidad de los robots móviles con los sistemas industriales actuales. El protocolo Modbus se ha utilizado con robots móviles en forma específica y limitada para el control de motores, lectura de sensores y comunicación con equipos de control, típicamente PLC. Es decir, Modbus se ha configurado más como un elemento de interfaz industrial que

como un elemento de un sistema robótico. La novedad de esta investigación está en estructurar un conjunto de instrucciones sobre Modbus RTU, es decir, incluir el protocolo como parte del lenguaje de movimiento del robot para facilitar su integración con sistemas automáticos. Asociada a cada instrucción de movimiento, se tiene una programación que busca precisión en el control de posición y orientación. Tecnológicamente, este trabajo contribuye a una alternativa de bajo costo a robots industriales propietarios y a una plataforma educativa industrial realista.

El artículo se organiza de la siguiente manera. La segunda sección describe la arquitectura del sistema, presenta los componentes principales, y explica la metodología de trabajo aplicada dada la naturaleza mecatrónica del proyecto. La tercera sección muestra el desarrollo de los modos de operación y las estrategias de control y comunicación utilizadas. En la cuarta sección se muestra el desempeño del robot móvil y se discuten cuestiones prácticas de implementación. Finalmente, en la quinta sección se presentan conclusiones respecto al avance logrado y trabajo futuro para mejorar el prototipo.

II. Materiales y Metodología

El robot móvil propuesto es la implementación física del robot móvil virtual reportado en [23]. El robot móvil integra una plataforma con tracción diferencial, un sistema de control de los actuadores y una interfaz de usuario para su operación y monitoreo. Estos dos últimos elementos se comunican con un protocolo serial con el fin de probar la funcionalidad básica del prototipo a bajo costo. La Fig. 1 muestra la arquitectura del robot móvil.

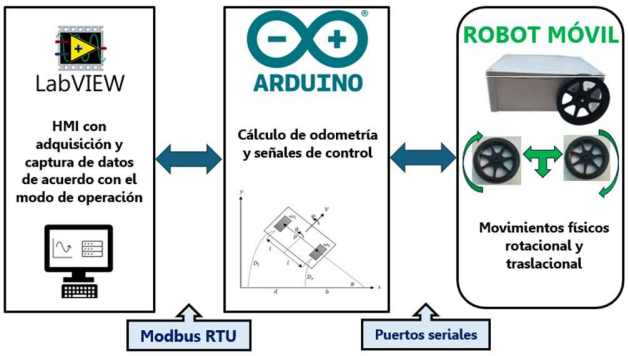

Fig. 1. Arquitectura del robot móvil.

El robot móvil de tracción diferencial se construyó con los componentes listados en la Tabla I.

Se utilizó un controlador Arduino para realizar el control de los motores a pasos e implementar los modos de operación. La interfaz gráfica del usuario se elaboró en LabVIEW®. La comunicación se configuró de acuerdo con el protocolo Modbus RTU. En la Fig. 2 puede verse físicamente el prototipo. En la Fig. 3 se muestran las conexiones de los componentes de hardware.

TABLA I
Componentes eléctricos, electrónicos y mecánicos

Elemento	Cantidad	Descripción
Arduino Uno R3	1	Microcontrolador de prototipado rápido
Driver TB6560	2	Controlador robusto para motores a pasos
Motor de pasos 29SMK250P1V	2	Marca de motores Matsushita
Regulador LM25-96	1	Regulador eficiente de voltaje DC conmutado ajustable
Batería de 12Vcc-7.2Ah	1	Batería recargable compacta de capacidad de 7.2Ah
Carcasa	1	Caja de IP67 de 29 cm de ancho, 34 cm de largo y 12.5 cm de alto
Llantas	2	Impresas con filamento de PLA y TPU con un diámetro de 15 cm
Rueda caster	1	Giro libre, 2.2 cm diámetro

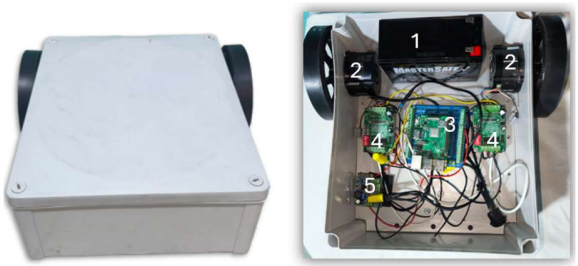

Fig. 2. Prototipo de robot móvil: batería (1), motores (2), Arduino (3), drivers de motores (4) y regulador de voltaje (5).

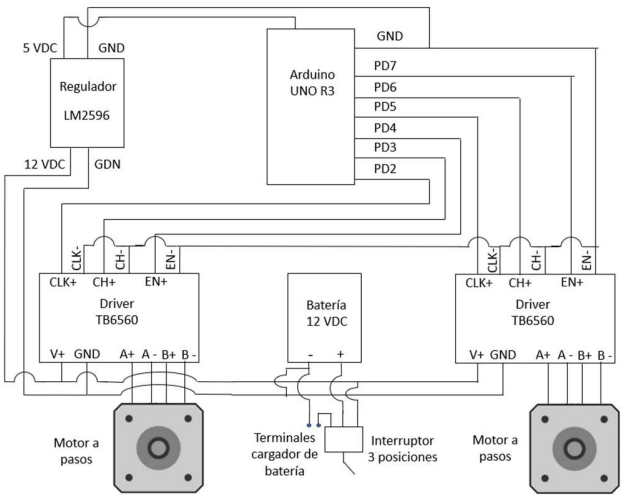

Fig. 3. Conexiones de componentes de hardware.

Los actuadores se simularon de acuerdo con las relaciones fundamentales de los motores a pasos. El ángulo de giro α producido por cada pulso aplicado al motor está dado por

$$\alpha = \frac{360°}{n_s}, \quad (1)$$

donde n_s es el número de pasos del motor o pulsos necesarios para una revolución. La frecuencia de pulsos por segundo f_p de entrada al motor desarrolla una velocidad angular de la flecha N en rev/s expresada por

$$N = \frac{f_p}{n_s}. \quad (2)$$

La cinemática del robot se describe con las siguientes ecuaciones [24]. La velocidad de cada rueda es proporcional a la velocidad angular de su motor:

$$v_i = 2\pi\, r\, N_i, \quad (3)$$

donde r es el radio de la rueda, y el subíndice $i \in \{r, l\}$ puede indicar rueda derecha $(i = r)$ o izquierda $(i = l)$. Las velocidades lineal V y angular Ω del robot están determinadas por las velocidades lineales de las ruedas:

$$V = \frac{v_r(k)+v_l(k)}{2}, \quad (4)$$

$$\Omega = \frac{v_r(k)-v_l(k)}{L}, \quad (5)$$

donde L es la separación entre las ruedas.

Se adoptó la metodología mecatrónica VDI 2206 [24] de la Asociación de Ingenieros Alemanes. Esta metodología establece un marco de desarrollo apropiado para proyectos interdisciplinarios que requieren la coordinación de las áreas de mecánica, electrónica y software. En este trabajo se aplicaron las diferentes fases de la metodología para la realización de especificación funcional del robot móvil, selección de instrumentación necesaria, diseño de modos de operación y set de instrucciones correspondientes, diseño de interfaz de operación, programación de comandos de movimiento, verificación del sistema mediante simulación hardware-in-the-loop y validación del sistema en un entorno real.

En la Fig. 4, puede visualizarse el esquemático en 'V' de la metodología VDI 2206 con los múltiples ciclos iterativos que asocian las etapas de integración con las etapas de diseño. Los distintos tipos de simulación permiten la ejecución simultánea de diferentes etapas de desarrollo: la etapa de modelación y simulación con ecuaciones (2) a (5), puede ir más allá de la representación del sistema, extendiéndose a una simulación software-in-the-loop que permite explorar el control sin interferir en la construcción del prototipo, y la etapa de integración de hardware se facilita con la simulación hardware-in-the-loop al probar el equipo de control programado con el sistema simulado previo a la interacción con el sistema real, dando oportunidad de terminar la construcción del sistema y probar la estrategia de control al mismo tiempo.

III. Operación, Control y Comunicación

Para el robot móvil se definieron cuatro modos de operación (Tabla II), los cuales están determinados por los datos especificados por el usuario: manual (frecuencia de pulsos a aplicar en cada motor), giro (ángulo de orientación), desplazamiento (distancia sin variación de ángulo), desplazamiento XY (punto cartesiano a alcanzar), y desplazamiento direccionado (ángulo y distancia).

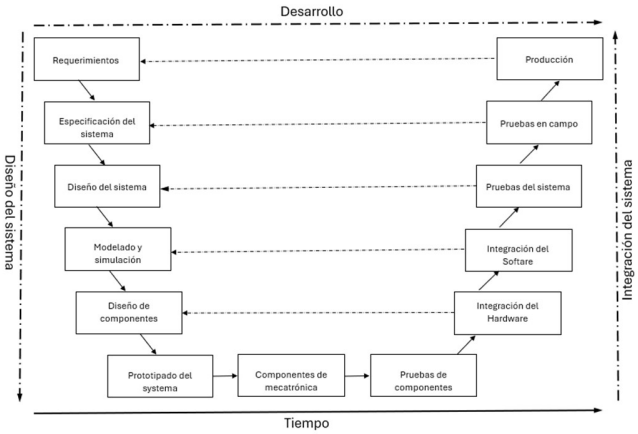

Fig. 4. Metodología VDI 2206 (adaptada de [24]).

TABLA II
Modos de Operación del Robot Móvil

Modo	Dato 1	Dato 2	Acción
Manual	Velocidad motor izquierdo	Velocidad motor derecho	Avance, giro horario o antihorario
Giro	Ángulo	-	Rotación sin avance
Desplazamiento	Distancia	-	Movimiento sin cambio de dirección
Desplazamiento XY	X	Y	Traslación hacia punto en plano XY
Desplazamiento y dirección	Ángulo	Distancia	Giro y traslación en dirección indicada

Para el cálculo de posición (x_k, y_k) y orientación θ_k, se implementó una función de odometría basada en las siguientes ecuaciones:

$$x_k = x_{k-1} + f_{int_d} cos\left(\frac{\pi\,\theta_k}{180}\right), \quad (6)$$

$$y_k = y_{k-1} + f_{int_d} sin\left(\frac{\pi\,\theta_k}{180}\right), \quad (7)$$

$$\theta_k = \theta_{k-1} \pm f_{int_a}, \quad (8)$$

donde f_{int_d} es el factor de integración de desplazamiento y f_{int_a} es el factor de integración de ángulo (ambos por pulso), las coordenadas se manejan en centímetros y el ángulo en grados. El factor de integración del desplazamiento está dado por

$$f_{int_d} = 2\pi r/n_s, \quad (9)$$

y el factor de integración de ángulo se calcula como

$$f_{int_a} = (2r/L)(360°/n_s). \quad (10)$$

El prototipo utiliza motores de n_s =200, con una separación de llantas L =32 cm, por lo que los factores de integración calculados fueron $f_{int_d} = 0.188$ y $f_{int_a} = 0.844$.

En el modo manual, el usuario determina la frecuencia de pulsos a aplicar en el motor de cada rueda; si las frecuencias especificadas son iguales el robot avanza sin cambio de

dirección, si la frecuencia del motor derecho es mayor que la frecuencia del motor de la izquierda, el robot gira hacia la izquierda, y si la frecuencia del motor izquierdo es mayor que la frecuencia del motor derecho, el robot gira a la derecha.

Los modos de operación distintos del manual utilizan un algoritmo retroalimentado para lograr el movimiento especificado. Aún y cuando los motores a pasos tengan un comportamiento bien definido y sea posible predeterminar los pulsos a aplicar para lograr determinado giro, el control de la posición y de la distancia recorrida se basa en el error o diferencia entre el valor deseado y el actual conforme se aplica cierta frecuencia de pulsos. Dicha frecuencia es el único parámetro de control y se asignó como 400 pulsos/s, el cual determina el tiempo de muestreo de 2.5 ms, valor razonable en control de movimiento y que puede modificarse. Este algoritmo es capaz de lograr con precisión el desplazamiento o el giro deseado ante la eventualidad de pérdida de pulsos, pues se basa en el error o diferencia detectada entre el movimiento especificado y el logrado, y no en la contabilización de pulsos enviados. El algoritmo es binario en su salida por basarse en la aplicación o no aplicación de una frecuencia predeterminada, sin embargo, ejerce un control continuo por responder al error en posición o ángulo, y logra su eliminación para alcanzar cualquier valor deseado con una tolerancia por el efecto natural de integración de velocidad tanto lineal como angular para obtener los correspondientes desplazamientos lineal y angular. La Fig. 5 muestra el algoritmo de control retroalimentado utilizado.

El protocolo Modbus RTU se selecciona para el desarrollo del conjunto de instrucciones por su aplicabilidad industrial. Es decir, el prototipo incluye el recurso industrial que implica la capacidad de transmisión de datos a largas distancias, métodos robustos para verificación de datos, como el método de redundancia cíclica, y la compatibilidad con redes de control y automatización, aún y cuando el funcionamiento del prototipo se verifique en distancias pequeñas y en una red de control con sólo dos elementos, el controlador y la interfaz desarrollada en LabVIEW®. Es decir, el prototipo incorpora el recurso de comunicación que le permitiría integrarse a un entorno industrial.

Para fines de prototipaje en la fase de desarrollo que se reporta se utiliza comunicación serial alámbrica para realizar las pruebas iniciales, por simplicidad y bajo costo.

La comunicación Modbus RTU se configura entre la interfaz de LabVIEW® o maestro y el controlador Arduino o esclavo, con baud rate de 19,200 y paridad par. Se define un banco de registros conformado por 3 registros de escritura ("holding registers" con uso de escritura y lectura) y 8 registros de entrada o de lectura para el maestro. El usuario utiliza los 3 registros de escritura para seleccionar el comando y enviar hasta dos parámetros para el movimiento deseado. Los registros de lectura son utilizados por el controlador para enviar los cálculos de posición requerida en las coordenadas X y Y, el ángulo requerido y la distancia requerida, así como las posiciones en X y Y actuales, el ángulo actual y el desplazamiento actual del robot. Los registros se manejan con los métodos *Hreg* para que el controlador adquiera los datos escritos en la interfaz o

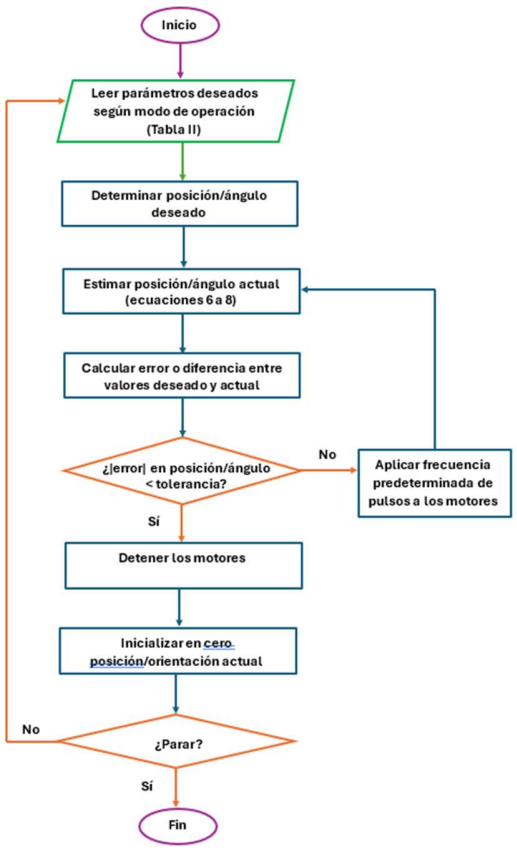

Fig. 5. Algoritmo de control retroalimentado.

maestro de la comunicación, y *setIreg* para que el controlador envíe los valores calculados a ser leídos por la interfaz.

En la Fig. 6 se aprecia una guía general que integra los elementos del sistema y describe su operación: en el primer paso se definen los parámetros de comunicación Modbus RTU para establecer la conexión; en los pasos 2 y 3 el usuario especifica el modo de operación y los datos necesarios para el movimiento deseado, respectivamente, y los motores de las llantas son controlados de manera que el robot ejecuta el movimiento especificado (según diagrama de Fig. 5); en el paso 4 se visualiza en tiempo real como cambian la posición y la orientación del robot, y pueden compararse contra sus valores deseados o especificados; y en el paso 5 se prepara el sistema para la ejecución de otro comando de movimiento asignando valores nulos a la posición y orientación.

La simulación hardware-in-the-loop permitió verificar en forma preliminar las distintas funciones de movimiento especificadas. Este tipo de pruebas también ayudó a verificar la comunicación Modbus RTU entre el microcontrolador y la interfaz. Una vez, habilitado el robot móvil, se validaron experimentalmente los modos de movimiento. En la interfaz en LabVIEW®, mostrada en la Fig. 7, se especificaron los comandos y se visualizaron en tiempo real las variables de posición y ángulo, tanto en la en la simulación como en la operación del robot físico.

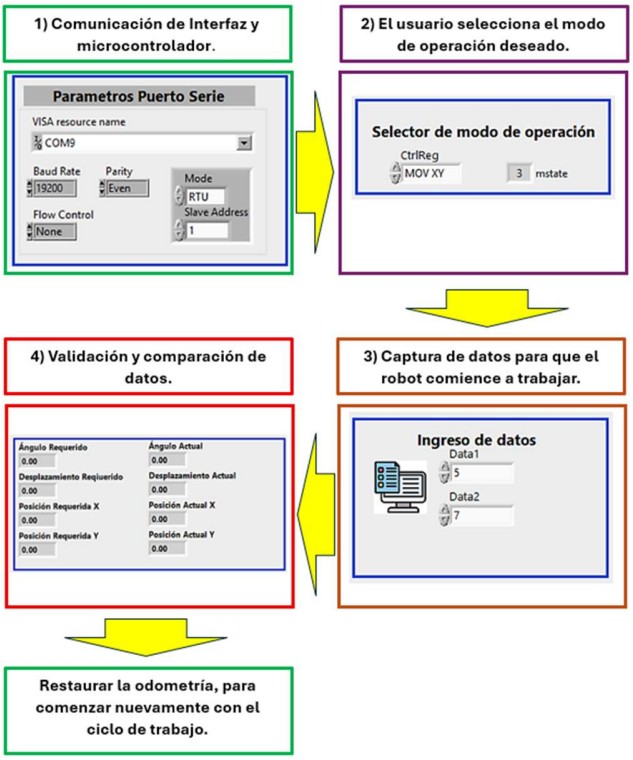

Fig. 6. Guía general operativa.

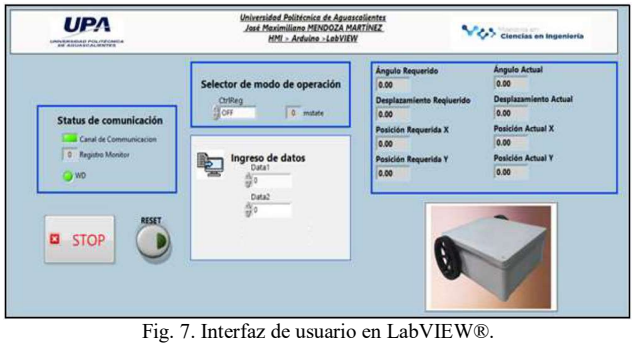

Fig. 7. Interfaz de usuario en LabVIEW®.

IV. DESEMPEÑO

El robot móvil prototipado logró operarse de acuerdo con los distintos modos propuestos. Las Fig. 8 a 10 muestran la evolución de la posición y la orientación en el tiempo de acuerdo con los comandos del usuario. En estas figuras, las verticales delimitan, en forma aproximada, las zonas de graficado que corresponden a los distintos comandos o pruebas. Las pruebas se enumeran en forma consecutiva a través de las Fig. 8 a 10. Las posiciones alcanzadas ante cada comando pueden verificarse mediante relaciones trigonométricas básicas, aún y cuando se tiene una exactitud limitada aritméticamente por redondeo y la tolerancia de error manejada, y físicamente por la resolución o el número de pasos de los motores.

En la Fig. 8 se aplica una secuencia de cuatro comandos o pruebas utilizando los modos *Giro* y *Desplazamiento*. En la prueba 1, partiendo de la posición en coordenadas $(0, 0)$ y de orientación de 0°, se pide un giro de 25° en el tiempo, t, de $0.7\ s$, el cual se logra prácticamente en $t = 6.6\ s$, con un error de 0.1°. En la prueba 2, en $t = 15.5\ s$ se solicita el segundo movimiento consistente en un desplazamiento de $50\ cm$ sin cambio de dirección, es decir, manteniendo la orientación de 25° lograda anteriormente, por lo que el robot parte de la posición $(0, 0)$ y llega a la posición $(45.3, 21.1)\ cm$ en $t = 25.2\ s$ con un error de desplazamiento de 0.027 cm. La prueba 3 o tercera instrucción del usuario consiste en adoptar la orientación de $-25°$ sin desplazarse, y se da en $t = 29.9\ s$ logrando el ángulo requerido en $t = 40.2\ s$ con un error de -0.1°. Finalmente, en la prueba 4, el comando especificado en $t = 43.7\ s$ es desplazar el robot $50\ cm$ en la dirección recién adquirida de $-25°$, lo cual se concreta en $t = 53.3\ s$ alcanzando el punto $(90.6, 0)$ con un error de desplazamiento de 0.027 cm. En estas pruebas se observa que la magnitud de los errores se repiten ante especificaciones con misma magnitud con una exactitud de 99.946% en desplazamiento (100-0.027*100/50) y 99.6% en ángulo (100-0.1*100/25).

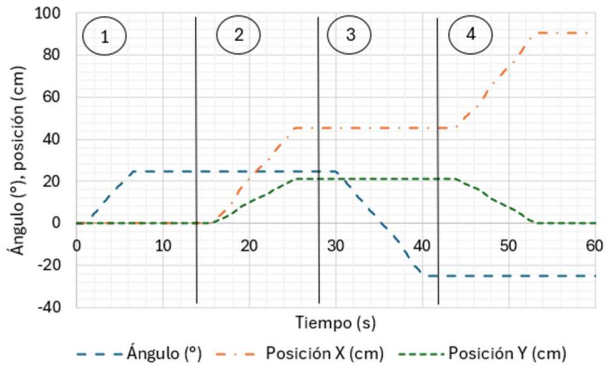

Fig. 8. Monitoreo del robot móvil alternando modos de *Giro y Desplazamiento*: 1) ángulo deseado: 25°; 2) desplazamiento deseado: 50 cm; 3) ángulo deseado: -25°; 4) desplazamiento deseado: 50 cm.

La Fig. 9 muestra la operación bajo el modo *Desplazamiento XY*. El usuario proporciona dos comandos, uno en la prueba 5 y otro en la prueba 6. En la prueba 5, en $t = 3.4\ s$ se solicita alcanzar las coordenadas $(25, 43.3)\ cm$ a partir del origen. El robot primero adquiere la orientación correspondiente de 59.8° (con error de 0.2° respecto al ángulo deseado de 60° implícito en el cambio del origen hacia el punto especificado) en $t = 17.6\ s$, e inmediatamente después se desplaza linealmente llegando al punto $(25, 42.9)$ en $t = 27.8\ s$ con un error en X de 0 cm y un error en Y de 0.4 cm. Luego, en la prueba 6, el usuario en $t = 32\ s$ especifica las coordenadas objetivo de $(43.3, 25)\ cm$, el robot adquiere la orientación de $-44.7°$ (con error de -0.3° respecto al ángulo deseado de -45° implícito en el cambio del punto (25, 43.3) al punto (43.3, 25)) en $t = 54.7\ s$, y llega a la posición $(42.9, 25)$ en $t = 60\ s$ con un error en X de 0.4 cm y un error en Y de 0 cm. En estas pruebas se obtiene una exactitud del 99.07% (100-0.4*100/43.3) considerando el error más grande obtenido en ambas coordenadas (0.4) con respecto al valor especificado (43.3).

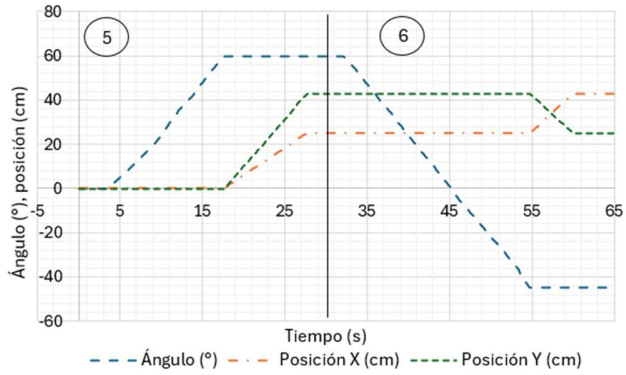

Fig. 9. Operación del robot móvil en modo *Desplazamiento XY*: 5) coordenadas deseadas (25, 43.3) cm; 6) coordenadas deseadas (43.3, 25) cm.

La Fig. 10 ilustra el uso del modo de operación *Desplazamiento y dirección*, mediante las pruebas 7 y 8. En la prueba 7, el usuario especifica en $t = 0.2\ s$ un ángulo de 60° y una distancia a recorrer de $100\ cm$. A partir de estos datos se calcula la posición deseada de (50, 86.6). Nuevamente, primero se adquiere la orientación de 59.9°, con un error de 0.1°, en $t = 15.7\ s$, y luego el robot se desplaza linealmente hasta el punto $(50, 86.5)$, alcanzándolo en $t = 37.9\ s$ con un error de desplazamiento de 0.089 cm (y errores en X y Y de 0 y 0.1 cm, respectivamente). En la prueba 8, en $t = 44.3\ s$, el usuario captura un ángulo de 30° y un desplazamiento de $100\ cm$. Con estas especificaciones se espera llegar a la posición deseada de (136.6, 136.5). El robot rota hasta alcanzar el ángulo de 30.1° en $t = 51.2\ s$, con un error de -0.1°, y después alcanza el punto $(136.5, 136.4)$ con un error de desplazamiento de 0.139 cm (y errores en X y Y de 0.1 cm). La exactitud de estas pruebas en dirección sería de 99.67% (100-0.1*100/30), y en desplazamiento de 99.86% (100-0.139*100/100).

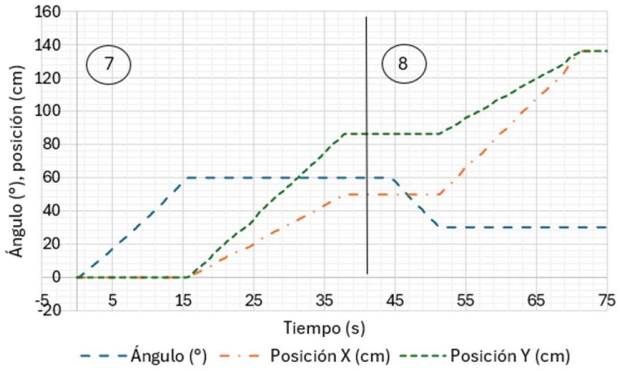

Fig. 10. Movimiento del robot en modo *Desplazamiento y dirección*: 7) valores deseados de desplazamiento de 100 cm y dirección de 60°; 8) valores deseados desplazamiento de 100 cm y dirección de 30°.

En las primeras pruebas con el prototipo, los movimientos se realizaban, pero sin alcanzar con exactitud la posición y orientación especificados. Esto era debido a la necesidad de calibrar los factores de integración tanto del giro sobre el eje vertical del robot como del desplazamiento rectilíneo para que coincidieran con los movimientos físicos del robot. Dicho procedimiento se realizó a partir de ajustar las posiciones alcanzadas de acuerdo con los comandos y sus parámetros de ángulo y distancia (modos *Giro* y *Desplazamiento*) aplicados, verificando las ecuaciones (9) y (10). Esto permitió también obtener exactitud en las coordenadas de posición X y Y para los modos *Desplazamiento XY* y *Desplazamiento y dirección*. Las gráficas mostradas (Fig. 8 a 10) reflejan movimientos correctamente medidos y controlados.

Respecto a los resultados mostrados es necesario advertir que las exactitudes cuantificadas no tienen un valor estadístico, por relacionarse solo con las pruebas descritas, las cuales son muy limitadas en cantidad. Las pruebas ilustran la ejecución de los movimientos de acuerdo con los modos de operación definidos. Los errores obtenidos reflejan suficiencia para la aplicación del robot en tareas de precisión moderada como traslado de materiales. Las pruebas se realizaron sobre superficies planas y que no propiciaran el deslizamiento de las llantas. Los modos de operación combinados o que implican giro y desplazamiento, *Desplazamiento XY* y *Desplazamiento y dirección*, realizan primero el cambio de orientación y luego la traslación lineal aplicando la misma frecuencia de pulsos a ambas llantas. Es decir, no se ajusta orientación y posición del robot al mismo tiempo.

V. CONCLUSIÓN

La presente investigación aporta al estudio de la integración entre sistemas de robótica móvil y protocolos de comunicación industrial, específicamente mediante el uso de Modbus RTU como medio de control y transmisión de instrucciones de movimiento. Modbus RTU continúa siendo ampliamente utilizado en entornos industriales debido a su robustez, bajo costo y facilidad de integración con dispositivos como PLC y variadores de frecuencia [25], lo que respalda su aplicación en sistemas de control automático. Asimismo, esta investigación se alinea con estudios recientes que destacan la importancia de los protocolos industriales tradicionales dentro de arquitecturas modernas de automatización e IoT [26]. La integración de protocolos como Modbus en sistemas robóticos y dispositivos industriales (como PLC o sistemas SCADA) permite mejorar la interoperabilidad entre hardware heterogéneo, facilitando el desarrollo de sistemas distribuidos de control [27]. Por lo tanto, el sistema desarrollado no solo representa una solución funcional de robótica móvil, sino también una plataforma escalable para su integración en sistemas de automatización y monitoreo industrial.

El trabajo aquí presentado detalla el desarrollo de un prototipo de robot móvil de tracción diferencial utilizando recursos de control y comunicación de bajo costo. El sistema de control se basa en un microcontrolador comercial de plataforma abierta. El uso de motores a pasos simplifica la manipulación del robot por su naturaleza predecible o bien determinada. La comunicación con Modbus permite aprovechar un protocolo abierto de amplio uso industrial. El uso de Modbus RTU, aún y cuando se utilizó cableado, facilitó las comunicaciones para las pruebas y operación del robot. Se utilizó una interfaz de LabVIEW® como recurso de prototipaje para la especificación de las instrucciones de movimiento.

El prototipo logra una funcionalidad básica y robusta de movimiento. Se procesan comandos de movimiento de giro y desplazamiento, ya sea con parámetros de dirección y distancia,

o bien, con coordenadas cartesianas X y Y. El control retroalimentado del movimiento mediante cálculos odométricos permitió dar la funcionalidad deseada en forma rápida, efectiva y repetitiva.

Este desarrollo ubica al prototipo de investigación en un nivel de maduración tecnológica TRL (del inglés Technology Readiness Level) de 4, lo cual corresponde a la realización de pruebas del prototipo en laboratorio para corroborar la integración y operabilidad del sistema.

El prototipo permitirá la implementación de movimiento simultáneo de posición y ángulo de acuerdo a una trayectoria de forma más general (curvilínea) ya sea especificada por el usuario o determinada por un algoritmo de optimización. También podrán incorporarse sensores de visión o sensores láseres (LiDAR) para apoyar modos avanzados de navegación. Respecto a la comunicación, el protocolo Modbus RTU permite la transmisión confiable de datos por radiofrecuencia para así obtener comunicación inalámbrica como posibilidad de mejora posterior. Con el desarrollo reportado se tiene una base que facilitará la transición a Modbus TCP por medio inalámbrico WiFi para aplicaciones remotas e incorporación del IoT. El prototipo del robot móvil podrá ser incorporado a aplicaciones prácticas de vigilancia, traslado de materiales, u otro tipo de servicio. Además, el robot móvil podrá utilizarse como una plataforma de investigación y experimentación de técnicas avanzadas de control.

REFERENCIAS

[1] X. Zhao and T. Chidambareswaran, “Autonomous Mobile Robots in Manufacturing Operations,” in *2023 IEEE 19th International Conference on Automation Science and Engineering (CASE)*, Aug. 2023, pp. 1–7. doi: 10.1109/CASE56687.2023.10260631.

[2] E. Guzmán Ortiz, B. Andres, F. Fraile, R. Poler, and Á. Ortiz Bas, “Fleet management system for mobile robots in healthcare environments,” *Journal of Industrial Engineering and Management (JIEM)*, vol. 14, no. 1, pp. 55–71, 2021, doi: 10.3926/jiem.3284.

[3] D. F. Yépez-Ponce, J. V. Salcedo, P. D. Rosero-Montalvo, and J. Sanchis, “Mobile robotics in smart farming: current trends and applications,” *Front. Artif. Intell.*, vol. 6, Aug. 2023, doi: 10.3389/frai.2023.1213330.

[4] L. B. Sobolev, “Robots in aerospace industry,” *Revista de la Universidad del Zulia*, vol. 13, no. 36, pp. 241–256, 2022.

[5] Gobierno de México, “Data México.” Accessed: Feb. 09, 2026. [Online]. Available: https://www.economia.gob.mx/datamexico/

[6] Modern Machine Shop México, “La demanda mundial de robots industriales se duplicó en 10 años.” Accessed: Feb. 09, 2026. [Online]. Available: https://www.mms-mexico.com/noticias/post/la-demanda-mundial-de-robots-industriales-se-duplica-en-10-anos

[7] J. L. T. Moreno, J. D. Á. Hervás, M. M. Rodríguez, and J. L. B. Claraco, “Arquitectura de control para robots móviles operando en edificios inteligentes,” *Simposios del Comité Español de Automática (CEA)*, vol. 1, no. 1, May 2025, doi: 10.64117/simposioscea.v1i1.11.

[8] D. Lopes *et al.*, “Integrated Fleet Management of Mobile Robots for Enhancing Industrial Efficiency: A Case Study on Interoperability in Multi-Brand Environments Within the Automotive Sector,” *Applied Sciences*, vol. 15, no. 13, Jun. 2025, doi: 10.3390/app15137235.

[9] A. P. Raut, A. Raj, and S. Patil, “Internet Controlled Techrobot using Raspberry Pi,” *IJERT*, vol. V9, no. 08, p. IJERTV9IS080275, Aug. 2020, doi: 10.17577/IJERTV9IS080275.

[10] N. K. A. Al-Sahib and M. Z. Azeez, “Build and Interface Internet Mobile Robot using Raspberry Pi and Arduino,” *Innovative Systems Design and Engineering*, vol. 6, no. 1, pp. 106–114, 2015.

[11] R. Farkh, M. Tabrez Quasim, K. Al Jaloud, S. Alhuwaimel, and S. Tabrez Siddiqui, “Computer Vision-Control-Based CNN-PID for Mobile Robot,” *Computers, Materials & Continua*, vol. 68, no. 1, pp. 1065–1079, 2021, doi: 10.32604/cmc.2021.016600.

[12] V. Yevsieiev and A. Skripkin, “Development of Architecture for Mobile Robot Control Based on Raspberry Pi Model 3 B+,” *InterConf*, no. 103, pp. 274–277, 2022.

[13] S. Chakraborty and P. S. Aithal, “Demonstration of Modbus Protocol for Robot Communication Using C#,” *IJAEML*, pp. 119–131, Nov. 2021, doi: 10.47992/IJAEML.2581.7000.0108.

[14] A. S. Sadun, J. Jalani, and J. A. Sukor, “Modbus RTU protocol and arduino IO package: A real time implementation of a 3 finger adaptive robot gripper,” *MATEC Web Conf.*, vol. 108, p. 05004, 2017, doi: 10.1051/matecconf/201710805004.

[15] R. Kellers, “Home Automation Network MODBUS,” Vaasa University of Applied Sciences, Finland, 2021.

[16] L. Xuan and L. Yongzhong, “Research and Implementation of Modbus TCP Security Enhancement Protocol,” *J. Phys.: Conf. Ser.*, vol. 1213, no. 5, p. 052058, Jun. 2019, doi: 10.1088/1742-6596/1213/5/052058.

[17] S. Pecolt, A. Błażejewski, T. Królikowski, P. Z. Trzebiatowski, and B. Schulz, “Autonomous control of a mobile robot to explore areas with difficult terrain,” *Procedia Computer Science*, vol. 192, pp. 3467–3476, Jan. 2021, doi: 10.1016/j.procs.2021.09.120.

[18] J. Chen and S. Huang, “Analysis and Comparison of UART, SPI and I2C,” in *2023 IEEE 2nd International Conference on Electrical Engineering, Big Data and Algorithms (EEBDA)*, Feb. 2023, pp. 272–276. doi: 10.1109/EEBDA56825.2023.10090677.

[19] M. Ohchi, T. Furukawa, M. Takayama, and S. Yanagi, “Development of autonomously mobile robot with control system connected by TCP/IP network,” in *IECON’01. 27th Annual Conference of the IEEE Industrial Electronics Society (Cat. No.37243)*, Nov. 2001, pp. 311–316 vol.1. doi: 10.1109/IECON.2001.976500.

[20] M. Albonico, M. Đorđević, E. Hamer, and I. Malavolta, “Software engineering research on the Robot Operating System: A systematic mapping study,” *Journal of Systems and Software*, vol. 197, p. 111574, Mar. 2023, doi: 10.1016/j.jss.2022.111574.

[21] B. Kazed and A. Guessoum, “A Lyapunov based posture controller for a differential drive mobile robot,” *IAES International Journal of Robotics and Automation (IJRA)*, vol. 13, no. 1, pp. 1–10, Mar. 2024, doi: 10.11591/ijra.v13i1.pp1-10.

[22] A. R. Khairuddin, M. S. Talib, and H. Haron, “Review on simultaneous localization and mapping (SLAM),” in *2015 IEEE International Conference on Control System, Computing and Engineering (ICCSCE)*, Nov. 2015, pp. 85–90. doi: 10.1109/ICCSCE.2015.7482163.

[23] M. M. Cardenas, J. Villalobos, A. F. Santacruz, C. A. Paredes, I. Y. Sanchez, and F. Martell, “Mobile Robot with Modbus RTU Connectivity to Arduino Microcontroller for Remote-Online Education in Control Systems,” in *2021 International Conference on Electrical, Computer and Energy Technologies (ICECET)*, Dec. 2021, pp. 1–6. doi: 10.1109/ICECET52533.2021.9698696.

[24] F. Martell and I. Y. Sanchez, *Practical Control Engineering for Mechatronics and Automation*. Boca Raton: CRC Press, 2024. doi: 10.1201/9781003357889.

[25] E. Flores García, J. C. Quezada Quezada, A. E. Solís Galindo, H. Calderón Medina, and E. Mendoza Jiménez, “Implementación de protocolo MODBUS RTU en la comunicación de un PAC y un variador de velocidad para el control automático de motores eléctricos,” *Boletín Científico INVESTIGIUM de la Escuela Superior de Tizayuca*, vol. 3, no. 5, Jul. 2017, doi: 10.29057/est.v3i5.2428.

[26] M. Tyrovolas and T. Hajnal, “Inter-communication between Programmable Logic Controllers using IoT technologies: A Modbus RTU/MQTT Approach,” Feb. 11, 2021, *arXiv*: arXiv:2102.05988. doi: 10.48550/arXiv.2102.05988.

[27] A. Wendt and T. Schüppstuhl, “A Solution to the Generalized ROS Hardware IO Problem -- A Generic Modbus/TCP Device Driver for PLCs, Sensors and Actuators,” in *2021 26th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*, Sep. 2021, pp. 1–8. doi: 10.1109/ETFA45728.2021.9613662.