



Recibido: 4 de febrero 2026 / Aceptado: 18 de mayo 2026

AGITADOR DE MUESTRAS BIOLÓGICAS CON CONTROL DE TEMPERATURA BASADO EN TECNOLOGÍAS IOT

BIOLOGICAL SAMPLE SHAKER WITH TEMPERATURE CONTROL BASED ON IOT TECHNOLOGIES

Dan Salvador García Guevara¹, Edgardo Tomas Martínez², Miguel Alcaraz Vázquez³,
Marco Antonio Adame Ramírez⁴, Arnulfo Catalán Villegas⁵, Gustavo A. Alonso-Silverio⁶

Resumen:

Los agitadores de laboratorio se utilizan en cultivos, fermentaciones y ensayos que requieren agitación continua y condiciones ambientales trazables. Sin embargo, en laboratorios con recursos limitados es común que estos equipos dependan de supervisión presencial, mediciones externas y registros manuales, lo que reduce la trazabilidad y dificulta la atención oportuna ante variaciones de operación. Para atender este vacío, se desarrolló un prototipo de agitador de muestras biológicas con control de temperatura y monitoreo remoto basado en tecnologías IoT. El sistema emplea un microcontrolador ESP32 de doble núcleo; el núcleo 0 gestiona las comunicaciones WiFi, Telegram, MQTT y HTTP, mientras que el núcleo 1 atiende el control del motor y la lectura de sensores. Se integraron un sensor DHT22, un motor a pasos 42BYGHWB09 con controlador A4988 y un controlador W3230 para la regulación térmica ON-OFF con histéresis. Las pruebas reportaron operación continua durante 72 h, disponibilidad de comunicación y registro de 99,2 %, precisión de velocidad de $\pm 0,3$ RPM en el intervalo de 5 a 30 RPM, estabilidad térmica de $\pm 0,5$ °C y tiempos de respuesta promedio de $2,2 \pm 0,8$ s para Telegram, $0,8 \pm 0,2$ s para MQTT y $2,5 \pm 0,1$ s

para la interfaz web local. Estos resultados muestran que el prototipo puede apoyar la automatización de laboratorio mediante componentes comerciales de bajo costo, registro de variables y comunicación remota para seguimiento de procesos experimentales.

Palabras claves: ESP32, IoT, Agitador, Telegram, MQTT, Automatización de laboratorio.

Abstract

Laboratory shakers are used in cultures, fermentations and assays that require continuous agitation and traceable environmental conditions. However, in resource-limited laboratories these devices often depend on on-site supervision, external measurements and manual records, which reduces traceability and hinders timely response to operating variations. To address this gap, a biological sample shaker prototype with temperature control and remote monitoring based on IoT technologies was developed. The system uses a dual-core ESP32 microcontroller; core 0 manages WiFi, Telegram, MQTT and HTTP communications, while

1 Ingeniero en computación, Universidad Autónoma de Guerrero, México, Chilpancingo. Afiliación institucional: Universidad Autónoma de Guerrero, México. Correo electrónico personal e institucional e-mail: danlachele@gmail.com 16282142@uagro.mx ORCID: <https://orcid.org/0009-0000-3751-4804>

2 Ingeniero en computación, Universidad Autónoma de Guerrero, México, Chilpancingo. Afiliación institucional: Universidad Autónoma de Guerrero, México. Correo electrónico personal e institucional e-mail: edgardo_tm@hotmail.com 24600063@uagro.mx ORCID: <https://orcid.org/0009-0004-9689-8392>

3 Ingeniero en computación, Universidad Autónoma de Guerrero, México, Chilpancingo. Afiliación institucional: Universidad Autónoma de Guerrero, México. Correo electrónico personal e institucional e-mail: vazquezmicky@gmail.com 19275295@uagro.mx ORCID: <https://orcid.org/0009-0006-8355-0287>

4 Ingeniero en Computación, Universidad Autónoma de Guerrero, México, Chilpancingo. Afiliación institucional: Universidad Autónoma de Guerrero, México. Correo electrónico personal e institucional e-mail: ragnarockgarnes86@gmail.com 18451084@uagro.mx ORCID: <https://orcid.org/0009-0003-2361-4945>

5 Doctorado en ciencias de la educación, Universidad Autónoma de Guerrero, México, Chilpancingo. Afiliación institucional: Universidad Autónoma de Guerrero, México. Correo electrónico personal e institucional e-mail: catalanvillegas@gmail.com 03180@uagro.mx ORCID: <https://orcid.org/0009-0001-0391-7960?lang=es>

6 Doctorado en Ciencias con especialidad en Ingeniería Eléctrica por parte del CINVESTAV-IPN, México. Afiliación institucional: Universidad Autónoma de Guerrero, México. Correo electrónico personal e institucional e-mail: gsilverio@uagro.mx ORCID: <https://orcid.org/0000-0002-2699-140X>

core 1 handles motor control and sensor acquisition. A DHT22 sensor, a 42BYGHW809 stepper motor with an A4988 driver and a W323D controller for ON-OFF thermal regulation with hysteresis were integrated. The tests reported 72 h of continuous operation, 99.2 % communication and logging availability, ± 0.3 RPM speed precision in the 5 to 30 RPM range, ± 0.5 °C thermal stability and average response times of 2.2 ± 0.8 s for Telegram, 0.8 ± 0.2 s for MQTT and 2.5 ± 0.1 s for the local web interface. These results indicate that the prototype can support laboratory automation through low-cost commercial components, variable logging and remote communication for monitoring experimental processes.

Keywords: ESP32, IoT, Shaker, Telegram, MQTT, Laboratory automation.

1. Introducción

Los agitadores de laboratorio son equipos empleados en investigación biotecnológica, microbiológica y química para mantener cultivos, fermentaciones y reacciones en condiciones de mezcla homogénea. La agitación continua permite reducir la sedimentación, favorecer el intercambio gaseoso y conservar la distribución uniforme de microorganismos o reactivos, por lo que influye directamente en la reproducibilidad experimental. En ensayos de varias horas o días, además de la velocidad de agitación, es necesario registrar variables ambientales como temperatura y humedad, ya que sus variaciones pueden modificar la viabilidad de cultivos o la estabilidad de las muestras. Los sensores DHT22 se han utilizado en aplicaciones IoT por su bajo costo y por sus especificaciones típicas de medición de temperatura y humedad relativa [1], [9]. No obstante, muchos agitadores disponibles en laboratorios académicos operan sin monitoreo remoto integrado, sin registro histórico de variables y con dependencia de supervisión presencial. Esta situación dificulta la trazabilidad, la detección temprana de desviaciones y la continuidad de procesos fuera del horario de trabajo. La incorporación de tecnologías IoT permite

conectar sensores, actuadores e interfaces de usuario para consultar el estado del equipo, recibir alertas y registrar datos de operación. El ESP32 es una alternativa adecuada para este tipo de sistemas por su conectividad WiFi, arquitectura de doble núcleo y capacidad para ejecutar tareas de comunicación y control en paralelo [2], [7]. En trabajos previos se han utilizado arquitecturas IoT para integración de sensores en biorreactores [3] y para sistemas de laboratorio inteligente basados en ESP32 y plataformas de visualización [4]. Además, MQTT se emplea en adquisición de datos IoT por su modelo de publicación-suscripción y bajo consumo de ancho de banda [5], mientras que los bots de Telegram ofrecen una interfaz accesible para enviar comandos y recibir notificaciones [6]. A pesar de estos avances, se identifica la necesidad de un prototipo de bajo costo que reúna en un mismo equipo la agitación controlada, la regulación térmica autónoma, el monitoreo ambiental, el registro de variables y la comunicación remota mediante Telegram, MQTT e interfaz web local. Por ello, este trabajo presenta el diseño, construcción y validación de un agitador de muestras biológicas con control de temperatura basado en tecnologías IoT, orientado a apoyar procesos experimentales que requieren seguimiento continuo y operación trazable.

2. Materiales y Métodos

2.1 Materiales y Componentes

Para el desarrollo del sistema se construyó un dispositivo electrónico integrado basado en el microcontrolador ESP32-WROOM-32. El prototipo se conectó con sensores ambientales, actuadores de agitación, un módulo de regulación térmica y canales de comunicación IoT para permitir el monitoreo y control remoto del agitador de laboratorio. Los materiales y componentes utilizados se presentan en la Tabla 1.

Tabla 1. Lista de materiales y componentes electrónicos utilizados.

Cantidad	Nombre	Imagen
1	Kit ESP32	
1	Motor a Pasos 42BYGHW809	
1	Sensor DHT22 de temperatura y humedad	
1	Driver de motor a pasos A4988	
1	Controlador de temperatura W3230	
1	Celda Peltier SP184827145	
1	Contrachapado de 7 mm de grosor	
5	Piezas de aluminio	
2	Disipadores de aluminio	
2	Ventiladores de 12 V, 0,7 A, DV07020B122U	
1	Placa fenólica perforada con pistas, de 7,5 cm x 4,5 cm	

El circuito integra el ESP32 como controlador principal para gestionar las comunicaciones IoT y coordinar los subsistemas de medición y actuación. El sensor DHT22 registra temperatura y humedad; el motor a pasos 42BYGHW809, controlado mediante el driver A4988, ejecuta la agitación con velocidad ajustable.

El sistema de control térmico utiliza el controlador W3230, configurado como termostato ON-OFF con histéresis. Este subsistema opera de forma independiente del ESP32, lo que permite mantener la regulación de temperatura aunque el controlador principal se reinicie o pierda temporalmente conectividad.

La interfaz local incluye indicadores LED para el estado del motor, conectividad de red y alimentación, así como botones para control manual del motor y configuración de red. El potenciómetro permite ajustar la velocidad de agitación de manera local, independientemente de los comandos remotos.

La fuente de 12 V y 10 A alimenta los componentes de potencia y los sistemas de control, con regulación apropiada para cada subsistema. La separación entre la sección lógica y los actuadores de potencia reduce interferencias y facilita el diagnóstico durante las pruebas.

2.2. Metodología del Sistema

En la Figura 2 se muestra la metodología del sistema IoT desarrollado.

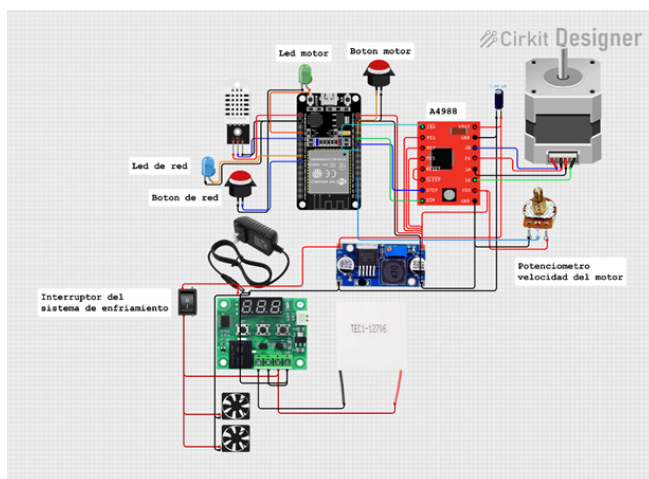


Figura 1. Esquema del circuito eléctrico del sistema IoT propuesto.

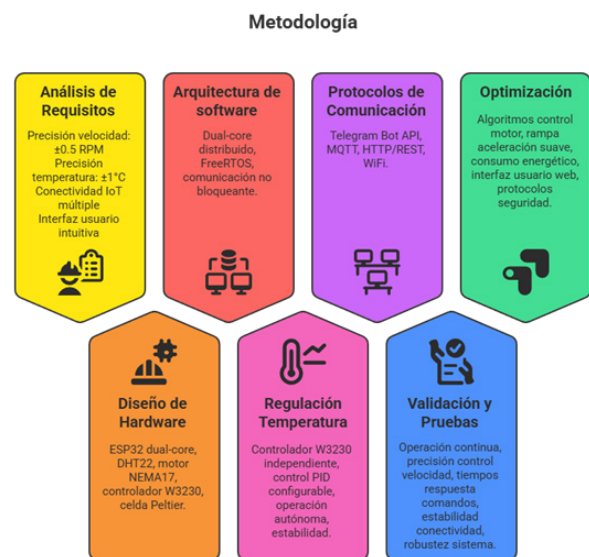


Figura 2. Metodología del sistema.

La metodología de desarrollo fue incremental y se organizó en siete fases interconectadas. Cada fase generó insumos para la siguiente y permitió verificar el desempeño del prototipo antes de integrarlo por completo.

Fase 1: Análisis de requisitos

Se definieron los requisitos funcionales y no funcionales del prototipo, considerando operación remota, ajuste local de velocidad, monitoreo de temperatura y humedad, regulación térmica y registro de eventos. Como métricas de aceptación se establecieron una precisión de velocidad de $\pm 0,5$ RPM, disponibilidad del sistema mayor a 99 %, estabilidad térmica cercana a ± 1 °C, tiempo de respuesta a comandos remotos menor a 3 s y operación continua durante 72 h sin fallos de operación.

Fase 2: Diseño de hardware

A partir de los requisitos se seleccionaron los componentes del sistema. El ESP32 se eligió como controlador principal por su conectividad WiFi y arquitectura de doble núcleo. El sensor DHT22 se utilizó para el monitoreo ambiental, el motor a pasos 42BYGHW609 con driver A4988 se empleó para la agitación, de acuerdo con el uso común de motores a pasos en sistemas controlados por microcontrolador [8], y el controlador W3230 se integró como módulo autónomo para la regulación térmica. La selección también consideró disponibilidad comercial, facilidad de reemplazo y compatibilidad eléctrica con una alimentación principal de 12 V.

Fase 3: Arquitectura de software

El firmware se organizó en tareas separadas para reducir bloqueos entre comunicación y control. El núcleo 0 del ESP32 se asignó a las comunicaciones WiFi, Telegram, MQTT y HTTP, mientras que el núcleo 1 se dedicó a la lectura de sensores, control de velocidad y gestión de estados del motor. Esta distribución permitió mantener la respuesta del sistema aun cuando existieran consultas remotas o reconexiones de red.

Fase 4: Regulación de temperatura

La regulación de temperatura se implementó mediante el controlador W3230 en modo ON-OFF con histéresis configurable. El controlador activa o desactiva el subsistema térmico con base en la temperatura medida y el punto de ajuste definido. Al operar de manera independiente, este módulo conserva la regulación térmica aunque el ESP32 se reinicie,

se actualice el firmware o se interrumpa temporalmente la comunicación inalámbrica.

Fase 5: Protocolos de comunicación

Se implementaron tres canales de comunicación. Telegram permite consultar el estado del equipo, enviar comandos y recibir alertas desde una interfaz conversacional. MQTT publica variables de operación para integración con sistemas externos o tableros de monitoreo. La interfaz HTTP local permite visualizar parámetros del prototipo desde un navegador dentro de la misma red. Los tres canales fueron diseñados para evitar bloqueos en el control del motor y en la lectura de sensores.

Fase 6: Validación y pruebas

La validación incluyó pruebas de operación continua durante 72 h, medición de precisión de velocidad en el intervalo de 5 a 30 RPM, evaluación de tiempos de respuesta por Telegram, MQTT e interfaz web, comparación de lecturas del DHT22 con instrumentos de referencia, pruebas de recuperación ante pérdida de conectividad y registro de fallos de operación. La disponibilidad se calculó a partir del tiempo operativo efectivo respecto al tiempo total de prueba.

Fase 7: Optimización

Después de las pruebas iniciales se ajustaron los parámetros de aceleración del motor, los intervalos de lectura, la frecuencia de publicación MQTT, los mensajes de Telegram y la presentación de datos en la interfaz web. La optimización se orientó a reducir retardos de comunicación, evitar reinicios por sobrecarga de tareas y mejorar la claridad de las alarmas para el usuario final.

2.3. Tecnologías y Herramientas de Desarrollo

El desarrollo del sistema utilizó Arduino IDE 2.0.x y lenguaje C++ para el firmware del ESP32. La gestión de tareas se apoyó en FreeRTOS 10.4.x. Para la comunicación remota se utilizaron Telegram Bot API 6.x, MQTT 3.1.1 y un servidor HTTP/REST 1.1 para la interfaz local de monitoreo. La configuración persistente del equipo se almacenó mediante el sistema Preferencias del ESP32, lo que permitió conservar parámetros después de reinicios.

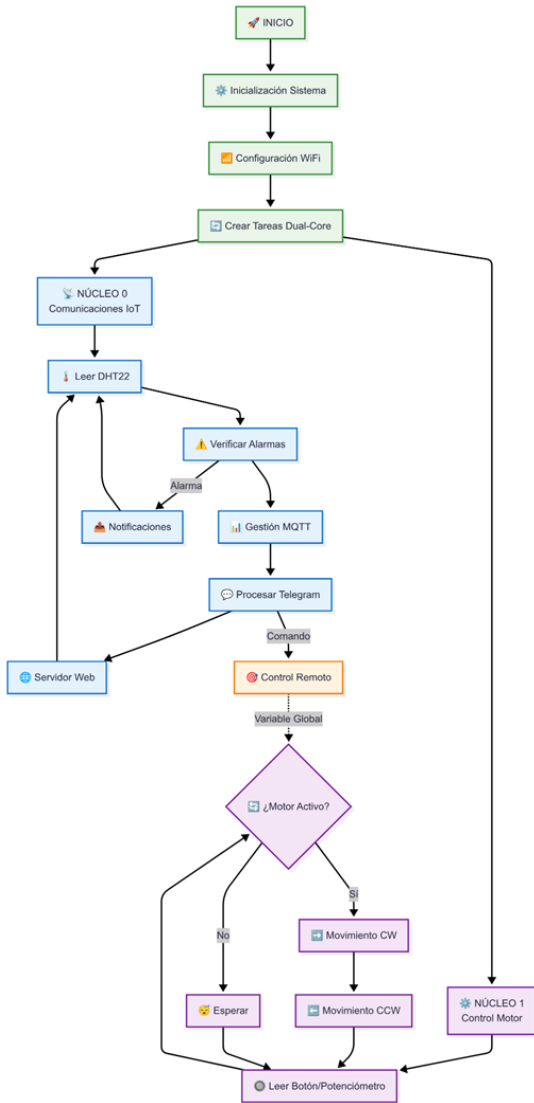


Figura 3. Diagrama de flujo de la arquitectura de doble núcleo del sistema IoT propuesto.

3. Resultados

El sistema desarrollado se evaluó mediante pruebas de operación continua, precisión de velocidad, respuesta de comunicación, monitoreo ambiental y recuperación ante pérdida de conectividad. En conjunto, el prototipo operó durante 72 h, alcanzó 99,2 % de disponibilidad de comunicación y registro, no registró fallos de operación del motor ni del control térmico y conservó la comunicación por los tres canales implementados.

La integración de los componentes descritos en la Tabla 1 y del circuito de la Figura 1 dio como resultado el prototipo físico mostrado en la Figura 4. La Figura 3 corresponde a la arquitectura de control de doble núcleo, por lo que la evidencia del dispositivo construido se presenta en la Figura 4.

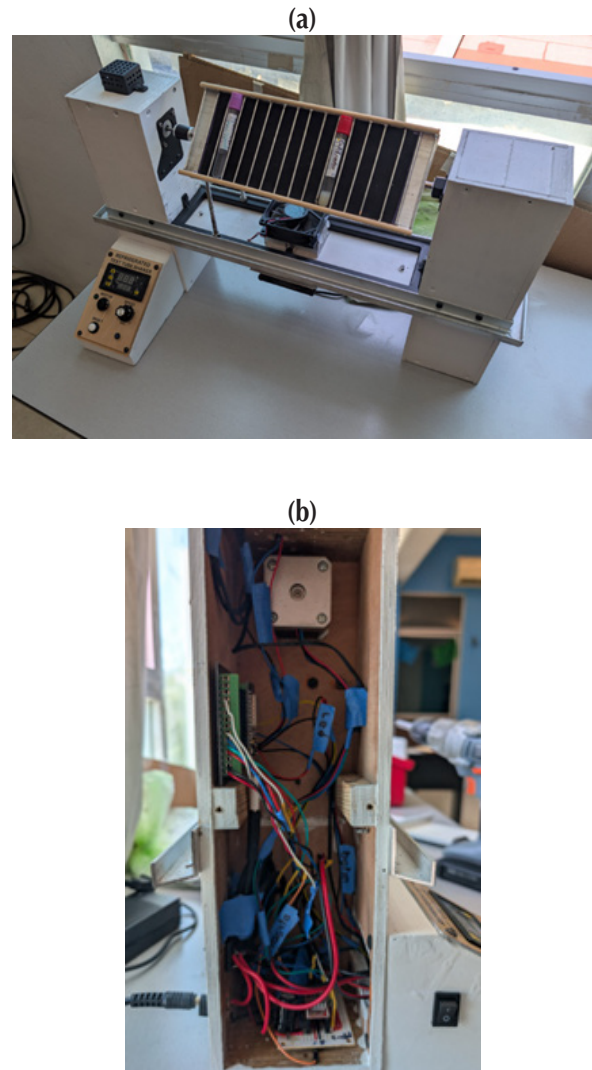


Figura 4. a) Producto final del sistema IoT propuesto, b) interior del dispositivo propuesto.

Fuente: elaboración propia.

3.1. Evaluación de Precisión y Desempeño

El control de velocidad del motor alcanzó una precisión de $\pm 0,3$ RPM en el rango de operación de 5 a 30 RPM, por debajo del límite de

aceptación de $\pm 0,5$ RPM (Figura 5). Este desempeño se asocia con la rampa de aceleración, el ajuste de pasos del driver A4988 y la separación de tareas entre comunicación y control del motor.

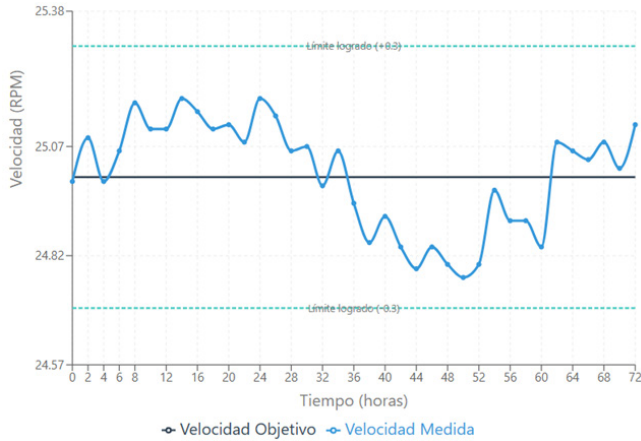


Figura 5. Validación de precisión de control de velocidad durante operación continua.

Las pruebas de conectividad se realizaron en condiciones controladas y con variaciones de red para observar la respuesta del sistema. Los tiempos de respuesta promedio fueron $2,2 \pm 0,8$ s para comandos de Telegram, $0,8 \pm 0,2$ s para mensajes MQTT y $2,5 \pm 0,1$ s para la interfaz web local.

Durante la prueba de 72 h, la disponibilidad de comunicación y registro fue de 99,2 %. Para este cálculo se consideraron como indisponibilidad los intervalos en los que el sistema no respondió a consultas remotas o no publicó datos. Ante pérdidas de conectividad, el sistema recuperó la comunicación en 20 ± 5 s y no se registraron fallos de operación del motor ni del control térmico durante el periodo evaluado.

3.2. Sistema de Monitoreo Ambiental

El sistema de monitoreo ambiental entregó lecturas consistentes al compararse con instrumentos de referencia. Las diferencias observadas fueron menores a $\pm 0,5$ °C para temperatura y menores a 2 % para humedad relativa. El sistema de alarmas notificó eventos fuera de rango por Telegram y MQTT en menos de 30 s desde la detección. El control térmico con W3230 mantuvo una estabilidad aproximada de $\pm 0,5$ °C durante las pruebas, lo que permitió regular las condiciones ambientales sin depender del estado de conexión del ESP32.

3.3. Validación de Protocolos de Comunicación

La interfaz de Telegram ejecutó correctamente el 95 % de los comandos en el primer intento e incorporó mensajes de ayuda para reducir errores de operación. La interfaz web local mostró los parámetros del sistema con actualización automática cada 2 s y visualización adaptable a dispositivos móviles.

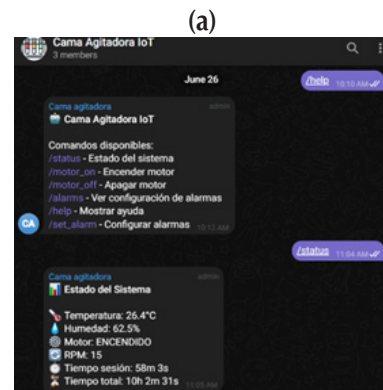


Figura 6. a) Monitoreo usando comandos de Telegram, b) panel de control en funcionamiento.

El protocolo MQTT permitió la transmisión de variables con una tasa de entrega de mensajes de 99,8 % y conservó mensajes en cola durante desconexiones de corta duración, lo que facilitó la recuperación de datos al restablecer la red.

4. Conclusiones

El sistema IoT desarrollado para control y monitoreo de un agitador de muestras biológicas integró agitación controlada, regulación térmica, lectura ambiental y comunicación remota mediante Telegram, MQTT e interfaz HTTP local. Las pruebas realizadas validaron la arquitectura de doble núcleo del ESP32 para separar tareas de comunicación y control en una aplicación de laboratorio.

El controlador W3230 permitió regular la temperatura de manera autónoma mediante control ON-OFF con histéresis, manteniendo una estabilidad aproximada de $\pm 0,5$ °C durante las pruebas. Esta independencia respecto al ESP32 aporta continuidad al control térmico cuando existen reinicios o interrupciones temporales de conectividad.

El prototipo alcanzó 99,2 % de disponibilidad de comunicación y registro durante 72 h de operación continua, precisión de velocidad de $\pm 0,3$ RPM y tiempos de respuesta promedio menores a 3 s en los canales de comunicación evaluados. Estos valores respaldan el uso del sistema como herramienta de apoyo para procesos de laboratorio que requieren seguimiento remoto y registro de variables.

Las interfaces Telegram, MQTT y HTTP ofrecen opciones de interacción e integración con distintos entornos de trabajo. Como trabajo futuro se recomienda ampliar el número de pruebas con muestras reales, comparar el desempeño frente a agitadores comerciales y evaluar el sistema en periodos de operación más prolongados.

5. Reconocimientos

Se agradece a la Universidad Autónoma de Guerrero y al programa de maestría en ingeniería para la innovación y desarrollo tecnológico (MIIDT) por proporcionar las facilidades y recursos necesarios para el desarrollo de este proyecto. Especial reconocimiento al laboratorio de Tecnologías de Cómputo y Electrónica por el apoyo técnico durante la implementación del sistema y al laboratorio de Ciencias Químicas por el apoyo en los requerimientos específicos para el sistema.

6. Referencias

- [1] Y. A. Ahmad, T. S. Gunawan, H. Mansor, B. A. Hamida, A. F. Hishamudin, and F. Arifin, "On the Evaluation of DHT22 Temperature Sensor for IoT Application," in Proc. 2021 8th International Conference on Computer and Communication Engineering (ICCE), Jun. 2021, pp. 131-134, doi: 10.1109/ICCE50029.2021.9467147. [Online]. Available: <https://doi.org/10.1109/ICCE50029.2021.9467147>
- [2] D. Hercog, T. Lerher, M. Truntič, and O. Težak, "Design and Implementation of ESP32-Based IoT Devices," Sensors, vol. 23, no. 15, Art. no. 8739, Aug. 2023, doi: 10.3390/s23158739. [Online]. Available: <https://doi.org/10.3390/s23158739>
- [3] L. M. Baicu, M. Andrei, G. A. Irim, and L. T. Dimitrievici, "Embedded IoT Design for Bioreactor Sensor Integration," Sensors, vol. 24, no. 20, Art. no. 6587, Oct. 2024, doi: 10.3390/s24206587. [Online]. Available: <https://doi.org/10.3390/s24206587>
- [4] F. Wibowo, S. Suheri, M. Diponegoro, and B. Hermanto, "Design and Implementation of IoT-Based Smart Laboratory Using ESP32 and Thingsboard to Improve Security and Safety in POLNEP Informatics Engineering Laboratory," Jurnal ELIT, vol. 3, no. 2, pp. 13-21, 2022, doi: 10.31573/elit.v3i2.402. [Online]. Available: <https://doi.org/10.31573/elit.v3i2.402>
- [5] R. A. Atmoko, R. Riantini, and M. K. Hasin, "IoT real time data acquisition using MQTT protocol," Journal of Physics: Conference Series, vol. 853, no. 1, Art. no. 012003, May 2017, doi: 10.1088/1742-6596/853/1/012003. [Online]. Available: <https://doi.org/10.1088/1742-6596/853/1/012003>
- [6] A. Rahmatulloh and A. Firmansah, "Telegram Bot for Automation of Academic Information Services with The Forward Chaining Method," Sinkron, vol. 3, no. 2, pp. 73-78, Apr. 2019, doi: 10.33395/sinkron.v3i2.10023. [Online]. Available: <https://doi.org/10.33395/sinkron.v3i2.10023>
- [7] P. Foltýnek, M. Babiuch, and P. Šuránek, "Measurement and data processing from Internet of Things modules by dual-core application using ESP32 board," International Journal of Distributed Sensor Networks, vol. 15, no. 6, Jun. 2019, doi: 10.1177/1550147719857748. [Online].

Available: <https://doi.org/10.1177/1550147719857748>

[8] I. Virgala, M. Kelemen, T. Lipták, L. Miková, and F. Božek, "Control of Stepper Motor by Microcontroller," *Journal of Automation and Control*, vol. 3, no. 3, pp. 131-134, 2015, doi: 10.12691/automation-3-3-19. [Online]. Available: <https://doi.org/10.12691/automation-3-3-19>

[9] W. Adhiwibowo, A. F. Daru, and A. M. Hirzan, "Temperature and Humidity Monitoring Using DHT22 Sensor and Cayenne API," *Jurnal Transformatika*, vol. 17, no. 2, pp. 209-214, Jan. 2020, doi: 10.26623/transformatika.v17i2.1820. [Online]. Available: <https://doi.org/10.26623/transformatika.v17i2.1820>