

# Protocolo MQTT para conectividad del sistema electrónico de un robot colaborativo



## Colaboración

Mariana Natalia Ibarra Bonilla; Raúl Eusebio Grande, Tecnológico Nacional de México / Instituto Tecnológico Superior de Atlixco

Fecha de recepción: 15 de noviembre de 2022

Fecha de aceptación: 17 de diciembre de 2022

El avance actual de las tecnologías y comunicaciones ha introducido una nueva tendencia en la industria inteligente que está generando que los robots y los humanos colaboren juntos, mediante los Robots Colaborativos (Cobots), en la que el Internet de las Cosas (IoT) ha presentado arquitecturas de conectividad entre varios dispositivos con resultados prometedores. El presente artículo describe el desarrollo de una arquitectura de conexión basada en el protocolo MQTT, en la que se conectan como dispositivos unas tarjetas de desarrollo que incorporan un microcontrolador ARM que transmiten los datos obtenidos de un sensor hacia un servidor. Los resultados han sido satisfactorios y han demostrado oportunidades para su aplicación en un ecosistema de robots colaborativos educativos.

**PALABRAS CLAVE:** ARM, cobots, comunicación, IoT, microcontrolador, MQTT.

**ABSTRACT:** The current advancement of technologies and communications has introduced a new trend in the smart industry in which robots and humans work together, through Collaborative Robots (Cobots), in which the Internet of Things (IoT) has presented connectivity architectures between multiple devices with promising results. This paper describes the development of a connection architecture based on the MQTT protocol, in which development board that incorporate an ARM microcontroller are connected as devices that transmit the data obtained from a sensor to a server. The results have been satisfactory and have shown opportunities for its application in an ecosystem of collaborative educational robots.

**KEYWORDS:** ARM, cobots, communication, IoT, microcontroller, MQTT.

## INTRODUCCIÓN

A partir de la Cuarta Revolución Industrial o Industria 4.0 los Robots Colaborativos, conocidos como Cobots se han introducido de forma masiva dentro de los procesos industriales. Los Cobots son sistemas diseñados para desempeñar funciones complementarias a las que realiza un operador humano dentro de un ambiente industrial o de trabajo. Las tareas de estos robots deben realizarse de manera segura para los humanos, es por ello que integran sensores y sistemas de seguridad avanzados que eviten posibles colisiones que puedan dañar al obrero. Estos robots generalmente son fabricados en forma de brazos robóticos que se incluyen dentro de las cadenas de manufactura de las industrias, pero con

la diferencia de estos robots no emplean barreras de protección lo que permite una interacción más directa con el operador humano ampliando de esta manera las posibilidades de ejecutar una tarea con múltiples estrategias. Por ello, una de las razones de su éxito es la combinación de fuerza y resistencia del robot con la destreza y flexibilidad del humano [1].

Los Cobots se han incluido en procesos que integran la interconexión de diferentes dispositivos [2, 3]. En este aspecto, las tecnologías del Internet de las Cosas (IoT) proporcionan grandes beneficios frente a las redes convencionales, debido a que se basan en protocolos de bajo consumo energético y alta seguridad [4]. Por consiguiente, en la industria se busca que los Cobots dentro de la cadena de producción se conecten a una red de datos centralizados, con el fin de poder monitorear su desempeño, sincronizar sus movimientos e incluso comandar a distancia en tiempo real. Por tal razón, un protocolo de comunicación industrial que ha tomado relevancia debido a que alcanza rápidamente la comunicación automática entre varios dispositivos es MQTT. El protocolo MQTT usa un paquete de datos de menor tamaño que otros protocolos, por lo que aumenta la velocidad de transmisión de datos o la baja latencia [5]. MQTT es usado en aplicaciones de Máquina a Máquina (Machine to Machine, M2M) y Máquina a Nube (Machine to Cloud, M2C) [6, 7], así como también en aplicaciones de agricultura inteligente para el monitoreo del clima o del aire [8, 9, 10]. MQTT también se puede usar en aplicaciones para la adquisición de datos y brinda mayores beneficios que el protocolo HTTP [11, 12].

El objetivo de este artículo es presentar un sistema de conectividad inalámbrica y el protocolo MQTT, mediante la infraestructura de la red de telefonía celular. Esta red presenta el beneficio de cobertura a nivel nacional, así como disponibilidad continua las 24 horas del día. Así mismo se presenta un esquema de 3 dispositivos inalámbricos basados en el chip SIM7000 del fabricante SIMCOM. A través de este módulo se despliega un esquema de comunicación celular para establecer una micro-red de colaboración entre máquina-humano. Inicialmente se estableció un servidor llamado: Broker, mediante el que se recibe y almacena información proveniente de los dispositivos. La información recabada de los dispositivos corresponde a las variables climáticas: temperatura, humedad relativa y presión atmosférica. Si bien este tipo de información no está directamente relacionada con el funcionamiento de los Cobots, sirvió para probar la estabilidad de la micro-red implementada. Este protocolo y la infraestructura utilizada en el presente trabajo, se emplea posteriormente para intercomunicar los sistemas de Cobots dentro de un laboratorio escolar. Esta propuesta propone una comunicación confiable y precisa, lo cual es necesario para su aplicación con los Cobots.

## MATERIALES Y MÉTODOS

### Comunicación por el protocolo MQTT

El protocolo MQTT, del inglés Message Queuing Telemetry Transport, es un protocolo estándar para IoT y M2M que tiene el objetivo de ofrecer una plataforma de comunicación ligera basada en el modelo de publicación-suscripción (Pubsub). Los sistemas Pubsub tienen tres componentes principales: publicista, suscriptores y un bróker (servidor). Los publicistas son los sensores y dispositivos que se conectan al bróker para enviar sus datos. Los suscriptores son aplicaciones o dispositivos que están interesados en cierto tema o tópico, por lo que se conectan al bróker para mantenerse informados cada vez que se reciben nuevos datos. Los brokers clasifican los datos en temas y los envía a los suscriptores interesados en esos temas. Un dispositivo puede comportarse como publicista y suscriptor al mismo tiempo, publicando para ciertos temas y suscribiéndose a otros, el término cliente-MQTT se usa para distinguir los publicistas/suscriptores de otros brokers [13].

La Figura 1 muestra un ejemplo de tres clientes-MQTT que se comunican entre sí a través de un bróker-MQTT usando el modelo publicación-suscripción.

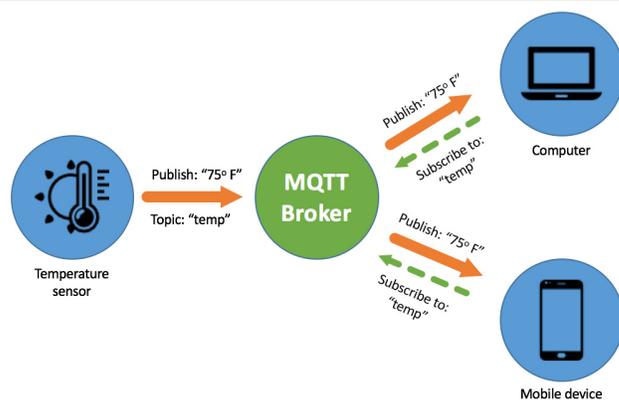


Figura 1. Modelo de publicación/suscripción MQTT entre sensor y dispositivos.

Fuente: Recuperado de [14].

El sensor de temperatura (cliente) publica la temperatura actual en el tema "temp". Los clientes, computadora y dispositivo móvil, reciben la lectura de la temperatura porque están suscritos al tema "temp". El bróker dirige las conexiones y la comunicación de los mensajes. Es importante aclarar entonces, que bajo un esquema MQTT todos los clientes pueden publicar en cualquier tema; pero solamente los dispositivos que estén suscritos a un tema específico recibirán las actualizaciones de datos o información de los clientes que publiquen en este tema.

### Módulo de comunicación SIM7000

El SIM7000, mostrado en la Figura 2, es un dispositivo de comunicaciones inalámbrica para infraestructura celular, orientado a aplicaciones M2M como: rastreo

o seguimiento de activos, monitoreo remoto, y sensado inteligente. No permite comunicación por voz debido a la naturaleza de sus aplicaciones, sin embargo, sí permite una amplia variedad de protocolos de comunicación M2M. En la Tabla 1 se muestra un resumen de sus características más importantes. Se puede alimentar en un rango de 3.0 a 4.3 V, lo que permite utilizar baterías como alimentación principal. Su consumo en un estado de inactividad es de 11mA y de hasta 7uA cuando se encuentra apagado. El consumo pico por un breve lapso de tiempo, cuando está en operatividad máxima, puede sobrepasar los 1000mA dependiendo las condiciones de enlace de con las antenas.

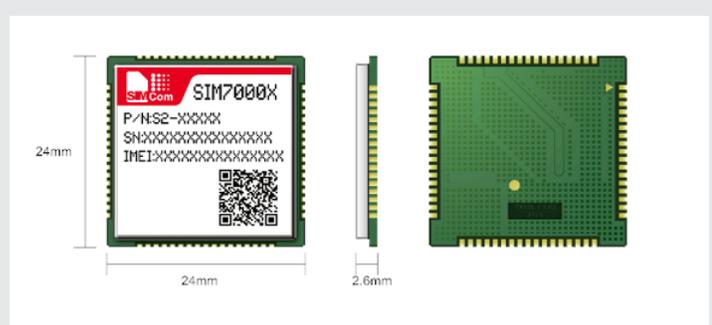


Figura 2. Dispositivo SIM7000 del fabricante SIMCOM en un encapsulado LCC de 68pines SMD.  
Fuente: Recuperado de [15].

La empresa que fabrica este dispositivo es SIMCOM y ha sacado al mercado diferentes versiones de la serie, que se pueden identificar mediante una letra mayúscula al final de la matrícula. De manera que el SIM7000G es la versión global de la empresa, diseñado para que pueda ser desplegado en cualquier parte del mundo, la serie SIM7000A es la versión orientada al continente americano. La versión global tiene la capacidad para comunicarse en los estándares CAT-M1 y CAT-NB1oT que corresponden a tecnologías desarrolladas para IoT de bajo consumo.

Tabla 1. Características principales del SIM7000.

Descripción	SIM7000X
Versiones	G=Global, A=América, E=Europa, JC=Japón, C=China
Bandas de frecuencia en CAT-M1 y NB1oT	B1/B2/B3/B4/B5/B8/B12/B13/17/B18/B19/B20/B25/B26/B28/B39
Voltaje de trabajo	3.0 a 4.3 volts
Consumo de Corriente	7 uA (Off), 11 mA (Idle) 1mA (Sleep), >1A(Active).
Puertos de comunicación	SIM Card, USART, USB, I2C.
Firmware OS	Sistema operativo por comandos AT.
Protocolos de comunicación	TCP/UDP/LWM2M/COAP/MQTT/FTP/HTTP/TLS/DTLS/NTP

Fuente: Elaboración propia

El módulo integra un sistema operativo (OS) basado en comandos AT y a través de este se puede controlar el comportamiento completo del dispositivo. Los comandos son enviados mediante protocolo serial asíncrono: 8 bits de datos, 1 bit de inicio y 1 bit de paro. El módulo posee un puerto serial físico RS-232 dedicado para dicho propósito a 9600 o 115200 baud/s. La comunicación con el módulo es mediante una cadena de caracteres ASCII iniciando siempre con la palabra "AT+" seguido del nemónico del comando correspondiente [16].

El dispositivo es comercializado por medio de diferentes plataformas electrónicas, como el que se muestra en la Figura 3 y fue el que se utilizó en el presente trabajo. Esta plataforma es de manufactura china y contiene el sistema básico de entradas y salidas para que pueda ser integrado a un microcontrolador. Este permite insertar el chip SIM-celular para obtener el servicio del operador. También tiene el puerto serial disponible mediante pines macho para su conexión al exterior.



Figura 3. Módulo de desarrollo SIM7000G Breakout LTE Kit.  
Fuente: Recuperado de [17].

### Tarjeta de desarrollo Ophyra

La Figura 4 presenta a la tarjeta Ophyra, la cual es una plataforma de desarrollo creada para proyectos electrónicos de alto desempeño, diseñada y construida por la empresa mexicana Intesc Electrónicos & Embebidos. La tarjeta está diseñada con el microcontrolador STM32F407VGT6 de STMicroelectronics. Este microcontrolador integra un ARM Cortex-M4° de 32 bits a 168MHz que ofrece características especiales, con lo cual se busca reducir el tiempo en la ejecución de tareas y además se puede programar en lenguaje C y Python. La Tabla 2 presenta las principales características de la tarjeta Ophyra que son útiles para este proyecto [19].

### Descripción de la propuesta de conexión

El esquema general de conectividad que se propone en este proyecto se muestra en la Figura 5, donde los elementos que componen cada dispositivo se presentan en el esquema de la Figura 6. El dispositivo consiste en la conexión del módulo SIM7000G con la tarjeta de

desarrollo Ophyra a través del puerto serial configurado a 115200 baud/s. El objetivo del sistema de cada dispositivo es adquirir datos del sensor BME280 integrado en la tarjeta Ophyra. Este sensor puede medir temperatura, humedad relativa y presión atmosférica. Los datos se obtienen mediante el protocolo de comunicación I2C a 400 Kb/s. La tarjeta ofrece una completa integración con el sensor y mediante librerías genéricas escritas en lenguaje en C se realiza la adquisición de las mediciones. El microcontrolador ARM-STM32F407 es el encargado de leer al sensor y controlar el módulo de comunicación SIM7000. Para controlar este último fue necesario crear una librería extensa con los principales comandos AT de enlace, telecomunicación y protocolo MQTT, y así enviar los datos recabados del sensor hacia el servidor o Broker-MQTT.

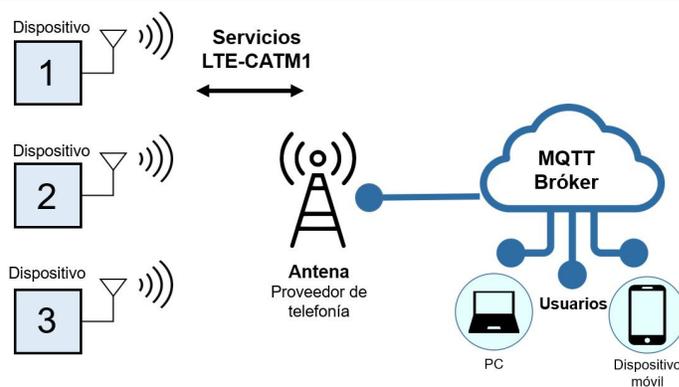


Figura 5. Esquema general de la propuesta de conexión. Fuente: Elaboración propia.

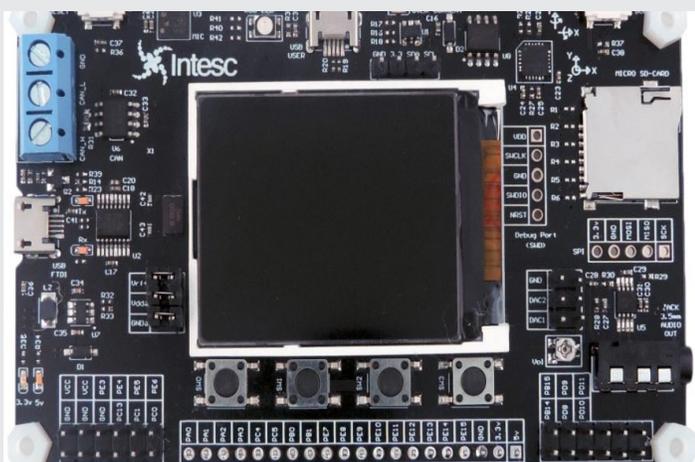


Figura 4. Tarjeta de desarrollo Ophyra. Fuente: Recuperado de [19].

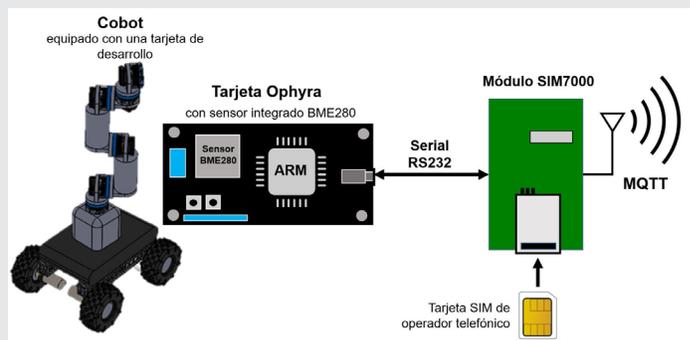


Figura 6. Esquema general de conexión para cada dispositivo. Fuente: Elaboración propia.

Es importante mencionar que un objetivo en progreso es que los componentes que conforman cada dispositivo se monten respectivamente en la base de un robot colaborativo, de tal manera que estos se conecten al Broker y se esté monitoreando su desempeño para sincronizar sus movimientos. El diseño del Cobot propuesto consiste en un brazo robot de 6 GDL, basado en el MyCobot-Robot de la empresa Elephant Robotic [19], donde cada articulación corresponde a un sistema de engranes planetarios. El diseño CAD en SolidWorks de este Cobot se visualiza en la Figura 7 y es importante mencionar que la construcción de tres de estos se encuentra en progreso.

Tabla 2. Características principales de Ophyra.

Descripción	Característica
Microcontrolador	STM32F407VGT6/168Mhz. Núcleo de 32-bit ARM Cortex®-M4. Unidad de punto flotante (FPU). 1-Mbyte de memoria Flash. 192-Kbyte de memoria RAM.
Puerto COM virtual	FTDI: FT230XS. Interfaz USB a serial. Velocidad de transferencia de hasta 3 Mbaudios.
Fuente de alimentación	Principal de 5V Secundaria de 3.3 V
Sensores	Acelerómetro y Giroscopio MP6050. Temperatura/Humedad Relativa/Presión Atmosférica BME280.
Conector tarjeta SD	Interfaz SDIO de 4 bits. Detección de tarjeta.

Fuente: Elaboración propia con información recuperada de [19].

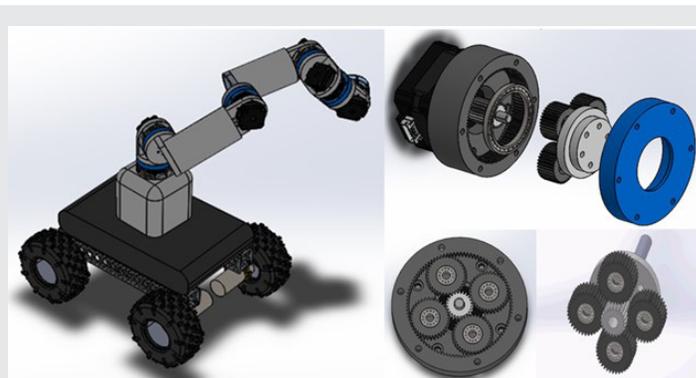


Figura 7. Diseño CAD del Cobot móvil en SolidWorks. Fuente: Elaboración propia.

Para validar el esquema de conexión propuesto, el módulo SIM7000 primero se enlaza a las antenas de telefonía celular mediante los servicios de LTE-CATM1. Posteriormente, inicia un enlace mediante el protocolo MQTT para establecer una comunicación bidireccional con el Broker-MQTT. Una vez establecida la comunicación el microcontrolador principal envía una cadena corta de información conformada por los datos del sensor, la fecha y la hora. En total se activaron tres dispositivos idénticos al mismo tiempo. Estos dispositivos fueron programados con la capacidad para suscribirse y publicar a los temas del Broker-MQTT. El Broker-MQTT propuesto, es un servicio internacional llamado MyQTTHub [20] que provee diferentes planes de servicios, configurables por el usuario. La plataforma MyQTTHub provee de un plan gratuito cuyas características principales se presentan la Tabla 3 y que fue el que se utilizó en este trabajo.

Tabla 3. Resumen del plan gratuito de MyQTTHub.

Descripción	Cantidades
Total de Usuarios Activos	100
Paneles de Información	3
Conexiones (máx.)	50
Almacenamiento	10MB
Suscripciones (máx. Conectados)	50
Mensajes por día	10 000
Mensajes al mes	310 000
Tamaño de mensaje (máx.)	4KB

Fuente: Elaboración propia con información recuperada de [20].

En este plan el usuario puede configurar o dar de alta los dispositivos que se conectarán a él, mediante un token o key como contraseña y hasta 100 diferentes contraseñas. También se pueden crear los temas donde los dispositivos se suscriben o publicarán sin límite de temas, pero se debe tener en cuenta que solamente se aceptarán 50 suscripciones al mismo tiempo. El máximo de mensajes que se pueden enviar en este plan es de 10000, con una longitud máxima de 4 kB cada mensaje.

Para este proyecto se configuraron en total tres temas, denominados: Configuración, Tareas y Datos. Los primeros dos son para que cada dispositivo pueda recibir comandos de configuración y de tareas a desempeñar, por lo tanto, los dispositivos deben suscribirse a éstos siempre que establecen comunicación con el Broker-MQTT. El tema Datos es creado para que todos los sistemas publiquen sus datos sensados. Para evaluar la viabilidad de una red de dispositivos por celular, los dispositivos fueron separados y puestos en diferentes edificios del Instituto Tecnológico Superior de Atlisco, donde fueron llevadas a cabo las pruebas.

## RESULTADOS

Las pruebas consistieron en establecer enlace con el Broker-MQTT de manera automática por parte de los tres dispositivos. En cada uno de ellos se configuró una ventana de tiempo entre cada mensaje, de manera que cada dispositivo se enciende por un lapso de 5 minutos. Durante este tiempo los dispositivos pueden enviar y recibir información, y posteriormente se duermen por un lapso de 30 minutos. Con ello se alcanzó un total de 144 mensajes por día. La longitud máxima de cada mensaje fue de 50 Bytes, sin tomar en cuenta los metadatos que se generan al establecer comunicación con el Broker-MQTT.

La cadena de datos enviada consiste de una cabecera que señala el nombre del sensor que envía la información, seguido de los datos de temperatura (°C), humedad relativa (%), presión atmosférica (hPa), fecha y hora. En la cadena se utiliza el carácter ASCII de la coma como separador de datos. A continuación, se muestra un ejemplo de la cadena enviada por cada dispositivo:

`DISPX,25.5,70.0,790.6,dd/mm/aa,hr/min`

Todos los dispositivos enviaron información con un desfase de tiempo promedio de 40 s. entre cada dispositivo, esto se debe a que cada dispositivo se despierta a la hora establecida, pero el tiempo que requieren para establecer enlace con la antena es aleatorio. Los dispositivos pueden recibir información del Broker-MQTT al suscribirse a los temas "Configuración" y "Tareas". Durante las pruebas un usuario encargado del sistema puede reconfigurar el reloj interno de cada dispositivo mediante una cadena de datos con el comando correspondiente. Los comandos que se implementaron para el proyecto se muestran en la Tabla 4. Se debe tomar en cuenta que estos comandos se envían a través del Tema "Configuración" y todos los dispositivos suscritos al tema pueden recibir el comando. Sin embargo, únicamente el dispositivo que coincide con el identificador es el que ejecuta y responde al comando. La cadena de respuesta que envía el dispositivo se puede observar en la última fila de la Tabla 4. Esta respuesta es enviada al tema "Configuración". El tema "Tareas" es empleado para asignar tareas simples a cada dispositivo. De manera que sí un dispositivo recibe una tarea a realizar durante el tiempo que se encuentra activo este debe realizarla antes de apagarse. Los comandos previstos para las tareas se muestran en la Tabla 5.

Las respuestas a estas tareas son enviadas al tema de "Datos" con la estructura de cadena de datos ya establecido previamente. Si un dispositivo recibe una tarea de muestreo para un solo sensor, este responderá con una cadena de datos, omitiendo los datos no requeridos.

Tabla 4. Comandos creados para Configuración.

Cadena enviada	Descripción
DISPX,reløj,dd/mm/aa,hr/min	Configura la fecha y hora del dispositivo.
DISPX,off/on,Temperatura	Permite apagar la lectura de un sensor en particular (Temp, Hum, Pres).
DISPX,ventana,mm,mm	Configura la ventana de tiempo (en minutos) que permanecerá encendido y apagado.
DISPX,iden,DIPSY	Permite cambiar el nombre de identificación de un dispositivo.
DISPX,com,OK	Respuesta del dispositivo con el comando (com) ejecutado y un OK.

Fuente: Elaboración propia.

Tabla 5. Comandos creados para Tareas.

Cadena enviada	Descripción
DISPX,mues,num,seg	Ejecuta el muestreo de los datos un determinado número (num) de veces (máx. 20 muestras), en intervalos de número de segundos (seg)
DISPX,temp,num,seg	Ejecuta el muestreo de la temperatura un determinado número (num) de veces, en intervalos de número de segundos (seg).
DISPX,hum,num,seg	Ejecuta el muestreo de la humedad un determinado número (num) de veces, en intervalos de número de segundos (seg).
DISPX,pres,num,seg	Ejecuta el muestreo de la presión atmosférica un determinado número (num) de veces, en intervalos de número de segundos (seg).

Fuente: Elaboración propia.

Las Figuras 8, 9 y 10 presentan las gráficas de los datos almacenados en el Broker-MQTT de las variables ambientales. Durante la recepción de los datos se detectaron tres eventos en los que el dispositivo 1 y el 2 no enviaron datos dentro de la ventana de tiempo. No fue posible establecer con precisión cual fue la falla exacta, pero se cree que se debe a que ocurrieron durante una hora en la que las condiciones del clima registraron lluvias y nubosidad densa en la región. Y se considera que estas condiciones pudieron contribuir a una desconexión temporal por parte del proveedor de servicio de telefonía celular.

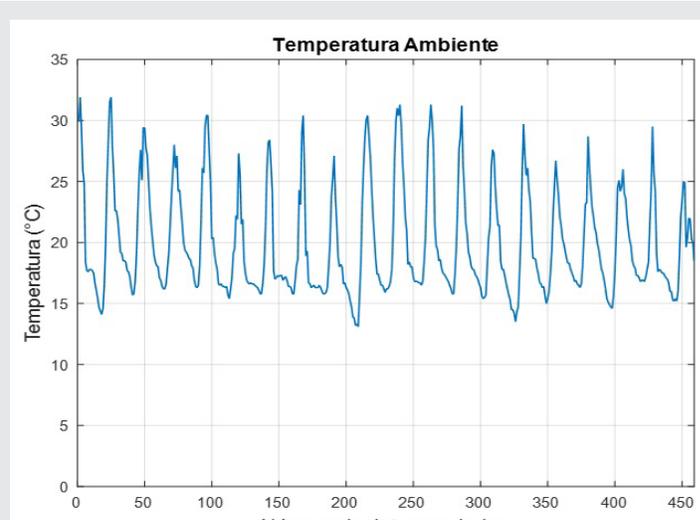


Figura 8. Gráfica de los datos de temperatura almacenados en el Broker-MQTT durante 1 mes con 460 datos, donde cada onda representa la temperatura registrada en 1 día. Fuente: Elaboración propia.

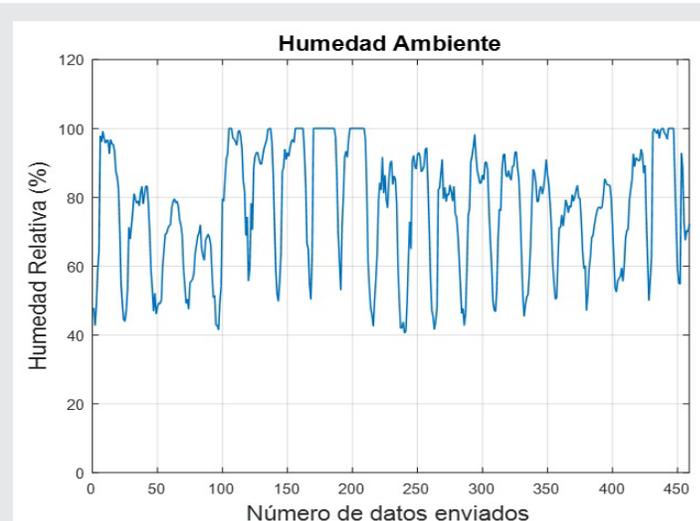


Figura 9. Gráfica de los datos de humedad relativa almacenados en el Broker-MQTT durante 1 mes con 460 datos. Fuente: Elaboración propia.

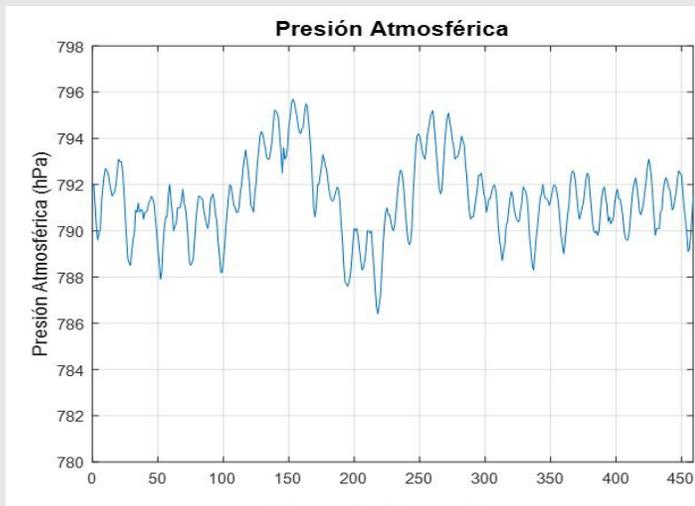


Figura 10. Gráfica de los datos de presión atmosférica almacenados en el Broker-MQTT durante 1 mes con 460 datos.  
Fuente: Elaboración propia.

Todos los dispositivos respondieron y realizaron las tareas de muestreo asignadas por el servidor. Las tareas asignadas solamente se programaron una vez por día, y con un máximo de 20 muestras por cada sensor, esto con el fin de no sobrepasar el número de mensajes máximo que el Broker-MQTT permite por día. Las configuraciones se realizaron 1 vez cada 15 días, con el fin de modificar el reloj interno de los dispositivos. En ninguno de los casos, fue necesario reconfigurar la ventana de tiempo de cada dispositivo.

Con respecto a las conexiones físicas, cada dispositivo se alimenta con una batería de Li-Ion, con una carga de 10000 mAh. Esto permitió una autonomía de 1 mes de pruebas continuas en cada dispositivo. Cada dispositivo se encendió unas 1440 veces durante el mes, lo que permite inferir que el consumo promedio del dispositivo por cada evento de encendido fue de 6.9 mA.

Se adquirieron chips SIM de tres proveedores de telefonía celular diferentes disponibles en México (Telcel, AT&T y Movistar) con el fin de evaluar cuál de estos presenta la mejor infraestructura y estabilidad de sus servicios IoT. Cada chip fue instalado en un dispositivo. Se pudo notar que el servicio de prepago de la compañía Telcel no permite la recuperación de la hora y fecha correctamente. Los 3 eventos en donde no se enviaron datos, correspondieron a los dispositivos que tenía instalados los chips de AT&T y Movistar.

El enlace solamente se pudo realizar mediante el estándar LTE-CATM1, por lo que el estándar LTE-NB-IoT no fue posible probarlo. Esto se debió a que en el momento de realizar las pruebas los tres proveedores únicamente ofrecen los servicios de IoT mediante el estándar CATM1.

## CONCLUSIONES

Este artículo presentó una propuesta para comunicar dispositivos a través del protocolo MQTT. Este trabajo en particular corresponde a una de las etapas de desarrollo de un proyecto de investigación, el cual consiste en desarrollar un ecosistema de robots móviles colaborativos que se interconecten entre sí y ejecuten aplicaciones educativas. Por lo que la propuesta de comunicación presentada en este trabajo es apropiada para que los profesores expliquen sobre las aplicaciones de la Industria 4.0 y del IoT en sistemas educativos.

Las pruebas de evaluación se realizaron utilizando la tarjeta de desarrollo Ophyra como un dispositivo de conexión. Se eligió esta tarjeta porque incorpora un microcontrolador ARM y este será la unidad central de control con la que se equipará cada Cobot. Los resultados que se obtuvieron demostraron que todos los dispositivos respondieron y realizaron las tareas de nuestro asignadas por el servidor; por lo que el progreso obtenido es la base para continuar con el proyecto de investigación aplicado a los Cobots.

## BIBLIOGRAFÍA

- [1] Aliev, K., Antonelli, D., Awouda, A., & Chiabert, P. (2019). *Key performance indicators integrating collaborative and mobile robots in the factory networks*. In *Working Conference on Virtual Enterprises* (pp. 635-642). Springer, Cham.
- [2] Kianoush, S., Savazzi, S., Beschi, M., Sigg, S., & Rampa, V. (2020). *A multisensory edge-cloud platform for opportunistic radio sensing in cobot environments*. *IEEE Internet of Things Journal*, 8(2), 1154-1168.
- [3] Sivan, M., Hagari, G., & Baranwal, E. (2021). *Cobots for FinTech*. In *2021 International Conference on Electrical, Computer, Communications and Mechatronics Engineering (ICECCME)* (pp. 1-4). Mauritius, Mauritius.
- [4] González, D. E., Verdugo, A. C., Escobar, L. F., y Loza, D.C. (2018). *Implementation of an IoT Architecture based on MQTT for a Multi-Robot System*. In *2018 IEEE Third Ecuador Technical Chapters Meeting (ETCM)* (pp. 1-6). Cuenca, Ecuador.
- [5] Garcia, C. A., Montalvo-Lopez, W., & Garcia, M. V. (2020). *Human-robot collaboration based on cyber-physical production system and MQTT*. *Procedia manufacturing*, 42, 315-321.
- [6] Mishra, B. y Kertesz, A. (2020). *The use of MQTT in M2M and IoT systems: A survey*. *IEEE Access*, 8, 201071-201086.
- [7] Köksal, Ö. y Tekinerdogan, B. (2017). *Feature-driven domain analysis of session layer proto-*

- cols of internet of things. In 2017 IEEE International Congress on Internet of Things (ICIOT) (pp. 105-112). Honolulu, HI, USA.
- [8] Bauer, J., y Aschenbruck, N. (2017). Measuring and adapting MQTT in cellular networks for collaborative smart farming. In 2017 IEEE 42nd conference on local computer networks (LCN) (pp. 294-302). Singapore.
- [9] Pooja, S., Uday, D. V., Nagesh, U. B., & Talekar, S. G. (2017). Application of MQTT protocol for real time weather monitoring and precision farming. In 2017 International Conference on Electrical, Electronics, Communication, Computer, and Optimization Techniques (ICECCOT) (pp. 1-6). Mysuru, India.
- [10] Hapsari, A. A., Hajamydeen, A. I., Vresdian, D. J., Manfaluthy, M., Prameswono, L., & Yusuf, E. (2019, December). Real time indoor air quality monitoring system based on IoT using MQTT and wireless sensor network. In 2019 IEEE 6th International Conference on Engineering Technologies and Applied Sciences (ICETAS) (pp. 1-7). Kuala Lumpur, Malaysia.
- [11] Atmoko, R. A., Riantini, R., y Hasin, M. K. (2017). IoT real time data acquisition using MQTT protocol. In *Journal of Physics: Conference Series*, 853(1), 1-6. IOP Publishing.
- [12] Atmoko, R. A., y Yang, D. (2018). Online monitoring & controlling industrial arm robot using MQTT protocol. In 2018 IEEE International Conference on Robotics, Biomimetics, and Intelligent Computational Systems (Robionetics) (pp. 12-16). Bandung, Indonesia.
- [13] Ayerra González, A. (2021). Integración de robot colaborativo en sistema de fabricación flexible en el contexto de la Industria 4.0. Obtenida el 15 de septiembre de 2022, de la página electrónica: <https://addi.ehu.es/handle/10810/53936>.
- [14] Brenman L. (2018). API Builder and MQTT for IoT. Obtenida el 10 de septiembre de 2022, de la página electrónica: <https://blog.axway.com/product-insights/amplify-platform/api-builder/api-builder-and-mqtt-for-iot-part-1>.
- [15] SIMCOM Wireless Solution (2022). SIM7000X. Obtenida el 1 de agosto de 2022, de la página electrónica: <https://www.simcom.com/product/SIM7000X.html>.
- [16] SIMCOM Wireless Solution (2019). SIM700 Serie AT Command Manual v1.06. Obtenida el 1 de agosto de 2022, de la página electrónica: <https://www.simcom.com/product/SIM7000X.html>.
- [17] Aliexpress (2022). NB IoT módulo SIM7000G Breakout LTE SIM7000G Kits. Obtenida el 18 de septiembre de 2022, de la página electrónica: <https://es.aliexpress.com/item/1005002837681411.html>.
- [18] Intesc Electronics & Embbebed (2022). Manual de Usuario de Ophyra. Obtenida el 17 de septiembre de 2022, de la página electrónica: <https://intesc.mx/wp-content/uploads/2021/08/ManualOphyraRevE.pdf>.
- [19] MyCobot280 (2022). Obtenida el 20 de septiembre de 2022, de la página electrónica: <https://www.elephantrobotics.com/en/mycobot-en/>.
- [20] MyQttHub.com (2022). Planes MyQttHub.com. Obtenida el 1 de agosto de 2022, de la página electrónica: <https://myqttHub.com/>.