

Creación de un sistema IoT embebido en el microcontrolador Esp32 para la gestión de sensores y actuadores en modo AP (Punto de acceso) y STA (Estación)

Luis Carlos Triviño Torres

Universidad Santo Tomás

Ingeniería informática

Bogotá, Colombia

luistrivino@usta.edu.co

Resumen— Este proyecto, desarrollado por el semillero de investigación Ciberondas del programa de Ingeniería Informática de la Universidad Santo Tomás, se centró en la creciente relevancia del Internet de las Cosas (IoT), transformando la interacción con dispositivos y el entorno. Su objetivo era diseñar un sistema IoT basado en el microcontrolador Esp32, que gestionara eficientemente sensores y actuadores, adaptándose a diferentes entornos de conectividad con modos de operación punto de acceso AP y STA. La metodología incluyó la integración de una API REST para comunicación entre el Esp32 y servicios externos, y el uso del protocolo MQTT, compatible con el broker EMQX en entornos locales y en la nube. La interfaz de usuario se desarrolló en HTML, CSS y JavaScript, implementada en un dashboard web para monitoreo y control en tiempo real. Los resultados destacaron la versatilidad del sistema, empleando el Esp32 como unidad central y sensores de alta precisión, contribuyendo significativamente al campo del IoT.

Palabras clave: IoT, Esp32, Monitoreo, MQTT, Conectividad

Introducción

El Internet de las Cosas (IoT, por sus siglas en inglés) ha emergido como un paradigma innovador que conecta dispositivos físicos a través de internet, permitiendo la automatización y el control remoto en diversos

entornos (Atzori, Iera & Morabito, 2010). Este proyecto, desarrollado por el semillero Ciberondas de la Universidad Santo Tomás, tiene como objetivo diseñar e implementar un sistema IoT embebido en el microcontrolador ESP32 para la gestión de sensores y actuadores en tiempo real.

El sistema está diseñado para operar tanto en modo punto de acceso (AP) como en modo STA, utilizando el protocolo MQTT y una API REST para facilitar la comunicación entre dispositivos y servicios externos. La metodología incluyó el desarrollo de un dashboard web para el monitoreo y control del sistema, así como pruebas de rendimiento en entornos controlados.

Los resultados demostraron la versatilidad y eficiencia del sistema en la gestión de sensores y actuadores, destacándose por su eficiencia energética y capacidad de adaptación a diferentes escenarios de conectividad.

Este proyecto contribuye significativamente al campo del IoT al ofrecer una solución robusta y escalable para aplicaciones que requieren monitoreo y control en tiempo real (Al-Fuqaha et al., 2015; Gubbi, Buyya, Marusic & Palaniswami, 2013).

Justificación

El proyecto se fundamenta en la creciente demanda de soluciones IoT que sean robustas tanto técnica como operativamente, además de flexibles y escalables para diversos entornos de uso. Los sistemas IoT convencionales enfrentan desafíos significativos en términos de

conectividad, gestión de datos e integración con sistemas existentes (Patel & Patel, 2016).

Al desarrollar un sistema basado en ESP32 con una API REST y el protocolo MQTT, este proyecto aborda estas limitaciones, ofreciendo una plataforma adaptable para aplicaciones tanto domésticas como industriales. Los beneficios de este sistema se extienden a múltiples sectores, incluyendo el hogar, donde facilita la implementación de soluciones domóticas; la industria, mejorando la automatización de procesos; la salud, permitiendo un monitoreo más eficiente; el agro y la ganadería, optimizando la gestión de recursos y el monitoreo en tiempo real; y la automatización en general, donde se pueden integrar fácilmente en sistemas existentes para mejorar la eficiencia operativa.

Además, fomenta la adopción de tecnologías IoT en nuevos contextos, facilitando su acceso a una amplia gama de usuarios y potenciando su aplicabilidad en diversos sectores. El sistema tiene como objetivo mostrar toda la información del dispositivo al usuario a través del dashboard, permitiendo un mejor control sobre el mismo. El sistema también permite dos modos de operación: localmente y con acceso a internet, asegurando así su flexibilidad y utilidad en diferentes condiciones de uso.

Utiliza memoria y almacenamiento interno para el firmware embebido, lo que facilita el acceso del usuario final al dispositivo. Otra ventaja significativa es la capacidad de actualizar el firmware a través del dashboard, permitiendo al usuario adaptar el dispositivo a condiciones variables de uso de manera sencilla.

Materiales y Métodos

El diseño del sistema IoT se basa en una arquitectura modular centrada en el microcontrolador ESP32. Este dispositivo puede operar en dos modos: punto de acceso (AP) y STA. En el modo AP, el ESP32 actúa como un punto de acceso Wifi independiente, mientras que en el modo STA se conecta a redes Wifi existentes, integrándose en la infraestructura de red del usuario.

En cuanto al hardware, la selección de componentes incluyó el ESP32-WROOM-32D

como unidad central, así como los sensores DHT22 y DS18B20 para la medición de temperatura y humedad. También se incorporaron actuadores como LEDs RGB y relés de estado sólido para diversas funciones de control.

Para el desarrollo del software, se utilizó el entorno Arduino IDE para programar el ESP32. El sistema incluyó la implementación de una API REST para la gestión de datos y el uso del protocolo MQTT para la comunicación eficiente entre dispositivos.

El dashboard web fue desarrollado utilizando HTML, CSS y JavaScript. Este sigue una arquitectura de Single Page Application (SPA) y utiliza websockets para permitir actualizaciones en tiempo real de los datos recogidos por el sistema IoT.

Se llevaron a cabo pruebas exhaustivas de rendimiento en entornos controlados para evaluar la conectividad del sistema, su eficiencia energética y capacidad de respuesta en diferentes escenarios de operación. Estas pruebas aseguraron la fiabilidad y funcionalidad del sistema en diversos entornos de uso.

A continuación, se muestra las características principales de los dos modos usados.

TABLA I Características del modo AP y STA

Característica	Modo AP (Punto de acceso)	Modo STA (Estación)
Función Principal	El ESP32 actúa como un punto de acceso Wi-Fi, creando su propia red.	El ESP32 actúa como un cliente, conectándose a una red Wi-Fi existente.
Conectividad	Permite que otros dispositivos se conecten a su red Wi-Fi.	Se conecta a un punto de acceso externo, como un enrutador o hotspot.
Dirección IP	El ESP32 asigna direcciones IP a los dispositivos conectados.	El ESP32 recibe una dirección IP del punto de acceso al que se conecta.
Aplicaciones Comunes	Redes locales ad hoc, configuración de dispositivos, aplicaciones sin acceso a Internet.	Conexión a Internet, comunicación con servidores remotos, control de dispositivos a distancia.
Flexibilidad	Menos dependiente de infraestructura	Depende de la existencia de un punto de acceso

	externa, ideal para aplicaciones locales.	Wi-Fi para conectarse.
Acceso a Internet	No tiene acceso directo a Internet; depende de una conexión compartida si es necesario.	Acceso total a Internet y recursos de red una vez conectado.
Configuración de Red	Crea una nueva red Wi-Fi con SSID y contraseña definidos por el usuario.	Se conecta a una red existente proporcionando SSID y contraseña del AP.
Rango de Dispositivos Conectados	Puede soportar varios dispositivos conectados simultáneamente.	El ESP32 es uno de los muchos dispositivos conectados al AP.

El modo AP del ESP32 es ideal cuando se requiere que el microcontrolador cree su propia red Wi-Fi. En este modo, el ESP32 actúa como un punto de acceso, permitiendo que otros dispositivos se conecten directamente a su red. Esta configuración es particularmente útil en situaciones donde no se dispone de un enrutador o punto de acceso externo, ya que el ESP32 gestiona completamente la red, asignando direcciones IP y permitiendo la comunicación local entre los dispositivos conectados.

Por otro lado, el modo STA es utilizado cuando el ESP32 necesita conectarse a una red Wi-Fi existente. En este caso, el microcontrolador actúa como un cliente, buscando y uniéndose a un punto de acceso externo, como un enrutador o hotspot. Este modo es esencial cuando se requiere acceso a Internet o comunicación con otros dispositivos en la misma red, permitiendo que el ESP32 interactúe con servidores remotos, envíe y reciba datos a través de la red, o sea controlado a distancia.

Resultados

El sistema IoT desarrollado demostró un rendimiento en diversos entornos de conectividad, tanto en modo AP como en STA. Los resultados clave se presentan en la siguiente tabla:

TABLA 2 Rendimiento en los modos AP y STA

Parámetro	Modo AP	Modo STA
Cobertura efectiva	30 metros en espacios abiertos	Conectividad estable en redes 2.4GHz y 5GHz
Tiempo de	150 ms	80 ms

respuesta (API)		
Consumo energético (activo)	120 mA	120 Ma
Consumo energético (bajo consumo)	10 μ A	10 μ A
Latencia en MQTT	20 ms (broker local)	100 ms (broker en la nube)

La tabla compara el rendimiento del ESP32 en los modos AP y STA según varios parámetros clave.

En cobertura efectiva, el modo AP ofrece hasta 30 metros en espacios abiertos, creando su propia red Wi-Fi. En contraste, el modo STA se conecta a redes Wi-Fi existentes, aprovechando la estabilidad de las bandas 2.4GHz y 5GHz.

El tiempo de respuesta en el modo AP es de 150 ms, debido a la gestión de la red por parte del ESP32, mientras que en el modo STA es más rápido, con 80 ms, aprovechando la eficiencia de la red Wi-Fi.

Ambos modos consumen 120 mA en activo y solo 10 μ A en bajo consumo, mostrando un rendimiento energético similar.

La latencia en MQTT es de 20 ms en el modo AP (con un broker local) y 100 ms en el modo STA (con un broker en la nube), reflejando la mayor distancia de transmisión en el segundo caso.

Discusión

Los resultados obtenidos subrayan la capacidad del ESP32 para adaptarse eficazmente en la gestión de sistemas IoT, destacando su versatilidad en los modos de operación AP y STA. Mientras que el modo STA ofrece una latencia más baja, ambos modos mantienen un consumo energético similar, lo que es ventajoso para aplicaciones en entornos remotos. Esta eficiencia energética permite al ESP32 operar durante períodos prolongados con un bajo consumo, lo que es crucial para aplicaciones en el hogar, la industria, la salud, el agro, la ganadería, y la automatización de procesos.

A pesar de estos logros, se identificaron áreas donde es posible mejorar, como el

soporte para una gama más amplia de sensores y actuadores, así como la incorporación de medidas de seguridad más avanzadas para proteger la integridad de los datos. Estos aspectos podrían potenciar aún más la adaptabilidad y la seguridad del sistema, permitiendo su uso en una variedad aún mayor de aplicaciones. Además, el proyecto destaca por su capacidad para actualizar el firmware a través de un dashboard, lo que ofrece una ventaja significativa al usuario al permitirle ajustar el dispositivo a condiciones variables de uso con facilidad.

Conclusiones

1. El sistema IoT basado en ESP32 desarrollado en este proyecto demuestra ser una solución eficiente, flexible y adaptable para la gestión de sensores y actuadores, lo que lo convierte en una opción viable para una amplia variedad de entornos, desde aplicaciones domésticas hasta industriales.
2. La integración del protocolo MQTT y una API REST no solo facilita una comunicación fluida y escalable entre dispositivos, sino que también potencia la interoperabilidad del sistema con otros servicios y plataformas, lo que incrementa su aplicabilidad en sectores como la domótica, la salud, el agro y la automatización de procesos.
3. Aunque el sistema muestra un rendimiento robusto, se identifican áreas de mejora cruciales, como la implementación de medidas de seguridad avanzadas para proteger la integridad de los datos y la ampliación de la compatibilidad con un mayor número de dispositivos y sensores. Estos aspectos serán abordados en futuros desarrollos para fortalecer la eficacia y seguridad del sistema.

Referencias

- Al-Fuqaha, A., Guizani, M., Mohammadi, M., Aledhari, M., & Ayyash, M. (2015). Internet of Things: A survey on enabling technologies, protocols, and applications. **IEEE Communications Surveys & Tutorials**, 17(4), 2347-2376.
- Atzori, L., Iera, A., & Morabito, G. (2010). The Internet of Things: A survey. **Computer Networks**, 54(15), 2787-2805.
- Espressif Systems. (2022). ESP32 Series Datasheet. Retrieved from

https://www.espressif.com/sites/default/files/documentation/Esp32_datasheet_en.pdf

Fielding, R. T., & Taylor, R. N. (2000). Architectural styles and the design of network-based software architectures. University of California, Irvine.

Gubbi, J., Buyya, R., Marusic, S., & Palaniswami, M. (2013). Internet of Things (IoT): A vision, architectural elements, and future directions. **Future Generation Computer Systems**, 29(7), 1-17.

Patel, K. K., & Patel, S. M. (2016). Internet of Things-IOT: Definition, Characteristics, Architecture, Enabling Technologies, Application & Future Challenges. **International Journal of Engineering Science and Computing**, 6(5), 6122-6131.