

# Sesión de Laboratorio 8: IoT con M5Core2

---

**Tiempo estimado:** 1,5 h (1 sesión)

## 1. Trabajando con hardware real

### Consideraciones generales

En esta práctica se va a implementar una red MQTT formada por dos dispositivos M5Core2 conectados a través de un broker común. Dentro de la red, uno de los dispositivos tendrá el rol de publisher y publicará en un topic si se ha pulsado algún botón. El otro dispositivo hará de subscriber y, en función del mensaje recibido del publisher, cambiará el color de la pantalla tal y como se describe más abajo.

- La red WiFi a la que se tienen que conectar los dispositivos será la red local:
  - SSID: Inflnd
  - Password: InflndMQTT
- Como broker se puede utilizar alguna de las siguientes opciones:
  - Un broker público como [test.mosquitto.org](http://test.mosquitto.org) o [mqtt.eclipseprojects.io](http://mqtt.eclipseprojects.io).
  - Un broker privado creado con *mosquitto* si se han seguido los pasos de [esta página](#).
- **IMPORTANTE:** ¡Cuidado con los nombres! Si se va a trabajar en un broker público hay que tener cuidado con el nombre del cliente y de los topics:
  - En un mismo broker no puede haber dos clientes con el mismo nombre.
  - Si se utiliza un nombre muy genérico en un broker público (p.ej., "test" o "topic") es muy probable que haya algún otro cliente que publique o se subscriba a esos topics. En este caso, es conveniente utilizar nombres no genéricos (p.ej., usando tu nombre) para evitar interferir con otros clientes de la red.

### Parte 1: Publisher

Hacer un script para M5Core2 (publisher) que haga lo siguiente:

- Si se pulsa el botón A (`\texttt{M5.BtnA}`) publica en un topic (p.ej.: `\M5\color`) el mensaje "Rojo".
- Si se pulsa el botón B (`\texttt{M5.BtnB}`) publica en un topic (p.ej.: `\M5\color`) el mensaje "Amarillo".
- Si se pulsa el botón C (`\texttt{M5.BtnC}`) publica en un topic (p.ej.: `\M5\color`) el mensaje "Verde".
- **INFO:** Para ver si se ha pulsado algún botón se puede usar la función `wasReleased()` (p.ej., `M5.BtnA.wasReleased()`).

### Parte 2: Subscriber

Hacer un script para M5Core2 (subscriber) que haga lo siguiente

- Se subscribe al topic donde publica el otro M5Core2 (p.ej.: \M5\color)
- La pantalla se pone del color del mensaje que se ha recibido. Se puede utilizar la función \texttt{\{fillScreen()\}} (p.ej., M5.Lcd.fillScreen(RED)).

## Parte 3: Ampliación

- En el publisher: El mensaje no se publica hasta que el botón A se pulsa durante más de un segundo. Es decir, se debe almacenar el último botón que se ha pulsado (A, B, o C), pero el mensaje no se envía hasta que el botón A se pulsa durante más de un segundo. Se puede utilizar la función `pressedFor()`.
- En el subscriber: Si el mensaje no es ninguno de los anteriores, la pantalla se pone de color negro.
- En el subscriber: Publicar en el topic \confirmacion el mensaje "Mensaje recibido" cuando se haya recibido un mensaje y se cambie el color de la pantalla. De esta forma, el dispositivo será subscriber y publisher al mismo tiempo.
- En ambos: Escribir en la pantalla del dispositivo que hace de publisher las posibilidades (p.ej: *Pulsar A para "Rojo", o Mensaje "Rojo" enviado*). Escribir en la pantalla del dispositivo que hace de subscriber el estado de los mensajes recibidos (p.ej: *Mensaje ``Rojo'' recibido, cambiado color*).

## 2. Trabajando en simulación

El ejercicio se puede realizar en simulación siguiendo las instrucciones que se describen en [esta página](#)