**MQTT -Arduino Wifi rev2 ou MKR 1010**

Tutoriel : https://docs.arduino.cc/tutorials/uno-wifi-rev2/uno-wifi-r2-mqtt-device-to-device

Bibliothèque à installer : ArduinoMqttClient et WiFiNINA

Ajouter la carte Arduino Wifi rev2 ou Arduino MKR Wifi 1010 suivant votre cas.

Code :

```
#include <ArduinoMqttClient.h>
#include <WiFiNINA.h>
#include "arduino_secrets.h"

///////please enter your sensitive data in the Secret tab/arduino_secrets.h
char ssid[] = SECRET_SSID;        // your network SSID (name)
char pass[] = SECRET_PASS;    // your network password (use for WPA, or use as key for WEP)

WiFiClient wifiClient;
MqttClient mqttClient(wifiClient);

const char broker[] = "test.mosquitto.org";
int        port     = 1883;
const char topic[]  = "real_unique_topic";
const char topic2[]  = "real_unique_topic_2";
const char topic3[]  = "real_unique_topic_3";

//set interval for sending messages (milliseconds)
const long interval = 8000;
unsigned long previousMillis = 0;

int count = 0;

void setup() {
  //Initialize serial and wait for port to open:
  Serial.begin(9600);
  while (!Serial) {
    ; // wait for serial port to connect. Needed for native USB port only
  }
```

```
  // attempt to connect to Wifi network:
  Serial.print("Attempting to connect to WPA SSID: ");
  Serial.println(ssid);
  while (WiFi.begin(ssid, pass) != WL_CONNECTED) {
    // failed, retry
    Serial.print(".");
    delay(5000);
  }

  Serial.println("You're connected to the network");
  Serial.println();

  Serial.print("Attempting to connect to the MQTT broker: ");
  Serial.println(broker);

  if (!mqttClient.connect(broker, port)) {
    Serial.print("MQTT connection failed! Error code = ");
    Serial.println(mqttClient.connectError());

    while (1);
  }

  Serial.println("You're connected to the MQTT broker!");
  Serial.println();
}

void loop() {
  // call poll() regularly to allow the library to send MQTT keep alives
which
  // avoids being disconnected by the broker
  mqttClient.poll();

  unsigned long currentMillis = millis();

  if (currentMillis - previousMillis >= interval) {
    // save the last time a message was sent
    previousMillis = currentMillis;

    //record random value from A0, A1 and A2
    int Rvalue = analogRead(A0);
    int Rvalue2 = analogRead(A1);
```

```
    int Rvalue3 = analogRead(A2);

    Serial.print("Sending message to topic: ");
    Serial.println(topic);
    Serial.println(Rvalue);

    Serial.print("Sending message to topic: ");
    Serial.println(topic2);
    Serial.println(Rvalue2);

    Serial.print("Sending message to topic: ");
    Serial.println(topic2);
    Serial.println(Rvalue3);

    // send message, the Print interface can be used to set the message
contents
    mqttClient.beginMessage(topic);
    mqttClient.print(Rvalue);
    mqttClient.endMessage();

    mqttClient.beginMessage(topic2);
    mqttClient.print(Rvalue2);
    mqttClient.endMessage();

    mqttClient.beginMessage(topic3);
    mqttClient.print(Rvalue3);
    mqttClient.endMessage();

    Serial.println();
  }
}
```