

Modifications 2020

9.3 Connaître l'état des capteurs en se connectant au serveur MQTT

Intégrer la fonctionnalité MQTT à votre programme :

Ajouter les fichiers `mqtt.h` et `mqtt.cpp` sur le modèle des autres fonctionnalités.

Modifier la fonction `reconnect()` pour la rendre non bloquante : ne faire qu'une tentative de connexion et retourner l'état de connexion. Appeler la dans la boucle `mqtt` et mettre fin à la fonction si la connexion échoue. On fera une tentative au prochain tour si nécessaire. Le but est de ne pas bloquer les autres fonctionnalités si on perd la connexion au broker.

L'ESP publie périodiquement l'état de chaque capteur (toutes les 30 secondes par exemple).

Au départ utiliser un broker local.

Ajouter des champs dans la structure de configuration pour la période de publication et pour l'adresse (ou le nom dns) du broker :

```
struct CONFIG {  
    char ssid[32] ; // ssid of wifi network to connect to  
    char passwd[64] ;  
    char myhostname[32] ; // the name of this ESP  
    char mqttServerIp[128] ; // ip ou nom du serveur mqtt  
    unsigned long mqttPublishPeriod ; // milliseconds  
    unsigned long sensorsPeriod ; // milliseconds  
};
```

Schéma des topics capteurs (XXYY sont les deux derniers octets de l'adresse MAC de l'ESP, il est important d'identifier correctement l'ESP pour éviter la confusion lorsque vous serez connecté au même broker)

Topic	Format des messages
/espXXYY/temperature	21.4
/espXXYY/humdity	62.2
/espXXYY/light	243

- Vérifier que vous pouvez utiliser un broker en ligne. Par exemple `test.mosquitto.org`
- Installer l'application `mqtt-explorer` : <http://mqtt-explorer.com/>
- Abonnez vous à vos topic et tracer les graphiques avec `mqtt-explorer`.
- Abonnez vous aux topics de vos camarades.

Pendant le Tp, je mettrai un service un ESP qui publiera des données issues de véritables capteurs sur le broker `test.mosquitto.org` . Les topics sont identiques avec l'adresse MAC **776A**.

9.4 Commande d'actionneur en passant par un serveur MQTT

On souhaite commander le relais (RELAY simulé par une led) , la LED_ANT par des messages MQTT. Quand on commande un actionneur il est important de s'assurer que la commande a bien été exécutée¹. Pour cela on va créer deux topics par actionneur :

un topic **set** qui permet de commander l'actionneur,

un topic **state** qui permet de connaître l'état courant de l'actionneur.

L'ESP va s'abonner au topic **set** de chaque actionneur. S'il reçoit une commande valide sur ce topic, il exécute cette commande et publie le nouvel état de l'actionneur sur le topic **state** correspondant. En l'absence de commande, il publie périodiquement l'état de l'actionneur sur la topic **state**².

Schéma des topics actionneurs (XXYY sont les deux derniers octets de l'adresse MAC de l'ESP)

Topic	Format des messages ³
/espXXYY/led/set	0 , 1
/espXXYY/led/state	0 , 1
/espXXYY/relay/set	0 , 1
/espXXYY/relay/state	0 , 1

- En vous accordant avec un autre étudiant, commander sa led en appuyant sur le bouton FLASH de votre ESP. Attention il faut publier un seul message lors d'un appui sur le bouton et inverser la commande lors du prochain appui.

Utiliser une machine d'état avec le pseudo code ci dessous :

msg ← "1"

E0 : Si appui passer à l'état E1

E1 : envoyer msg, changer msg pour la prochaine fois puis passer à E2

E2 : Si relâché passer à E0

- Si vous avez un téléphone Android commander la led avec l'application `mqttDash` :

<https://play.google.com/store/apps/details?id=net.routix.mqttdash&hl=fr>

1 Avec les QOS de MQTT on peut s'assurer que le message a bien été transmis mais on ne pas être certain que la commande a bien été exécutée. Seul un retour fonctionnel permet de s'en assurer.

2 On peut aussi utiliser le mode *retain* pour qu'un client qui vient de s'abonner soit informé de l'état sans attendre la publication périodique.

3 Il vaut mieux prendre une logique normale (1 allumée, 0 éteint) dans les messages mqtt et prendre en compte l'inversion au moment de piloter la led (`digitalWrite()`) ou de lire son état (`digitalRead()`).

9.5 Une page web exploitant mqtt.

Faire une page qui affiche les données réelles issues de l'esp 776A par mqtt (ou les données simulés d'un camarade).

Vous pouvez soit vous inspirer de ce qui a été fait avec AJAX ou WebSocket : Votre ESP souscrit aux topics des capteurs et met à jour des variables lorsqu'il reçoit les valeurs. La page envoie périodiquement une requête pour demander la valeur de ces variables.

Ou alors vous pouvez écrire un code JS qui se connecte directement au broker et s'abonne aux topics. Voir par exemple :

<http://www.steves-internet-guide.com/using-javascript-mqtt-client-websockets/>

Ajouter un bouton pour commander la led à distance.

9.6 Un chat à travers MQTT ?

Avec une page web ou en utilisant le moniteur série vous pouvez envoyer et recevoir des messages texte à travers MQTT.

Attention si vous passez par l'ESP à la limitation de taille : 123 octets pour topic+payload.