

**KONKARLAB**  
**Configuration (mode OTAA) d'un premier node**  
**sur le réseau "The Things Network"**

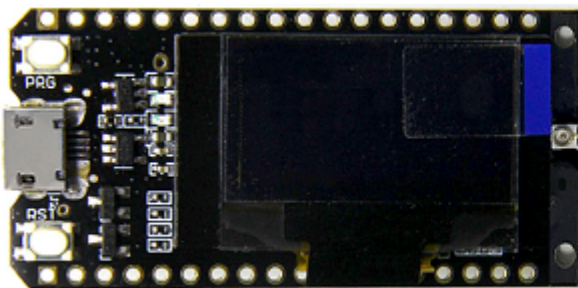
(dernière mise à jour : 18-01-2018 par Tix)

**Résumé des différentes étapes :**

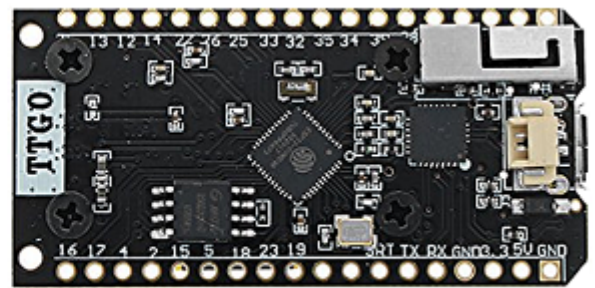
1. Version de l'ESP utilisée pour ce test
2. S'inscrire sur le réseau "The Things Network"
3. Se logger et accéder à la console
4. Créer une application
5. Ajouter un device dans cette application
6. Personnaliser votre code (en y précisant les clés fournies à l'étape précédente)
7. Télécharger le code dans l'ESP
8. Une gateway est-elle en service pour mes essais ?
9. Mettre le node en service (le temps de prendre votre pain à la boulangerie Leroux par exemple !)
10. S'assurer de la bonne transmission (coup d'œil sur votre console et/ou moniteur série Bluetooth sur un Android)

-----  
**Les étapes (plus en détail) :**

1/ Le sketch joint à cet email permet de tester un premier node  
il fonctionne sur un module TTGO Lora32 Version 1.2 : quelques modifs à prévoir (gpio) pour la version 2 (dans le forum de TTN chercher "big esp32" )



TTGO LoRa32 V1, (433MHz or) 868MHz, top



TTGO LoRa32 V1, (433MHz or) 868MHz, bottom

(je dispose du même modèle que ci-dessus mais sans afficheur oled)

Mais avant de s'intéresser davantage à l'ESP32 petit détour vers le réseau :

2/ Il vous faut être connu par les serveurs de "The Things Network" ("TTN") :  
L'inscription se passe à partir de <https://www.thethingsnetwork.org/> (Signup en haut sur la droite)

Il y a de nombreux tutos sur le net. J'ai trouvé que le plus simple reste de commencer par la doc officielle ici :

<https://www.thethingsnetwork.org/docs/devices/registration.html>

C'est en anglais mais si nécessaire google translate fait l'affaire ;) )

Il y a aussi un forum qui regorge d'infos :

<https://www.thethingsnetwork.org/forum>

3/ Lorsque vous serez enregistré chez TTN vous pourrez ouvrir la page

<https://console.thethingsnetwork.org/>

*"This is where the magic happens. Here you can work with your data. Register applications, devices and gateways, manage your integrations, collaborators and settings."*

4/ C'est le moment de s'occuper d'une première "application" : elle est à créer ( <https://console.thethingsnetwork.org/applications> )

*Remarque : dans une "application" vous pourrez, par la suite, enregistrer plusieurs "devices" (mais si j'ai bien suivi un device ne peut être enregistré que dans une seule application !)*

5/ Puis vous pourrez ajouter un device : lorsqu'il sera créé vous aurez (automatiquement) les clés qu'il vous faudra alors entrer dans le sketch .ino  
Lorsque vous validerez l'enregistrement de votre device les clés seront automatiquement créées (ne pas s'inquiéter des cases restant vides donc lorsque vous avez choisi "generate" !)

ce petit logo =====>



*Remarque 1 : dans une "application" vous pourrez, par la suite, enregistrer plusieurs "devices" (mais si j'ai bien suivi un device ne peut être enregistré que dans une seule application !)*

*Remarque 2 : le code en pièce jointe correspond à un device configuré en mode OTAA (le choix OTAA/ABP vous est proposé lorsque vous créez le device, )*

Pour ces points 3 et 4 voir la doc :

<https://www.thethingsnetwork.org/docs/devices/registration.html> et plus généralement :

<https://www.thethingsnetwork.org/docs/>

6/ c'est maintenant le moment de s'intéresser au sketch et d'entrer les valeurs correspondant à votre device.

copié/collé sans oublier les inversions (avec les flèches, voir vidéo, dernier lien en bas de cette page ) si LSB demandé (précisé dans les commentaires dans le

sketch ; voir aussi annexe 1 en fin de ce document "Personnalisation du sketch pour config OTAA")

7/ c'est fini : ne pas oublier de connecter l'antenne puis télécharger le code ...

8/ assurez-vous que la passerelle est en service (!) ... car actuellement celle du fablab vient par exemple de partir en week-end à Quiberon (mibs days #2)

Si elle n'est pas connectée elle apparaît en rouge sur cette carte :

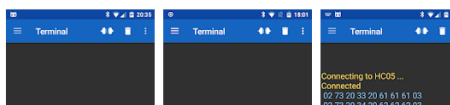
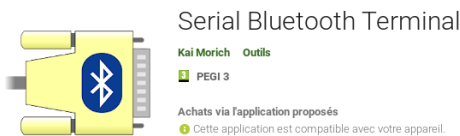
<https://www.thethingsnetwork.org/map> ... après que vous ayez zoomé ;)

9/ Sortez votre escabeau et montez sur la dernière marche : gardez une main pour l'escabeau et l'autre pour le module à bout de bras !

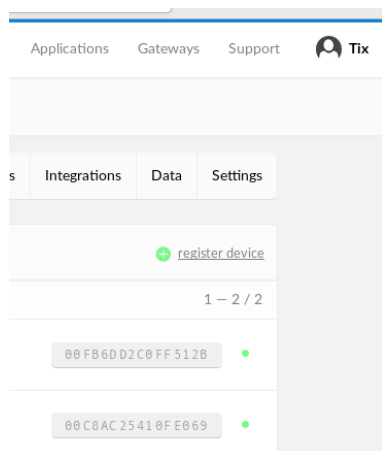
(j'ai aussi testé -sans l'escabeau- depuis le parking "Bureau Vallée" et depuis la boulangerie Leroux !) avec la petite antenne "quart d'onde" !

c'est le moment de surveiller les échanges entre votre node et la passerelle en vous connectant par bluetooth et en ouvrant un moniteur série.

J'ai téléchargé [https://play.google.com/store/apps/details?id=de.kai\\_morich.serial\\_bluetooth\\_terminal&hl=fr](https://play.google.com/store/apps/details?id=de.kai_morich.serial_bluetooth_terminal&hl=fr) mais il y en a d'autres !



10/ Respectez les paliers pour la descente -de l'escabeau- puis vérifiez que votre device est bien enregistré (vous aviez laissé, pour ce faire, votre pc en service et votre console ouverte (<https://console.thethingsnetwork.org>) Si le module a été reçu (dans la config OTAA) il apparaît avec un point vert dans la console.



Personnellement j'ai 2 devices enregistrés : en fait je n'en ai physiquement qu'un seul : une config est en "OTAA" et une seconde correspond à un autre sketch avec configuration "ABP" .

**Accessoire** : j'ai ajouté du bluetooth dans le code (je n'ai pas su utiliser le bluetooth ble ) pour pouvoir suivre via un moniteur série sur Android (j'ai téléchargé Bluetooth serial sur google play)

**Bibliothèque nécessaire à la compilation de ce code sous Arduino IDE :**

<https://github.com/matthijskooijman/arduino-lmic>

**À propos du code :**

a) le même code mais en configuration ABP est dispo dans les exemples de la bibliothèque indiquée ci-dessus.

b) prochaine étape en local : transmettre les données issues d'un capteur ... puis les retrouver sur le net !

(voilà le montage de Patrice que j'évoquais ce soir et qui utilise LPP Cayenne et qui pourrait nous servir d'aide : [https://wikifab.org/wiki/Mise\\_en\\_service\\_d%27un\\_thermom%C3%A8tre\\_connect%C3%A9](https://wikifab.org/wiki/Mise_en_service_d%27un_thermom%C3%A8tre_connect%C3%A9) )



c) Trouver un site pour la passerelle : antenne dégagée ... et à l'extérieur (quelque part à proximité du cinéma ? je dis ça comme ça ... au hasard ;)

d) il y aura alors la cartographie TTN Mapper à découvrir ... et les antennes à affuter ...

**Liens** : Si vous trouvez des liens sympas pour -mieux- décrire ces étapes, merci de partager :-)

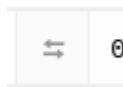
## Annexe 1 : Personnalisation du sketch pour config OTAA

Rendez-vous dans votre console (/applications/votre device) et copier-coller les éléments suivants vers votre sketch .ino

Astuce : la bascule Hex / C-style se fait par un clic sur



Attention : lorsque LSB est demandé l'inversion est facile avec un simple clic sur



```
// This EUI must be in little-endian format, so least-significant-byte
// first. When copying an EUI from ttntcl output, this means to reverse
// the bytes. For TTN issued EUIs the last bytes should be 0xD5, 0xB3,
// 0x70.
static const u1_t PROGMEM APPEUI[8]={ 0xCE, 0xCE, 0xCE, 0xCE, 0xCE, 0xCE, 0xCE, 0xCE }; //LSB first <===== VOIR Doc.pdf page 5 (Annexe 1) add Tix
void os_getArtEui (u1_t* buf) { memcpy_P(buf, APPEUI, 8);}

// This should also be in little endian format, see above.
static const u1_t PROGMEM DEVEUI[8]={ 0xCE, 0xCE, 0xCE, 0xCE, 0xCE, 0xCE, 0xCE, 0xCE }; //LSB first <===== VOIR Doc.pdf page 5 (Annexe 1) add Tix
void os_getDevEui (u1_t* buf) { memcpy_P(buf, DEVEUI, 8);}

// This key should be in big endian format (or, since it is not really a
// number but a block of memory, endianness does not really apply). In
// practice, a key taken from ttntcl can be copied as-is.
// The key shown here is the semtech default key.
static const u1_t PROGMEM APPKEY[16] = { 0xCE, 0xCE, 0xCE, 0xCE, 0xCE, 0xCE, 0xCE, 0xCE, 0xCE, 0xCE, 0xCE, 0xCE, 0xCE, 0xCE, 0xCE, 0xCE }; // <===== VO
void os_getDevKey (u1_t* buf) { memcpy_P(buf, APPKEY, 16);}
```

**APPEUI, DEVEUI et APPKEY** sont , pour une configuration OTAA, les données à récupérer depuis votre console et à recopier dans votre sketch.

... à suivre ...