

Application d'innovation

Positionnement en intérieur

Modification de l'architecture

Enseignants responsables du défi :

- Stéphane Huet
- Philippe Gozlan

Rappel et Constat.

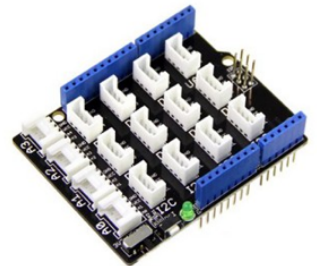
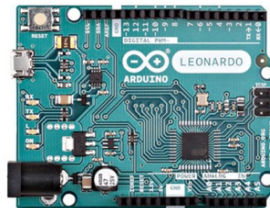
Lors du TP "défi" chaque groupe a pu travailler de façon complètement autonome avec son "objet" (coordinateur) et une ou plusieurs autres balises (end device) car vous étiez chacun dans un PAN différent donc "isolés" les uns des autres (paramètre ID dans XCTU). Malheureusement depuis le lancement de l'application d'innovation sur la géolocalisation d'intérieur les mesures et donc le stockage des valeurs des RSSI se sont trouvés perturbés voire impossibles. En effet le contexte est différent en ce sens que chaque groupe auquel est attribué un "objet" doit communiquer avec les mêmes balises (au nombre de 5). Dans une configuration de réseau Zigbee il ne peut y avoir *qu'un seul coordinateur*. Dans notre application nous privilégierons naturellement une topologie en étoile.

Solution matérielle proposée.

Nous devons, pour que chacun d'entre vous, puissent "récupérer" le niveau de réception des 5 balises, modifier la structure des "Xbeecons".

La miniaturisation de ces dernières n'étant pas à l'ordre du jour nous vous proposons d'opérer les modifications structurelles des balises avec le matériel ci-dessous

- Carte Arduino léonardo + shield V2 grove
- Un adaptateur grove Xbee +cordon



- Un Power Bank RS 5Ah.



Comment allez-vous faire ?

Coté objet.

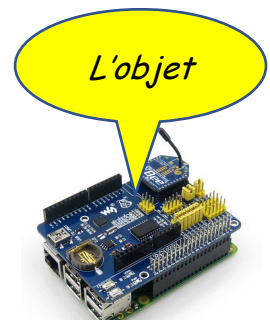
Tout d'abord intéressons-nous à votre coordinateur(l'objet).

Dans XCTU vous aller le configurer à minima de la manière suivante :

CE=1→ Coordinateur

AP =0→ Mode transparent ⁽¹⁾

⁽¹⁾ Il s'agit d'une vraie liaison RS232 mais "over the air"



DH=0 et DL=FFFF → Diffusion générale(broadcast)
CH=B ;
ID = Cn33 ou "n" représente votre N°de groupe (1 à 6).
Seul ce paramètre (PANID) vous différenciera.
NI= Le nom de votre objet.

Coté balises (déjà configurées).

Également dans XCTU :

CE=0 → End device

AP=0 → Mode transparent ou AT.

MY=FFFF

DH=DL=0 → accès au coordinateur

NI= BALn ou n=1 à 5.

Les "Xbeecons" ne posséderont plus de cavalier. La "loopback" va se faire autrement.
C'est ce que nous allons voir.

La Léonardo possède une UART avec laquelle nous pouvons communiquer via les broches repérées par TXD1 et RXD1.

Sur le Shield Grove, cette UART est sérigraphiée UART. La léonardo est déjà programmée pour "changer " automatiquement en boucle de PANID.

Mise en place de l'échange et récupération du paramètre DB (RSSI)

Les différents objets doivent récupérer le RSSI lu(reçu) par chacune des balises mais aussi la provenance du RSSI.

Pour ce faire les objets (coordinateurs) vont envoyer une simple commande AT selon le protocole classique à savoir l'envoi de "+++" suivi de la commande AT considérée, à toutes les balises. Cette commande AT **est la commande ND** pour Node Discovered. En retour on récupérera l'adresse 64bits (SH, SL), le RSSI (DB) et le "petit nom" de la balise (NI). Notre "loopback" est assurée.

Coté balises et pour chacune d'elle l'implémentation d'un code sur la carte Leonardo utilisant l'UART de cette dernière (Serial1), modifiera leur (propre) paramètre ID en boucle afin qu'il corresponde à l'un de ceux des coordinateurs. Cet ID devra rester un temps suffisant pour que l'objet émetteur récupère les RSSI de toutes les balises.

Donc le code de la RBPi "objet" qui émettra la commande ATND récupérera, après s'être assuré de la réception de la totalité des infos provenant de chacune des balises, les différentes RSSI qu'il stockera.

On rappelle que pour initier une commande AT il faut produire "+++" attendre la réception de "OK" et ensuite on peut en toutes lettres éditer les commandes AT nécessaires, acquérir leur valeur, les modifier et enfin les émettre.

Plusieurs problèmes lors de la "rotation" des PANID peuvent survenir dont l'un concerne la synchronisation des 5 balises. Nous ne pourrons pas éviter la désynchronisation. Voilà pourquoi il faudra trouver une solution de codage coté RBpi fiable et pérenne...qui assure une réponse sûre de chacun des Xbeecons.

Ca aussi c'est un défi !