

**TP SIN****Bluetooth****Support : Carte Arduino et shield BLE****<http://www.commentcamarche.net/contents/107-fonctionnement-du-bluetooth>****Support : Carte arduino Méga et shield BLE****1. Présentation**

Le nom « Bluetooth » est directement inspiré du surnom anglicisé du roi danois Harald à la dent bleue (en danois Harald Blåtand, en anglais Harald Bluetooth), connu pour avoir réussi à unifier les tribus danoises au sein d'un même royaume, introduisant du même coup le christianisme.

C'est une technologie de réseau personnel sans fils (noté WPAN pour Wireless Personal Area Network), c'est-à-dire une technologie de réseaux sans fils d'une faible portée permettant de relier des appareils entre eux sans liaison filaire. Contrairement à la technologie IrDa (liaison infrarouge), les appareils Bluetooth ne nécessitent pas d'une ligne de vue directe pour communiquer, ce qui rend plus souple son utilisation et permet notamment une communication d'une pièce à une autre, sur de petits espaces.

**2. Caractéristiques**

Le Bluetooth permet d'obtenir des débits de l'ordre de 1 Mbps, correspondant à 1600 échanges par seconde en full-duplex, avec une portée d'une dizaine de mètres environ avec un émetteur de classe II et d'un peu moins d'une centaine de mètres avec un émetteur de classe I.

Le standard Bluetooth définit en effet 3 classes d'émetteurs proposant des portées différentes en fonction de leur puissance d'émission :

| Classe | Puissance (affaiblissement) | Portée       |
|--------|-----------------------------|--------------|
| I      | 100 mW (20 dBm)             | 100 mètres   |
| II     | 2,5 mW (4 dBm)              | 15-20 mètres |
| III    | 1 mW (0 dBm)                | 10 mètres    |

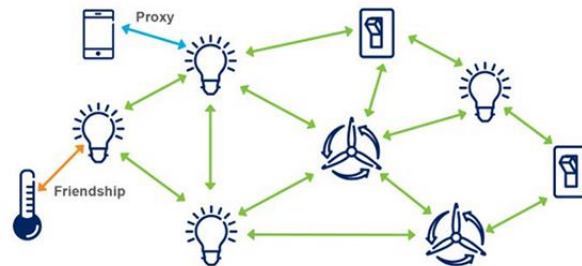
On trouve différentes versions de Bluetooth qui sont les suivantes :

- Bluetooth v1.0 et v1.0B ;
- Bluetooth v1.1, normalisé en 2002 sous le nom IEEE 802.15.1-2002 ;
- Bluetooth v1.2, normalisé en 2005 sous le nom IEEE 802.15.1-2005 ;

Nom : .....

Prénom : .....

- Bluetooth v2.0 + EDR, rendue publique en 2004 ;
- Bluetooth v2.1 + EDR, rendue publique en 2007 ;
- Bluetooth v3.0 + HS, rendue publique en 2009 ;
- Bluetooth v4.0, rendue publique en 2010, débit de données brutes de 1 Mbps ;
- Bluetooth v4.1, rendue publique en 2013.
- Bluetooth v4.2, rendue publique le 2 Décembre 2014. Il est 2.6 fois plus rapide et à une gestion de sécurité plus élevée.
- Bluetooth v5 rendue publique décembre 2016. Augmentation de la bande passante, d'où une augmentation du débit et de la portée.
- Bluetooth Mesh rendue publique en 2017. Mise en place du Bluetooth en réseau.



Dans un réseau maillé, au lieu d'un dispositif central communiquant avec des périphériques individuels, un message est relayé d'un point du réseau à un autre par saut de canaux bidirectionnels, reliant ainsi plusieurs nœuds. Il permet un contrôle simultané de dizaines de périphériques connectés.

### 3. Etablissement des connexions

Pour établir une connexion et échanger des données, il faut un côté serveur (device qui reçoit la connexion) et un côté client (device qui envoie la connexion). Un serveur peut avoir plusieurs clients mais pas l'inverse.

L'établissement d'une connexion entre deux périphériques Bluetooth suit une procédure relativement compliquée permettant d'assurer un certain niveau de sécurité, selon le déroulé suivant :

#### Remarque :

Pour les applications android pensez activer les autorisations Bluetooth.

#### **3.1. Bluetooth BR / EDR**

- Mode passif
- Phase d'inquisition : Découverte des périphériques distants

Seul l'un des périphériques peut être découvert, l'autre périphérique doit découvrir le périphérique détectable.

- Appairage avec un périphérique distant : Pairage à l'aide d'un code PIN (sécurité)

Dès que votre application obtient la liste des périphériques distants, vous pouvez commencer une requête d'appairage avec l'un de ces périphériques

- Découverte des services du point d'accès

Dès que le périphérique exécutant votre application est appairé avec un périphérique distant, vous pouvez vous connecter aux applications s'exécutant sur le périphérique distant.

- L'une des applications (serveur) doit publier un service.
- L'autre application (client) doit découvrir le service de l'application serveur et s'y connecter.
- On va chercher à obtenir un socket à connecter au service distant. Pour spécifier le service distant, on doit connaître la propriété UUID

UUID : Cet identificateur est représenté par une chaîne délimitée par des accolades. Les UUID doivent être au format "{xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx}", où chaque x est un chiffre hexadécimal (de 0 à 9 ou de A à F). Par exemple : "{00000000-0000-1000-8000-00805F9B34FA}".

- Puis démarrer la connexion avec le serveur.
- Une fois la connexion réalisée, on peut envoyer ou recevoir des données.

### 3.2. **Bluetooth Low Energy (BLE) v4.0**

Le **BLE** est principalement destiné aux objets connectés pour lesquels le besoin en débit est faible et l'autonomie cruciale ainsi qu'aux équipements nomades tels que les smartphones, les tablettes, les montres, etc. La portée se compte en quelques dizaines de mètres.

- Phase d'inquisition : Découverte des périphériques distants

Le BLE ne fonctionne pas comme un classique bluetooth, sur Android vous n'avez pas besoin d'appairer les périphériques. Les clients doivent découvrir les serveurs.

**Remarque :** Les applications Windows doivent appairer les périphériques avant la découverte des serveurs. Pour plus d'informations, voir [Windows Bluetooth FAQ \(EN\)](#).

- Une fois que vous avez découvert le périphérique, on doit commencer à obtenir les services et les caractéristiques du profil standard
- Découverte et obtention des services :

Les services sont des collections de caractéristiques et de relations avec d'autres services qui encapsulent le comportement d'une partie d'un périphérique.

Un profil GATT inclut un ou plusieurs services standard qui sont identifiés par un numéro. Par exemple, le numéro officiel suivant a été attribué au service Heart Rate (Fréquence cardiaque) : "0x180D".

Vous trouverez des informations concernant les services standard sur la page officielle des [services BluetoothLE standard \(EN\)](#).

Pour obtenir un service, vous devez utiliser l'UUID complet, c'est-à-dire le numéro attribué au service standard + le code Bluetooth de base, par exemple le code complet du service Heart Rate est 0000180D-0000-1000-8000-00805F9B34FB.

- Découverte et obtention des caractéristiques :

Un service peut contenir une ou plusieurs caractéristiques qui fournissent des informations pour le service particulier. Chaque caractéristique est identifiée par un numéro officiel. Par exemple, le service Heart Rate possède une caractéristique permettant de mesurer la fréquence cardiaque, Heart Rate Measurement avec l'UUID 00002A37-0000-1000-8000-00805F9B34FB, et une autre caractéristique permettant d'identifier l'emplacement du capteur corporel, Body Sensor Location avec l'UUID 00002A38-0000-1000-8000-00805F9B34FB.

Nom : .....

Prénom : .....

Chaque caractéristique a des propriétés (lecture, écriture) spécifiques. Ces propriétés sont importantes car elles permettent de connaître les différentes connexions possibles (mode écriture, lecture etc).

- Lecture, écriture ou abonnement à une caractéristique

Quand vous avez obtenu la caractéristique du service sur le serveur, vous pouvez lire ou écrire (suivant les propriétés) les informations contenues. Vous pouvez également vous abonner à une caractéristique pour obtenir les informations chaque fois qu'elle change. Par exemple, pour commencer à surveiller le capteur de fréquence cardiaque.

### 3.3. **Beacon**

Beacon signifie simplement balise. Une balise est un boîtier de quelques centimètres qui peut être installé où l'on veut et qui émet dans un rayon jusqu'à quelques dizaines de mètres en Bluetooth Low Energy. Ces balises se signalent aux smartphones qui sont dans leurs environnement et leur permettent de se géolocaliser précisément. On parle de micro-localisation (surtout à l'intérieur des bâtiments où le GPS ne fonctionne pas) mais au fond c'est une vieille technique : celle des phares qui aident les navires à se situer.

Une application spécifique doit être installée pour pouvoir recevoir les informations et les actions de l'appli déclenchées par la balise. Le Bluetooth doit être activé préalablement.

Lire <http://www.atlantico.fr/rdv/minute-tech/revolution-beacon-vrai-faux-technologie-geolocalisation-commerciale-qui-explose-christophe-benavent-1728832.html#AgS13Yz4BYfrEwM8.99>

Le Beacon intègre des informations dans ses données de service, en particulier les données spécifiques au fabricant ("Manufacturer Specific Data") décrites dans "Bluetooth Specification Version 4.1, page 2023 (EN)" <https://www.bluetooth.org/en-us>. Grâce à ces informations, n'importe quel périphérique Bluetooth Low Energy se trouvant à proximité peut identifier le beacon et calculer la distance qui les sépare, sans qu'il soit nécessaire de l'appairer ni d'établir une connexion.

Caractéristiques de la trame :

02 | 15 | E2 0A 39 F4 73 F5 4B C4 A1 2F 17 D1 AD 07 A9 61 | 00 00 | 00 00 | C8

- ID (uint8\_t) - Ce sera toujours 0x02
- Longueur des données (de uint8\_t) - Le nombre d'octets dans le reste de la charge utile = 0x15 (21 dec)
- 128-bit UUID (uint8\_t [16]) - L'ID 128 bits identifie votre entreprise / magasin / etc.
- Major (uint16\_t) - La valeur principale. Ce code correspond à un bâtiment de cette société (par exemple : major 1 = AS du Louvre, major 2 = AS de la Défense, major 3 = AS de la cinquième avenue)
- Minor (uint16\_t) - La valeur mineure. Ce code correspond à un emplacement dans le bâtiment (par exemple : minor 1 = rayon des iPad, minor 2 = rayon des accessoires, minor 3 = rayon des iMac, etc).
- TX Puissance (uint8\_t) - Cette valeur est utilisée pour tenter d'estimer la distance sur la base de la valeur RSSI

Le TX power est la force du signal mesurée à 1m de l'appareil (RSSI : Received Signal Strength Indication). Sachant que la puissance du signal diminue au fur et à mesure que l'on s'éloigne et en sachant la valeur RSSI courante (cette information est transmise avec le signal reçu), il est possible de calculer la différence.

- Vous devez d'abord connaître le format utilisé pour le beacon que vous souhaitez surveiller :

Nom : .....

Prénom : .....

Il existe deux formats disponibles le iBeacon et le **AltBeacon** :

- **iBeacon** : c'est le format défini par Apple. Pour l'API, l'iBeacon est considéré comme le **mode standard**.
- **AltBeacon** : il s'agit d'un format ouvert, connu sous le nom de beacon alternatif. Vous trouverez les spécifications complètes sur <http://altbeacon.org> (EN). Pour l'API, le beacon alternatif est considéré comme le **mode alternatif**.
- **Eddystone** (variante du iBeacon) : Cette balise en plus des informations transmises par un iBeacon, elle peut émettre une adresse URL (Eddystone-URL) ou des valeurs télémétriques du Beacon (Eddystone-TLM ) (température, tension de la batterie etc...).

- Informations sur le fabricant

Les périphériques Bluetooth Low Energy en mode service envoient à maintes reprises les données des services par voie radio sous la forme de différentes structures de type AD (**Advertisement Data types**). L'un des types AD des données de service est le type "Manufacturer Specific Data" qui est utilisé par les beacons. Les données incluses dans le type "Manufacturer Specific Data" fournies par l'API des beacons sont les suivantes :

- **UUID** : un identificateur unique qui identifie un groupe de beacons, par exemple les beacons d'une entreprise spécifique.
- **MajorID, MinorID** : identifie les régions au sein d'un UUID spécifique. MajorID identifie une sous-région au sein d'une région plus étendue définie par un UUID. MinorID spécifie une sous-division supplémentaire au sein d'un MajorID.
- **TxPower** : TxPower est la puissance mesurée à 1 mètre de distance. La distance approximative d'un beacon est calculée à l'aide de TxPower et du **RSSI** (Received Signal Strength Indicator).

- Recensement des régions à surveiller

Vous devez recenser les régions des beacons que vous souhaitez surveiller. La **région** est un groupe de beacons dont les UDID, MajorID et MinorID sont spécifiés.

- Démarrage et arrêt de l'analyse des beacons

Une fois les beacons recensés, vous pouvez commencer à les surveiller.

- Obtention d'informations à partir des beacons entrants

**Remarque** : Pour des raisons de compatibilité multiplateforme, si vous recensez plusieurs éléments région avec le même UUID et que vous souhaitez les surveiller, vous ne pourrez obtenir que les événements du dernier élément de région que vous avez recensé. Dans l'éventualité où vous souhaiteriez surveiller différentes régions, il existe deux façons de faire selon votre scénario :

- Recensez une région avec son **UUID** spécifique, afin de recevoir des événements associés à tous les beacons ayant le même **UUID**.
- Recensez une région avec son **UUID** spécifique et concevez votre app de façon à filtrer selon vos valeurs personnalisées et les exigences relatives à votre app.

#### **Remarque :**

Les services de localisation permettent aux apps basées sur la localisation d'utiliser les informations des réseaux cellulaires, Wi-Fi, GPS et des beacons pour déterminer votre emplacement approximatif. Pour détecter les beacons, vous devez activer les services de localisation.

Pour Android, pensez activer les autorisations Bluetooth.

Nom : .....

Prénom : .....

Pour la plate-forme iOS, si le mode alternatif est activé, vous n'avez pas besoin d'activer les services de localisation. En mode standard (iBeacons), la demande d'activation de la localisation doit être acceptée lors du premier démarrage de l'app.