Technologie Bluetooth

par Xavier LAGRANGE
Ingénieur de l'École Centrale Paris
Professeur à l'École nationale supérieure des télécommunications Bretagne (ENST Bretagne)

et Laurence ROUILLÉ
Docteur en informatique de l'université de Rennes - 1
Ingénieur recherche-développement, NeoSoft

Un réseau sans fil personnel, ou WPAN™ (Wireless Personal Access Network), a vocation à faire communiquer tous les appareils numériques situés dans le POS (Personal Operating Space). Le POS correspond à une sphère de l'ordre de 10 m de rayon centrée sur un individu. Des réseaux ne nécessitant pas d'infrastructure préalable (réseau ad hoc) se forment alors à

Toute reproduction sans autorisation du Centre français d'exploitation du droit de copie est strictement interdite. – © Editions T.I.

Dossier délivré pour
Madame, Monsieur
08/09/2008

l'intérieur même des POS ou à l'intersection, pour faire communiquer différents appareils numériques portatifs. L'autre caractéristique forte des WPAN est un faible coût par unité, condition essentielle à un déploiement massif assurant l'adoption d'un tel système.

Bluetooth peut être considéré comme le premier WPAN à avoir été spécifié et développé. Les applications supportées sont principalement le transfert de données et la transmission audio à 64 kbit/s dans des contextes variés :

— connexion sans fil entre un ordinateur, un clavier, une souris, une imprimante, etc. ;

— synchronisation à distance d'un téléphone mobile, d'un PDA et d'un ordinateur : transfert de fichiers ;

— utilisation d'un téléphone portable comme modem pour obtenir une connexion Internet ;

— téléphone 3 en 1 : talkie-walkie, téléphone sans fil, téléphone cellulaire ;

— applications de domotique, pilotage par un ordinateur de différentes installations : lave-linge, arrosage automatique, etc. ;

— applications automobiles : récepteur GPS Bluetooth, kit main libre pour l'utilisation d'un téléphone portable, etc.

Nous présentons tout d'abord le contexte de normalisation. Ensuite, nous décrivons l'architecture générale du système et ses différentes couches : couche physique, bande de base, etc., les mécanismes de sécurité, les profils et les performances.

Sigles et abréviations

Abréviation	Signification
ACL	Asynchronous Connection Less
ACO	Authenticated Ciphering Offset
AFH	Adaptive Frequency Hopping
AM_ADDR	Active Member Address
ARQ	Automatic Repeat reQuest
BD_ADDR	Bluetooth Device Address
BER	Bit Error Rate
CAC	Channel Access Code
CRC	Cyclic Redundancy Check
DAC	Device Access Code
DH	Data High
DIAC	Dedicated Inquiry Access Code
DM	Data Medium
DQPSK	Differential Quadrature Phase Shift Keying
DV	Data Voice
EDR	Enhanced Data Rate
ESCO	Extended SCO
FEC	Forward Error Correction
FHS	Frequency Hopping Synchronization
FHSS	Frequency Hopping Spread Spectrum
GFSK	Gaussian Frequency Shift Keying
GIAC	General Inquiry Access Code
GPS	Global Positioning System
HCI	Host Controller Interface
HEC	Header Error Code
HV	High-quality Voice
IAC	Inquiry Access Code
IEEE	Institute of Electrical and Electronics Engineers
ISM	Industrial Scientific and Medical
L2CAP	Logical Link Controller and Adaptation Protocol
LAP	Low Address Part

Sigles et abréviations

Abréviation	Signification
LC	Link Controller
LLC	Logical Link Control
LM	Link Manager
LMP	Link Manager Protocol
LSB	Least Significant Bit
MSC	Message Sequence Charts
MTU	Maximum Transmit Unit
NAP	Non significant Address Part
OBEX	Object Exchange
OFDM	Orthogonal Frequency Division Multiplexing
PDA	Personal Digital Assistant
PDU	Protocol Data Unit
PIN	Personal Identification Number
PM_ADDR	Parked Member Address
POS	Personal Operating Space
PPP	Point to Point Protocol
PSK	Phase Shift Keying
RSSI	Received Signal Strength Indicator
SAR	Segmentation And Reassembly
SCO	Synchronous Connection Oriented
SDP	Service Discovery Protocol
SIG	Special Interest Group
TCS	Telephony Control protocol Specification
TDD	Time Division Duplex
UAP	Upper Address Part
USB	Universal Serial Bus
UUID	Universally Unique IDentifier
UWB	Ultra Wide Band
WAP	Wireless Application Protocol
WLAN	Wireless Local Access Network
WPAN	Wireless Personal Access Network

Toute reproduction sans autorisation du Centre français d'exploitation du droit de copie est strictement interdite. – © EditionsT.I.

Dossier délivré pour Madame, Monsieur 08/09/2008

Tableau 1 – Récapitulatif des versions de Bluetooth SIG

Nº de version SIG	Approbation	Norme IEEE correspondante	Approbation par IEEE	Nouveautés
1.0B	Juillet 1999	Non repris par IEEE		Version de base
1.1	Avril 2002	802.15.1-2002	Avril 2002	Mesures RSSI (Received Signal Strength Indicator)
1.2	Novembre 2003	802.15.1-2005	Juin 2005	AFH (Adaptive Frequency Hopping), pour réduire les interférences avec Wi-Fi, ESCO (extended SCO), amélioration de la qualité radio par la retransmission des paquets corrompus.
2.0	Novembre 2004			EDR (Enhanced Data Rate), transmission 3 à 10 fois plus rapide et réduction de la consommation Nouveaux types de paquets

1. Contexte de normalisation

La spécification du système radio Bluetooth [1] est la résultante des efforts conjugués d'un groupe d'industriels réunis autour d'Ericsson en février 1998 dans le SIG (Special Interest Group) Bluetooth. Il a initialement été conçu comme un système de remplacement des câbles de connexion utilisés entre les appareils portatifs tels que les téléphones portables, les PDA (Personal Digital Assistant) et les ordinateurs portables. En particulier, il avait pour but de s'affranchir des problèmes d'interconnexion et de configuration liés à l'utilisation d'interfaces filaires propriétaires.

L'étude des WPAN™ au sein de l'IEEE (Institute of Electrical and Electronics Engineers) a commencé en 1997. Cette activité, démarrée dans le groupe 802.11 WLAN (Wireless Local Access Network) sous l'appellation *wearable computing*, a donné naissance en 1998 au groupe 802.15. Celui-ci est subdivisé en plusieurs sous-groupes dont le premier, appelé TG1 (ou 802.15.1), a eu pour mission de reprendre la spécification initialement produite par des industriels au sein du SIG Bluetooth, afin d'en faire un standard IEEE. La correspondance entre les documents du SIG Bluetooth et des normes IEEE est donnée dans le tableau 1.

Par ailleurs, en mars 2006, le SIG Bluetooth s'est engagé vers l'implémentation de l'UWB (Ultra Wide Band), pour contrer la menace représentée par l'USB 2.0 sans fil (ou WUSB) et ses débits de 60 Mbit/s. À l'origine de cette annonce, un accord signé entre le SIG Bluetooth et la WiMedia Alliance, l'association à l'origine d'une des deux normes UWB du marché, basée sur l'OFDM (Orthogonal Frequency Division Multiplexing). Les produits Bluetooth nouvelle version offriront des débits théoriques de 100 Mbit/s, soit un bond majeur comparé à la version 2.0 cantonnée à 3 Mbit/s théorique. La nouvelle norme 2.1 EDR devrait être approuvée au cours du premier semestre 2007 [4].

2. Architecture générale du système

2.1 Concept de piconet

Dans la suite, le terme station est utilisé pour désigner tout appareil (PDA, appareil photo numérique, etc.) susceptible de contenir un module de transmission Bluetooth.

Bluetooth a inauguré le concept de piconet : réseau éphémère se créant au gré des besoins. La gestion du piconet est centralisée et confiée à une station appelée maître. Par extension, toutes les stations non maître incluses dans un piconet sont appelées esclaves.

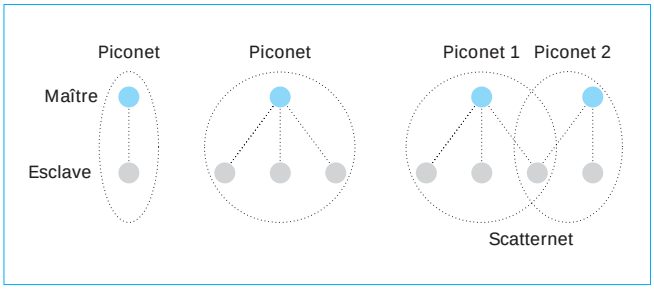


Figure 1 – Topologies de réseaux Bluetooth

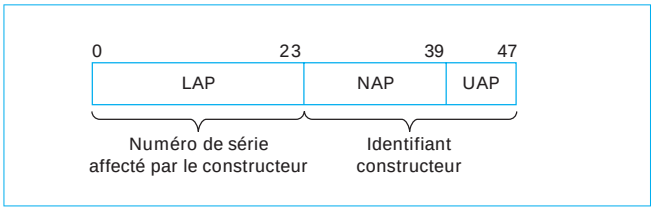


Figure 2 – Structure de l'adresse BD_ADDR compatible IEEE

Un maître ne peut communiquer qu'avec un des esclaves de son piconet et un esclave ne communique qu'avec son maître. Cette limitation peut en partie être levée grâce à l'interconnexion de plusieurs piconets par une station charnière, ou station relais. L'ensemble constitué d'au moins deux piconets interconnectés est appelé *scatternet* (figure 1). On peut avoir jusqu'à 10 piconets interconnectés. On peut cependant noter que le cas d'utilisation le plus courant est celui où le piconet est réduit à deux stations : un maître et un esclave (ce dernier étant en général un périphérique).

2.2 Identification

Chaque équipement Bluetooth possède un identifiant unique appelée BD_ADDR (Bluetooth Device Address). Il est codé sur 48 bits et reprend la même structure que pour Ethernet (spécification IEEE 802-2001). Il est constitué de trois parties (figure 2) :

— LAP (Low Address Part) sur 24 bits, correspond au numéro de série du module Bluetooth affecté par le constructeur ;
— NAP (Non significant Address Part) sur 16 bits ;
— UAP (Upper Address Part) est formée des 8 bits de poids forts.

Toute reproduction sans autorisation du Centre français d'exploitation du droit de copie est strictement interdite. – © Editions T.I.

Dossier délivré pour
Madame, Monsieur
08/09/2008

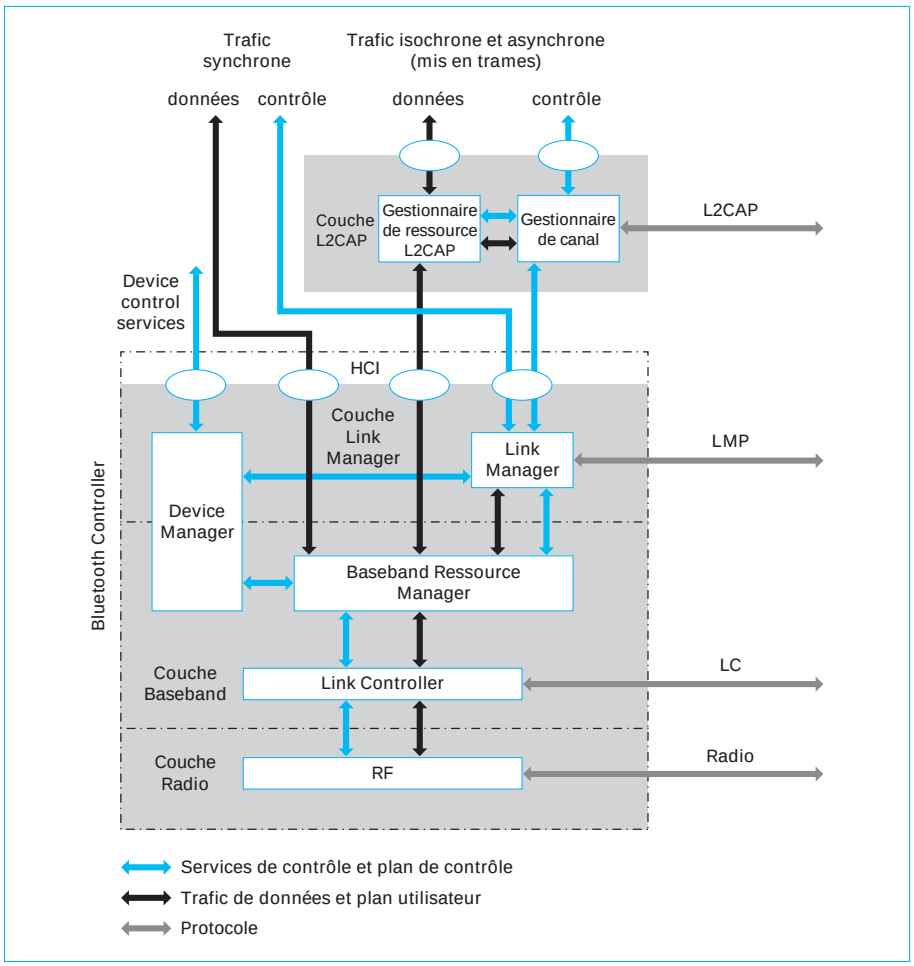


Figure 3 – Modèle architectural Bluetooth

Les deux parties NAP et UAP forment donc un ensemble de 24 bits (3 octets) qui constituent l'identifiant du constructeur du module Bluetooth (figure 2). Seules les parties LAP et UAP de l'adresse BD_ADDR sont utilisées dans les algorithmes (principalement l'algorithme de saut de fréquences). C'est cette particularité qui explique le nom NAP (partie non signifiante de l'adresse) attribué aux 16 bits du milieu.

2.3 Modèle architectural

Bluetooth propose un modèle architectural qui comprend un empilement de protocoles au sens du modèle OSI, mais qui suggère également une architecture physique, certaines fonctions étant placées dans un module spécifique Bluetooth et d'autres étant placées dans l'équipement (par exemple un ordinateur). Le modèle place également dans l'architecture les fonctions de gestion interne (figure 3).

Au sens purement protocolaire, l'architecture en couches est la suivante :

— la couche radio spécifie comment sont transmis et reçus les signaux, c'est-à-dire les fonctions de modulation et démodulation ;

— le protocole de contrôle de liaison (LC : Link Controller) comprend les échanges nécessaires pour l'établissement d'un piconet, son maintien ; il gère principalement le multiplexage des différentes transmissions sur une liaison radio unique ;

— le protocole de gestion du lien (LMP : Link Manager Protocol) permet l'établissement et le contrôle du lien (authentification, chiffrement, *polling*, gestion du mode économie de batterie) ;

— L2CAP (Logical Link Control and Adaptation Protocol) fournit des services orientés connexion ou non, aux protocoles de niveaux supérieurs.

Le multiplexage au niveau LC est géré par une entité interne appelée Baseband Resource Manager. Les autres fonctions de gestion sont prises en charge par l'entité Device Manager. Typiquement les couches basses (radio, LC, LMP) et les gestionnaires (Baseband Resource Manager et Device Manager) sont dans un même module physique appelé Bluetooth Controller. L'interface de commande de ce module est spécifiée précisément et constitue le HCI (Host Controller Interface).

Plusieurs protocoles standard peuvent être utilisés avec Bluetooth :

— SDP (Service Discovery Protocol) offre aux applications clientes les moyens de découvrir l'existence des services fournis par les applications serveurs, ainsi que les propriétés (attributs) de ces services ; il utilise les services de L2CAP ;

— TCS (Telephony Control protocol Specification) est un protocole de signalisation téléphonique au-dessus de L2CAP qui permet d'échanger des commandes (envoi d'un appel, décroché, raccroché, etc.) ;

— RFCOMM, permet l'émulation du port série au-dessus du protocole L2CAP ;

Toute reproduction sans autorisation du Centre français d'exploitation du droit de copie est strictement interdite. – © EditionsT.I.

Dossier délivré pour Madame, Monsieur 08/09/2008

— LCC (Logical Link Control) permet de disposer d'une couche liaison de données IEEE 802.2 au-dessus de L2CAP et donc d'être conforme aux standards 802 ;

— la couche Audio gère le codage/décodage numérique de la voix, qui est transportée directement par le LC.

La figure 3 donne un exemple d'implémentation d'une architecture Bluetooth. Les fonctions liées aux couches basses sont intégrées dans une carte Bluetooth spécifique. Il s'agit de la partie émission/réception radio (Radio Transceiver), du contrôle de lien en bande de base (Link Controller/Baseband) et de la gestion de lien (Link Manager). Les fonctions des couches plus hautes sont supposées être purement logicielles et intégrées par exemple dans le PC. Il s'agit de la liaison de données et de services spécifiques. Dans le présent document, le terme « module Bluetooth » est utilisé pour désigner l'entité matérielle hébergeant les fonctions des couches basses (Radio Transceiver, Link Controller/Baseband et LM).

3. Couche physique

3.1 Gammes de fréquences

La couche radio Bluetooth utilise la bande de fréquences sans licence ISM (Industrial Scientific and Medical) à 2,4 GHz. Cette bande de fréquences qui s'étale de 2 400 à 2 483,5 MHz est découpée en 79 canaux de 1 MHz. Il n'y a jamais de longue transmission sur une même fréquence, mais une série de transmissions de courts paquets sur une suite pseudo-aléatoire de fréquences.

3.2 Modulation

La modulation utilisée est GFSK (Gaussian Frequency Shift Keying) avec un débit de 1 Mbit/s sur la voie radio. C'est une modulation de fréquences de type FSK à deux états, dont le signal modulant est filtré par un filtre gaussien. Les données sont transmises sur une fréquence centrale F_c. La modulation FSK fait correspondre à un élément binaire de valeur 1, une déviation de fréquence positive (transmission sur la fréquence $F_c + \Delta f$), et à un élément binaire de valeur 0, une déviation négative (transmission sur la fréquence $F_c - \Delta f$). L'indice de modulation ($2 \times \Delta f / D$ où D est le débit) est compris entre 0,28 et 0,35, ce qui correspond à 140 kHz $\leqslant \Delta f \leqslant$ 175 kHz.

Bluetooth EDR (Enhanced Data Rate) défini dans la version 2.0 du SIG augmente le débit par un facteur de 2 à 3 grâce à des modulations plus évoluées. Avec la modulation PSK (Phase Shift Keying) de type 4-DQPSK (Differential Quadrature PSK), on dispose de 2 bits par symbole et on multiplie le débit par 2. Avec une modulation 8-DPSK, on dispose de 3 bits par symbole et on multiplie donc le débit par 3. Pour que l'augmentation de débit soit réelle au niveau applicatif, le taux d'erreur doit rester modéré et donc le rapport signal/bruit du signal doit être plus important si la modulation a plus de points dans la constellation. Par conséquent, l'augmentation de débit n'est pas assurée dans tous les cas de figure.

3.3 Caractéristiques radio : puissance et sensibilité

Trois classes de puissance ont été définies : classe 1 à 100 mW (20 dBm), classe 2 à 2,5 mW (4 dBm) et classe 3 à 1 mW (0 dBm). Les classes de puissance correspondent aux puissances maximales des équipements. Il est possible d'implémenter un algorithme de contrôle de façon à réduire la puissance réellement utilisée (tableau 2).

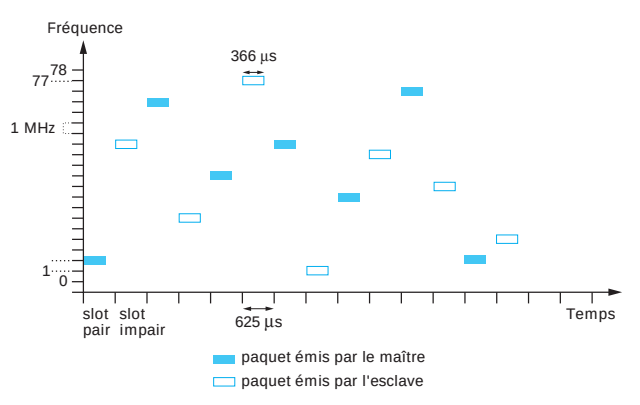


Figure 4 – Saut de fréquences

Tableau 2 – Différentes classes de puissance

Classe	Puissance maximale	Puissance minimale	Puissance nominale	Contrôle de puissance	Portée
Classe 1	100 mW (20 dBm)	1 mW (0 dBm)	N/A	Obligatoire entre 4 et 20 dBm	100 m
Classe 2	2,5 mW (4 dBm)	0,25 mW (– 6 dBm)	1 mW (20 dBm)	Optionnel	10 m
Classe 3	1 mW (0 dBm)	N/A	N/A	Optionnel	1 m

À l'intérieur, la puissance d'émission dans la bande de fréquences de 2,4 GHz reste limitée à 100 mW, tandis qu'à l'extérieur, l'usage de 100 mW est autorisé entre 2,4 et 2,454 GHz et réduit à 10 mW entre 2,454 et 2,483 5 GHz, afin d'éviter les risques de brouillage avec les appareils de communication militaires [17].

Le contrôle de la puissance est obligatoire pour les équipements de classe 1 et optionnel pour les autres classes. L'intérêt de contrôler la puissance d'émission est double : réduire l'interférence sur les autres utilisateurs du réseau et réduire la consommation d'énergie.

Le niveau de sensibilité du récepteur est défini pour un taux d'erreur bit (BER : Bit Error Rate) de 0,1 %. La norme impose une sensibilité au moins égale à – 70 dBm. Dans un milieu avec des obstacles, la portée maximale pour cette sensibilité et une puissance de 100 mW est d'environ 100 m.

3.4 Principe général de transmission

Nous considérons le cas d'un échange déjà établi entre un maître et un esclave. Nous supposons que l'esclave est déjà synchronisé sur le maître et connaît son adresse et décrivons comment les deux stations s'échangent des données *via* une liaison Bluetooth.

Le canal de transmission est divisé temporellement en slots de taille fixe (625 µs). Il y a 1 600 slots par seconde. Chaque équipement découpe les données à transmettre en courts paquets, un paquet étant transmis dans un slot. Un duplexage temporel TDD (Time Division Duplex) est employé pour permettre les transmissions bidirectionnelles. Le maître transmet pendant un slot et un esclave pendant le slot suivant.

Chaque paquet à transmettre est émis sur une fréquence différente de celle utilisée précédemment (figure 4). La séquence de sauts est déterministe mais possède une très longue période

Toute reproduction sans autorisation du Centre français d'exploitation du droit de copie est strictement interdite. – © Editions T.I.

Dossier délivré pour
Madame, Monsieur
08/09/2008

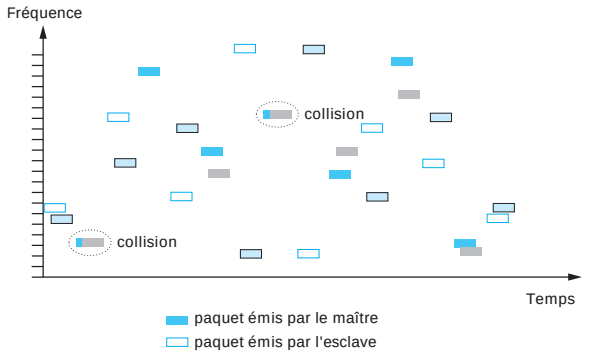


Figure 5 – Possibilités de collisions entre piconets

(elle est donc pseudo-aléatoire) ; elle dépend de la partie LAP de l'adresse du maître et de son horloge (en mode nominal) et est parfaitement connue du maître comme de l'esclave. Deux réseaux distincts n'ont pas la même séquence de sauts et ne sont pas synchronisés. Des collisions peuvent se produire. Comme le nombre de fréquences est important (79 fréquences), le risque de collision (figure 5) est faible mais non nul. Dans le cas où il y a collision, un paquet émis ne sera pas correctement reçu mais sera retransmis par l'émetteur, si nécessaire (mécanisme ARQ : Automatic Repeat reQuest).

Dans la version 1.0 de Bluetooth, le saut de fréquences se fait, pour une liaison établie sur les 79 fréquences disponibles. La version 1.2 du SIG Bluetooth introduit la technique AFH (Adaptive Frequency Hopping).

AFH n'utilise qu'un sous-ensemble de *N* fréquences (minimum 20) parmi les 79 fréquences définies sur la bande. C'est le maître qui décide quelles sont les fréquences à utiliser, en fonction des informations dont il dispose sur l'occupation des canaux. Cela permet, par exemple, d'éviter des fréquences qui sont très utilisées par un autre système.

Dans le cas le plus courant, un paquet dure 366 µs et occupe un slot. Il est cependant possible dans le mode ACL de transmettre des paquets plus longs (sur 3 ou 5 slots). Cela permet une plus grande efficacité car le ratio « taille de l'en-tête »/« longueur des données » est plus favorable. Dans certains états (phase de découverte par exemple), les équipements transmettent des paquets plus courts (1/2 slot).

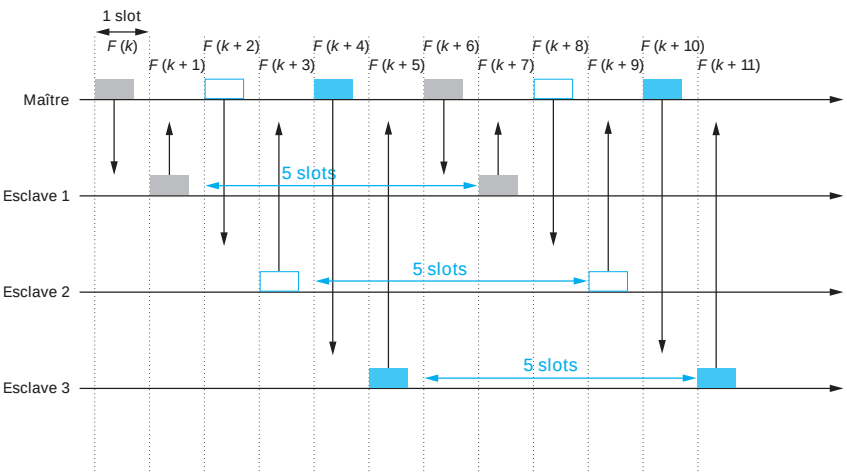
4. Principes de transmission sur un lien établi

La couche bande de base comprend l'ensemble des mécanismes de transmission dans le piconet, une fois qu'il est établi. Elle va fournir un service de transmission de données aux couches supérieures, soit de type synchrone SCO (Synchronous Connection-Oriented), soit de type asynchrone ACL (Asynchronous Connection-Less).

4.1 Liens physiques

Le lien SCO est un lien point-à-point entre le maître et un esclave. Il consiste à réserver régulièrement un slot pour le service considéré et il est donc adapté aux services comme la transmission de la voix. Il peut y avoir jusqu'à trois liens SCO par piconet (figure 6). Le maître réserve à des instants périodiques (tous les 2, 4 ou 6 slots), deux slots (un dans chaque sens) qui sont consacrés aux flux SCO. Lorsque le lien synchrone transporte des paquets utilisant une modulation PSK, il est appelé eSCO (extended SCO).

Le lien ACL est un lien point-à-multipoint entre le maître et l'ensemble des esclaves. Il n'en existe qu'un par piconet. Les liens directs entre esclaves ne sont pas possibles, mais peuvent éventuellement être simulés au niveau application. Le lien ACL utilise les slots non réservés par les liens SCO. Le maître attribue les slots afin de satisfaire au mieux les besoins des différents esclaves. Un esclave est habilité à émettre sur le canal s'il a reçu, dans le slot précédent, une trame du maître contenant son adresse



Pour la représentation, on fait abstraction du saut de fréquence.
F (*k*) donne la fréquence à utiliser pour le slot *k* selon l'algorithme de saut.

Figure 6 – Trois liens SCO au maximum dans un piconet

Toute reproduction sans autorisation du Centre français d'exploitation du droit de copie est strictement interdite. – © Editions T.I.

Dossier délivré pour Madame, Monsieur 08/09/2008

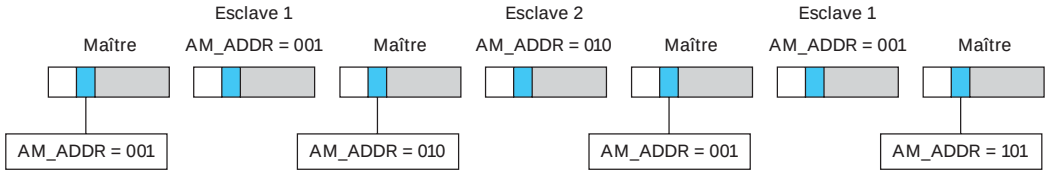


Figure 7 – Principe d'allocation de ressources pour le lien ACL

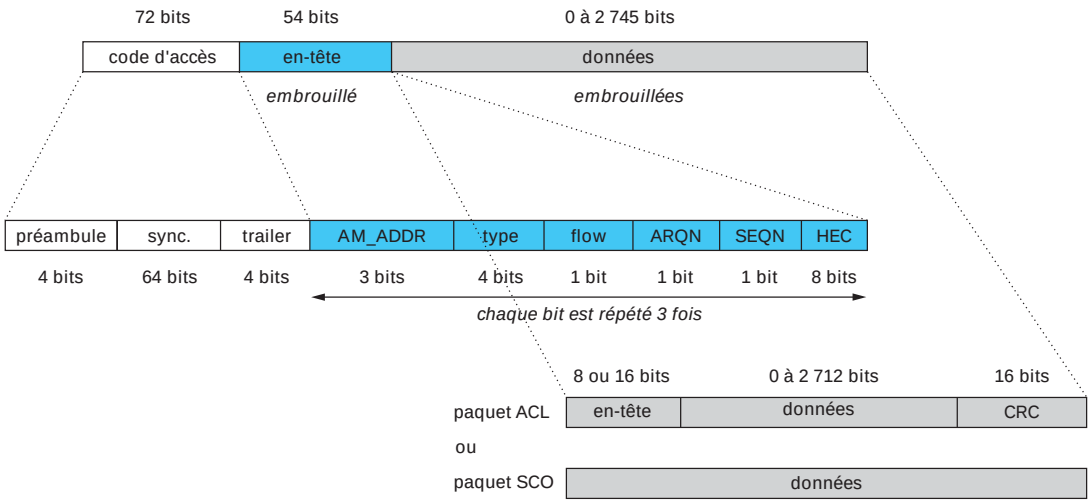


Figure 8 – Structure d'un paquet de données (mode normal)

comme adresse destination. Pour éviter des tailles d'en-tête importantes, on utilise une adresse courte codée sur 3 bits appelée AM_ADDR qui est allouée lors de la constitution du piconet. La figure 7 illustre le principe d'attribution des slots pour le lien ACL.

Le lien ACL est utilisé pour le transfert de données en tout genre. Les débits sur le lien sont rendus variables par l'utilisation de paquets plus ou moins longs. Ceux-ci peuvent occuper jusqu'à 5 slots. Le service ACL permet de disposer d'une fonction de contrôle de flux. Si son tampon de réception est plein, une station peut demander l'arrêt des émissions en affectant le bit FLOW.

Le tableau 4 résume les caractéristiques principales des paquets disponibles sur le lien ACL.

4.2 Format général des paquets

Un paquet est composé de trois parties : un code d'accès, un en-tête et le corps (figure 8). Le code d'accès est utilisé comme identifiant de piconet et permet la synchronisation du récepteur sur l'émetteur. L'en-tête de paquet qualifie le paquet. Il contient des informations telles que le type de paquet et l'adresse du destinataire. Le corps contient les données à transmettre. Dans certaines situations particulières, des paquets ne contiennent que le code d'accès ou bien le code d'accès et l'en-tête mais sans données.

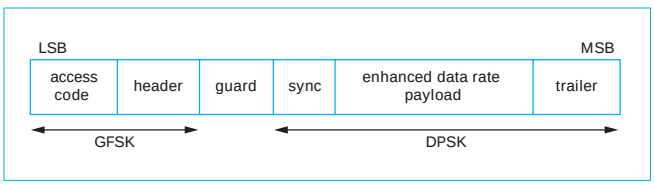


Figure 9 – Structure d'un paquet de données en mode EDR

Lorsque le mode EDR est utilisé (figure 9), la transmission commence avec le code d'accès et l'en-tête comme dans le mode normal pour assurer la compatibilité avec le mode normal. Elle se poursuit par une durée de garde, un code de synchronisation spécifique et quelques bits de fin (*trailer*) pour assurer un dernier symbole entier.

4.3 Chaîne de transmission

La chaîne de transmission est représentée sur la figure 10. Les éléments binaires de chaque paquet sont embrouillés pour éviter l'apparition trop fréquente de certaines suites de bits (*whitening*). L'embrouillage s'applique sur les champs en-tête et données (et pas sur le code d'accès).

Toute reproduction sans autorisation du Centre français d'exploitation du droit de copie est strictement interdite. – © Editions T.I.

Dossier délivré pour
Madame, Monsieur
08/09/2008

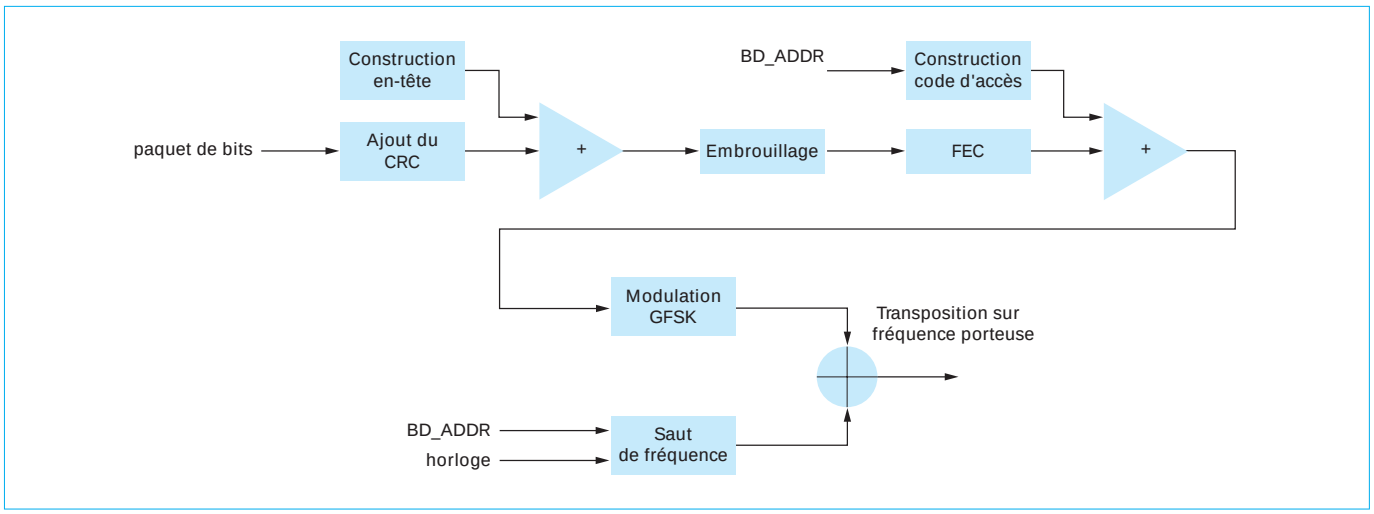


Figure 10 – Chaîne de transmission

L'« embrouillage » est basé sur une séquence pseudo-aléatoire générée par un registre à décalage. Le registre à décalage utilise la fonction $g(D) = D^7 + D^4 + 1$. On obtient donc une séquence de longueur $2^8 - 1 = 127$. Le registre est initialisé en fonction de l'horloge du maître.

4.4 Format détaillé d'un paquet

4.4.1 Composition du code d'accès

Le code d'accès contient un préambule, un mot de synchronisation et une séquence de transition. Le préambule sur 4 bits garantit des transitions 0-1 permettant au récepteur de se caler sur le rythme d'émission de l'émetteur. La séquence de synchronisation est calculée à partir d'une adresse (en général, l'adresse BD_ADDR du maître) de façon à posséder de bonnes propriétés d'autocorrélation. La séquence de transition sur 4 bits, appelée *trailer*, garantit également des transitions 0-1. Si le paquet est réduit au seul code d'accès, le *trailer* est absent.

Le code d'accès utilisé dépend de la phase au sein du processus de dialogue.

Pour un piconet actif, le code d'accès est bâti à partir de l'adresse BD_ADDR de la station maître. Il est appelé alors CAC (Channel Access Code). Cela signifie que le même code d'accès est utilisé par toutes les stations actives faisant partie du même piconet et que c'est une caractéristique du piconet. Cela évite qu'une station reçoive par erreur un paquet émis au sein d'un autre piconet.

Pendant la phase initiale de recherche de stations (phase *inquiry*, § 5.1.1), le code d'accès est bâti à partir de la LAP fixe de valeur 9E 8B 33. Le code d'accès résultant est appelé GIAC (General Inquiry Access Code). Toutes les stations sont donc susceptibles de se découvrir car elles sont aptes à se synchroniser sur le paquet émis.

Une procédure de recherche de stations particulières est envisagée dans la norme. Soixante-quatre valeurs de LAP (9E 8B 00 à 9E 8B 3F) sont réservées pour cela. On obtient des codes d'accès appelés DIAC (Dedicated Inquiry Access Code). Une valeur particulière correspond à la recherche d'un type spécifique d'équipements.

Pour la phase d'appel d'une station (*page*), le code utilisé est construit à partir de l'adresse de la station appelée (future station esclave). Le code est alors appelé DAC (Device Access Code).

4.4.2 En-tête

L'en-tête contient 18 bits protégés par un code FEC à répétition de taux 1/3 (chaque bit est répété trois fois). L'en-tête est constitué de l'AM_ADDR (Active Member Address), du type du paquet transmis (TYPE), d'un indicateur de contrôle de flux (FLOW), d'un indicateur d'acquittement (ARQN) et d'un numéro de séquence (SEQN). L'ensemble constitue 10 bits protégés par un code détecteur d'erreurs sur 8 bits de types CRC appelé HEC (Header Error Code).

4.4.3 Données ou charge utiles

La zone des données, *payload*, d'un paquet transitant sur un lien ACL contient un en-tête, les données et un code détecteur d'erreurs de type CRC. L'en-tête permet principalement d'indiquer l'entité utilisatrice de niveau supérieur (LM ou L2CAP), si le paquet est un fragment final ou non lorsqu'il transporte du L2CAP et la longueur des données utiles.

La zone des données d'un paquet SCO ne contient que des données. Pour un paquet transportant à la fois de la voix et des données (DV : Data Voice), la zone de données contient tout d'abord une partie réservée pour la voix, puis on retrouve la structure de données précédemment décrite pour un paquet de données ACL.

4.4.4 Types des paquets

Le tableau 3 rassemble tous les types de paquets définis dans Bluetooth, en précisant leur utilisation et leurs caractéristiques.

4.5 Mécanismes protocolaires

4.5.1 Mécanismes de correction d'erreurs

Comme la plupart des interfaces radio, Bluetooth permet de combiner la correction d'erreurs par redondance (FEC : Forward Error Correction) et la correction par un mécanisme protocolaire par répétition (ARQ : Automatic Repeat reQuest).

4.5.1.1 Mécanismes FEC

Deux types de FEC sont possibles :

— code à répétition de taux 1/3 où chaque bit est répété 3 fois (ce codage est peu efficace mais présente l'avantage d'être extrêmement simple) ;

Toute reproduction sans autorisation du Centre français d'exploitation du droit de copie est strictement interdite. – © Editions T.I.

Dossier délivré pour Madame, Monsieur 08/09/2008

Tableau 3 – Les différents types de paquets de la bande de base

Types	Lien	Nom	Signification	Slot	FEC (sur données)	CRC (sur données)	Caractéristiques
Contrôle		ID	Identify	1/2	sans objet	sans objet	Utilisé pour l'*inquiry* et le *paging* Code d'accès uniquement
		NULL	Null	1	sans objet	sans objet	Code d'accès du canal et un en-tête, sans acquittement, pour envoyer de l'information sur le lien
		POLL	Poll	1	sans objet	sans objet	Idem NULL, mais avec acquittement, envoyé par le maître à ses esclaves pour tester les liens
		FHS	Frequency Hopping Synchronisation	1	2/3	oui	Adresse du maître (donc séquence de sauts) et horloge du maître, pour la synchronisation du piconet, avec FEC et CRC
Voix	SCO	HV	High Quality Voice	1	oui	non	Voix avec correction d'erreur 1/3 ou 2/3 FEC sans CRC
Données/voix	SCO	DV	Data Voice	1	2/3 pour données	oui pour données	Voix, données avec correction 2/3 FEC et CRC
Données	ACL	DM	Data Medium	1, 3, 5	2/3	oui	Données avec CRC et correction 2/3 FEC
		DH	Data High	1, 3, 5	non	oui	Idem DM, sans correction d'erreur
		AUX	Auxiliaire	1	non	non	Idem DH, sans CRC

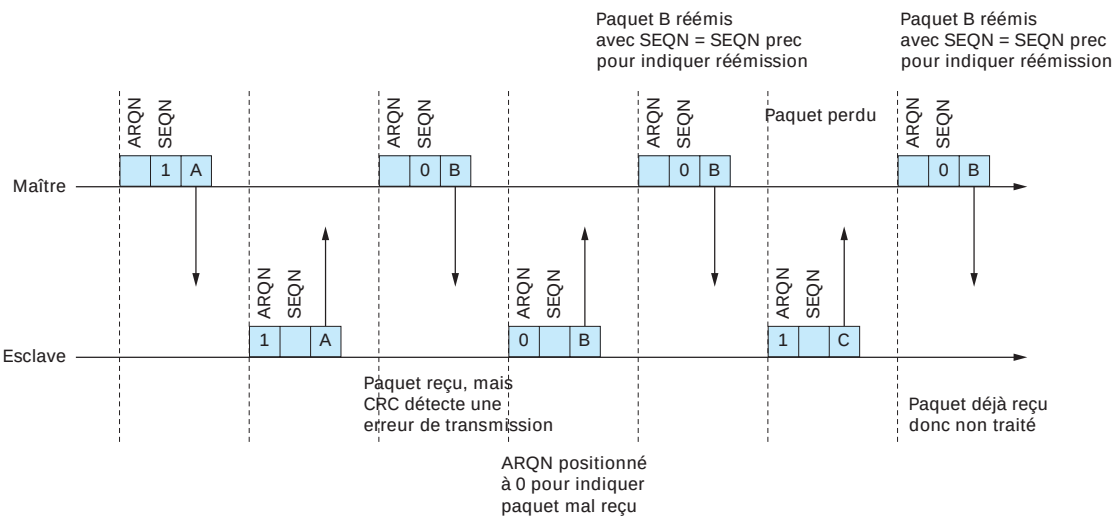


Figure 11 – Utilisation des bits ARQN et SEQN

— code de Hamming de taux 2/3 ; on utilise un code de Hamming court (15, 10). Ce code peut corriger toutes les erreurs simples et détecter toutes les erreurs doubles.

Le code à répétition est utilisé pour l'en-tête des paquets et pour les paquets de type HV1 (High Quality Voice) (voir tableau 5). Le code de Hamming est utilisé pour la transmission en mode protégé pour les services synchrones de voix (paquets DV : Data Voice) et pour les services de données ACL (paquets DM : Data Medium). Il est possible de n'utiliser aucun code FEC (on a alors des paquets DH : Data High).

4.5.1.2 Mécanisme ARQ

Dans le mécanisme ARQ, les paquets sont retransmis jusqu'à ce qu'un acquittement soit reçu, ou que le quantum de temps soit dépassé. Les paquets sont numérotés sur un bit grâce au champ SEQN. En cas de répétition d'un même paquet, le champ SEQN n'est pas modifié. En cas d'émission d'un nouveau paquet, la valeur de SEQN est modifiée. Dans le sens de transmission opposé, le champ ARQN indique un acquittement positif ou négatif. Un paquet de retour perdu est considéré comme un acquittement négatif (figure 11).

Toute reproduction sans autorisation du Centre français d'exploitation du droit de copie est strictement interdite. – © Editions T.I.

Dossier délivré pour
Madame, Monsieur
08/09/2008

Tableau 4 – Principaux paquets et débits disponibles sur les liens ACL

Type de paquet	Nombre de slots occupés	Taille des données utilisateurs sans l'en-tête ACL et le CRC (octets)	Débit symétrique max. (kbit/s)	Débit asymétrique max. (kbit/s)	
DM1	1	0 à 17	108,8	108,8	108,8
DH1		0 à 27	172,8	172,8	172,8
2-DH1		0 à 54	345,6	345,6	345,6
3-DH1		0 à 83	531,2	531,2	531,2
DM3	3	0 à 121	258,1	387,2	54,4
DH3		0 à 183	390,4	585,6	86,4
2-DH3		0 à 367	782,9	1 174,4	172,8
3-DH3		0 à 552	1 177,6	1 766,4	235,6
DM5	5	0 à 224	286,7	477,8	36,3
DH5		0 à 339	433,9	723,2	57,6
2-DH5		0 à 679	869,7	1 448,5	115,2
3-DH5		0 à 1 021	1 306,9	2 178,1	177,1
AUX	1	0 à 29	185,6	185,6	185,6

Tableau 5 – Paquets et débits disponibles sur le lien SCO

Type de paquet	Nombre de slots occupés	Taille des données utilisateurs (octets)	FEC	CRC	Débit symétrique max. (kbit/s)	Version Bluetooth
HV1	1	10	1/3	Non	64	1
HV2	1	20	2/3	Non	64	1
HV3	1	30	Non	Non	64	1
DV	1	10 (voix) 0 à 9 (données)	2/3 pour données	pour données	64 + 57,6 (données)	1
EV3	1	1 à 30	Non	Oui	96	2
EV4		1 à 120	2/3	Oui	192	2
EV5		1 à 180	Non	Oui	288	2
2-EV3		1 à 60	Non	Oui	192	2
2-EV5		1 à 360	Non	Oui	576	2
3-EV3		1 à 90	Non	Oui	288	2
3-EV5		1 à 540	Non	Oui	864	2

4.6 Exemples de débits

L'utilisation des différents modes de transmission (protégé ou non, sur un ou plusieurs slots) permet une grande variété de débits. Dans les tableaux 4 et 5, nous indiquons les débits correspondant aux paquets sur les liens ACL et SCO. Le chiffre avant le nom du paquet indique le mode : aucun chiffre pour le mode normal, 2 lorsque la 4-DPSK est utilisé (2 bits par symboles) et 3 lorsque le 8-DPSK est utilisée (3 bits par symbole).

Exemple 1 : en utilisant des paquets de type DM1 (sur 1 slot) (figure 12), on peut transmettre 17 octets de données (cf. tableau 4) par paquet. On dispose donc d'un débit (17×8) bits/$(2 \times 625\ \mu s)$ = 108,8 kbit/s dans un sens de transmission et dans de bonnes conditions.

Exemple 2 : en utilisant des paquets sans protection FEC de type DH5 (sur 5 slots) (figure 13), on peut transmettre 339 octets de données (cf. tableau 4) par paquets, soit 339 octets tous les 6 slots. Le débit est de (339×8) bits/$(6 \times 625\ \mu s)$ = 723,2 kbit/s dans un sens de transmission. On dispose seulement d'un slot dans le sens opposé tous les 6 slots. On a donc (27×8) bits/$(6 \times 625\ \mu s)$ = 57,6 kbit/s dans le sens retour.

Toute reproduction sans autorisation du Centre français d'exploitation du droit de copie est strictement interdite. – © Editions T.I.

Dossier délivré pour Madame, Monsieur 08/09/2008

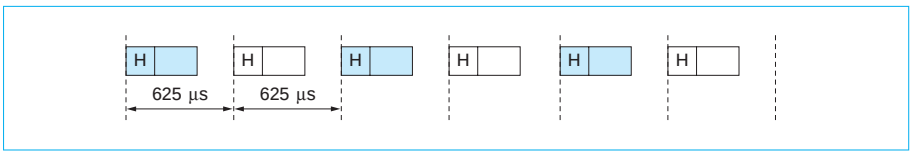


Figure 12 – Exemple de transmission ACL avec des paquets DM1

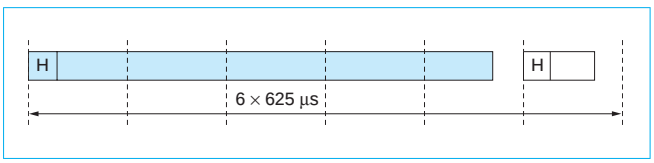


Figure 13 – Exemple de transmission ACL avec des paquets DH5

5. Mécanismes de découverte et d'établissement de lien

La couche LC abrite les processus fondamentaux permettant l'établissement du piconet. Celui-ci s'effectue en trois phases : *inquiry*, *paging*, connexion.

5.1 Établissement d'un piconet

Par défaut, une station mise sous tension est en veille (*standby state*). Lorsqu'une station se connecte à un piconet, elle quitte cet état pour parvenir à l'état connecté (*connection state*). Pour l'aider à atteindre cet état, deux familles de procédures ont été définies : d'une part, les procédures d'*inquiry* utilisées pour permettre à une station Bluetooth de découvrir son environnement, c'est-à-dire les autres stations sous sa portée ; d'autre part, les procédures de *page* destinées à établir la connexion entre deux stations qui sont connues l'une de l'autre. L'ensemble des activités réalisables par une station en veille est représenté sur la figure 14.

5.1.1 Procédures d'*inquiry*

Le mécanisme de découverte est composé de deux mécanismes complémentaires : l'*inquiry* et l'*inquiry scan*.

Une station procédant à un *inquiry* envoie sur un canal radio une rafale de paquets ID de courte durée (68 µs) pour inviter les stations à l'écoute à envoyer leurs informations personnelles. Ce paquet diffusé ne contient pas d'information sur la station émettrice. Il indique juste quelles sont les classes de station recherchées. Le code d'accès du paquet ID peut être de type GIAC, recherche générale de toutes les stations, ou DIAC, recherche d'un type de station (par exemple une oreillette).

Les stations ne sont pas encore synchronisées et pour augmenter les chances de succès, seulement 32 fréquences d'émission sur les 79 sont utilisées. La séquence de saut de fréquences est déterminée à partir de la partie LAP (Low Address Part) du GIAC ; elle est donc la même pour toutes les stations. Les 32 fréquences sont divisées en deux ensembles de 16 canaux. Une station émet 16 paquets ID de la façon suivante : elle émet deux paquets ID sur deux fréquences parmi un ensemble de 16 puis écoute pendant deux demi-slots sur deux fréquences pour détecter d'éventuelles réponses. Ce processus est répété 8 fois et constitue un « train » d'une durée de $8 \times 2 \times 2 \times 0{,}312\,5 = 10$ ms. La station répète 256 fois un train puis recommence l'émission de 256 trains constitués avec 16 autres fréquences du groupe de 32. Les deux trains sont appelés train A et train B.

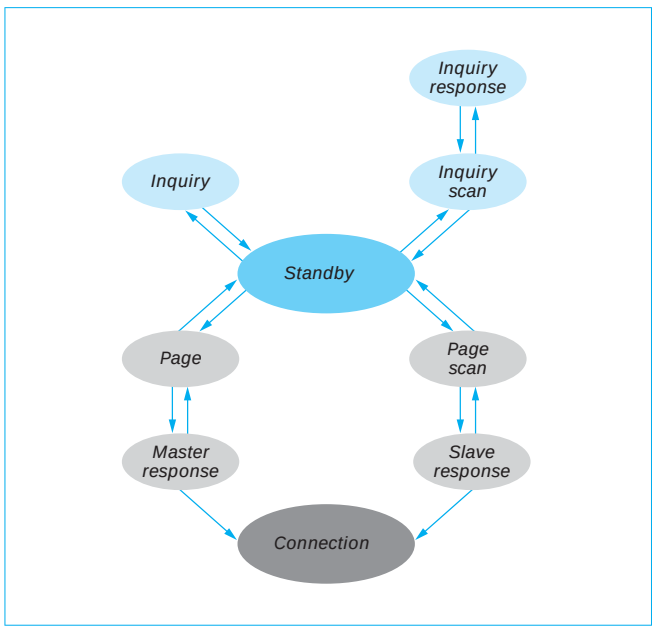


Figure 14 – Diagramme d'états à partir de l'état de veille

Une station procédant à un *inquiry scan* balaye les mêmes fréquences mais beaucoup plus lentement, toutes les 1,28 s. Elle reste 11,25 ms sur la même fréquence. Cela permet de recevoir un paquet ID parmi le groupe de 16 envoyés, s'il y a coïncidence des trains. Sinon, il faut attendre le train suivant.

Une fois que la rencontre a eu lieu, la station procédant à l'*inquiry scan* repasse dans l'état *standby* ou *connection* pendant une durée aléatoire (entre 0 et 1 023 slots) pour éviter les collisions avec les autres stations qui répondent. À la fin de cette période, elle passe dans l'état d'*inquiry response*. Après la réception à nouveau d'un paquet ID du maître, elle envoie après une durée d'un slot, ses informations personnelles à la station procédant à l'*inquiry*, sur une fréquence associée à la fréquence de réception de l'ID. Ces informations comprennent l'adresse BD_ADDR, la classe de la station et l'horloge interne. Elles sont envoyées dans un paquet FHS (Frequency Hopping Synchronization).

À la fin de la procédure d'*inquiry* (figure 15), la station maître connaît, grâce à la réception du paquet FHS :

— l'horloge de l'esclave ;
— l'adresse BD_ADDR de l'esclave.

La station esclave n'a pas d'information sur le maître, elle n'a reçu que des paquets ID.

Toute station doit faire des *inquiries* périodiques pour découvrir son environnement ou mettre à jour ses informations. La durée recommandée est de 10,24 s toutes les 60 s. L'*inquiry* peut s'arrêter prématurément, si la station considère qu'elle a obtenu assez de réponses ou si elle veut réaliser une connexion (*paging*) avec une station qu'elle vient de découvrir.

Toute reproduction sans autorisation du Centre français d'exploitation du droit de copie est strictement interdite. – © Editions T.I.

Dossier délivré pour
Madame, Monsieur
08/09/2008

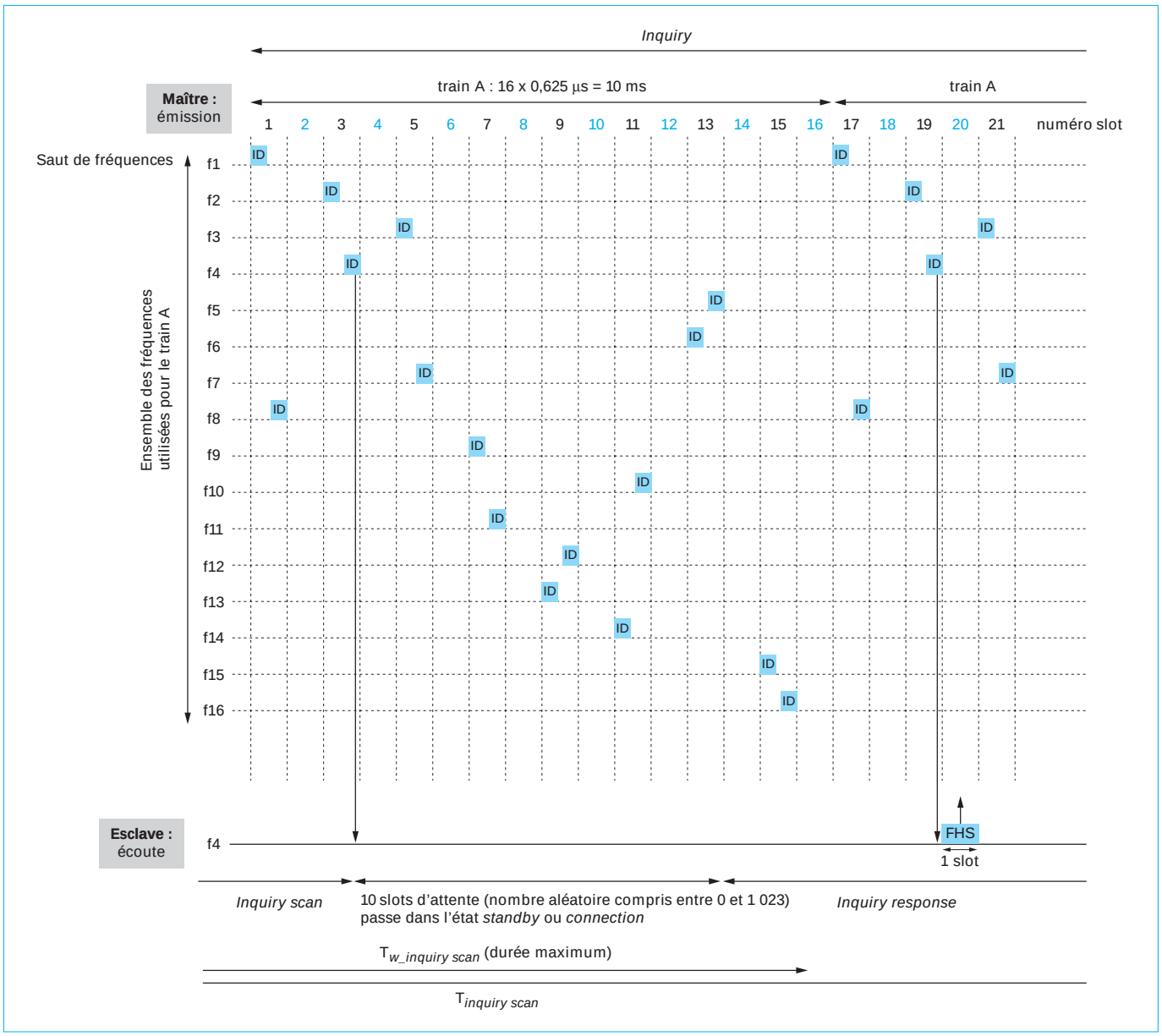


Figure 15 – Exemple d'*inquiry*

Une station peut être en mode découvrable ou non découvrable. Dans le premier cas, elle doit faire des *inquiries scan* périodiques pour se faire connaître des autres stations. Dans le second cas, elle ne procède à aucun *inquiry scan*. Un équipement peut être par défaut en mode non découvrable et passer en mode découvrable sur action de l'utilisateur.

5.1.2 Procédures de *page*

Les procédures de *page* et de *page scan* permettent l'établissement d'une connexion entre deux stations. Elles fonctionnent sur le même principe que les procédures d'*inquiry*. En particulier, un sous-ensemble de 32 fréquences radio est utilisé, mais il est distinct des 32 fréquences des procédures d'*inquiry*.

La station qui initie l'appel est celle qui a réalisé l'*inquiry*. La station appelée est celle qui a fait un *inquiry scan*. La station appelante connaît l'horloge et l'adresse de la station qu'elle cherche à connecter (paquet FHS reçu lors d'un *inquiry* précédent). Pour accélérer la procédure de connexion, la station va exploiter ces informations, en se basant sur l'horloge du récepteur. En particulier, elle va tenter de prédire le canal sur lequel le récepteur sera à l'écoute lors de la transmission de la demande de connexion. Ainsi, les paquets de demande de connexion seront envoyés en premier lieu sur le canal radio prédit. Le paquet *paging* est un paquet ID qui contient un mot de synchronisation de type code d'accès DAC calculé à partir de l'adresse BD_ADDR de la station distante, obtenue lors d'un *inquiry* précédent.

Toute reproduction sans autorisation du Centre français d'exploitation du droit de copie est strictement interdite. – © EditionsT.I.

Dossier délivré pour Madame, Monsieur 08/09/2008

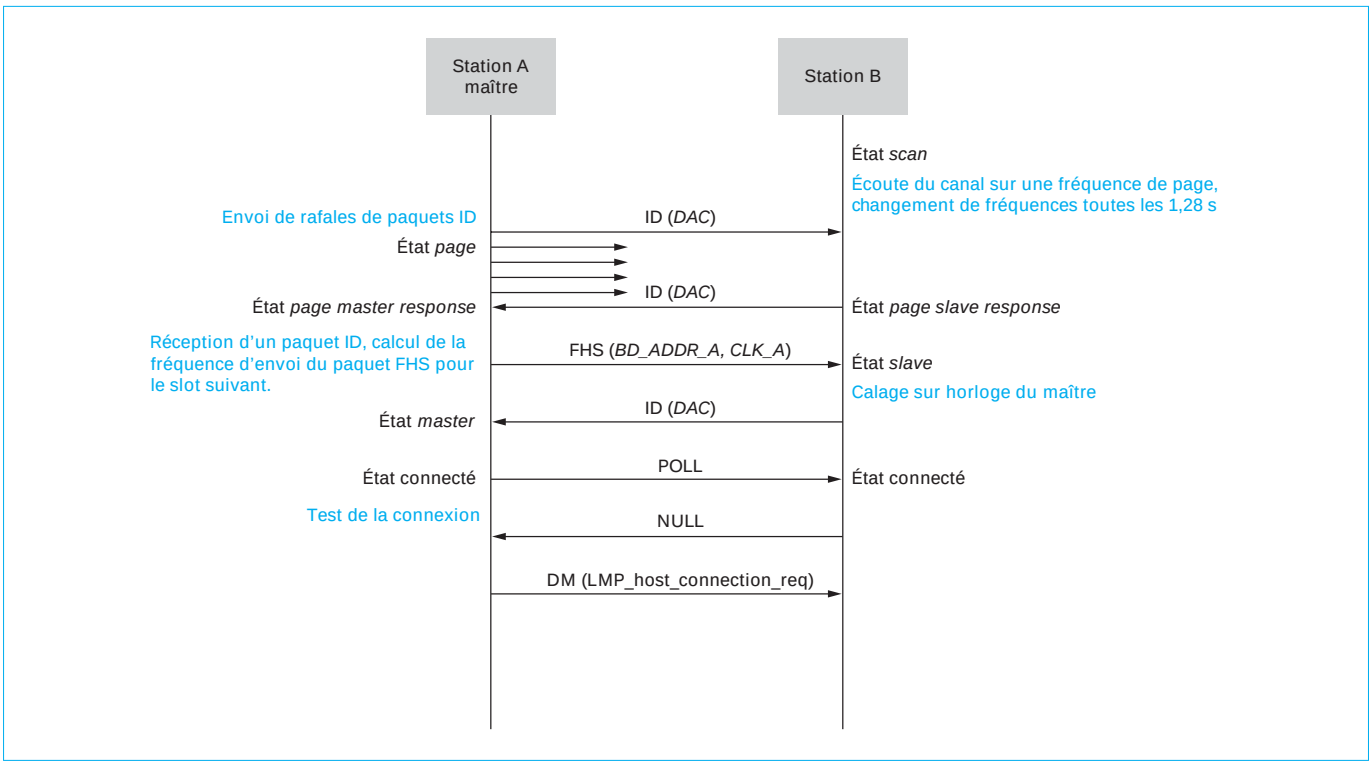


Figure 16 – MSG procédure de *paging*

Si la station n'effectue pas un *page* directement après l'*inquiry*, ses prédictions risquent d'être moins bonnes et les temps de connexion plus longs, parce qu'il y aura un décalage entre les deux horloges (pour réduire les coûts, il n'y a pas de forte exigence sur la précision des horloges des stations Bluetooth). Elle utilise également, comme pour l'*inquiry*, deux trains A et B de 16 fréquences chacun, qui, eux, sont propres à chaque station à connecter.

Une station, qui est en mode *page scan*, recherche sur une fréquence un paquet ID avec son propre DAC pendant 1,28 s puis passe à la fréquence suivante. Les fréquences de *scan* sont sélectionnées selon une séquence de saut propre à la station et fonction de son horloge. Quand elle reçoit un paquet ID, la station ID répond par un paquet ID un slot plus tard. Si elle ne reçoit pas de réponse du maître, elle envoie alors un nouveau paquet ID, jusqu'à obtenir une réponse du maître.

À l'issue de l'échange de paquets ID, l'appelant transmet ses informations personnelles dans un paquet FHS : la station appelée se synchronise alors sur la station appelante et renvoie un paquet ID pour confirmer la bonne réception du FHS. Elle quitte l'état de veille pour passer à l'état connecté. Elle est alors intégrée, en tant qu'esclave, au sein du piconet dont le maître n'est autre que l'initiateur de la procédure de *page* (figure 16).

5.2 État connecté et ses différents modes

Aussitôt connectée, une station peut intervenir dans le piconet. Elle suit les règles d'accès et de transmission qui ont été établies. En particulier, elle doit écouter le lien ACL tous les deux slots même si elle n'a pas à intervenir.

Afin d'optimiser les ressources énergétiques, des modes adaptés à ce type de situation ont été définis, libérant la station des contraintes d'écoute régulière du canal.

Dans l'ordre décroissant de consommation d'énergie, nous avons : le mode actif, le mode *sniff*, le mode *hold* et finalement le mode *park*.

Les changements d'état sont effectués sur commande du gestionnaire de liens (LM).

5.2.1 Mode actif

Le mode actif est le mode par défaut où une station doit impérativement écouter toutes les trames transmises par le maître sur le lien ACL. Elle doit bien évidemment répondre aux obligations des liens SCO si elle en est pourvue.

5.2.2 Mode *sniff*

Le mode *sniff* permet à une station de n'écouter le canal qu'à des intervalles prédéfinis et périodiques, pendant un nombre de slots donnés, ou jusqu'à échéance d'un *timer*. Il s'agit d'une opération répétitive adaptée pour les événements périodiques. La période dépend de l'application.

5.2.3 Mode *hold*

Le mode *hold* permet à une station de se libérer de toutes ses obligations au sein du piconet pour une période donnée, déterminée entre le maître et l'esclave. Elle peut néanmoins conserver une liaison SCO. Ce mode peut être activé à la demande de l'esclave ou du maître. Cette période de liberté est utilisée pour procéder à des opérations d'*inquiry* ou de *page*. Les paramètres peuvent être négociés entre les deux parties.

Le mode *hold* est typiquement utilisé dans le cas de connexions avec plusieurs piconets ou quand l'échange de données n'est pas fréquent.

Toute reproduction sans autorisation du Centre français d'exploitation du droit de copie est strictement interdite. – © Editions T.I.

Dossier délivré pour
Madame, Monsieur
08/09/2008

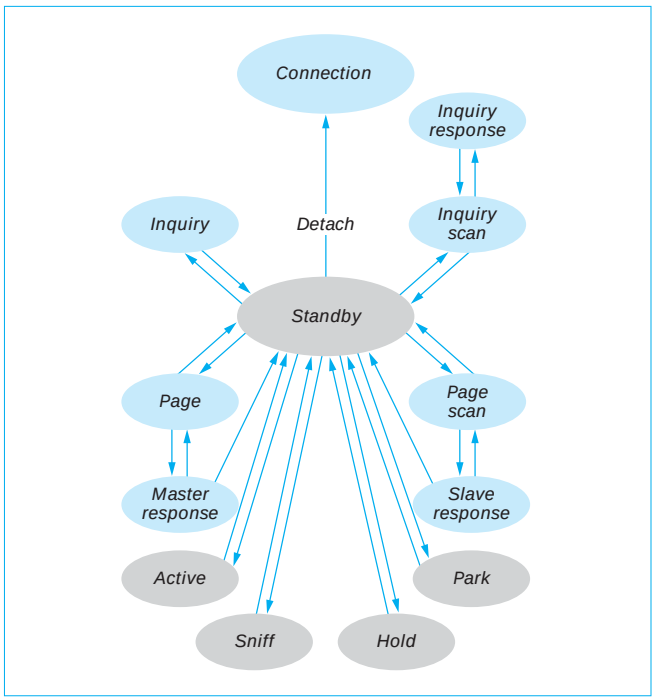


Figure 17 – État connecté

5.2.4 Mode *park*

Le mode *park* est utilisé pour les stations n'ayant pas à intervenir dans l'immédiat au sein du piconet, mais qui ont besoin de rester synchronisées. Au maximum 255 stations peuvent ainsi rester synchronisées au maître sans toutefois pouvoir intervenir au sein du piconet. Ce mode a l'avantage de permettre une réintégration rapide des stations. Il permet également de lever partiellement la limitation à 7 esclaves par piconet. En effet, un esclave en mode *park* libère son adresse AM_ADDR. Le maître lui assigne une adresse de parking PM_ADDR et une adresse AR_ADDR, qu'il utilisera pour faire sortir cet esclave du parking.

La station parquée se réveille à intervalles réguliers pour écouter le canal pour se resynchroniser avec le maître et surveiller les messages de diffusion. Le maître établit un canal pour l'envoi de balises pour les esclaves parqués. Les balises sont envoyées régulièrement.

Des fenêtres d'accès sont définies, pour que les esclaves puissent envoyer des requêtes au maître pour repasser dans l'état actif. Les slots des fenêtres d'accès peuvent être également utilisés par le trafic applicatif du piconet.

5.3 Diagramme d'état d'une station Bluetooth

La figure 17 illustre les différents états et les différents modes accessibles depuis l'état connecté par une station Bluetooth.

Une fois connectée, une station peut continuer à effectuer les opérations de *page* et d'*inquiry*. L'opération d'*inquiry* permet à une station déjà connectée de rafraîchir les informations concernant son environnement. Elle doit également effectuer périodiquement des *inquiries scan*, pour fournir aux autres stations les informations la concernant.

Les procédures de *page* ou *page scan* sont également accessibles aux stations connectées. Elles permettent la création de scatternet. Lorsque le maître actuel d'un piconet répond à un *page* (en effectuant un *page scan*), il devient esclave dans un nouveau piconet. Lorsque le maître actuel du piconet effectue un *page* avec succès, il ajoute un esclave à son piconet. De même, lorsqu'un esclave répond à un *page*, il devient esclave dans deux piconets à la fois. Lorsqu'un esclave effectue un *page* avec succès, il reste esclave dans son piconet d'origine et devient maître dans le piconet nouvellement créé.

La procédure *detach* permet à tout moment à une station de quitter l'état connecté. Concrètement, la station quitte le piconet dans lequel elle a effectué cette procédure.

5.4 Connexion d'une station à un piconet existant

Quand une station initie une connexion, elle devient le maître du piconet ainsi créé. Si une station A veut se connecter à une station B, déjà maître d'un piconet, il y aura deux piconets : le piconet de B et le piconet nouvellement créé entre A et B, A étant le maître. Dans cette situation, le piconet initial de B est indisponible quand B communique avec A. B peut alors demander le changement de rôle, de manière à ce qu'il n'y ait plus qu'un seul piconet ayant B comme maître, A devenant esclave de B.

Quand il y a un seul piconet, c'est souvent l'ordinateur qui est choisi comme maître, le clavier, le casque, le téléphone, etc., étant ses esclaves. Dans le cas de réseaux Bluetooth plus complexes, faisant intervenir plusieurs piconets, le choix des maîtres peut être plus délicat. Tous les équipements Bluetooth ne disposent pas non plus de la capacité à être maître d'un piconet, comme par exemple un clavier ou une oreillette.

5.5 Scatternet

Plusieurs piconets peuvent se connecter pour former un scatternet. La formation de ce scatternet est un problème complexe pour avoir un réseau performant.

Par exemple, si le maître d'un premier réseau devient actif, en tant qu'esclave, dans un deuxième réseau, tous ses esclaves du premier réseau sont, pendant ce temps, bloqués, et le débit diminue.

Un esclave et un maître ne peuvent communiquer qu'après avoir négocié une « période *sniff* ». Si l'esclave est alors en communication dans un autre piconet, le maître termine la connexion. Il faut donc que ces périodes soient différentes et qu'elles ne se recouvrent pas, quand une station appartient à plusieurs piconets.

Les stations décident de leur rôle dans les piconets en fonction des informations qu'elles disposent en local. Il faut des algorithmes efficaces pour que ces choix soient judicieux, et qu'elles puissent changer de rôle, si cela est nécessaire.

Il existe plusieurs algorithmes pour la formation des scatternets, BlueTrees, Bluenet, Bluestars, BlueMesh, TSF, etc. C'est un sujet de recherche important [11] [12]. Bluetooth et la norme SIG ne proposent pas d'algorithme spécifique.

Un réseau Bluetooth est prévu pour avoir au maximum 10 piconets interconnectés, soit 72 stations actives. Tout matériel Bluetooth n'a pas la possibilité technique d'appartenir à plusieurs piconets, c'est précisé dans les caractéristiques techniques du matériel.

6. Couche gestion des liens

Le gestionnaire de liens (LM) traduit les commandes reçues du HCI en opérations pour la bande de base. Il gère le piconet (ajout d'esclave, contrôle de la consommation d'énergie), établit les liens

Toute reproduction sans autorisation du Centre français d'exploitation du droit de copie est strictement interdite. – © Editions T.I.

Dossier délivré pour Madame, Monsieur 08/09/2008

SCO et ACL, et assure des fonctions de sécurité. Le gestionnaire de liens communique avec les autres gestionnaires sur les autres unités Bluetooth à travers le protocole LMP (Link Management Protocol). Les PDU LMP sont transmis dans des paquets DM1 ou DV. Le contrôleur de liens doit avoir au préalable établi un lien entre les deux unités par une phase de *paging*.

Le LMP assure également d'autres gestions :

— gestion de la puissance d'émission des stations ;
— gestion du piconet et des passages dans les différents modes d'économie de batterie ;
— gestion des connexions, changement de rôle maître/esclave, paramétrage, QoS, etc.

7. L2CAP

Le protocole d'adaptation et de contrôle de lien logique (L2CAP) est le protocole minimal d'échange de données de la spécification Bluetooth. C'est en utilisant L2CAP que sont implémentées les plus hautes couches du protocole Bluetooth, comme SDP ou RFCOMM.

L2CAP permet :

— la segmentation et le réassemblage des paquets. Les paquets de la bande de base ont une taille limitée et L2CAP doit segmenter les paquets des couches supérieures pour respecter cette taille ; à la réception, L2CAP doit réassembler les paquets ;
— le multiplexage. L2CAP distingue les paquets des différents protocoles des couches supérieures comme SDP ou TCS ;
— la qualité de service. Lors de l'établissement d'une connexion L2CAP, les deux stations doivent négocier la QoS ;
— le concept de groupe. Il existe des protocoles qui utilisent la notion de groupe, L2CAP doit implémenter la notion de groupe en permettant la diffusion de paquets.

Il existe trois types de canaux L2CAP :

— bidirectionnel pour la signalisation : établissement, gestion et déconnexion d'un canal L2CAP ;
— orienté connexion, pour une connexion ACL point-à-point ;
— sans connexion, pour la diffusion de paquets pour implémenter la notion de groupes.

L2CAP ne gère pas les connexions voix SCO. Celles-ci sont gérées directement par la bande de base.

8. Mécanismes de sécurité

8.1 Modes de sécurité

Bluetooth définit trois modes de sécurité optionnels pour les stations :

— 1 : pas de sécurité. Une station, dans ce mode, n'initie pas de procédure de sécurité ; elle peut supporter ou non l'authentification. C'est le mode de sécurité par défaut ;
— 2 : sécurité au niveau application. Une fois la connexion L2CAP établie, la station décide des mécanismes de sécurité à utiliser ;
— 3 : sécurité au niveau de la connexion réalisée par le gestionnaire de liens par échange de messages LMP.

Il y a deux niveaux de sécurité pour les appareils : fiable ou non fiable, et trois niveaux pour les services : autorisé et authentifié, authentifié, accès libre.

Bluetooth permet l'authentification, l'autorisation et le chiffrement des données. Il ne fournit pas de vérification de l'intégrité des données.

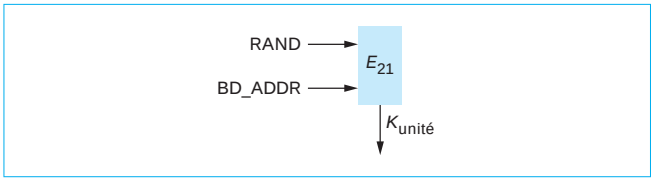


Figure 18 – Génération de la clé unité

8.2 Gestion des clés

8.2.1 Clé unité

Chaque entité a sa propre clé unité, qu'elle génère une seule fois : la première fois qu'elle est allumée. Cette clé est stockée en mémoire. Elle peut être régénérée ultérieurement si, par exemple, la station change de propriétaire.

La clé unité est calculée avec l'algorithme E_{21}, qui prend en entrée l'adresse BD_ADDR de l'entité et un nombre aléatoire (figure 18).

8.2.2 Clé d'initialisation et clé maître

La clé d'initialisation est générée la première fois que deux stations veulent communiquer (figure 19). Cette clé est commune à deux stations. Elle impose qu'un même code PIN (Personal Identification Number) soit entré dans les deux stations, par exemple directement par l'utilisateur.

La station A génère un nombre aléatoire qu'elle transmet en clair à B. Les deux stations A et B génèrent la clé d'initialisation K_{init} avec l'algorithme E_{22}, qui prend comme paramètres d'entrée, ce nombre aléatoire, le code PIN et la longueur du code PIN.

La procédure de génération de la clé d'initialisation s'appelle la phase de couplage ou *pairing*. Cette phase dite de couplage n'est pas nécessaire à chaque nouvelle connexion. Elle intervient seulement lors d'une première connexion entre deux entités, si la sécurité est au niveau de la connexion. Par la suite, elle peut être relancée, lorsqu'une des deux stations ne dispose plus en mémoire d'une clé de liaison pour la station distante.

La clé maître est générée de la même façon qu'une clé d'initialisation. Elle sert lorsqu'une station maître veut faire du point-à-multipoint. Le nombre aléatoire est transmis à tous les esclaves et il est nécessaire qu'ils aient tous le même code PIN.

8.2.3 Clé combinée

La clé combinée est utilisée entre deux stations si les deux parties l'ont décidé. Elle est générée en parallèle par les deux stations avec l'algorithme E_{21}, un nombre aléatoire et les adresses des deux stations (figure 20). En effet, chaque station connaît la BD_ADDR de l'autre station, car elle a obtenu l'information lors de la phase de *paging* nécessaire avant toute procédure de sécurité.

La procédure suppose que l'on dispose déjà d'une clé commune aux deux stations (par exemple, la clé d'initialisation K_{init}). Les nombres aléatoires sont échangés d'une manière sécurisée en utilisant un XOR avec la clé commune.

8.2.4 Clé de liaison

Par définition, la clé de liaison (*link key*) est une clé commune aux deux stations. Elle est utilisée pour calculer la clé de chiffrement et pour la procédure d'authentification. Le type de clé de liaison générée dépend de l'application et des capacités des stations. La clé de liaison est soit une clé unité, soit une clé combinée. Si une station a une mémoire limitée, elle utilisera une

Toute reproduction sans autorisation du Centre français d'exploitation du droit de copie est strictement interdite. – © Editions T.I.

Dossier délivré pour
Madame, Monsieur
08/09/2008

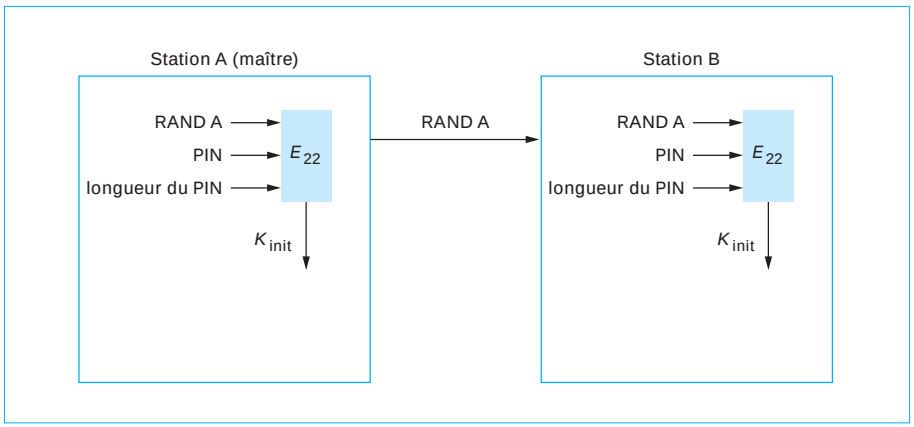


Figure 19 – Génération de la clé d'initialisation

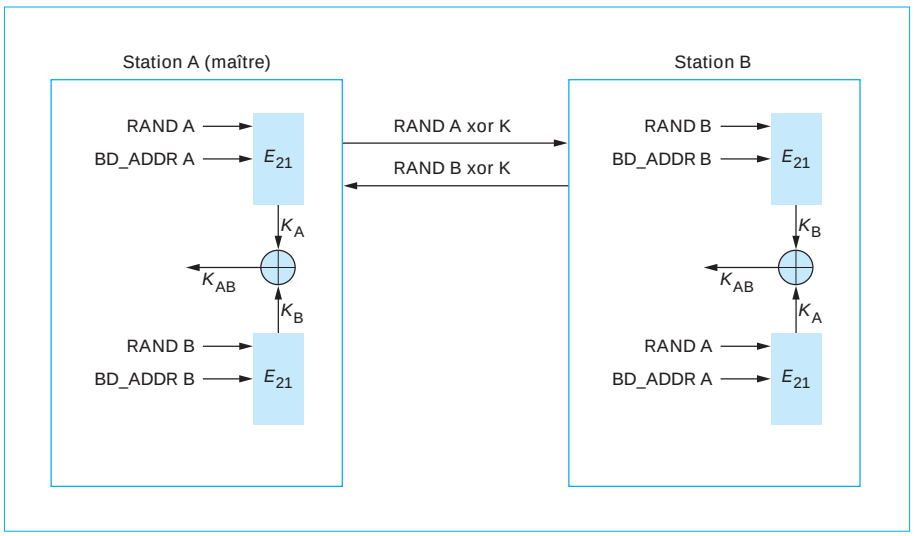


Figure 20 – Génération de la clé combinée

clé unité comme clé de liaison avec toutes les autres stations, plutôt que de mémoriser une clé combinée par liaison. Cependant, la clé étant la même pour toutes les stations, toute station peut écouter les communications des autres stations. Si une sécurisation plus importante est nécessaire, la clé de liaison utilisée est une clé combinée.

Dans le cas d'une liaison point-à-multipoint, la clé de liaison peut être une clé appelée clé maître. Cette dernière est générée à partir d'un code PIN comme la clé d'initialisation ; le code PIN est commun au maître et à l'ensemble des esclaves.

8.3 Authentification

L'authentification est le processus de vérification de l'entité Bluetooth distante. Lorsque par exemple l'équipement A souhaite authentifier B, le mécanisme suivant est utilisé (figure 21) :

— tout d'abord A envoie à B un nombre aléatoire en clair (RAND A) ;

— ensuite A et B calculent grâce à la fonction d'authentification E_1 un résultat XRES pour A (eXpected Result) et SRES pour B (Signed Result). La fonction E_1 prend en paramètres le nombre aléatoire RAND A, transmis par A à B, la BD-ADDR de B (connue de A puisqu'une liaison est établie avec B) ainsi que la clé de liaison (*link key*) ;

— une fois le calcul effectué de part et d'autre, B envoie son résultat SRES à A ;

— A peut alors vérifier si B possède ou non la clé de liaison en comparant les valeurs de SRES avec XRES. Si les deux valeurs sont égales, cela signifie que B possède la clé de liaison et l'authentification a réussi, sinon c'est un échec, B ne dispose pas de la clé de liaison et les stations ne peuvent pas s'authentifier.

En plus du résultat servant à authentifier B, la fonction E_1 produit un nombre appelé ACO (Authenticated Ciphering Offset). Ce nombre peut, par exemple, servir de base pour les calculs de chiffrement/déchiffrement qui ont généralement lieu après une authentification.

C'est l'application qui indique qui devra être authentifié. Par conséquent, le vérifieur n'est pas toujours le maître. Il peut y avoir également des authentifications mutuelles.

Un délai est introduit si l'authentification échoue avant qu'on ne puisse la retenter. Il double après chaque échec de la même adresse jusqu'à ce que le maximum, fixé par l'application, soit atteint.

Toute reproduction sans autorisation du Centre français d'exploitation du droit de copie est strictement interdite. – © Editions T.I.

Dossier délivré pour Madame, Monsieur 08/09/2008

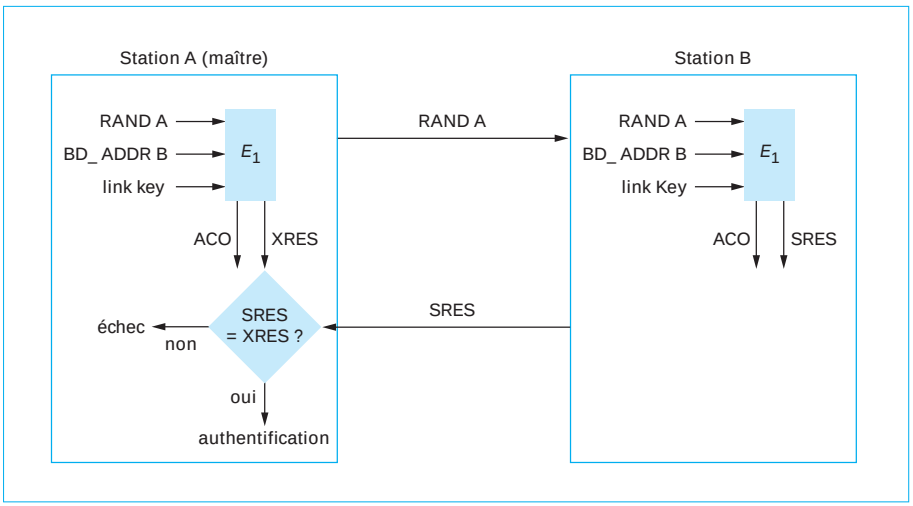


Figure 21 – Authentification d'une station esclave par le maître

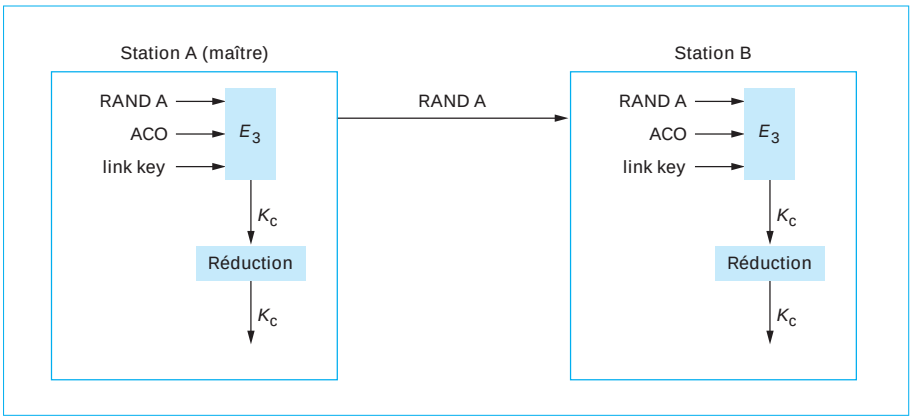


Figure 22 – Génération de la clé de chiffrement

8.4 Chiffrement

La procédure de chiffrement s'effectue suivant trois phases :

— négociation du mode de chiffrement, pas de chiffrement, chiffrement des paquets point-à-point, chiffrement des paquets point-à-point et point-à-multipoint ;

— négociation de la taille de la clé ;

— chiffrement.

La clé de chiffrement (K_c) est obtenue avec l'algorithme E_3 à partir de la clé de liaison, d'un nombre aléatoire généré par la station maître et transmis en clair, et du nombre ACO calculé lors de l'authentification (figure 22). Le nombre de bits de la clé de chiffrement entre 8 et 128 bits doit être négocié entre les deux stations. C'est l'application qui détermine un minimum acceptable pour éviter de baisser le niveau de sécurité.

Le mécanisme de chiffrement/déchiffrement est basé sur l'algorithme E_0 qui produit une clé de chiffrement/déchiffrement (K_{str}) à partir de la clé K_c, de l'adresse du maître et de son horloge (figure 23). La valeur de l'horloge étant différente à chaque slot, la clé K_{str} produite change à chaque paquet transmis. Il suffit alors de faire un XOR (ou exclusif) entre les données du paquet en clair et la clé K_{str} pour avoir les données chiffrées. Le processus de déchiffrement est identique : on réapplique la clé K_{str} aux données chiffrées pour avoir les données en clair.

Toute reproduction sans autorisation du Centre français d'exploitation du droit de copie est strictement interdite. – © Editions T.I.

Dossier délivré pour
Madame, Monsieur
08/09/2008

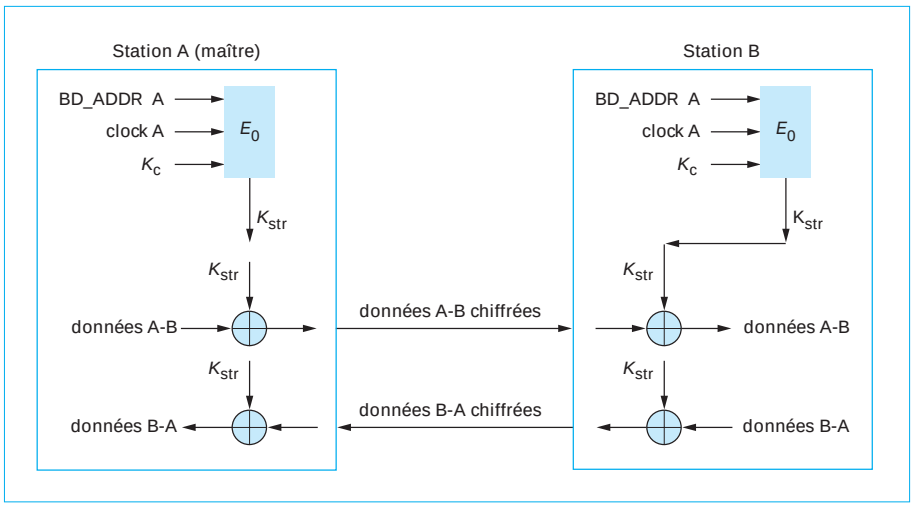


Figure 23 – Chiffrement et déchiffrement des données

Références bibliographiques

Normes

[1] *Specification of the Bluetooth System.* Bluetooth Special Interest Group.

[2] *Wireless Medium Access (MAC) and Physical Layer Specification (PHY) for Wireless Personal Access Network (WPAN).* IEEE 802.15.1 (2002).

[3] *Wireless Medium Access (MAC) and Physical Layer Specification (PHY) for Wireless Personal Access Network (WPAN).* IEEE 802.15.1 (2005).

[4] *Draft Bluetooth Core Specification v 2.1+EDR.* Bluetooth Special Interest Group. http://www.bluetooth.org/spec/

Ouvrages de base

[5] MILLER (B.A.) et BISDIKIAN (C.). – *Bluetooth Revealed : The insider's Guide to an Open Specification for Global Wireless Communications.* Prentice Hall (2000).

[6] HAARTSEN (J.C.). – *The Bluetooth Radio System.* IEEE Personal Communications, *7*, nº 1, 28-36 (2000).

[7] BRAY (J.) et STURMAN (C.F.). – *Bluetooth Connect Without Cables.* Prentice Hall (2000).

[8] HAARSTEN (J.C.). – *Bluetooth : A new radio interface providing ubiquitous connectivity.* VTC 2000 Spring, Tokyo (2000).

Articles

[9] HOLE (K.J.). – *Bluetooth – part 10 : Introduction to Wireless Security.* http://www.kjhole.com, avr. 2005.

[10] LINDBÄCK (R.) et SALOMON (A.). – *La sécurité de Bluetooth*. Présentation dans le cadre du cours « Security protocols and applications », École polytechnique fédérale de Lausanne, LASEC, avr. 2004.

[11] VERGETIS (E.), GUÉRIN (R.), SARKAR (S.) et RANK (J.). – *Can Bluetooth Succeed as a Large-Scale Ad Hoc Networking Technology ?* IEEE Journal on Selected Areas in Communications, *23*, nº 3 (2005).

[12] McDERMOTT-WELLS (P.). – *Bluetooth scatternet models.* IEEE Potentials, déc. 2004/janv. 2005.

[13] JIANG (J.), LIN (B.) et TSENG (Y.). – *Analysis of Bluetooth Device Discovery and some Speedup Mechanisms.* International Journal of Engineering Education, *11*, nº 4, 301-310 (2004).

Serveurs Web

[14] Palowireless, Bluetooth Resource Center. – http://www.palowireless.com/bluetooth

[15] Bluetooth SIG. – http://www.bluetooth.org

[16] Bluetooth SIG. – http://www.bluetooth.com

Réglementation

[17] Décision nº 03-909 de l'Autorité de régulation des télécommunications en date du 22 juillet 2003 portant modification de la décision nº 2002-1031 de l'Autorité de régulation des télécommunications en date du 7 novembre 2002 portant adoption des lignes directrices relatives à l'expérimentation de réseaux ouverts au public utilisant la technologie RLAN. http://www.art-telecom.fr/uploads/tx_gsavis/03-909.pdf

Toute reproduction sans autorisation du Centre français d'exploitation du droit de copie est strictement interdite. – © Editions T.I.

Dossier délivré pour Madame, Monsieur 08/09/2008