Android BLE4.0 应用

|  |  |  |
| --- | --- | --- |
| 序号 | 描述 | 姓名/时间 |
| 01 | Android BLE4.0技术应用 | 刘志保 / 2019-03-28 |
|  |  |  |
|  | https://github.com/MMLoveMeMM/AngryPandaBTBLE.git |  |

BLE4.0 相对于以前的蓝牙版本的,具有数据传输量大,传输距离更远(可以达到将近100米),低功耗.这些都奠定了它在短距离通信领域的地位.以及在物联网方面的支持.

第一 : GAP,GATT协议

<1> : GAP(**Generic Access Profile**) : GAP（Generic Access Profile）主要用来控制设备连接和广播，GAP 使你的设备被其他设备可见，并决定了你的设备是否可以或者怎样与交互设备进行通讯。GAP层有4种不同类型的广播：通用的、定向的、不可连接的以及可发现的。设备每次广播时，会在3个广播信道上发送相同的报文。这些报文被称为一个广播事件。除了定向报文以外，其他广播事件均可以选择20ms - 10.28s不等的间隔。通常，一个广播中的设备会每一秒广播一次，广播事件之间的时间称为广播间隔，主机可以控制该间隔。例如 Beacon 设备就只是向外广播，不支持连接，小米手环等设备就可以与中心设备连接。

大部分情况下，外设通过广播自己来让中心设备发现自己，并建立 GATT 连接，从而进行更多的数据交换。也有些情况是不需要连接的，只要外设广播自己的数据即可，用这种方式主要目的是让外围设备，把自己的信息发送给多个中心设备。使用广播这种方式最典型的应用就是苹果的 iBeacon.

GAP Advertising 简述:

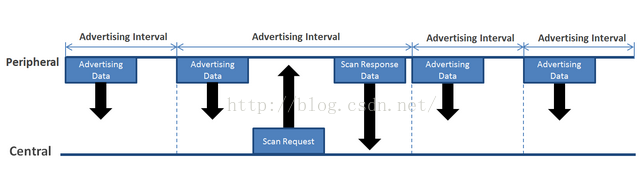
蓝牙技术中定义了GAP的4种角色：Broadcaster, Observer, Central, Peripheral(Section 6.2 Vol 1 Part A)和Linker Layer(LL)的5种状态：Standby, Advertising, Scanning, Initiating, Connection(Section 1.1 Vol 6 part B),一种设备可以同时拥有多种角色和多种状态

A、Advertising

有两种类型的数据包可以发送：广播数据包(Advertising packet) 和 扫描响应包(Scan Response packet),它们的最大数据量都为31字节。

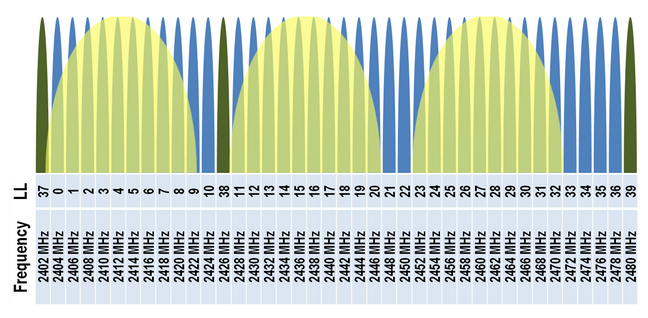
广播者在广播时间间隔内会一直广播数据，这个时间间隔是可以在广播的时候改变的，但也必须在最小时间间隔和最大时间间隔之间，正常情况下，不定向广播的时间间隔在20ms到10.24s之间(Section 7.8.5 Part E Vol2)。扫描响应包的发送是在收到扫描请求以后才发送。

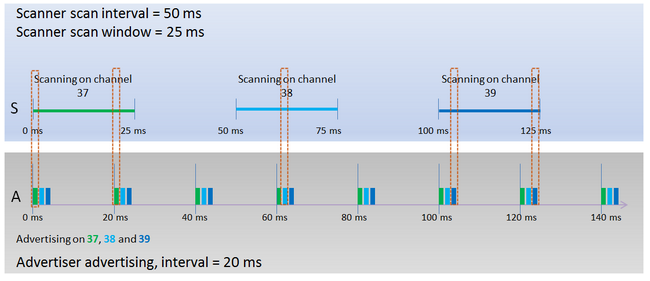
        下图描述了广播包和扫描响应包的发送过程



在一些应用场景中，我们可能不希望设备之间建立连接，那我们可以广播无连接广播数据，这样就可以跳过扫描请求，不回复扫描响应，这样可以节省电量，Beacon设备就是其中的一种。

 注意，所有的广播包，扫描请求包和扫描响应包都用同一个允许接入地址0x8E89BED6，这个通用的地址可以允许任何的蓝牙设备进行扫描和接收数据。

低功耗蓝牙设备用40个无线通道，中心频率是在2402+k\*2MHZ(k在0到39之间)，但是广播通道只有3个，就是37(2402MHZ), 38(2426MHZ), 39(2480MHZ)，选择这3个通道是为了避免与wifi通道冲突，下图所示就是蓝牙广播通道和wifi通道，其中扇形的是wifi

在nordic的协议栈中，默认情况下每一次广播事件都会在3个通道广播数据，扫描者会从3个中的1个通道抓取，如下图示

1、Broadcast

注意，一个设备可以同时扫描和广播，扫描者能发送给广播者唯一的信息是扫描请求包，它只包含了扫描者的自身地址。

2、广播类型和白名单广播

有4种广播类型(Section 2.3.1 Vol 6 Part B)

ADV\_IND: 可连接的非定向广播，这是通常的广播模式，任何设备能发送扫描响应和连接请求

ADV\_DIRECT\_IND:可连接的定向广播，这种模式定向的发送广播包给指定主设备去请求连接，其他的扫描者会忽略此包，同时此时的广播者也会忽略其他扫描者的连接请求和扫描请求。

ADV\_SCAN\_IND: 可扫描的非定向广播，这种广播包不接收连接请求，只接收扫描请求

ADV\_NONCONN\_IND: 不可连接的非定向广播，广播者不接收任何连接请求和扫描请求，在这种模式下可节省电量。Beacon就是最大的应用实例

白名单广播

广播者可以利用白名单来限制扫描者的互动。白名单列表包含一些设备的固定地址或者IRK numbers(主设备用的是随机地址)，如果不是白名单列表里的设备，它就会拒绝。所以白名单列表能被用来过滤扫描请求包，连接请求包。

B、Start advertising with nodrdic Softdevice

1、关于广播包和扫描响应包的设置可以参考(Chapter 11 Part C Vol 3)

   开始广播之前需要配置：

   广播时间间隔

   广播超时时间：在广播超时时间到了之后，会收到BLE\_GAP\_EVT\_TIMEOUT事件

  设备地址：如果设置的是定向广播

  白名单列表：如果有的话

   过滤选择： 过滤扫描请求还是连接请求或者两者都过滤

   广播通道选择：可以选择1个，2个或者3个都广播

  2、广播开始时可能收到的GAP事件

BLE\_GAP\_EVT\_TIMEOUT: 在超时时间到了出现，可以选择继续广播或者进入休眠模式。超时时间源可以是广播超时，连接超时，扫描超时，安全请求超时.

BLE\_GAP\_EVT\_SCAN\_REQ\_REPORT: 这个事件是在收到扫描请求以后才出现，只包含扫描者的地址和RSSI值，注意，只有你通过sd\_ble\_opt\_set()使能之后才有用.

BLE\_GAP\_EVT\_CONNECTED: 主端发送连接请求建立连接.

常见广播内容类型AD Type :

flag

AD Type是0x1，代表广播模式。LElimited Discoverable（AD data是0x0）是有限时间广播，而General Discoverable（AD data是0x1）是无限广播，直到被主动停止广播

Manufacture specific data

AD Type是0xff，代表厂商数据，自定义的数据可以写到这里。很多第三方场景应用都是利用这个字段来进一步定义应用协议格式，如微信的airsync协议和ibeacon协议。

service

代表服务的UUID，一般用16bit的UUID。如微信airsync协议的ServiceUUID是0xFEE7.

TX POWER LEVEL

发射功率，主机根据它和RSSI就可以计算主机到从机的距离，ibeacon室内定位即是利用这点.

local name

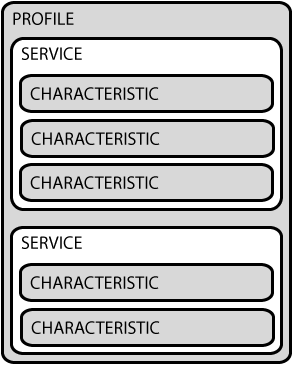
设备名称

<2> : GATT(**Generic Attribute Profile**) : GATT（Generic Attribute Profile）在ATT(Attribute Protocol)的基础上构建，为属性协议传输和存储数据建立了一些通用操作和框架。

GATT 通信的双方是 C/S 关系。外设作为 GATT 服务端（Server），它维护了 ATT 的查找表以及 service 和 characteristic 的定义。中心设备是 GATT 客户端（Client），它向 Server 发起请求。需要注意的是，所有的通信事件，都是由客户端（也叫中心设备，Master）发起，并且接收服务端（也叫外围设备，Slave）的响应。

一旦两个设备建立起了连接，GATT 就开始起作用了，这也意味着，你必需首先完成前面的 GAP 协议。这里需要说明的是，GATT 连接必需先经过 GAP 协议。实际上，我们在 Android 开发中，可以直接使用设备的 MAC 地址，发起连接，可以不经过扫描的步骤。这并不意味不需要经过 GAP，实际上在芯片级别已经给你做好了，蓝牙芯片发起连接，总是先扫描设备，扫描到了才会发起连接。

GATT 事务是建立在嵌套的Profiles, Services 和 Characteristics之上的的，如下图。



1）Profile Profile 并不是实际存在于 BLE 外设上的，它只是一个被 Bluetooth SIG 或者外设设计者预先定义的 Service 的集合。例如心率Profile（Heart Rate Profile）就是结合了 Heart Rate Service 和 Device Information Service。

2）Service Service 是把数据分成一个个的独立逻辑项，它包含一个或者多个 Characteristic。每个 Service 有一个 UUID 唯一标识。例如Heart Rate Service包含 3 个 Characteristic：Heart Rate Measurement, Body Sensor Location 和 Heart Rate Control Point。

3）Characteristic 在 GATT 事务中的最低界别的是 Characteristic，Characteristic 是最小的逻辑数据单元，当然它可能包含一个组关联的数据，例如加速度计的 X/Y/Z 三轴值。

实际上，和 BLE 外设打交道，主要是通过 Characteristic。你可以从 Characteristic 读取数据，也可以往 Characteristic 写数据。这样就实现了双向的通信。所以你可以自己实现一个类似串口（UART）的 Sevice，这个 Service 中包含两个 Characteristic，一个被配置只读的通道（RX），另一个配置为只写的通道（TX）。