

Dialog SDK 5.0.3 培训材料1

2016.3 数据广播



...personal
...portable
...connected

数据广播内容



数据广播

什么是广播？

barebone工程

barebone工程实例

源代码分析

输出显示

数据广播内容

你将会学会...

- 开始前 ...
 - 安装/解压 SDK5.0.3 到路径 C:\ drive
 - 快速浏览SDK目录下的Keil barebone 工程

- 将要学到的内容 ...
 - 理解BLE数据广播内容的基本概念
 - 总体了解 Dialog BLE SDK 5.0.3
 - 如何修改数据广播内容



数据广播内容

什么是广播？

- 低功耗蓝牙主要有两种操作模式：
 - 由服务定义的连接模式（查看下一个培训材料）
 - 无连接广播模式广播者定义了广播模式下的数据格式。

- 对于广播模式来说，有两种常见的应用：
 - 传输信标
 - 发布你的身份识别和服务信息

- 服务定义了在和设备建立连接之后所用到通信接口。

数据广播内容

什么是广播 ...

- 广播包长度较短，格式限定，只能携带少量用户数据。
- 广播模式同时支持在扫描响应的包内存储额外的数据。这些数据在无需建立连接的情况下，可以被使用扫描请求包的设备获得。
- 广播包是由一系列的AD字段组成，典型的如
 - 设备名字
 - 设备所支持的部分或者全部的服务类型
- 广播包也会包含厂商的特定字段和设备属性的标识。

数据广播内容

信标模式

- 信标通常只工作在广播模式，采用BLE的广播包来传输静态或者动态的数据，这些数据可以用来推送地址相关或上下文相关的内容用于使能手机端应用。
- 关于信标，有很多种规范，包含来自Apple的iBeacon, 和Google的Eddystone Beacon等. Dialog SDK可以在同一设备里支持其中一种或者多种的组合
- 尽管会有一些例外，但通常来说，信标模式是无法被连接的，所以不会广播服务信息。

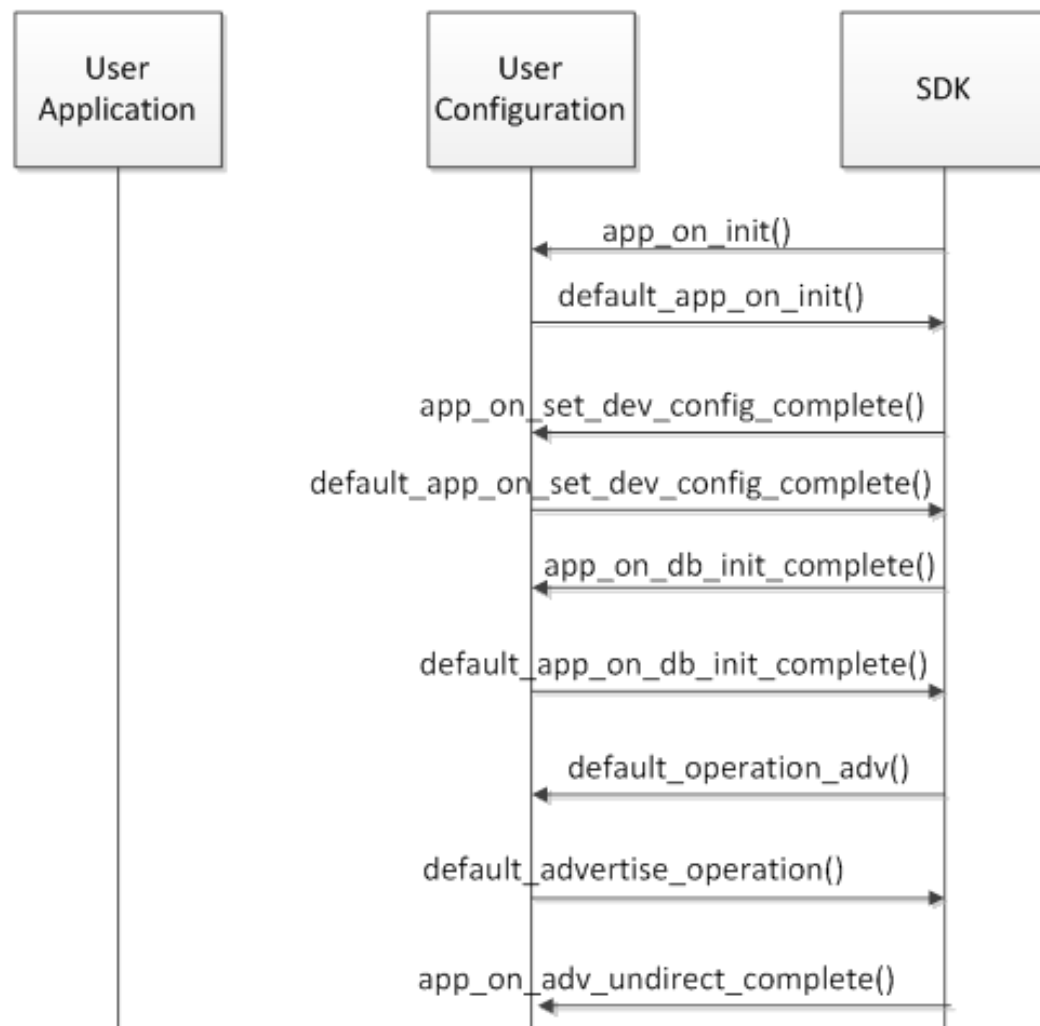
数据广播内容

Barebone 实例

- 第一个实例说明如何在Barebone工程的基础上构建自定义工程，用于广播带有特定的字段的位置信息.
- 工程路径(实际路径会有稍许差异):
C:\DA1458x_SDK\5.0.3\projects\target_apps\ble_examples\ble_app_barebone
- 每一个信标类型数据包里，都包含特定的厂商字段，用于广播自定义信息
- 接下去将介绍需要在源码上所做的修改.

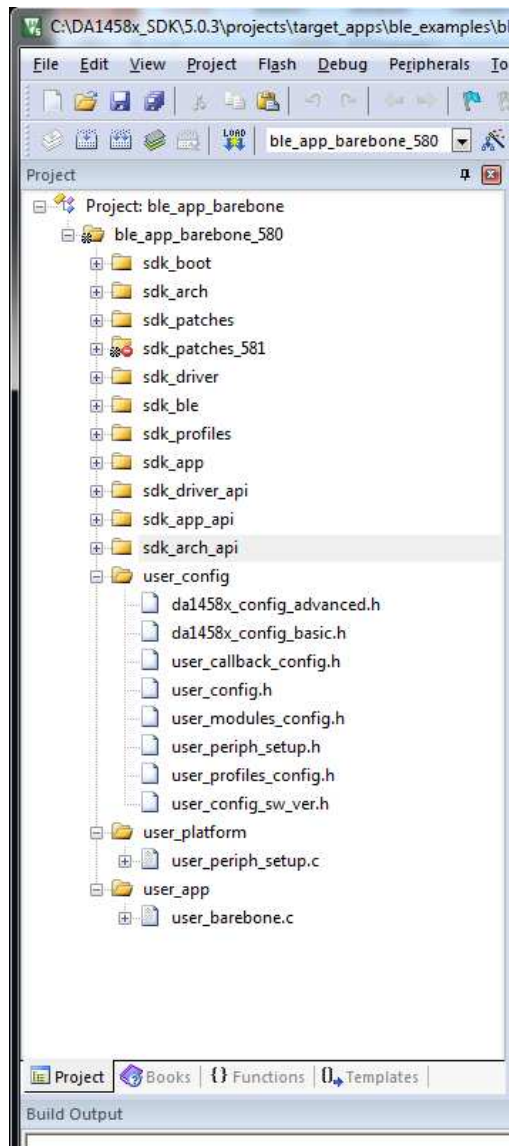
数据广播内容

应用代码流程图



数据广播内容

Barebone工程概览(工程文件的位置)



数据广播内容

源文件讨论: user_config/user_config.h

- 修改USER_DEVICE_NAME
- 练习: 修改名字“DIALOG-TRAINING”.

```
/// Advertising name
#define USER_DEVICE_NAME    ("DIALOG-TRAINING")
```

- 修改广播间隔.
- 练习: 修改广播间隔到1s

```
static const struct advertise_configuration user_undirected_advertise_conf ={
    /// Advertise operation type.
    .advertise_operation=ADV_NON_CONN,
    /// Own BD address source of the device:
    .address_src=GAPM_PUBLIC_ADDR,
    /// Advertise interval
    .intv = 1600,          //(1600 * 0.625ms) = 1 sec
```

数据广播内容

源文件讨论: user_app/user_barebone.c/user_barebone.h

- 广播包的填充数据部分里包含厂商的特定字段，这些字段可以被修改。在 user_barebone.h 文件里进行修改。
- 练习: 修改厂商字段为 “AAAA”.

```
/* Manufacturer specific data constants */  
#define APP_AD_MSD_COMPANY_ID          (0xAAAA)  
#define APP_AD_MSD_COMPANY_ID_LEN      (2)  
#define APP_AD_MSD_DATA_LEN            (sizeof(uint16_t))
```

- 厂商字段在特定时间里面可以自动定时修改.
- 练习: 修改定时器时间间隔到1s.

```
/* Advertising data update timer */  
#define APP_ADV_DATA_UPDATE_T0         (100)    // 100*10ms = 1 second
```

数据广播内容



涉及到的源文件: `user_callback_config.h`

- 当设备开始广播时，我们想要调用自定义的函数。下面的例子将说明如何利用默认的函数接口来实现。修改涉及文件” `user_callback_config.h`”

```
// Default Handler Operations
static const struct default_app_operations user_default_app_operations = {
    .default_operation_adv = user_app_adv_start,
};
```

数据广播内容

涉及源文件: user_app/user_barebone.c

用户自定义函数 user_app_adv_start()

- 当设备开始广播时，会调用这个函数来设置一些用于广播内容的标准参数。
- 广播内容就是直接从之前定义的结构里拷贝过来的。
- 最后，设备会要求基于这些配置来开始广播。

```
void user_app_adv_start(void)
{
    // Schedule the next advertising data update
    app_adv_data_update_timer_used = app_easy_timer(APP_ADV_DATA_UPDATE_TO, adv_data_update_timer_cb);

    struct gapm_start_advertise_cmd* cmd;
    cmd = app_easy_gap_undirected_advertise_get_active();

    // add manufacturer specific data dynamically
    mnf_data_update();
    app_add_ad_struct(cmd, &mnf_data, sizeof(struct mnf_specific_data_ad_structure));

    app_easy_gap_undirected_advertise_start();
}
```

数据广播内容

源文件讨论: user_app/user_barebone.c

- 函数 mnf_data_update() 用于在自定义的时间段里更新厂商字段内容（之前的章节已描述）。

```
static void mnf_data_update()
{
    uint16_t data;

    data = mnf_data.proprietary_data[0] | (mnf_data.proprietary_data[1] << 8);
    data += 1;
    mnf_data.proprietary_data[0] = data & 0xFF;
    mnf_data.proprietary_data[1] = (data >> 8) & 0xFF;

    if (data == 0xFFFF) {
        mnf_data.proprietary_data[0] = 0;
        mnf_data.proprietary_data[1] = 0;
    }
}
```

数据广播内容



涉及的源文件: da1458x_config_advanced.h

- 修改广播包的蓝牙地址.
- 练习: 修改默认蓝牙地址为01:02:03:04:05:06.

```
#define CFG_NVDS_TAG_BD_ADDRESS          {0x01, 0x02, 0x03, 0x04, 0x05, 0x06}
```

数据广播内容



在输出界面会看到：

- 当运行起来后，在手机上的BLEscanner应用软件上会显示设备名称（DIALOG-TRAINING)和地址(06:05:04:03:02:01)。这个设备不会处于连接模式
- 选中设备后，在界面上会显示厂商特定字段，大多数情况下会是一组16进制数，在这个例子里面会是4个字节。在厂商字段里会看到之前在config结构里定义的特定UUID字段，且会每1秒递增。
- 深入阅读：
 - <http://support.dialog-semiconductor.com/connectivity/product/da14580>
 - <http://support.dialog-semiconductor.com/resource/um-b-050-da1458x-software-development-guide-sdk5>

Bluetooth LE Scanner的屏幕截图



The Power To Be...■■



...personal
...portable
...connected