GR551x AMS Profile示例手册

版本： **1.8**

发布日期： **2021-08-09**

深圳市汇顶科技股份有限公司

前言

编写目的

本文档介绍了如何使用和验证GR551x SDK中的AMS Client示例，旨在帮助用户快速进行二次开发。

读者对象

本文适用于以下读者：

- GR551x用户
- GR551x开发人员
- GR551x测试人员
- iOS开发工程师
- 开发爱好者
- 文档工程师

版本说明

本文档为第6次发布，对应的产品系列为GR551x。

修订记录

版本	日期	修订内容
1.0	2019-12-08	首次发布
1.3	2020-03-16	更新“测试验证”章节中串口输出时对象ID的名称
1.5	2020-05-30	更正“应用详解”章节中的代码缩进格式
1.6	2020-06-30	基于SDK刷新版本
1.7	2021-04-20	优化“初次运行”和“应用详解”章节
1.8	2021-08-09	更新“准备工作”章节

版权所有 © 2021 深圳市汇顶科技股份有限公司

目录

前言 I

1 简介 1

2 Profile概述 2

3 初次运行 4

3.1 准备工作 4
3.2 固件烧录 4
3.3 测试验证 5

4 应用详解 11

4.1 运行流程 11
4.2 关键代码 12
4.2.1 远程控制 12
4.2.2 设置关注对象 12
4.2.3 设置展示对象 13
4.2.4 读取展示对象 13

5 常见问题 14

5.1 串口终端没有输出信息 14
5.2 手机搜索不到蓝牙广播 14
5.3 手机无法连接AMS Client设备蓝牙广播 14

版权所有 © 2021 深圳市汇顶科技股份有限公司

1 简介

苹果媒体服务（Apple Media Service，AMS）应用于蓝牙设备，如手环、手表等智能设备。蓝牙设备可通过BLE连接获取iOS设备的媒体通知，也可以向iOS设备发送AMS相关控制命令。

本文档主要介绍如何基于GR551x平台实现AMS Client，包括示例的运行方法及关键源码的说明。

在进行操作前，可参考以下文档。

表 1-1 文档参考

名称	描述
AMS Reference	AMS协议：Apple Media Service(AMS) Specification
GR551x开发者指南	GR551x软硬件介绍、快速使用及资源总览
Bluetooth Core Spec	Bluetooth官方标准核心规范
Bluetooth GATT Spec	Bluetooth Profile和Service的详细信息查看地址：www.bluetooth.com/specifications/gatt
J-Link用户指南	J-Link使用说明：www.segger.com/downloads/jlink/UM08001_JLink.pdf
Keil用户指南	Keil详细操作说明：www.keil.com/support/man/docs/uv4/

版权所有 © 2021 深圳市汇顶科技股份有限公司

2 Profile概述

AMS Profile定义了以下两种设备角色：

- Server端：iOS设备为中央设备（Central），提供Service以及数据源。
- Client端：蓝牙设备为外围设备（Peripheral），连接iOS设备后，发现iOS设备上的服务以及读写其数据。

Server端与Client端的交互流程如图 2-1所示：

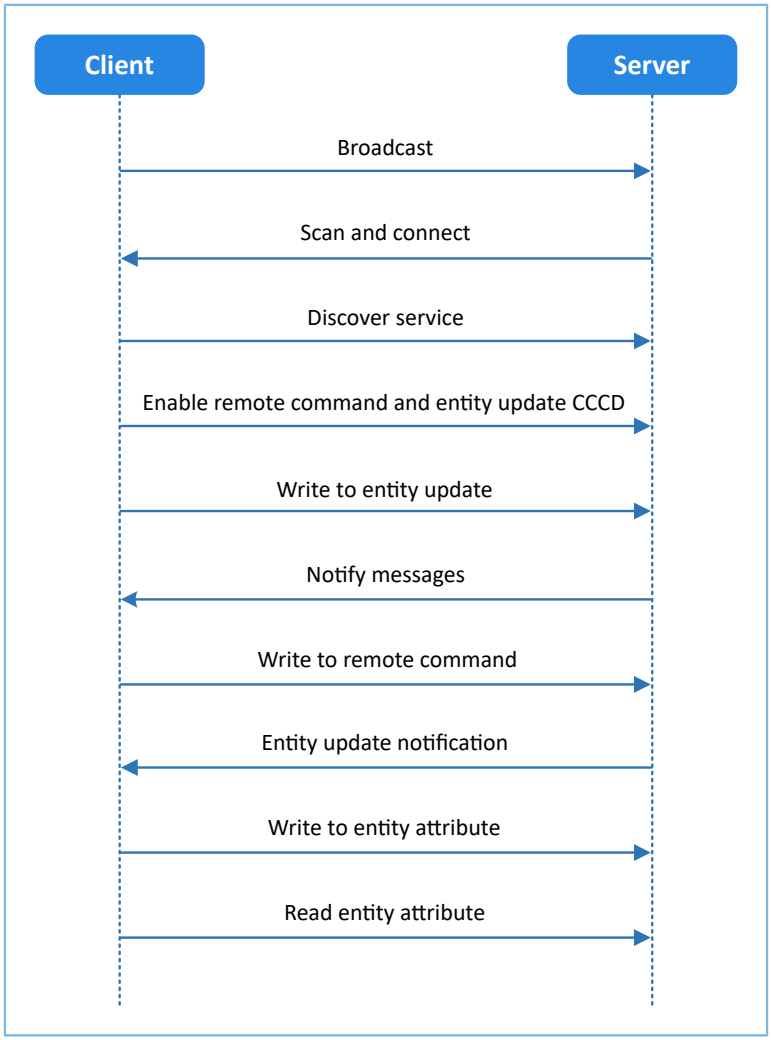

图 2-1 Client和Server 交互流程图

AMS 的特性（Characteristic）有：Remote Command、Entity Update、Entity Attribute，如下表 2-1 所示。

表 2-1 AMS Characteristic说明

Characteristic	UUID	Type	Support	Security	Properties
Remote Command	9B3C81D8-57B1-4A8A-B8DF-0E56F7CA51C2	128 bits	Mandatory	None	Write, Notify
Entity Update	2F7CABCE-808D-411F-9A0C-BB92BA96C102	128 bits	Mandatory	None	Write, Notify
Entity Attribute	C6B2F38C-23AB-46D8-A6AB-A3A870BBD5D7	128 bits	Mandatory	None	Write, Read

各特性的作用如下：

版权所有 © 2021 深圳市汇顶科技股份有限公司

- Remote Command：发送远程命令并接收可用远程命令更新。Write用于从蓝牙设备端发送远程命令到iOS设备端来实现远程控制。Notify用于当iOS设备端的可用远程命令发生变化时，iOS设备端向蓝牙设备端通知更新后的可用远程命令。
- Entity Update：设置关注对象并接收关注对象更新。Write用于在蓝牙设备端设置关注对象。Notify用于当iOS设备端的关注对象发生变化时，iOS设备端向蓝牙设备端通知关注对象更新后的值（受限于MTU长度，该值可能是不完整的）。
- Entity Attribute：设置和读取展示对象。Write用于在蓝牙设备端设置展示对象，Read用于在设置展示对象后，从iOS设备端读取展示对象的完整值。

版权所有 © 2021 深圳市汇顶科技股份有限公司

3 初次运行

本章将介绍如何使用开发板作为AMS Client、iOS设备作为AMS Server，初次运行AMS Client示例。

说明:

SDK_Folder为GR551x SDK的根目录。

3.1 准备工作

运行AMS Client示例之前，需要完成以下准备工作。

- 硬件准备

表 3-1 硬件准备

名称	描述
开发板	GR5515 Starter Kit开发板（以下简称开发板）
数据线	Micro USB 2.0数据线
iOS设备	支持BLE 4.0及以上的iOS设备，如iPhone 4s及其以上版本、iPad 3及其以上版本

- 软件准备

表 3-2 软件准备

名称	描述
Windows	Windows 7/Windows 10操作系统
J-Link Driver	J-Link驱动程序，下载网址：www.segger.com/downloads/jlink/
Keil MDK5	IDE工具，支持MDK-ARM 5.20 及以上版本，下载网址：www.keil.com/download/product/
GProgrammer（Windows）	Programming工具，位于`SDK_Folder\tools\GProgrammer`
GRUart（Windows）	串口调试工具，位于`SDK_Folder\tools\GRUart`

3.2 固件烧录

AMS Client示例工程的源码位于`SDK_Folder\projects\ble\ble_peripheral\ble_app_ams_c`。

用户可通过GProgrammer将*ble_app_ams_c_fw.bin*固件烧录至开发板，GProgrammer烧录固件的具体操作方法，请参考《GProgrammer用户手册》。

说明:

- *ble_app_ams_c_fw.bin*位于：`SDK_Folder\projects\ble\ble_peripheral\ble_app_ams_c\build\`。
- GProgrammer位于`SDK_Folder\tools\GProgrammer`。

版权所有 © 2021 深圳市汇顶科技股份有限公司

3.3 测试验证

开发板、iOS设备及GRUart准备就绪后，即可开始AMS通讯测试。在本示例中针对AMS服务的测试，包括：

- 开发板获取iOS设备的媒体通知。
- 开发板向iOS设备发送AMS控制命令。

用户可根据GRUart界面显示的打印信息，验证AMS服务是否运行正常。（如需进一步了解AMS，请参考Apple Media Service(AMS) Specification）。

具体步骤如下：

1. 建立连接。

   开发板上电。打开iOS设备蓝牙功能，搜索周边蓝牙设备，发现广播名为“Goodix_AMS_C”的开发板，如图 3-1所示。

图 3-1 发现广播名Goodix_AMS_C的设备

说明:

本文档测试用iOS设备为iPhone 6S，系统版本为12.4.1，可能与实际用户界面有差异。

点击“Goodix_AMS_C”连接开发板，自动弹出如图 3-2所示的配对请求窗口。在该窗口中，输入代码“123456”（配对所需代码的设置方法，可参考《GR551x鼠标示例手册》“关键代码”）后点击“配对”。

版权所有 © 2021 深圳市汇顶科技股份有限公司

图 3-2 输入配对信息

配对成功后，提示“已连接”信息，如图 3-3所示。

图 3-3 设备配对成功

iOS设备与开发板连接成功后，连接信息将显示在串口调试工具GRUart的界面上，如图 3-4所示。

版权所有 © 2021 深圳市汇顶科技股份有限公司

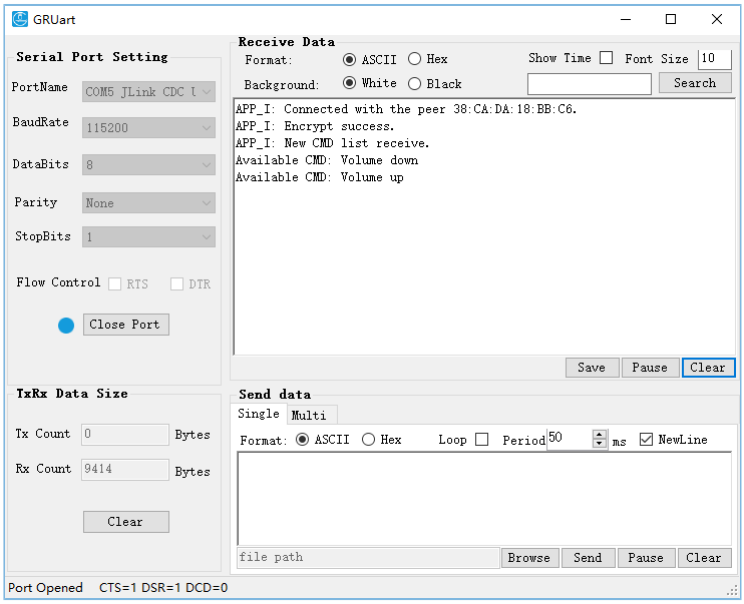

图 3-4 串口打印信息

2. 获取iOS设备媒体通知。

   开发板获取iOS设备媒体通知的测试步骤如下：

   (1) 在iOS设备上打开任意音乐播放器App。

   (2) 使用音乐播放器App播放任意歌曲，如图 3-5所示，示例中歌曲名为The Sound Of Silence，歌手名为Pat Metheny。

图 3-5 iOS设备上播放歌曲

   (3) 查看PC端GRUart的界面上的打印信息。

   在本示例中，GRUart的界面上显示的打印信息（如图 3-6所示），包括：

版权所有 © 2021 深圳市汇顶科技股份有限公司

- 该音乐播放器App支持的远程命令列表。
- 关注对象的更新信息。如关注了iOS设备媒体的Track-Title和Track-Artist对象，该对象发生改变时会发送相应的通知。（关于对象的详细信息，请参考Apple Media Service(AMS) Reference）。

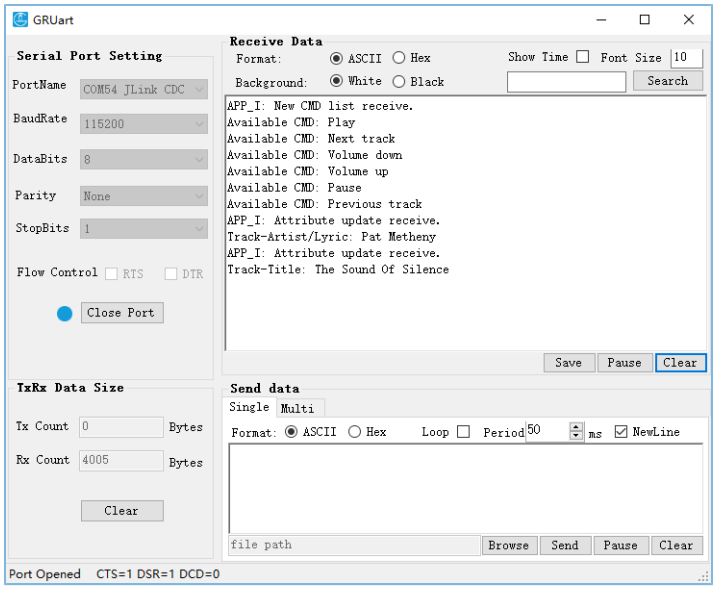

图 3-6 获取iOS设备媒体通知的打印信息

串口输出信息说明如下：

表 3-3 通知信息说明

名称	描述
APP_I: New CMD list receive	收到新的可用远程命令列表。
Available CMD: Next track	表明当前Next track命令可用，可发送“下一首”远程命令。
APP_I: Attribute update receive	关注的某个对象发生变化，收到该对象新的值。
Track-Artist/Lyric: Pat Metheny	Track-Artist对象发生变化，其新的值为“Pat Metheny”。

说明:

根据Apple Media Service(AMS) Reference的规定，Track-Artist的对象ID对应当前播放曲目的演唱者信息，但目前多数的音乐播放器App还通过该ID传递歌词信息。因此，在串口输出信息中，该ID对应的名称显示为：Track-Artist/Lyric。

通过图 3-6的打印信息，可确定开发板和iOS设备之间的AMS服务运行正常。

3. 向iOS设备发送控制命令。

   开发板向iOS设备发送控制命令的步骤如下：

   (1) 在iOS设备上打开任意音乐播放器App。

   (2) 使用音乐播放器App播放任意一首歌曲。

版权所有 © 2021 深圳市汇顶科技股份有限公司

(3) 在音乐播放过程中，按下开发板的“RIGHT”按键（即发送播放“下一首”的命令）。

(4) 查看音乐播放器是否切换到播放列表中的下一首歌曲进行播放，以及GRUart界面显示的信息。

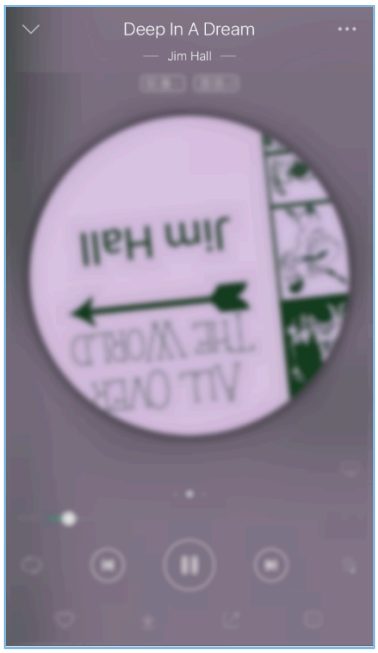

图 3-7 iOS设备上播放歌曲

在本示例中，GRUart的界面上显示的打印信息（如图 3-8所示），包括：

- 远程命令发送成功的信息。
- 当前已更新的歌曲名和演唱者信息（表征远程命令已经被执行）。

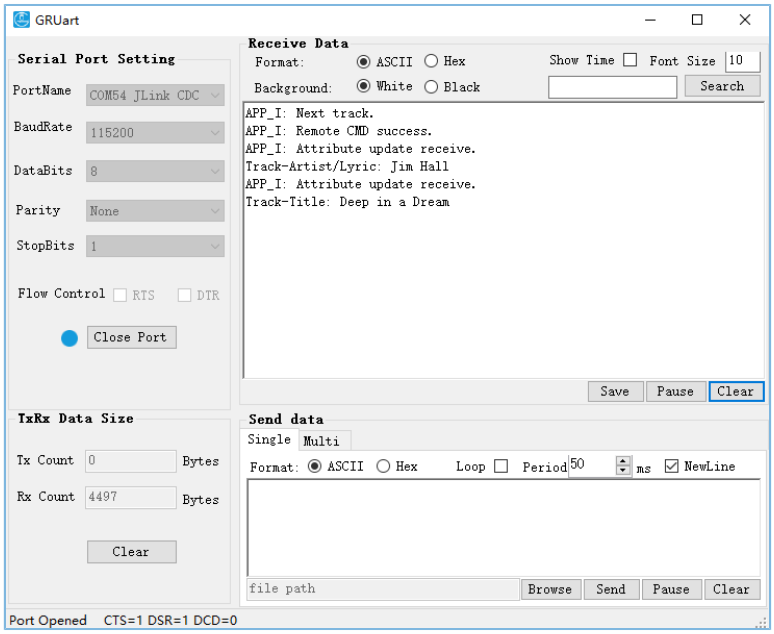

图 3-8 向iOS设备发控制命令的打印信息

通过图 3-8的打印信息，可确定开发板和iOS设备之间的AMS服务运行正常。

版权所有 © 2021 深圳市汇顶科技股份有限公司

说明:

开发板按键的详细信息，请参考用户对应的Starter Kit用户指南。

表 3-4 按键与命令对应关系说明

按键操作	对应命令
OK键（短按一次）	Play
OK键（短按两次）	Pause
OK键（长按一次）	Toggle play/pause
RIGHT键	Next track
LEFT键	Previous track
UP键	Volume up
DOWN键	Volume down

版权所有 © 2021 深圳市汇顶科技股份有限公司

4 应用详解

本章将介绍AMS Client示例的运行流程和关键代码。

4.1 运行流程

AMS Profile示例开始运行后，将依次执行配对与绑定、发现服务、CCCD（Client Characteristic Configuration Descriptor）使能、通知使能、通知消息处理和读写交互等操作。本节以AMS Client为例，列举出的交互流程如图 4-1所示。

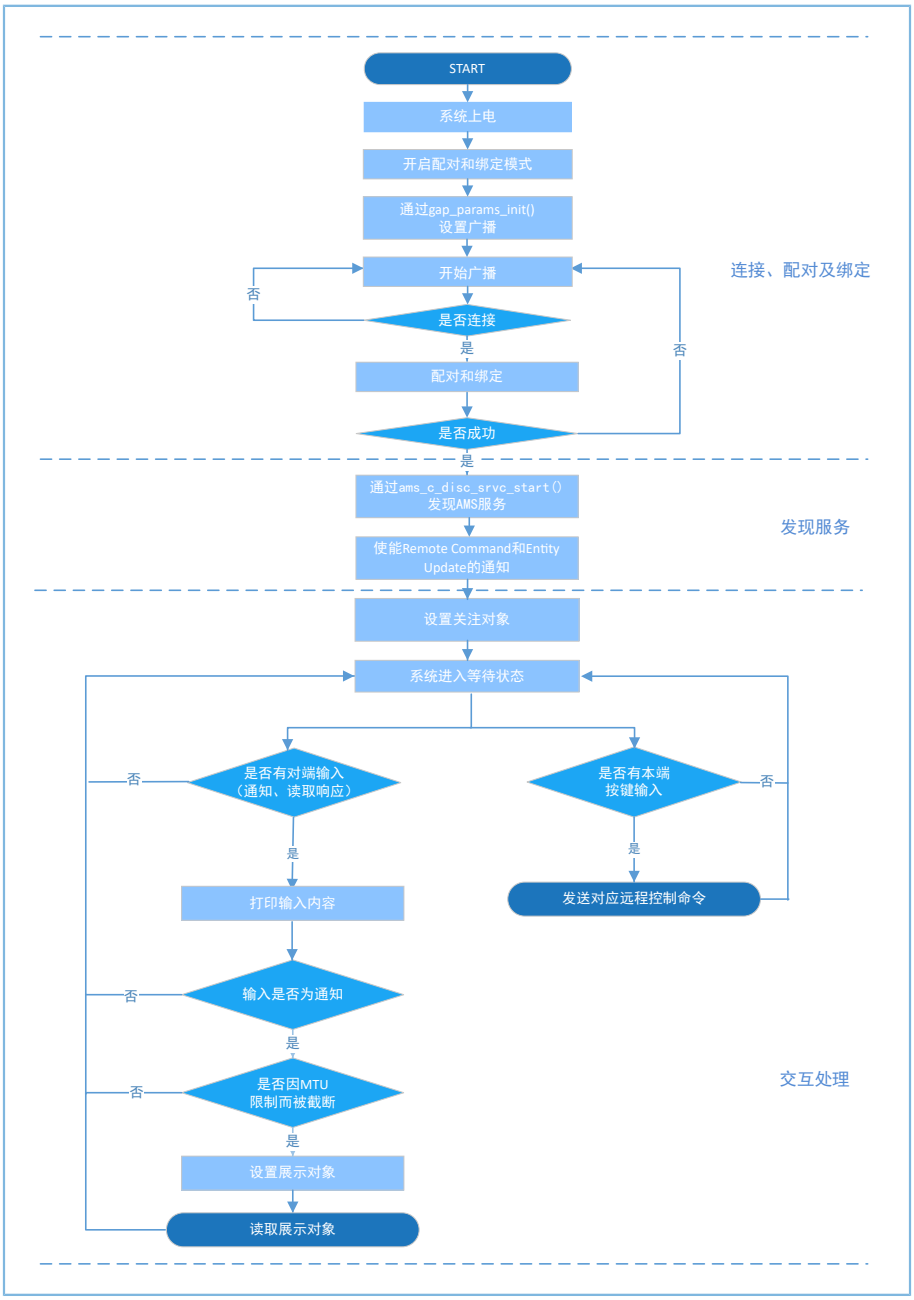

图 4-1 AMS交互流程图

版权所有 © 2021 深圳市汇顶科技股份有限公司

4.2 关键代码

蓝牙设备端（即开发板）和iOS设备经过扫描、配对与绑定、发现服务和使能相关CCCD后，蓝牙设备可通过向iOS设备端AMS的各特性写入或读取信息实现远程交互。下文描述了AMS支持的读写交互操作的主要代码。

4.2.1 远程控制

当用户在蓝牙设备端进行按键类操作时，可通过ams_c_cmd_send()向iOS设备端的AMS Remote Command特性发送命令以实现远程控制。命令内容为远程命令ID，ams_c_cmd_id_t结构体定义了各项远程命令ID的枚举值。

路径：工程目录下的gr_profiles\ams_c.c

名称：ams_c_cmd_send();

```
sdk_err_t ams_c_cmd_send(uint8_t conn_idx, uint8_t cmd_id)
{
    ...
    gattc_write_attr_value_t write_attr_value;
    write_attr_value.handle = s_ams_c_env.handles.ams_cmd_handle;
    write_attr_value.offset = 0;
    write_attr_value.length = 1;
    write_attr_value.p_value = (uint8_t *)&cmd_id;

    return ble_gattc_prf_write(s_ams_c_env.prf_id, conn_idx, &write_attr_value);
}
```

路径：工程目录下的gr_profiles\ams_c.h

名称：ams_c_cmd_id_t;

```
typedef enum
{
    AMS_CMD_ID_PLAY,                        /**< Command index of play. */
    AMS_CMD_ID_PAUSE,                       /**< Command index of pause. */
    AMS_CMD_ID_TOGGLE_PLAY_PAUSE,           /**< Command index of toggle. */
    AMS_CMD_ID_NEXT_TRACK,                  /**< Command index of next track. */
    AMS_CMD_ID_PREVIOUS_TRACK,              /**< Command index of previous track. */
    ...
} ams_c_cm
```

4.2.2 设置关注对象

蓝牙设备可通过attr_focus_set()向Entity Update特性写入对象ID来设置关注对象（对象ID包括Entity ID和Attribute ID两部分，如需了解对象ID详情请参考AppleMedia Service(AMS) Reference）。

路径：工程目录下的user_app\user_app.c

名称：attr_focus_set();

```
static void attr_focus_set(uint8_t conn_idx)
```

版权所有 © 2021 深圳市汇顶科技股份有限公司

```
{
    sdk_err_t   error_code;
    ams_c_ett_attr_id_t track_attr_id =
    {
        .ett_id     = AMS_TRACK_ID,
        .attr_id    = {AMS_TRACK_ARTIST_ID, AMS_TRACK_TITTLE_ID},
        .attr_count = 2
    };
    error_code = ams_c_attr_focus_set(conn_idx, &track_attr_id);
    APP_ERROR_CHECK(error_code);
}
```

4.2.3 设置展示对象

蓝牙设备通过ams_c_attr_display_set()向Entity Attribute特性写入对象ID来设置展示对象。读取Entity Attribute特性的值来获得该对象的完整数据。

路径：工程目录下的 gr_profiles\ams_c.c

名称：ams_c_attr_display_set();

```
sdk_err_t ams_c_attr_display_set(uint8_t conn_idx, const ams_c_attr_info_t *p_attr_info)
{
    ...
    gattc_write_attr_value_t write_attr_value;
    write_attr_value.handle = s_ams_c_env.handles.ams_attr_display_handle;
    write_attr_value.offset = 0;
    write_attr_value.length = 2;
    write_attr_value.p_value = (uint8_t *)&(p_attr_info->ett_id);
    return ble_gattc_prf_write(s_ams_c_env.prf_id, conn_idx, &write_attr_value);
}
```

4.2.4 读取展示对象

由于MTU的限制，特性Entity Update通知的对象数据可能是被截断的。要获得完整的对象数据，需要使用特性Entity Attribute。当用户成功向Entity Attribute特性写入对象ID设置展示对象后，可通过ams_c_cplt_attr_read()读取Entity Attribute特性的值来获得该对象的完整数据。

路径：工程目录下的gr_profiles\ams_c.c

名称：ams_c_cplt_attr_read()

```
sdk_err_t ams_c_cplt_attr_read(uint8_t conn_idx)
{
    if (BLE_ATT_INVALID_HDL == s_ams_c_env.handles.ams_attr_display_handle)
    {
        return BLE_ATT_ERR_INVALID_HANDLE;
    }
    return ble_gattc_prf_read(s_ams_c_env.prf_id, conn_idx,
                              s_ams_c_env.handles.ams_attr_display_handle, 0);
}
```

版权所有 © 2021 深圳市汇顶科技股份有限公司

5 常见问题

本章描述了在验证及应用AMS Client示例时，可能出现的问题、原因及处理方法。在以下描述中，手机指的是iOS设备。

5.1 串口终端没有输出信息

- 问题描述

  串口终端GRUart没有打印信息，或者打印乱码。

- 问题分析

  固件没有正确烧录到开发板运行，或者串口波特率设置不正确导致信息不能正常打印在终端。

- 处理方法

  1. 请确认GRUart的串口波特率已设置为115200，数据位为8，停止位为1，无校验位，无流控。同时检查串口线接入是否正确。
  2. 如果串口正常，请重新烧录*ble_app_ams_c_fw.bin*固件，并确认工程代码没有被修改，使用GProgrammer下载固件。

5.2 手机搜索不到蓝牙广播

- 问题描述

  打开手机App后，无法搜索到以Goodix_AMS_C为名称的广播。

- 问题分析

  蓝牙天线连接或固件异常。

- 处理方法

  1. 请确定使用的手机是否为iOS设备，手机蓝牙功能是否打开。如果都符合，则可能存在硬件问题，请检开发板天线是否正常工作。
  2. 如果以上均为正常，用户可通过下载模板固件*ble_app_template_fw.bin*来检查硬件是否异常。模板固件位于：`SDK_Folder\projects\ble\ble_peripheral\ble_app_template\build\`。运行该固件后，若硬件正常，则能够搜索到以Goodix_Tem为名称的广播，否则可能存在硬件异常。

5.3 手机无法连接AMS Client设备蓝牙广播

- 问题描述

  名称为Goodix_AMS_C的广播，出现在蓝牙的“我的设备”列表中，但无法连接。

版权所有 © 2021 深圳市汇顶科技股份有限公司

- 问题分析

  手机曾经连接并绑定过开发板，但该设备上的数据可能经过擦除和重新烧写，已丢失了绑定信息，使得绑定失效。

- 处理方法

  1. 在蓝牙的“我的设备”列表中，选择“Goodix_AMS_C”设备，再选择忽略此设备。
  2. 重新进行扫描、配对和绑定流程。

版权所有 © 2021 深圳市汇顶科技股份有限公司