吐血推荐历史最全的蓝牙协议栈介绍

一. 声明

本专栏文章我们会以连载的方式持续更新，本专栏计划更新内容如下：

第一篇:蓝牙综合介绍，主要介绍蓝牙的一些概念，产生背景，发展轨迹，市面蓝牙介绍，以及蓝牙开发板介绍。

第二篇:Transport层介绍,主要介绍蓝牙协议栈跟蓝牙芯片之前的硬件传输协议,比如基于UART的H4,H5,BCSP，基于USB的H2等

第三篇:传统蓝牙controller介绍，主要介绍传统蓝牙芯片的介绍，包括射频层（RF），基带层（baseband），链路管理层（LMP）等

第四篇:传统蓝牙host介绍，主要介绍传统蓝牙的协议栈，比如HCI,L2CAP,SDP,RFCOMM,HFP,SPP,HID,AVDTP,AVCTP,A2DP,AVRCP,OBEX,PBAP,MAP等等一系列的协议吧。

第五篇：低功耗蓝牙controller介绍，主要介绍低功耗蓝牙芯片，包括物理层（PHY），链路层（LL）

第六篇：低功耗蓝牙host介绍，低功耗蓝牙协议栈的介绍，包括HCI,L2CAP,ATT,GATT,SM等

第七篇：蓝牙芯片介绍，主要介绍一些蓝牙芯片的初始化流程，基于HCI vendor command的扩展

第八篇：附录，主要介绍以上常用名词的介绍以及一些特殊流程的介绍等。

另外，开发板如下所示，对于想学习蓝牙协议栈的最好人手一套。以便更好的学习蓝牙协议栈，相信我，学完这一套视频你将拥有修改任何协议栈的能力（比如Linux下的bluez，Android下的bluedroid）。

CSDN学院链接（进入选择你想要学习的课程）：https://edu.csdn.net/lecturer/5352?spm=1002.2001.3001.4144

蓝牙交流扣扣群：970324688

Github代码：https://github.com/sj15712795029/bluetooth_stack

入手开发板：https://item.taobao.com/item.htm?spm=a1z10.1-c-s.w4004-22329603896.18.5aeb41f973iStr&id=622836061708

二. 前言

首先在介绍以下内容之前，我们先来介绍下我们的CSDN课程，以下介绍内容都会在 CSDN课程 手把手教你蓝牙协议栈入门（点击我）中第三小节介绍。

三. HCI蓝牙架构

在介绍架构之前我们先介绍下两个名词:

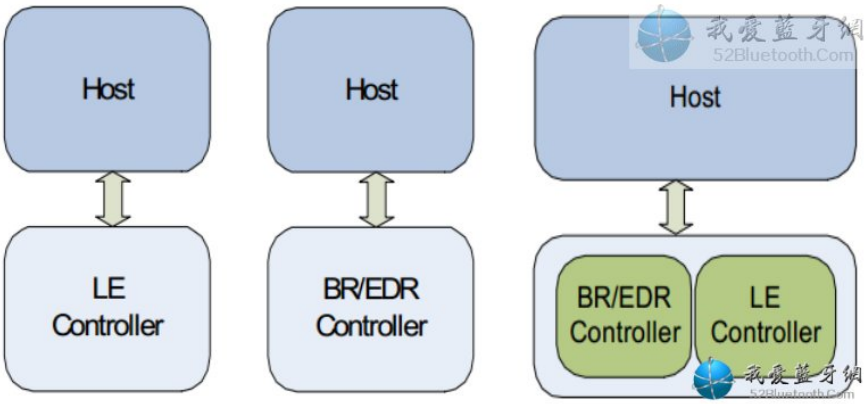
1）BT Controller：此部分指的是蓝牙芯片，包括BR/EDR芯片（蓝牙2.1芯片），AMP芯片（蓝牙3.0芯片），LE芯片（蓝牙4.0芯片），后续我们把4.0以下统称为传统蓝牙，4.0以上称为低功耗蓝牙，芯片层面会有2种模式，包括

单模蓝牙芯片：单一传统蓝牙芯片，单一低功耗蓝牙芯片

双模蓝牙芯片：同时支持传统蓝牙跟低功耗蓝牙的芯片

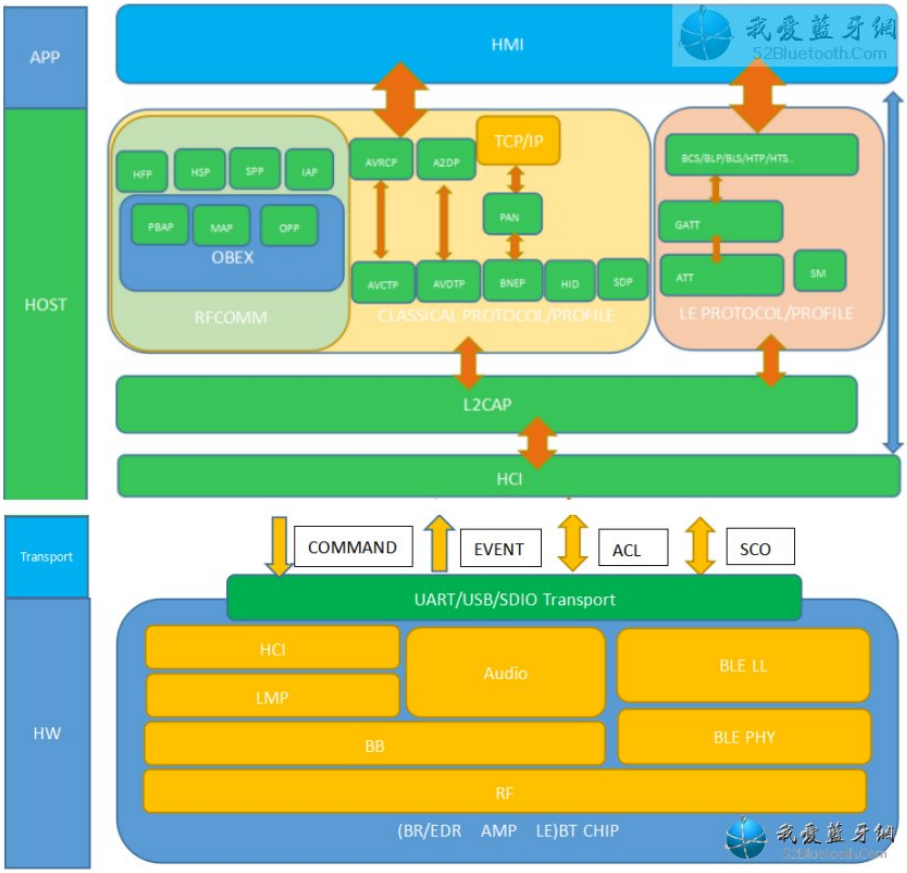
2）BT Host：蓝牙协议栈

所以HCI架构的蓝牙会有以下几种架构



四. HCI架构的蓝牙协议简要介绍

细展开以上的架构如下：



以上架构图从最底下开始大概说明，在后续章节也会逐一展开

HW层：这里就是蓝牙芯片层，包含以下几个部分

1）RF（RADIO）：射频层，本地蓝牙数据通过射频发送给远端设备，并且通过射频接收来自远端蓝牙设备的数据

2）BB（BASEBAND）：基带层，进行射频信号与数字或语音信号的相互转化，实现基带协议和其它的底层连接规程。

3）LMP（LINK MANAGER PROTOCOL）：链路管理层，负责管理蓝牙设备之间的通信，实现链路的建立、验证、链路配置等操作

4）HCI（HOST CONTROLLER INTERFACE）：主机控制器接口层，HCI层在芯片以及协议栈都有，芯片层面的HCI负责把协议栈的数据做处理，转换为芯片内部动作，并且接收到远端的数据，通过HCI上报给协议栈。

5）BLE PHY：BLE的物理层

6）BLE LL：BLE的链路层

TRANSPORT层：此部分在硬件接口（UART/USB/SDIO）实现HOST跟CONTROLLER的交互，此部分会分为以下几个协议，在后续章节会对transport协议做详细的说明

1）H2：USB的transport

2）H4：UART的transport

H4是UART传输种最简的一个Transport，只是在HCI raw data的前面加一个type就行，如下HCI一共有五种HCI data:

* HCI COMMAND:由蓝牙协议栈发送给芯片的命令

* HCI EVENT:由蓝牙芯片上报给蓝牙协议栈的事件

* HCI ACL:蓝牙协议栈跟蓝牙芯片双向交互的普通数据

* HCI SCO:蓝牙芯片跟蓝牙协议栈双向交互的通话/语音识别等音频数据

* HCI ISO（这部分是在core5.2才添加）:LE audio用的数据包格式

交互数据格式为：

```
|- 1 byte -|---- N byte HCI raw data ----|
| H4 TYPE -|---- HCI RAW DATA -----------|
```

其中H4 type定义如下：

HCI packet type	HCI packet indicator
HCI Command packet	0x01
HCI ACL Data packet	0x02
HCI Synchronous Data packet	0x03
HCI Event packet	0x04
HCI ISO Data packet	0x05

Table 2.1: HCI packet indicators

3）H5：UART的transport

4）BCSP: UART的transport

5）SDIO Transport,我不知道叫什么transport,但是有走SDIO的蓝牙芯片，比如Marvell8887，可以选择走SDIO或者UART

其中2,3,4的主要差别在于H4需要BT CHIP UART_TX/UART_RX/UART_CTS/UART_RTS/VCC/GND接到MCU，而H5,BCSP只需要BT CHIP的UART_TX/UART_RX/VCC/GND接到MCU就可以通信。

HOST层：此部分就是蓝牙协议栈，是我们本书的重点

1）HCI（HOST CONTROLLER INTERFACE）：主机控制层接口，主要负责透过transport把协议栈的数据发送给蓝牙芯片，并且接受来自蓝牙芯片的数据，数据主要分为HCI COMMAND(HOST->CONTROLLER),HCI EVENT(HOST<-CONTROLLER),HCI ACL(HOST<->CONTROLLER),HCI SCO(这个有点些微差异，因为部分芯片的SCO数据不是透过TRANSPORT直接跟HOST沟通，而是通过特殊的引脚，PCM IN/OUT/SYNC/CLK脚来传输数据),core文档HCI的架构如下：

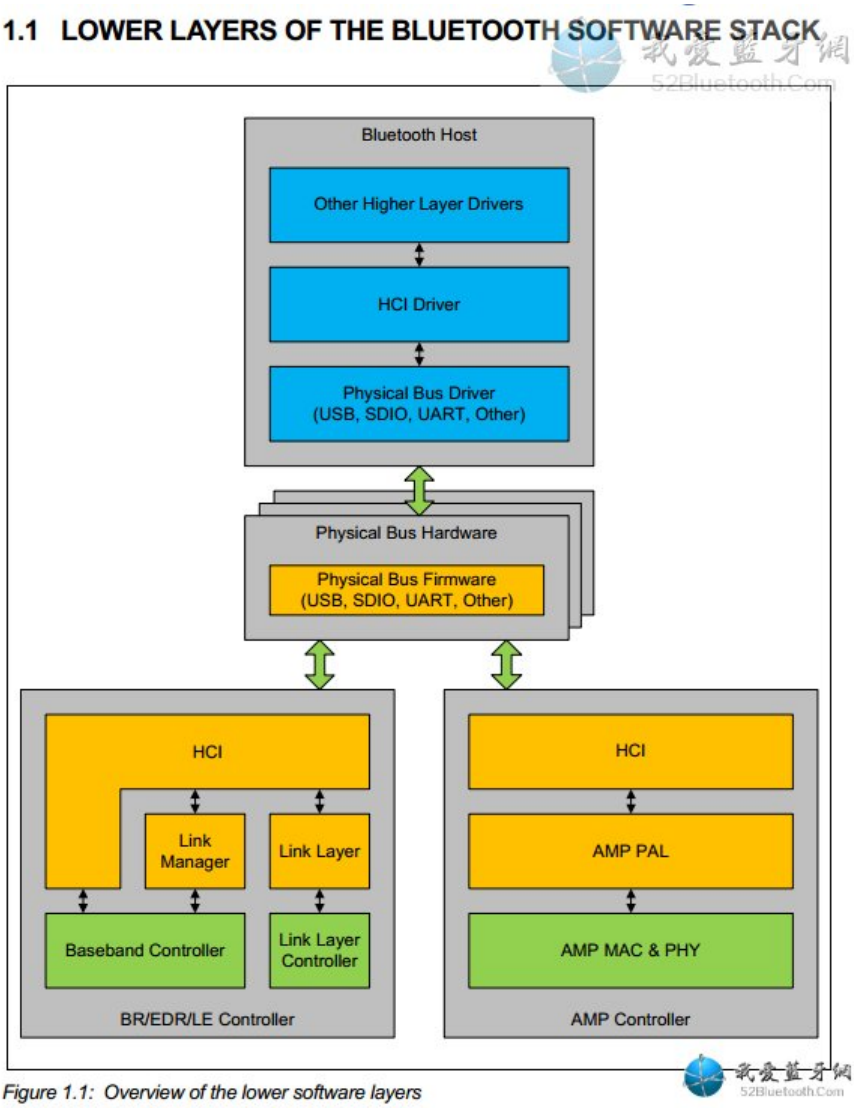


Figure 1.1: Overview of the lower software layers

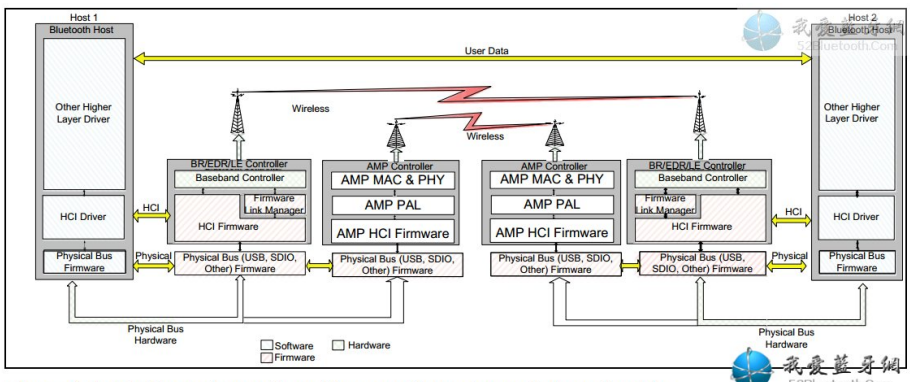


Figure 1.2: End to end overview of lower software layers to transfer data

2）L2CAP（Logical Link Control and Adaptation Protocol）：逻辑链路控制与适配协议，将ACL数据分组交换为便于高层应用的数据分组格式，并提供协议复用和服务质量交换等功能。

通过协议多路复用、分段重组操作和组概念,向高层提供面向连接的和无连接的数据服务,L2CAP还屏蔽了低层传输协议中的很多特性，使得高层协议应用开发人员可以不必了解基层协议而进行开发。架构如下:

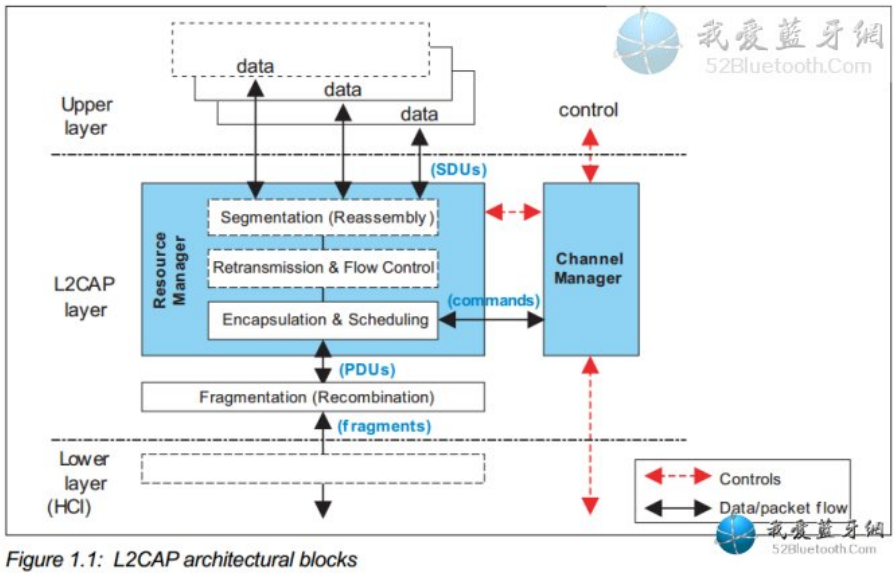


Figure 1.1: L2CAP architectural blocks

3）SDP（SERVICE DISCOVERY PROTOCOL）：服务发现协议，服务发现协议(SDP)为应用程序提供了一种方法来发现哪些服务可用，并确定这些可用服务的特征

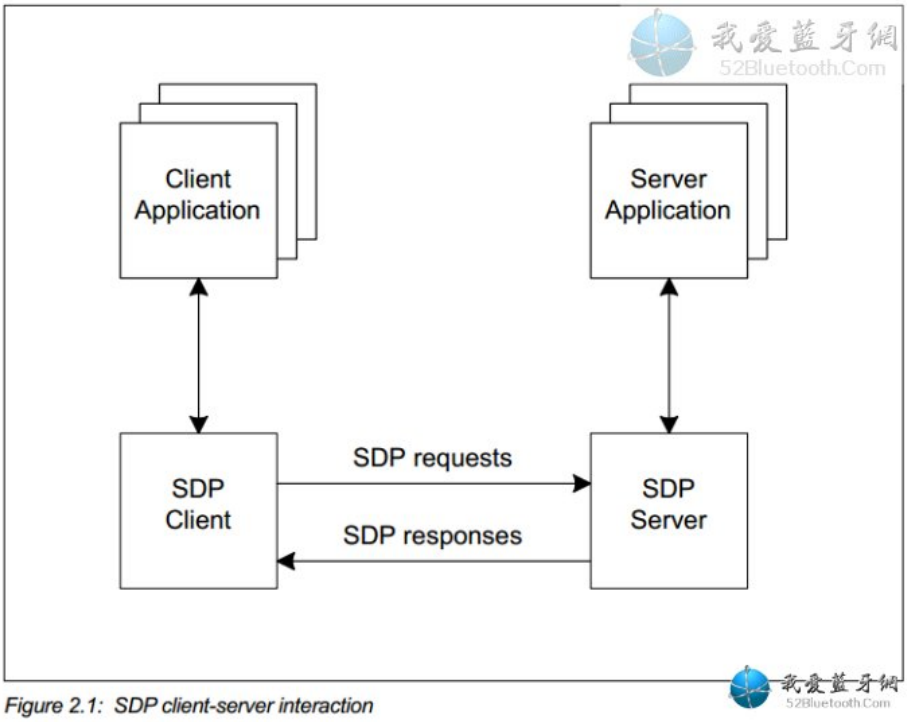


Figure 2.1: SDP client-server interaction

4）RFCOMM（Serial Port Emulation）：串口仿真协议，上层协议蓝牙电话，蓝牙透传SPP等协议都是直接走的RFCOMM

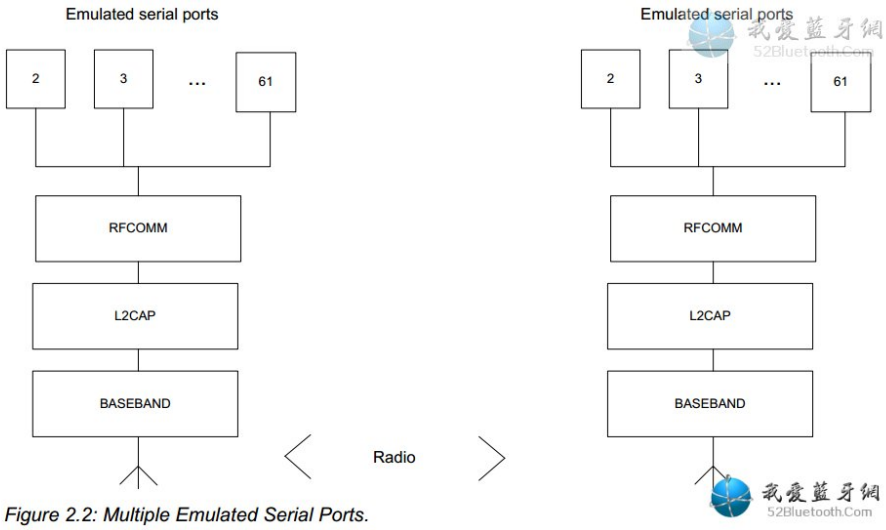


Figure 2.2: Multiple Emulated Serial Ports.

5）OBEX：对象交换协议，蓝牙电话本，蓝牙短信，文件传输等协议都是走的OBEX

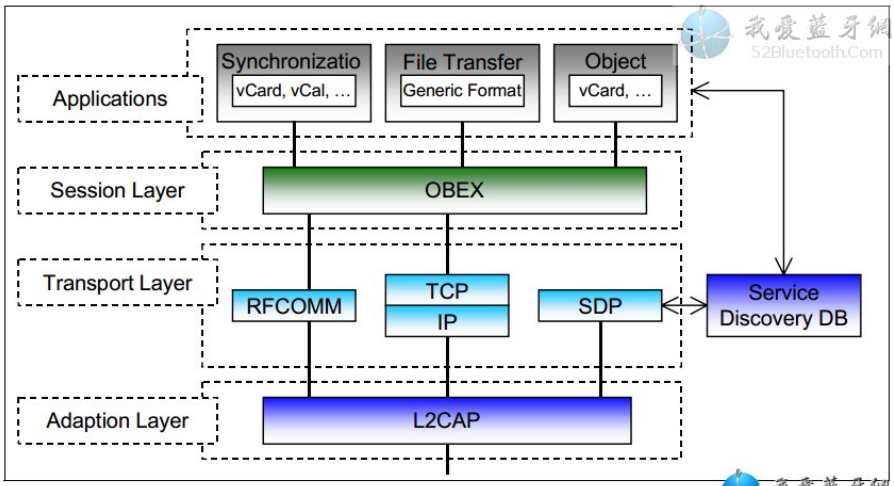


Figure 1.1: Part of Bluetooth Protocol Hierarchy

6）HFP（Hands-Free）：蓝牙免提协议

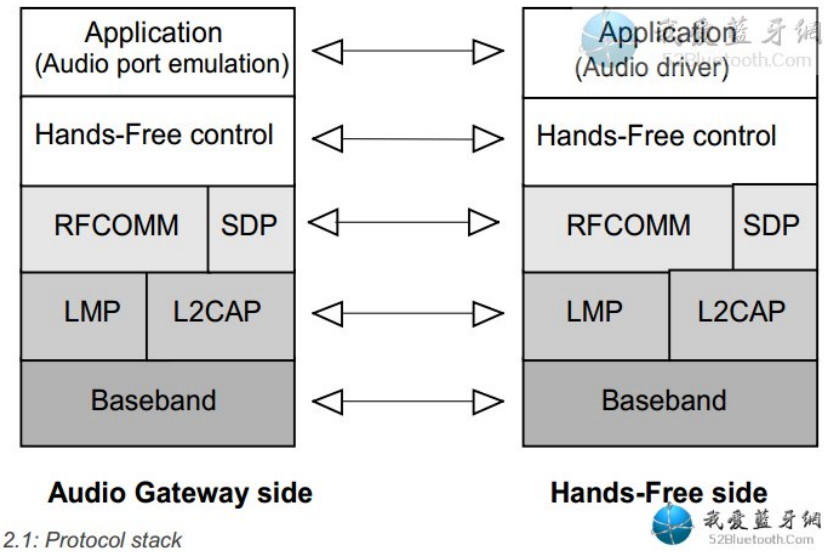


Figure 2.1: Protocol stack

一共分为两个角色：AG跟HF，举一个例子你一下就会懂，蓝牙耳机跟手机连接，那么手机的角色就是AG，蓝牙耳机的角色是HF

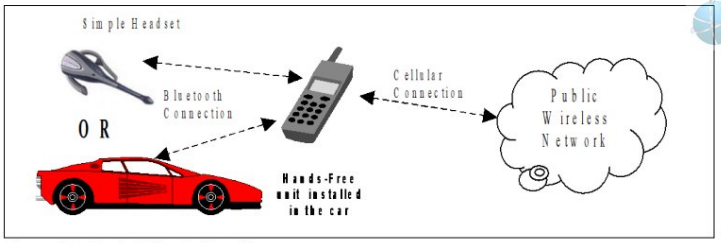


Figure 2.2: Typical Hands-Free Use

The following roles are defined for this profile:

Audio Gateway (AG) – This is the device that is the gateway of the audio, both for input and output. Typical devices acting as Audio Gateways are cellular phones.

Hands-Free unit (HF) – This is the device acting as the Audio Gateway's remote audio input/output mechanism. It also provides some remote control means.

7）HSP：蓝牙耳机协议，最开始的蓝牙耳机协议，目前已经没有产品在用这个了吧，至少我没有看到了。算是一个简化版的HFP。

8）SPP（SERIAL PORT PROFILE）：蓝牙串口协议，架构如下：

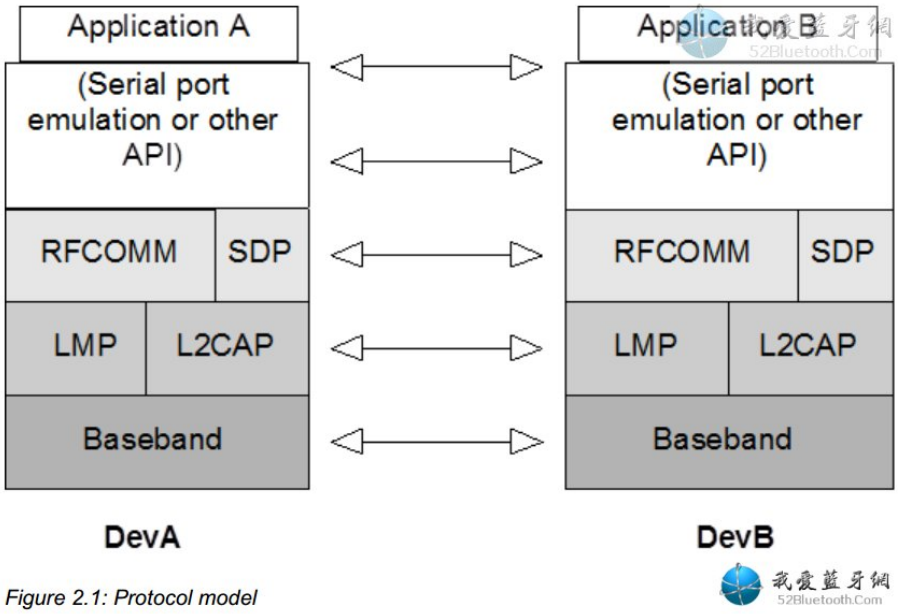


Figure 2.1: Protocol model

角色没有啥新奇古怪的，就是Device A/Device B

Figure 2.2shows one possible configuration of devices for this profile:

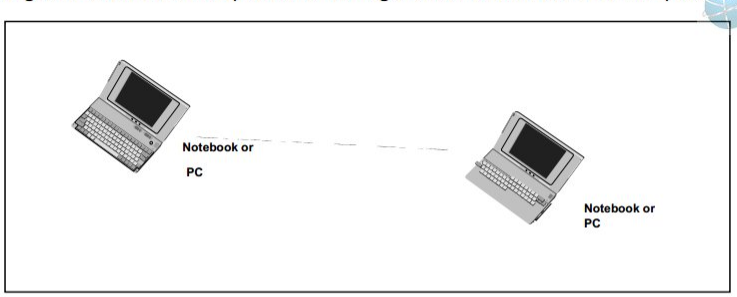


Figure 2.2: Serial Port profile, example with two notebooks.

The following roles are defined for this profile:

Device A (DevA) – This is the device that takes initiative to form a connection to another device (DevA is the *Initiator* according to Section 2.2 in GAP [9]).

Device B (DevB) – This is the device that waits for another device to take initiative to connect. (DevB is the *Acceptor* according to Section 2.2 in GAP [9]).

9）IAP：苹果的特有协议，分为IAP1/IAP2，一般做Carplay或者iPod功能的人肯定接触过这块，有需要这块的私下联系我

10）PBAP（Phone Book Access）：蓝牙电话本访问协议,架构如下：

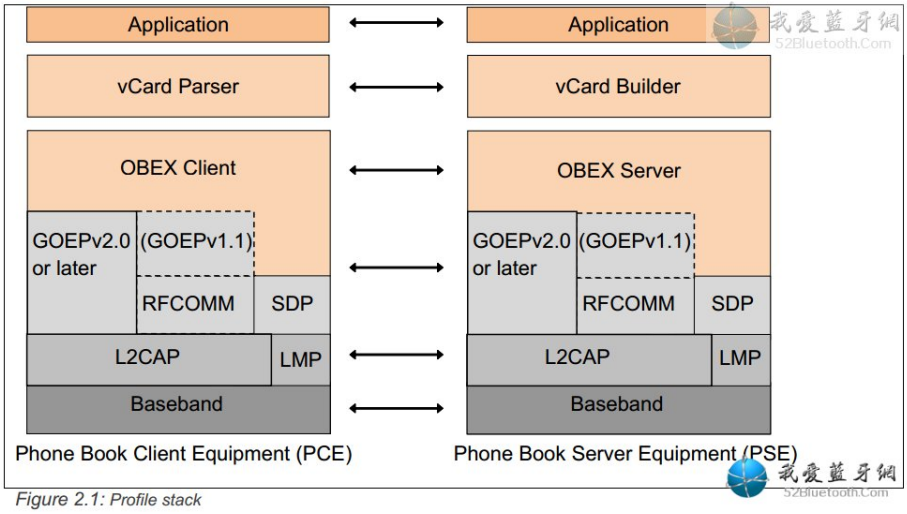


Figure 2.1: Profile stack

此部分尤其注意，PBAP在V1.2跟V1.1架构变化很大，V1.1 PBAP直接走的RFCOMM，在V1.2的时候如果GOEP是V2.0版本，那么PBAP是直接走的L2CAP，并且是L2CAP ERTM mode，不是basic mode.

角色如下：同样举例说明，我们车载蓝牙跟手机连接，车载蓝牙下载手机的电话本，那么手机的角色就是PSE，车载蓝牙就是PCE，多嘴提一句，我刚进公司的时候第一个协议是PBAP，所以对PBAP有额外的亲切感。

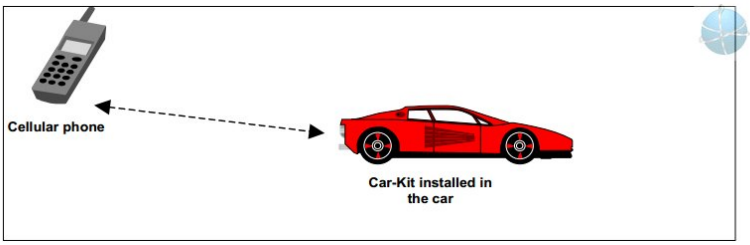


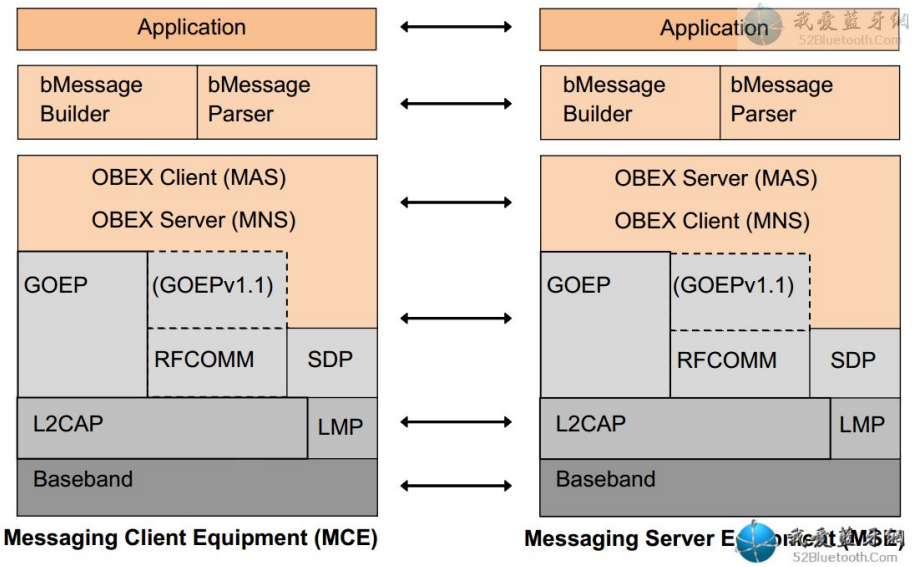
Figure 2.2: Phone Book Access Profile applied to the Hands-Free use case

The following roles are defined for this profile:

Phone Book Server Equipment (PSE) – This is the device that contains the source phone book objects.

Phone Book Client Equipment (PCE) – This is the device that retrieves phone book objects from the Server Equipment.

11）MAP（MESSAGE ACCESS PROFILE）：蓝牙短信访问协议，架构如下：



MAP跟PBAP很像，都是在V1.2的时候架构有变化，V1.1 MAP直接走的RFCOMM，在V1.2的时候如果GOEP是V2.0版本，那么MAP是直接走的L2CAP，并且是L2CAP ERTM mode，不是basic mode.

角色如下：

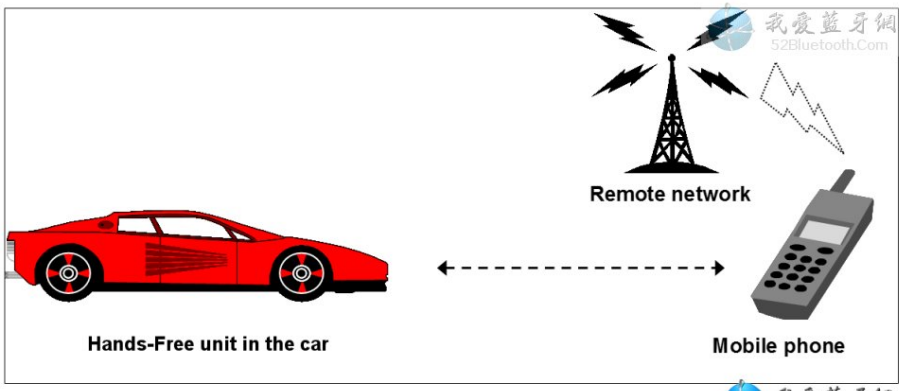
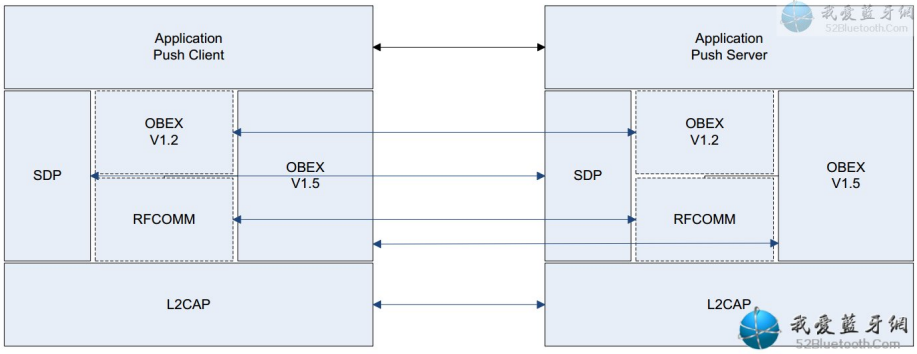


Figure 2.2: Message Access Profile applied to the Hands-Free use case

- **Message Server Equipment (MSE)** – is the device that provides the message repository engine (i.e., has the ability to provide a client unit with messages that are stored in this device and notifications of changes in its message repository).
- **Message Client Equipment (MCE)** – is the device that uses the message repository engine of the MSE for browsing and displaying existing messages and to upload messages created on the MCE to the MSE.

12）OPP（OBJECT PUSH PROFILE）：对象推送协议，架构如下



角色如下：

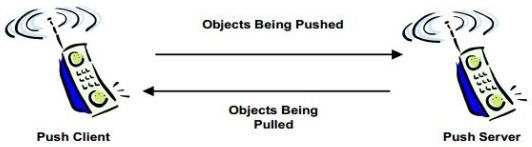


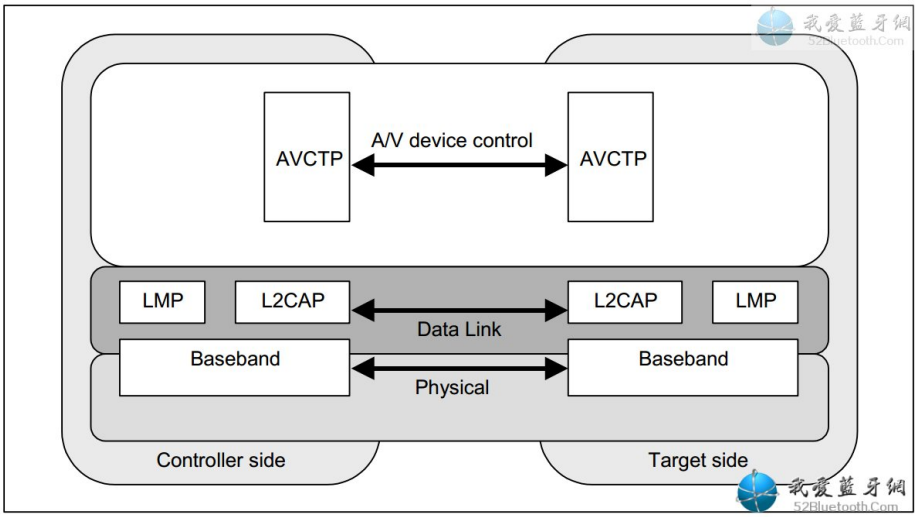
Figure 2.2: Push and Pull Example between Two Mobile Phones

The following roles are defined for this profile:

Push Server – This device provides an object exchange server. In addition to the interoperability requirements defined in this profile, the Push Server shall comply with the interoperability requirements for the server of the GOEP if not defined in the contrary.

Push Client – This device pushes and pulls objects to and from the Push Server. In addition to the interoperability requirements defined in this profile, the Push client shall also comply with the interoperability requirements for the client of the GOEP, if not defined to the contrary.

13）AVCTP（AUDIO/VIDEO CONTROL TRANSPORT PROTOCOL）：音视频控制传输协议，是AVRCP的地方，架构如下：



14）AVDTP（AUDIO/VIDEO DISTRIBUTION TRANSPORT PROTOCOL）：音视频分布传输协议，是A2DP的底层，架构如下

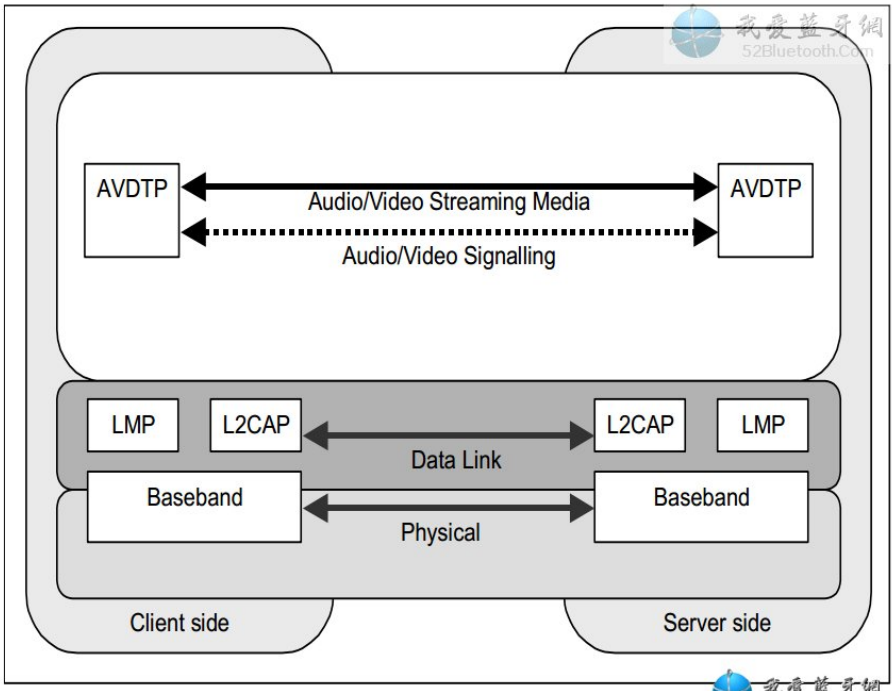
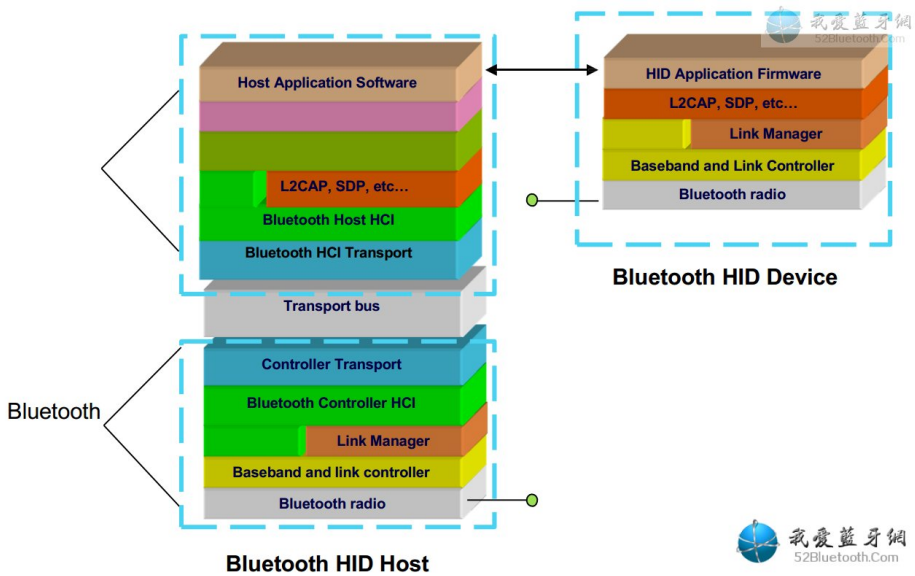


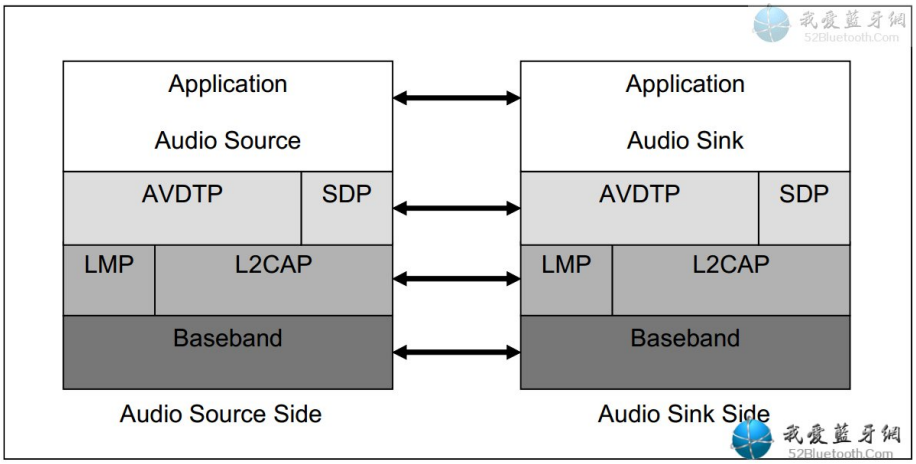
Figure 1.1: AVDTP and Bluetooth Protocol Stack

15）HID（HUMAN INTERFACE DEVICE）：人机接口协议，架构如下：

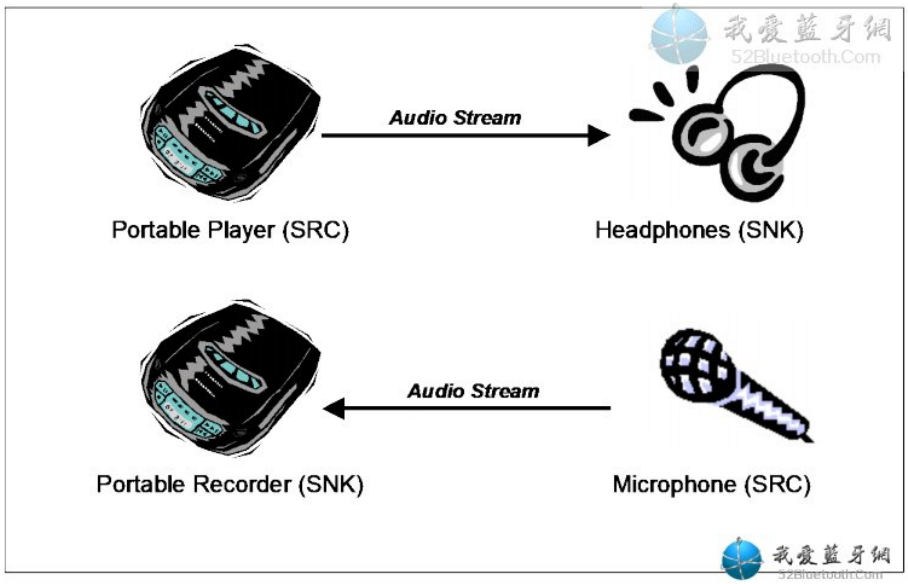


HID还是有很多广泛的用途的，比如蓝牙鼠标，蓝牙键盘，蓝牙自拍杆，蓝牙手柄等，学好HID还是能做很多产品的

16）A2DP（Advanced Audio Distribution）：蓝牙音乐协议，架构如下：



角色如下：举一个例子说明，还是拿蓝牙耳机跟手机连接，手机传输音乐给蓝牙耳机，那么手机就是A2DP source端，蓝牙耳机是A2DP sink端

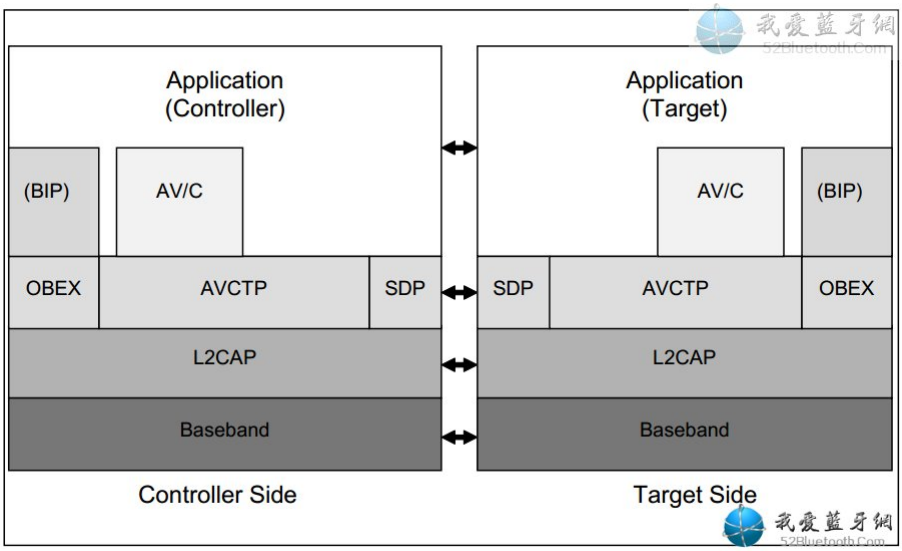


Source (SRC) – A device is the **SRC** when it acts as a source of a digital audio stream that is delivered to the **SNK** of the piconet.

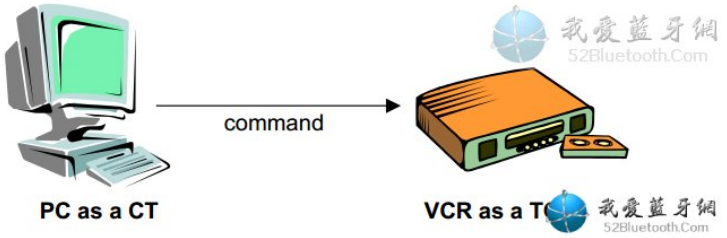
Sink (SNK) – A device is the **SNK** when it acts as a sink of a digital audio stream delivered from the **SRC** on the same piconet.

Examples of configurations illustrating the roles for this profile are depicted in Figure 2.2.

17）AVRCP（AUDIO/VIDEO REMOTE CONTROL PROFILE）：蓝牙音乐控制协议



角色如下:举例说明，哈哈，继续拿手机跟蓝牙耳机举例（前提是蓝牙耳机有上一首下一首的功能），那么蓝牙耳机就是controller(CT),手机就是target(TG)



- The controller (CT) is a device that initiates a transaction by sending a command frame to a target. Examples for CT are a personal computer, a PDA, a mobile phone, a remote controller or an AV device (such as an in car system, headphone, player/recorder, timer, tuner, monitor etc.).
- The target (TG) is a device that receives a command frame and accordingly generates a response frame. Examples for TG are an audio player/recorder, a video player/recorder, a TV, a tuner, an amplifier or a headphone.

18）ATT：蓝牙属性协议

19）GATT：蓝牙通用属性协议

20）SM: 蓝牙安全管理协议

后续的博客会慢慢每个都更新，敬请关注，感谢观看