

青风带你玩蓝牙 nRF52832 系列教程.....	2
-----作者: 青风.....	2
作者: 青风.....	3
出品论坛: www.qfv8.com	3
淘宝店: http://qfv5.taobao.com	3
QQ 技术群: 346518370.....	3
硬件平台: 青云 QY-nRF52832 开发板.....	3
2.10 蓝牙数据包和广播包分析:	3
10.1 nrf52832 蓝牙通信包解析:	3
10.2 广播包抓取:	3

青风带你玩蓝牙 nRF52832 系列教程

-----作者: 青风

出品论坛: www.qfv8.com 青风电子社区



作者: 青风

出品论坛: www.qfv8.com

淘宝店: <http://qfv5.taobao.com>

QQ 技术群: 346518370

硬件平台: 青云 QY-nRF52832 开发板

2.10 蓝牙数据包和广播包分析:

在使用 nrf52832 开发 BLE 应用中,有必要结合 nrf52832的例程向工程师介绍 BLE的广播和建立连接时通信数据包的相关内容。

10.1 nrf52832 蓝牙通信包解析:

使用 Packet Sniffer 软件,配合青风的usb dongle 抓包器进行抓包,首先我们分析一下BLE 例程下的广播包的构成,使得大家能够直观的认清蓝牙广播包的构成,同时对蓝牙的GAP 有一个 深入的认识。

首先大家需要下载任何一个BLE 蓝牙应用程序到我们的蓝牙开发板中,下列开发板都可以:

- 1.硬件开发板: (1): 青云nrf52832 开发板
- 2.软件: Packet Sniffer, TI 官方开发的蓝牙抓包软件,非常方便抓包,值得推荐:



10.2 广播包抓取:

打开软件点击下图三角号后,就可以开始抓包:



抓到的包显示如下:

P.nbr.	Time (us)	Channel	Access Address	Adv PDU Type	Adv PDU Header				AdvA
88	+42205 =3127096	0x25	0x8E89BED6	ADV_IND	Type	TxAdd	RxAdd	PDU-Length	0xE11D2123261C
1	2	3	4	5	6	7			

AdvData										CRC	RSSI (dBm)	FCS
0E	09	4C	65	64	42	75	74	6F	6E	0x3161DF	-42	OK
44	65	6D	6F	03	19	34	12	02	01	05		
8										9	10	11

我们把广播包按照不同组成标记为1 到11 个部分, 如上图所示。下面来一一分析:

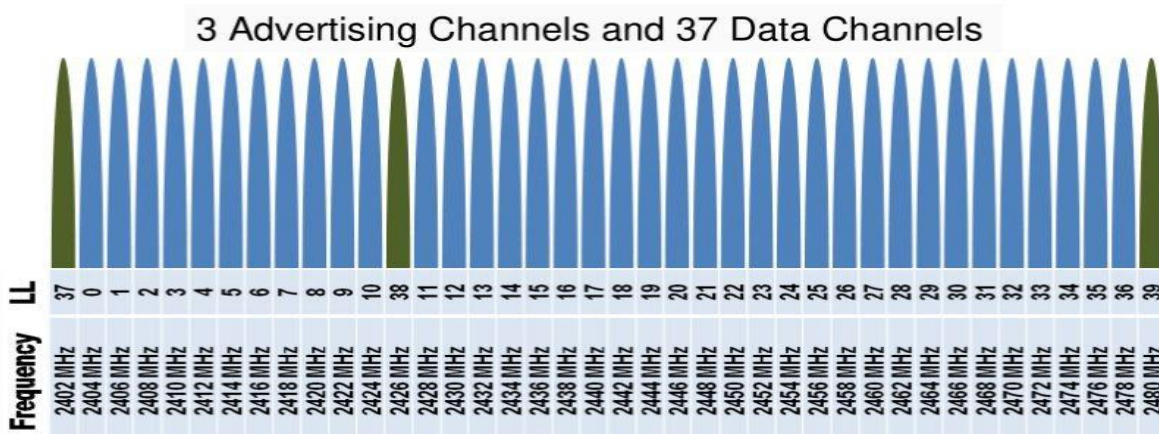
第 1 部分: P.nbr.指的是Packet Sniffer 抓的包的序列个数, 这个依次计数, 从1 开始, 依次计数。

第 2 部分: Time(us)指的是抓取包的时间延迟。

第 3 部分: 广播包表示的是**广播信道**, 数据包表示的是数据信道。数据链路层的 2 种道:

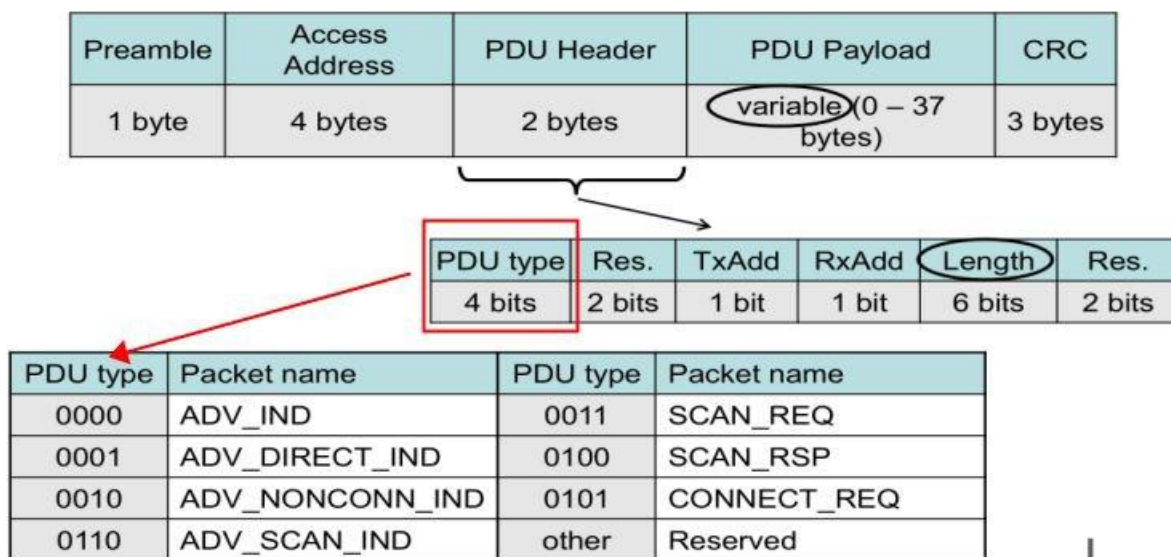
1)广播信道: 提供给还没有建立连接的蓝牙设备提供发射广播、扫描、建立连接的信道。BLE 有3 个广播信道:37、 38、 39,在每一个广播事件发生时, advertiser 分别在这3 个信道上各发送一 次广播信号。传统蓝牙的广播信道有 16-32 个, 而 BLE 只有3 个, 这就是为什么 BLE 的广播时间 比较短的原因。图中0x25 为37 信道。

2)数据信道: 提供给已经建立蓝牙连接的 master 和 slave 端提供可靠的数据通信信道。BLE 规 定, 数据信道有37 个。为加强通讯的可靠性, 避开干扰, BLE 设备通过自适应跳频的方式在这37 个信道上传。



第 4 部分: Access Address: 0xBE89BED6。所有BLE 设备的广播帧都是使用这个地址,

第 5 部分: 广播类型



PDU type 广播类型: 例子为普通可连接广播。

A) 以 ADV_开头的帧表示该帧是广播帧,是由 advertiser(蓝牙外设)发出的,它们有 4 种类型, 分别用在不同的蓝牙设备上。

- ADV_IND:通用的可以建立连接的广播, nrf52832 通常发送这种广播。
- ADV_DIRECT_IND:快速广播。广播最长发射时间为 1.28S。
- ADV_NONCONN_IND:不能建立连接的广播信号, ibeacon 发的就是这种类型的广播

B) ADV_SCAN_IND 为扫描帧,是由 scanner(手机、平板、PC)发出的。

C) ADV_SCAN_REQ 为扫描请求帧,是由 scanner(手机、平板、PC)发出的。只在scanner 想从 advertiser 获取更多的广播数据的时候才由 scanner 发出。相应的,当ADV_SCAN_REQ 被发出以后, advertiser

会以 SCAN_RSP 作为回应。

D) SCAN_RSP 为 ADV_SCAN_REQ 的回应。

E) CONNECT_REQ 为 scanner 向 advertiser。

第 6 部分: 广播 PDU 头:

Adv PDU Header			
Type	TxAdd	RxAdd	PDU-Length
0	1	0	21

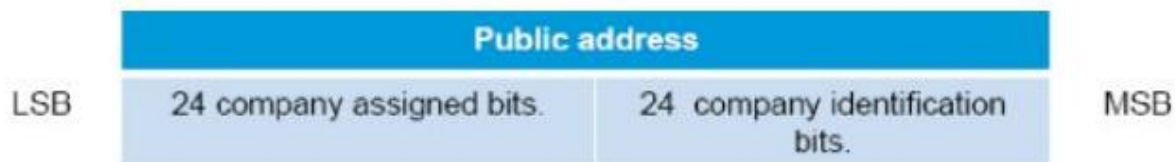
•TxAdd、 RxAdd 用来表示发送该广播帧的蓝牙设备的蓝牙地址类型。 1 表示 random address , 0 表示 public address 蓝牙地址的种类。

蓝牙协议规定,任何一个蓝牙设备必须拥有一个唯一的 48 bit 的地址,用以标识标识身份。而且,在广播的时候必须要把蓝牙地址广播出去。蓝牙地址有以下的四种:

1)Public address.

Public address 是公司通过IEEE 申请获得的称为 OUI(Organizationally Unique

Identifier)这个地址是固定的地址,全球唯一的,不可以修改。



Public address 的 24 bit LSB 用来表示公司名;另外24 bit 的 MSB 用来分配给不同的产品类型。

2) Random Static address

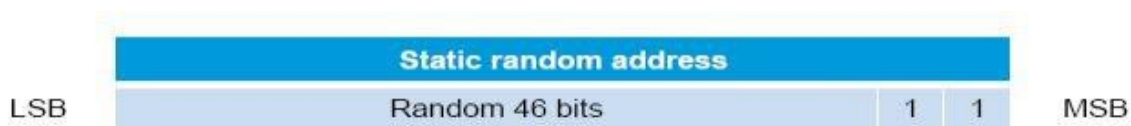
Random Static address 是设备在上电的时候随机生成或者是芯片厂家在生产芯片的时候随机烧录的不重复的 48 bit 的蓝牙地址。nrf52832 的蓝牙地址属于后者。该地址存放在 FICR 里面,用户不可以修改。

6.2.13 DEVICEADDR[0]

Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID (Field ID)	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Value after erase	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
ID	RW	Field	Value ID	Value	Description																											
A	R	ADDR			Device address bit 31-0.																											

6.2.14 DEVICEADDR[1]

Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID (Field ID)	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Value after erase	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
ID	RW	Field	Value ID	Value	Description																											
A	R	ADDR			Device address bit 47-32.																											



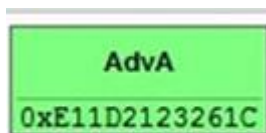
Random Static address 的最高位的后 2 bit 必须要为 1。


3) Private Non-Resolvable address和Private Resolvable address

这 2 种地址不常用,这里就不介绍了。

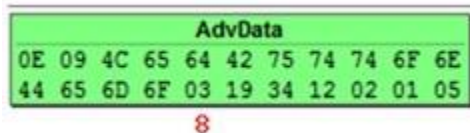
nrf52832 采用的是 Random Static address,在启动的时候协议栈从 FICR 里面读取作为设备的蓝牙地址。如果用户需要使 Public address,则需要使用 `sd_ble_gap_address_set()` 这个函数重新设定蓝牙地址。本例广播包是从设备 TxAdd 为 1,也就是设置为了 Random Static address。

•Length 表示后面 PDU Payload 的大小。



第7 部分： 蓝牙设备地址，前面已经谈过该地址的分类，也可以尝试自己读取一下 FICR 里面的蓝牙地址跟抓包得到的对比一下，看是否是一样的，后面教程我们会探讨如何修改该地址。

第8 部分：广播包内容：



PDU Payload 中 “0E 09 4C 65 64 42 75 74 74 6F 6E 44 65 6D 6F”为一组。0E 为本字段长度；09 为 LOCAL NAME，也就是蓝牙设备名称。通过查找 ASCII 码表，我们可以得到它的蓝牙设备名称为 “LedButtonDemo”。读者可以在 BLE 实验二：BLE LED 灯读写控制的项目工程中中找到其定义，如下所示。读者应该知道以后是在这个地方改蓝牙设备的名称了。

```
4C->L 65->e 64-> d 42-> B 75->u 74-> t 74->t 6F-> o 6E-> n 44->D 65->e

#define CONNECTED_LED_PIN_NO LED_1 /**< Is on when device is connected */
#define LEDBUTTON_LED_PIN_NO LED_0
#define LEDBUTTON_BUTTON_PIN_NO BUTTON_1
#define DEVICE_NAME "LedButtonDemo" /**< Name of device */
#define APP_ADV_INTERVAL 64 /**< The advertising interval in units of 0.625 ms */
#define APP_ADV_TIMEOUT_IN_SECONDS 180 /**< The advertising timeout in seconds */
#define APP_TIMER_PRESCALER 0 /**< Value of the timer prescaler */
#define APP_TIMER_MAX_TIMERS 2 /**< Maximum number of timers */
#define APP_TIMER_OP_QUEUE_SIZE 4 /**< Size of the timer operation queue */
```

6D->m 6F->o

“03193412”为一组。03 为本字段长度;19 为 APPEARANCE，因为蓝牙发送数据是低位在前，所以 “3412”其实是”1234”，0x1234 转换成十进制就是 4660。APPEARANCE 这个AD TYPE 是新添加的TYPE,所以在CORE4.0 核心协议里面是找不到的，这个在代码里认为是UNKNOWN

```

#define BLE_APPEARANCE_UNKNOWN 0 /**< Unknown. */
#define BLE_APPEARANCE_GENERIC_PHONE 64 /**< Generic Phone. */
#define BLE_APPEARANCE_GENERIC_COMPUTER 128 /**< Generic Computer. */
#define BLE_APPEARANCE_GENERIC_WATCH 192 /**< Generic Watch. */
#define BLE_APPEARANCE_WATCH_SPORTS_WATCH 193 /**< Watch: Sports Watch. */
#define BLE_APPEARANCE_GENERIC_CLOCK 256 /**< Generic Clock. */
#define BLE_APPEARANCE_GENERIC_DISPLAY 320 /**< Generic Display. */
#define BLE_APPEARANCE_GENERIC_REMOTE_CONTROL 384 /**< Generic Remote Control. */
#define BLE_APPEARANCE_GENERIC_EYE_GLASSES 448 /**< Generic Eye-glasses. */
#define BLE_APPEARANCE_GENERIC_TAG 512 /**< Generic Tag. */
#define BLE_APPEARANCE_GENERIC_KEYRING 576 /**< Generic Keyring. */
#define BLE_APPEARANCE_GENERIC_MEDIA_PLAYER 640 /**< Generic Media Player. */
#define BLE_APPEARANCE_GENERIC_BARCODE_SCANNER 704 /**< Generic Barcode Scanner. */
#define BLE_APPEARANCE_GENERIC_THERMOMETER 768 /**< Generic Thermometer. */
#define BLE_APPEARANCE_THERMOMETER_EAR 769 /**< Thermometer: Ear. */
#define BLE_APPEARANCE_GENERIC_HEART_RATE_SENSOR 832 /**< Generic Heart rate Sensor. */
#define BLE_APPEARANCE_HEART_RATE_SENSOR_HEART_RATE_BELT 833 /**< Heart Rate Sensor: Heart Rate Belt. */
#define BLE_APPEARANCE_GENERIC_BLOOD_PRESSURE 896 /**< Generic Blood Pressure. */
#define BLE_APPEARANCE_BLOOD_PRESSURE_ARM 897 /**< Blood Pressure: Arm. */
#define BLE_APPEARANCE_BLOOD_PRESSURE_WRIST 898 /**< Blood Pressure: Wrist. */
#define BLE_APPEARANCE_GENERIC_HID 960 /**< Human Interface Device (HID). */
#define BLE_APPEARANCE_HID_KEYBOARD 961 /**< Keyboard (HID Subtype). */
#define BLE_APPEARANCE_HID_MOUSE 962 /**< Mouse (HID Subtype). */

```

“020105”为一组。02 为本字段长度;01 是 FLAGS,06 表示本设备只支持 BLE，不支持传统蓝牙。05 的取值其实为 04+01，看下图可以明白。

Value	Description	Bit	Information
0x01	Flags	0	LE Limited Discoverable Mode
		1	LE General Discoverable Mode
		2	BR/EDR Not Supported (i.e. bit 37 of LMP Extended Feature bits Page 0)
		3	Simultaneous LE and BR/EDR to Same Device Capable (Controller) (i.e. bit 49 of LMP Extended Feature bits Page 0)
		4	Simultaneous LE and BR/EDR to Same Device Capable (Host) (i.e. bit 66 of LMP Extended Feature bits Page 1)
		5..7	Reserved

第9 部分: CRC 验证, 常见的一种校验方法。

第10 部分: RSSI 蓝牙信号强度, 表示抓到的广播包的信号强度。

第11 部分: 广播包接收判断, 如果显示 OK, 表示成功发现广播包。

1.2 数据包抓取:

当开发板 一旦和主机连接上后, 到这一行, 抓包就不在显示了, 这个时候, 如上填入地址, 并选好信道号:



然后再重新复位从机, 主机重新连接, 这个时候不一定 SmartRF Packet Sniffer 就能显示到 连接后的数据包。如果不能连接上, 就试试把信道改成 38、39 等等, 多试试几次, 就会出现 下面图了。(下图表明抓取到了ble 的数据包)

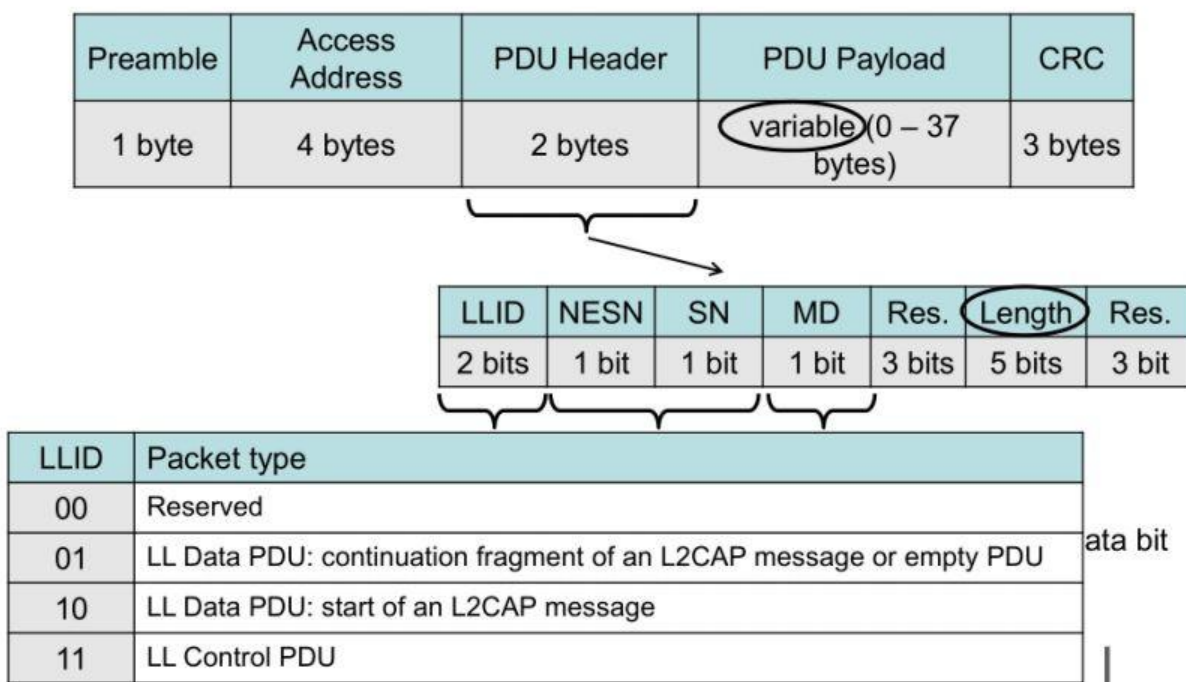
P.nbr.	Time (us)	Channel	Access Address	Direction	ACK Status	Data Type	Data Header	CRC	RSSI (dBm)	FCS
1117	+232 =16892462	0x19	0xE82E7A1E	S->M	OK	Empty PDU	LLID NESN SN MD PDU-Length 1 0 1 0 0	0x5A2767	-49	OK
1118	+48518 =16940980	0x23	0xE82E7A1E	M->S	OK	Empty PDU	LLID NESN SN MD PDU-Length 1 0 0 0 0	0x5A2AC1	-70	OK
1	2	3	4	5	6	7	8	9	10	11

前 3 个就不说明了, 和广播包的一样。第四个 Access Address 为数据接入地址, 接入地址由主设备来提供, 地址通过随机生成。但是也要遵循一定规律。不同与广播接入地址固定。

第5 个: 为连接方向, 是主机到从机 M→S, 还是从机到主机 S→M。

第7 个: 数据类型。Empty pdu 表示维持连接的报文, 也成为心跳报文。

第8 个: 报头, 如下图所示说明:



LLID 编码:

11: 链路层控制报文: 用于管理连接

10: 高层报文开始: 可用于一个完整报文

01: 高层报文延续

NESN:下一个预期序列号。SN: 序列号。MD:更多数据。PDU-Length:PDU 长度。没有数据的时候PDU 数据长度为0, 如果我们通信一下, 使用例子二: ble 的LED 读写, 读操作如下:

P.nbr.	Time (us)	Channel	Access Address	Direction	ACK Status	Data Type	Data Header	L2CAP Header	ATT_Read_Req	CRC	RSSI (dBm)	FCS
1104	+48518 =16599732	0x02	0xE82E7A1E	M→S	OK	L2CAP-S	LLID NESN SN MD PDU-Length 2 1 1 0 7	L2CAP-Length ChanId 0x0003 0x0004	Opcode AttHandle 0x0A 0x000E	0x903F38	-67	OK
1105	+288 =16600020	0x02	0xE82E7A1E	S→M	OK	Empty PDU	LLID NESN SN MD PDU-Length 1 0 1 0 0	CRC RSSI FCS 0x5A2767 -45 OK	读请求			
1106	+48462 =16648482	0x0C	0xE82E7A1E	M→S	OK	Empty PDU	LLID NESN SN MD PDU-Length 1 0 0 0 0	CRC RSSI FCS 0x5A2AC1 -66 OK				
1107	+232 =16648714	0x0C	0xE82E7A1E	S→M	OK	L2CAP-S	LLID NESN SN MD PDU-Length 2 1 0 0 6	L2CAP-Length ChanId 0x0002 0x0004	Opcode AttValue 0x0B 00	0xF9765B	-46	OK

00就是读取的数据

如果通过手机MCP 写入一个01，抓包如下：

Pnbr.	Time (us)	Channel	Access Address	Direction	ACK Status	Data Type	Data Header	L2CAP Header	ATT_Write_Req	CRC	RSSI (dBm)	FCS
1514	+48518 ~26739678	0x0A	0xE82E7A1E	M->S	OK	L2CAP-S	LLID NESN SN MD PDU-Length 2 0 0 0 8	L2CAP-Length ChanId 0x0004 0x0004	Opcode AttHandle AttValue 0x12 0x000E 01	0x9831B9	-72	OK
1515	+295 ~26739973	0x0A	0xE82E7A1E	S->M	OK	Empty PDU	LLID NESN SN MD PDU-Length 1 1 0 0 0	CRC RSSI (dBm) FCS 0x5A2C12 -46 OK				
1516	+48454 ~26788427	0x14	0xE82E7A1E	M->S	OK	Empty PDU	LLID NESN SN MD PDU-Length 1 1 1 0 0	CRC RSSI (dBm) FCS 0x5A21B4 -66 OK				
1517	+232 ~26788659	0x14	0xE82E7A1E	S->M	OK	L2CAP-S	LLID NESN SN MD PDU-Length 2 0 1 0 5	L2CAP-Length ChanId 0x0001 0x0004	ATT_Write_Rsp Opcode 0x13	0x7E9464	-48	OK

写反馈

写入1

写入成功

第9 个部分：CRC 验证，常见的一种校验方法。

第10 个部分：RSSI 蓝牙信号强度，表示抓到的数据包的信号强度。

第11 个部分：数据包接收判断，如果显示 OK，表示成功发现数据包。

在实际开发中，为了确定你的数据或者连接是否正确，抓包还是显得比较重要的，所以十分建议大家配一个抓包器进行开发。