LLSync 蓝牙设备接入协议

未经授权，请勿扩散

修订记录

修订日期	修订版本	修改描述	作者
20200817	V0.1.0	发布第一版 draft 规范 协议版本号为 0	willssong esma markyyao zekwang
20200910	V1.0.0	event 数据增加分片功能	esma
20200917	V1.1.0	协议版本号变更 增加数据分片	esma
20201102	V1.2.0	协议版本号变更 支持绑定、解绑数据分片	zekwang
20201119	V1.3.0	协议版本号变更 增加 OTA 功能	esma

修订流程

对于本文档中任何内容的增删改以及相关其它文档的创建，都应该知会作者或者相关接口人。

接口人

本文档中的任何信息都应该被仔细的阅读。如果有任何疑虑，意见或问题，请直接联系下表中的接口人。

姓名	邮箱	电话	组织
willssong	*willssong@tencent.com*		*腾讯云物联网产品中心*
esma	*esma@tencent.com*		*腾讯云物联网产品中心*
markyyao	*markyyao@tencent.com*		*腾讯云物联网产品中心*
zekwang	*zekwang@tencent.com*		*腾讯云物联网产品中心*

缩略语清单

缩略语	英文全名	中文解释
LLSync		*腾讯连连 Sync 协议*
BLE	*Bluetooth Low Energy*	*低功耗蓝牙*
LLDevice		*蓝牙 Sync 设备管理属性*
LLData		*蓝牙 Sync 数据属性*
LLEvent		*蓝牙 Sync 事件属性*
LLOTA		*蓝牙 Sync OTA 属性*

1. 引言

1.1 背景

腾讯连连是腾讯云面向物联网行业提供的一整套 C to B 开放平台服务，借助腾讯连连可以降低物联网产品的研发门槛以及加快研发速度，同时提供以微信小程序为载体的、面向消费者的应用入口，整合腾讯内部的品牌以及多项优势内容服务，助力万物互联时代真正到来。

BLE 设备在 IoT 设备中占比高，适用范围广，但由于 BLE 无法直接接入互联网，BLE 类设备上云比较困难，开发门槛较高。为解决此问题，腾讯云物联网开发平台制定 LLSync 协议帮助 BLE 设备快速简单的完成上云。

1.2 目的

本文档描述了 BLE 设备接入腾讯连连平台的流程标准，帮助开发者更好的理解 LLSync 协议。

2. 设备参数要求

参数项	要求
BLE ATT MTU	>= 23
BLE 协议	>= BLE 4.2

3. LLSync TLV 格式

腾讯云物联网为接入平台定义一套数据模板协议，将设备的接入形式通过 JSON 模板标准化。但是 BLE 设备受资源限制，无法承载 JSON 格式的数据交互，针对此定义了 TLV 格式的二进制数据包表示数据模板，最大程度的减少资源占用。如无特殊说明，本文所有数据均使用网络序传输。

LLSync TLV 二进制数据包中有用户数据、数据长度和数据类型，TLV 格式被广泛应用在 LLData 和 LLEvent 数据包中。

LLSync TLV 格式:

字段	type	length	value
长度	1 Byte	N Bytes	N Bytes
说明	Type 字段定义	可选	无

Type 字段说明：

Bit	7	6	5	4	3	2	1	0
字段	数据类型定义			ID 定义				

数据类型定义：

数据类型	值	数据长度	数据范围
布尔	0	1 Byte	0/1
整数	1	4 Bytes	-2^31 ~ 2^31 - 1
字符串	2	N(<= 2040)Bytes	用户自定义数据
浮点数	3	4 Bytes	1.2E-38 ~ 3.4E+38
枚举	4	2 Bytes	0 ~ 2^16 - 1
时间	5	4 Bytes	0 ~ 2^64 - 1

ID 含义说明：

1. 高 3 bit 表示用户数据的数据类型，当前支持**布尔、整数、字符串、浮点数、枚举、时间**等六种数据类型。
2. 低 5 bit 表示在不同的数据包中含义略有不同：
   - 属性(property)数据包，表示属性 ID(property id)。
   - 事件(event)数据包，表示事件的参数 ID(params id)。
   - 行为(action)数据包，表示行为的 input id 或 output id。

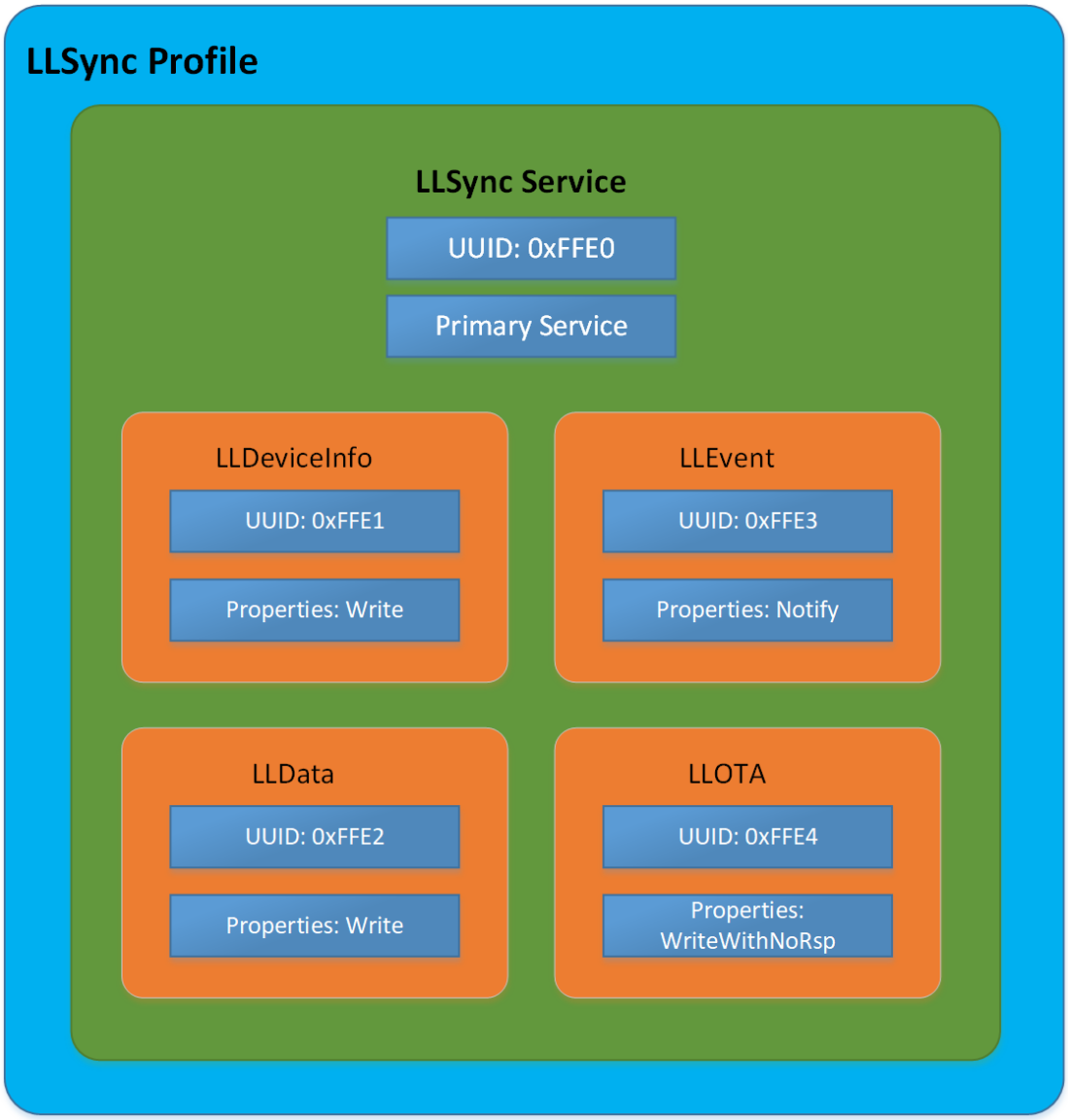
说明及限制：

1. *数据 ID 占据 5 bits，最大表示 2^5 - 1，即 LLSync 协议限制数据模版中(属性、事件、行为)的 **ID <= 63**。*
2. *浮点数指的是**单精度浮点数**。*
3. *除字符串外，其他数据类型数据长度固定，因此在 TLV 数据包中省略 length 字段。*
   示例如下：
   - *00 01 表示 id = 0， value = 1 的布尔数据。*
   - *41 00 05 68 65 6C 6C 6F 表示 id = 1，length = 5，value = hello 的字符串数据。*

   *即**只有字符串数据类型中含有 TLV 格式中的 length 字段**。*

4. LLSync Profile 定义

Profile 总架构如图：



LLSync Profile 包含 4 个 characteristics：
LLDeviceInfo：设备信息写入特征值，用于设备连接、绑定和身份确认。
LLData：数据模版操作特征值，用于通知设备端执行数据模版操作。
LLEvent：事件上报特征值，用于设备端向小程序上报数据。
LLOTA：升级数据特征值，用于控制设备进行版本更新。

LLSync 数据包最大长度为 **2048** 字节，包括数据包头和用户数据。同时支持数据分片，当数据包长度大于 ATT MTU 时，LLSync 协议会将数据分片发送，接收方收到分片数据后需要将数据组包后处理。

4.1 LLDeviceInfo

LLDeviceInfo 数据格式较为简单。由于历史原因，目前有两种格式不同的数据包，分别对应于 LLSync 协议版本号等于 0 和 LLSync 协议版本号大于 0 的 LLSync SDK。

LLDeviceInfo 数据格式(LLSync 协议版本 0)

字段	类型	数据
长度	1 Byte	N Bytes
说明	类型定义	数据格式定义

LLDeviceInfo 数据格式(LLSync 协议版本 1)

字段	类型	长度	数据
长度	1 Byte	2 Bytes	N Bytes
说明	类型定义	长度格式定义	数据格式定义

类型定义：

数据类型	类型值
时间同步	0
连接鉴权	1
绑定成功	2
绑定失败	3
解绑请求	4
连接成功	5
连接失败	6
解绑成功	7
解绑失败	8

数据格式定义：

数据格式字段由具体的数据类型定义。

数据类型	数据格式	说明
0	4 Bytes nonce + 4 Bytes Unix TS	请求绑定前向设备端发送计算签名所需信息
1	4 Bytes Unix Ts + 20 Bytes Hmac-sha1	使用 local psk 对 Ts 签名得到 Hmac-sha1
2	1 Byte bind result + 4 Bytes local psk + 8 Bytes bind string	绑定成功，小程序或网关生成 local psk 和 bind string，小程序或网关需要记录 local psk 和 bind string 与设备的对应关系
3	无	绑定失败
4	20 Bytes Hmac-sha1	使用 local psk 对“UnbindRequest”签名得到 Hmac-sha1
5	无	连接成功
6	无	连接失败
7	无	解绑成功
8	无	解绑失败

长度格式定义：

由于 ATT MTU 限制，当数据包长度大于 ATT MTU 时，需要对 LLSync 数据报文进行分片，分片标记记录在长度字段的 15 ～ 14 bit，数据长度记录在 13 ～ 0 bit，所以 LLSync 数据报文的最大长度为 2^13 - 1 字节。事实上，由于堆栈限制，LLSync 数据报文最大长度定义为 2^11 - 1 字节，已经可以满足绝大多数的使用场景。**长度格式定义适用于本文档中所有涉及到数据分片的 length 字段。**

Bit	15	14	13	12	11	10	…	1	0
说明	分片标记定义		数据长度						

分片标记定义：

分片值	00	01	10	11
说明	不分片	分片，首包	分片，中间包	分片，尾包

长度计算方式：分片标记 << 14 | 数据长度。

说明：

1. *数据分为 2 包时，只有分片标记为 01 和 11 的数据包。*
2. *数据长度指的是本包内有效载荷的长度。示例，数据包总长度 20 字节，包括 1 Byte Type 字段，2 Bytes Length 字段，17 Bytes 用户数据，数据长度应该填写十进制 17。*
3. *数据分片是将一段较长的用户数据切割为多个短数据后进行传输，每个短数据的传输格式都需要符合本文档中相应的报文格式。*

   示例：完整的连接鉴权信息共 24 字节，如下：

   0xA1, 0xA2, 0xA3, 0xA4, 0xB0, 0xB1, 0xB2, 0xB3, 0xB4, 0xB5, 0xB6, 0xB7, 0xB8, 0xB9, 0xBA, 0xBB, 0xBC, 0xBD, 0xBE, 0xBF, 0xC0, 0xC1, 0xC2, 0xC3。

   当 ATT MTU 为 23 字节时，单包用户数据长度最大为 20 字节。因此需要将连接鉴权信息分为 2 包发送，第一包数据如下：

   0x01, 0x40, 0x11, 0xA1, 0xA2, 0xA3, 0xA4, 0xB0, 0xB1, 0xB2, 0xB3, 0xB4, 0xB5,

   0xB6, 0xB7, 0xB8, 0xB9, 0xBA, 0xBB, 0xBC

   其中，0x01 表示连接鉴权信息，0x40，0x11 表示分片数据的第一包，数据长度 17 字节。

   第二包数据如下：

   0x01, 0xC0, 0x07, 0xBD, 0xBE, 0xBF, 0xC0, 0xC1, 0xC2, 0xC3

   其中，0x01 表示连接鉴权信息，0xC0，0x07 表示分片数据最后一包，数据长度 7 字节。

4.2 LLData

LLData 用于操作数据模版，不同的数据模版业务数据包格式相同，数据包中相同的字段含义略有不同。

LLData 格式定义：

字段	Fixed header	Payload
长度	1 Byte	N Bytes
说明	Fixed header 定义	Payload 定义

4.2.1 Fixed header 定义：

Bit	7	6	5	4	3	2	1	0
说明	数据模版类型定义		数据作用定义	ID 定义				

数据模版操作包括属性、事件、行为三大类，使用 bit 7 ～ 6 标记。

数据模版属性数据中 control 是向设备下发请求，reply 是应答设备，使用 bit 5 标记。

数据模版事件、行为需要指定 ID，使用 bit 4 ～0 标记。

数据模版类型定义：

字段	数值
property	0
event	1
action	2

数据作用定义：

字段	数值	说明
Request	0	向设备下发请求
Reply	1	应答设备

ID 定义：

数据模版类型	数据作用	ID	说明
0	0	0	无意义
	1	0	report_reply
		2	get_status_reply
1	1	event id	事件 id
2	0	action id	行为 id

说明：

event id/action id 在不得超过 63。

4.2.2 Payload 定义

Fixed header 由三部分组成，不同的组合对应不同的 Payload。

Payload 格式定义：

数据模版类型	数据作用类型	ID	Payload	说明
0	0	0	TLV 格式	见 6.4.2
	1	0	Reply Result 定义	见 6.4.1
		2	TLV 格式	见 6.4.3
1	1	event id	Reply Result 定义	见 6.4.4
2	0	action id	TLV 格式	见 6.4.5

Reply_Result 定义：

数值	说明
0	成功
1	失败
2	数据解析错误

Reply_Result 使用 1 Byte 表示，简要说明本次数据模版的操作结果。

4.3 LLEvent

LLEvent 用于设备主动上报报文，包括对 LLDeviceInfo、LLData 和 LLOTA 的回复。

LLEvent 格式定义：

字段	type	length	value
长度	1 byte	2 bytes	N bytes
说明	类型定义	长度定义	无

Event 类型定义：

字段	类型值	说明	Value
属性上报	0	数据模版中的 report	见 6.4.1
控制回复	1	数据模版中的 control_reply	见 6.4.2

获取最新信息	2	数据模版中的 get_status	见 6.4.3
事件上报	3	数据模版中的 event_post	见 6.4.4
行为响应	4	数据模版中的 action_reply	见 6.4.5
绑定鉴权信息	5	绑定后设备返回的信息	见 6.1
连接鉴权信息	6	连接后设备返回的信息	见 6.2
解绑鉴权信息	7	解绑后设备返回的信息	见 6.3
设备信息	8	上报 MTU 长度和协议版本	见 6.5
固件版本信息	9	上报固件版本信息	
升级请求回复	10	回复设备升级请求	
升级数据包回复	11	回复升级数据包	
升级校验结果回复	12	回复升级文件的校验结果	

4.4 LLOTA

LLOTA 用于对设备进行版本更新。

LLOTA 格式定义：

字段	type	length	value
长度	1 byte	1 byte	N bytes
说明	类型定义	无	无

OTA 类型定义：

类型定义	类型值
升级请求	0
升级数据包	1

说明：

1. *ota 报文中 length 字段长度 1 字节，表示 value 部分的长度。*
2. *ota 报文中 value 部分请参见 ota 章节。*

4.5 UUID 说明

LLSync Bluetooth Base UUID 为 00000000-65d0-4e20-b56a-e493541ba4e2。

按照 BLE 协议，16bit UUID 和 128bit UUID 转换关系为

128-bit value = 16-bit-value * 2^96 + BluetoothBaseUUID

即 0000xxxx-65d0-4e20-b56a-e493541ba4e2 中的 xxxx 替换为 16bit UUID，例如 Service 16bit UUID FFE0 转换为 128bit 的 UUID 为 0000ffe0-65d0-4e20-b56a-e493541ba4e2，Characteristic 的 UUID 的转换类似。

5. LLSync Advertisement 定义

自定义广播数据按照 BlueTooth 协议要求，添加到 0xFF Manufacturer Specific Data 的字段当中，company ID 使用 0xFEE7（Tencent Holdings Limited），0xFEE7 和 0xFEBA 均为腾讯申请的 Company ID。

广播包格式：

说明 / 状态	设备状态		设备标识		附加标识	
	长度	取值	长度	取值	长度	取值
未绑定	1	设备状态定义	6	MAC 地址	10	Product ID
绑定中	1		6	MAC 地址	10	Product ID
已绑定	1		8	设备标识计算	8	绑定标识计算

设备状态定义：

Bit	7	6	5	4	3	2	1	0
说明	协议版本				Reserved		绑定状态	

- 当前协议版本号为 1，不同协议版本号之间数据交互格式不同。
- 绑定状态为 0 表示未绑定，1 表示绑定中，2 表示已绑定。

设备标识计算：

Temp = md5sum(蓝牙设备的 product_id | 蓝牙设备的 device_name)
设备标识 Result = Temp 前 8 位 ^ Temp 后 8 位

比如：蓝牙设备的 ProductID 为 ABCDEFGHIJ，DeviceName 为 Dev01
那么 Temp = md5sum("ABCDEFGHIJDev01") = { 0x61, 0x2a, 0xf7, 0x9d, 0x50, 0x17, 0x93, 0x87, 0x2a, 0x4a, 0x97, 0xe8, 0xcb, 0xe4, 0x5a, 0x10}
Result 为 {0x4b, 0x60, 0x60, 0x75, 0x9b, 0xf3, 0xc9, 0x97}

绑定标识计算：

网关或小程序在绑定成功时提供，计算方式和说明 2 一致，使用网关的 product id 和 device name。

扫描看到的广播包如下图所示

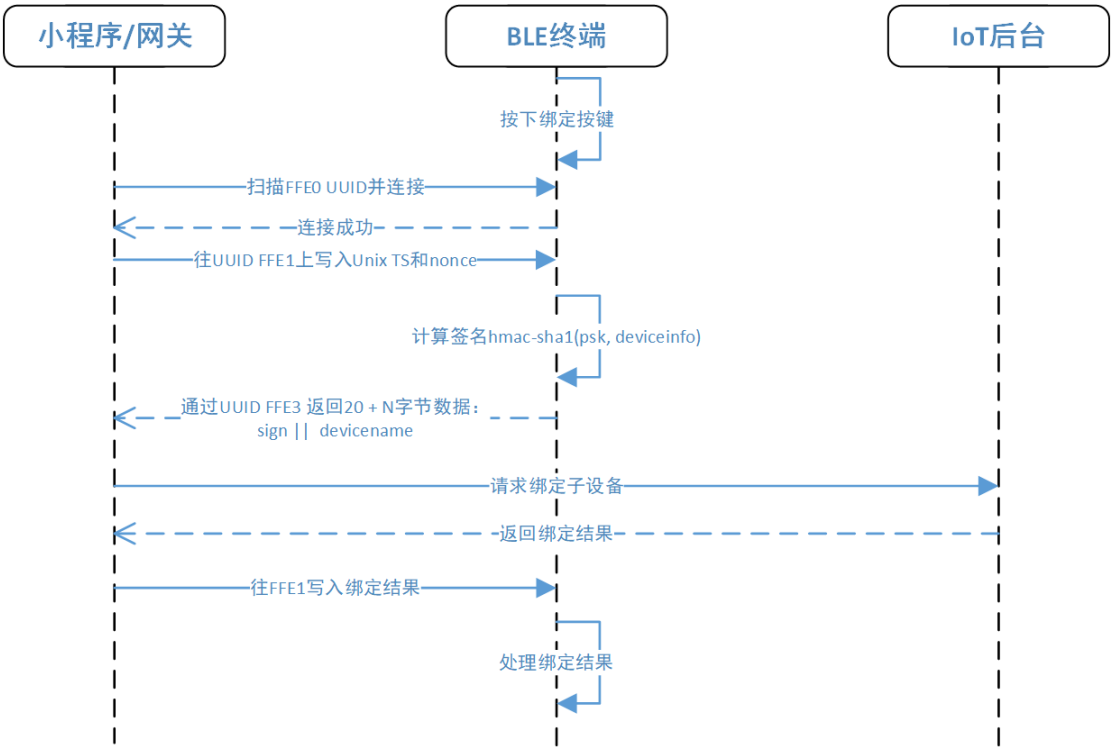
绑定状态说明：

状态	说明
未绑定	初始状态，可以选择不发送 BLE Advertisement 以省电
绑定中	按下绑定触发按键，在超时之前处于绑定中，超时前按照要求发送广播包
已绑定	正确完成绑定状态，需要持续发送广播包

6. BLE 通信数据流

6.1 子设备绑定

场景：BLE 终端尚未绑定需要进行绑定



1. 往 UUID FFE1 上写入 Unix TS 数据格式见下表

type	value	
	nonce	timestamp
00	4 Bytes nonce	4 Bytes timestamp

2. 设备验证签名后返回的 LLEvent 数据格式见下表

type	length	value	
		sign info	device name
05	2 Bytes value length	20 Bytes sign info	N Bytes device name

sign info 是通过设备的 psk 对设备信息签名得到，签名算法使用 hamc-sha1。

deviceinfo= product id + devicename + nonce + expiration time。

其中 expiration time = timestamp + 60。

说明：

计算签名时对于 timestamp，将其转换为字符串类型后再计算签名，避免大小端问题导致的签名错误。

示例：timestamp = 0x5f3279fa，转换为对应数值的字符串为“1597143546”。

3. 往 FFE1 写入绑定成功结果格式见下表

type	value		
	result	local psk	绑定标识符
02	02	4 Bytes Array local psk	8 Bytes

4. 往 FFE1 写入绑定失败结果格式见下表

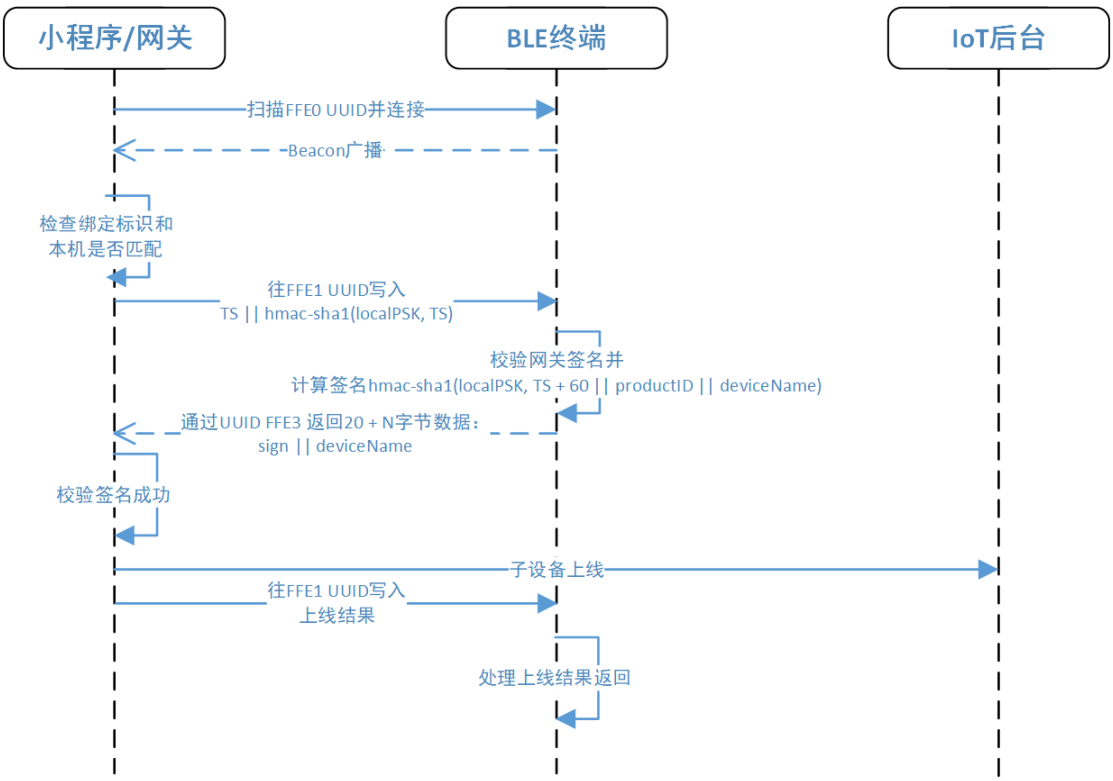
type	value
03	1 Bytes Reply_Result

说明：

1. *BLE 终端不会校验网关/小程序的身份，存在 BLE 终端被恶意绑定的可能，BLE 终端需要通过按键进入待绑定状态，2 分钟有效。*
2. *设备连接成功之后，不会再广播 beacon，小程序/网关无法再次扫描。*
3. *设备绑定成功之后，不会再响应 UUID FFE1 的读取请求。*
4. *如果绑定成功，需要在设备上存储 LocalPSK 用于后续的网关 + 子设备连接鉴权。*

6.2 子设备连接

场景：设备广播 Beacon 标识设备已绑定，需要重新连接



注：如果是小程序，写入上线结果认为是写入小程序和设备的连接结果。

1. 往 FFE1 写入签名信息数据格式见下表

type	value	
	timestamp	sign info
01	4 Bytes timestamp	20 Bytes sign info

sign info 是使用 local psk 对 timestamp 进行签名，算法选择 hmac-sha1。

2. 验签后返回的 LLEvent 信息数据格式如下表

type	length	value	
		sign info	device name
06	2 Bytes value length	20 Bytes sign info	N Bytes device name

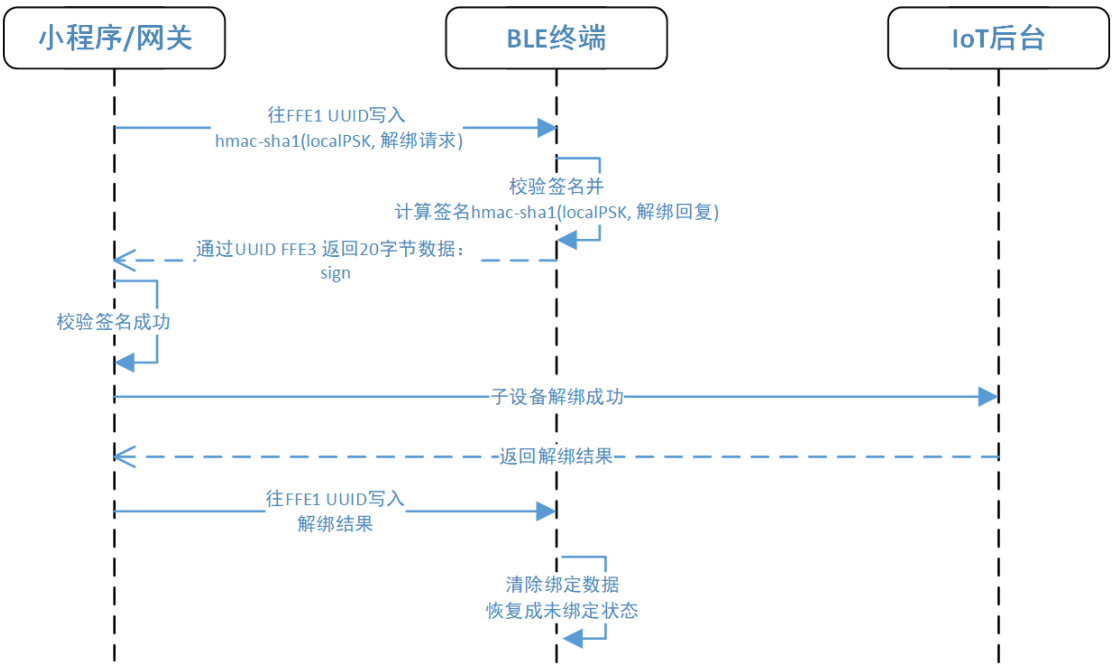
sign info 是使用 local psk 对设备信息进行签名，算法选择 hmac-sha1。设备信息包括 expiration time + product id + device name，其中 expiration time = timestamp + 60。

说明：

计算签名时对于 timestamp，将其转换为字符串类型后再计算签名，避免大小端问题导致的签名错误。示例：timestamp = 0x5f3279fa，转换为对应数值的字符串为“1597143546”。

6.3 子设备解绑

场景：子设备已经绑定且完成连接，小程序端请求解绑



1. 往 FFE1 写入解绑请求

type	value
04	20 Bytes sign info

2. 验签后返回的 LLEvent 信息数据格式如下表

type	length	value
07	2 Bytes value length	20 Bytes sign info

说明：

1. *解绑请求固定字符串 UnbindRequest，解绑回复固定字符串 UnbindResponse*
2. *使用 local psk 对固定字符串签名，算法选择 hmac-sha1*

3. 往 FFE1 写入解绑成功结果格式见下表

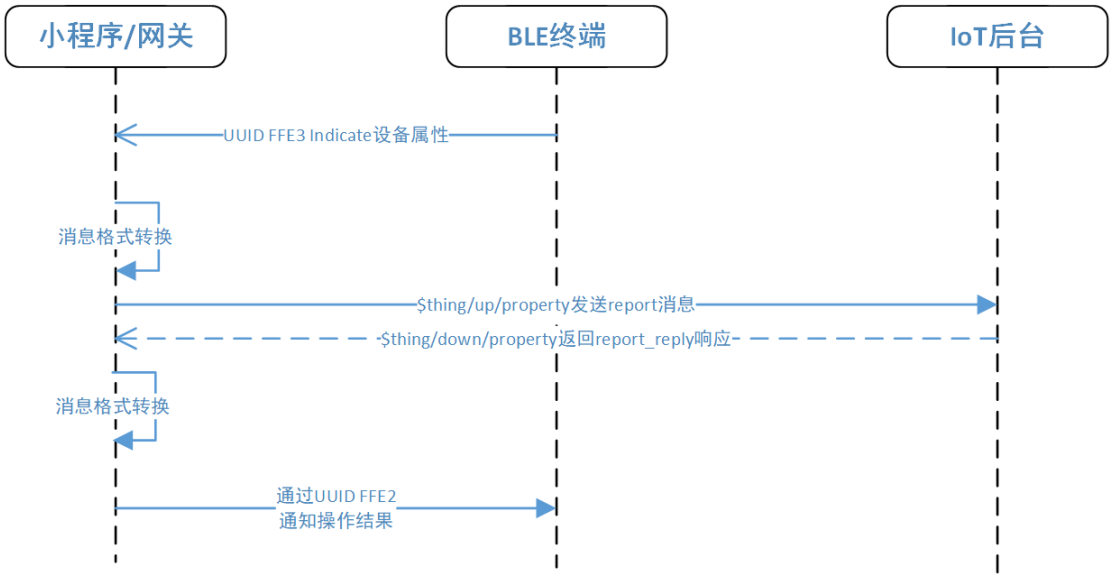
type	value
07	1 Byte Reply_Result

4. 往 FFE1 写入绑定失败结果格式见下表

type	value
08	1 Byte Reply_Result

6.4 数据模板协议交互

6.4.1 设备属性上报



1. 设备属性上报 LLEvent 数据格式，对应数据模版的 report 操作

type	length	property value
00	2 Bytes value length	tlv 数据

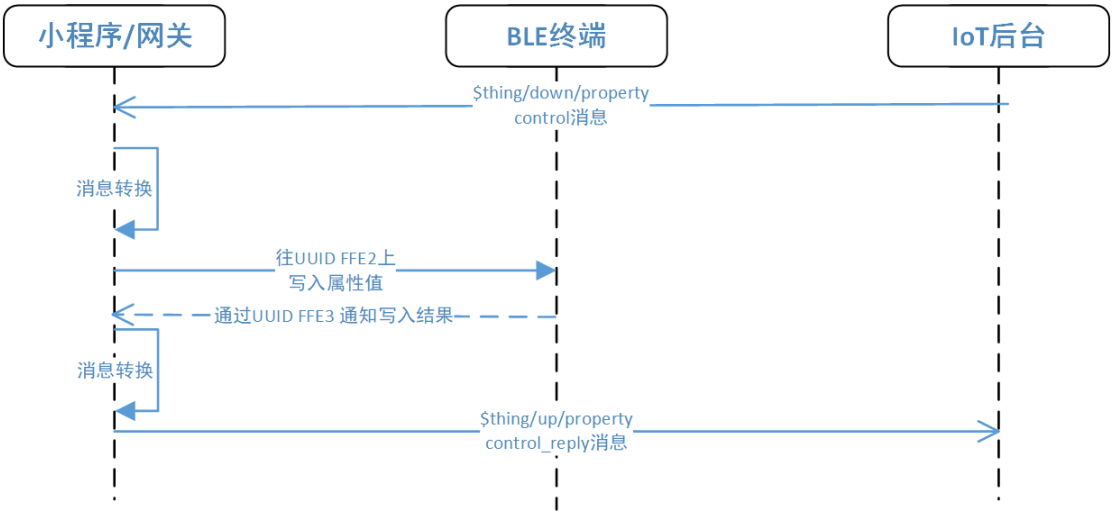
property value 中可以包含多个 property 的数据。示例

数值	描述
00	type
00, 0F	length
00，01	property power switch tlv data
81，00，01	property color tlv data
22，00，00，00，23	property brightness tlv data
43，00，02，31，32	property name tlv data

2. 属性上报结果通过 LLData 通知设备，对应数据模版的 report_reply 操作

header	value
0x20	1 Byte Reply_Result

6.4.2 设备远程控制



1. 通过 LLData 远程控制设备，对应数据模版的 control 操作.

header	length	proterpy value
00	2 Bytes value length	tlv 数据

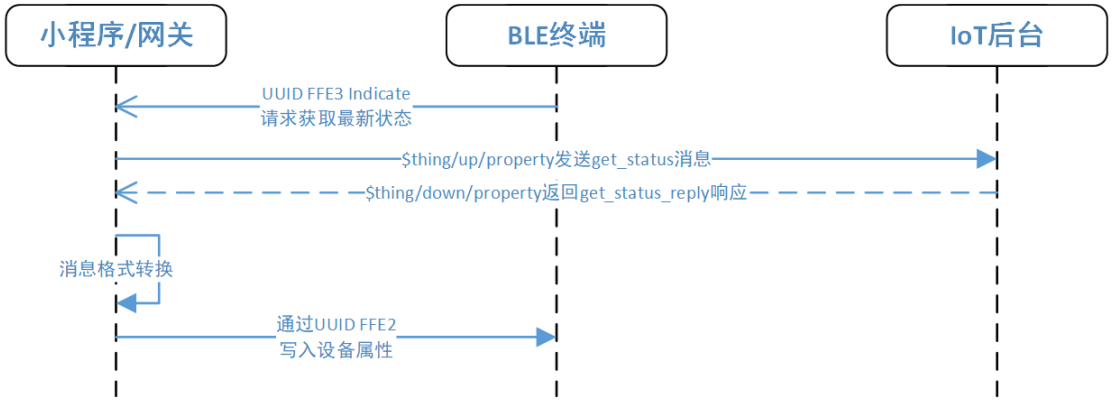
property value 中可以包含多个 property 的数据。示例

数值	描述
00	header
00，0F	value length
00，01	property power switch tlv data
81，00，01	property color tlv data
22，00，00，00，23	property brightness tlv data
43，00，02，31，32	property name tlv data

2. 设备通过 LLEvent 上报操作结果，对应数据模版的 control_reply 操作

type	length	result
01	2 Bytes value length	1 Byte Reply_Result

6.4.3 获取设备最新信息



1. 设备通过 LLEvent 上报最新信息，对应数据模版的 get_status 操作

type
02

get_status 操作不需要携带数据。

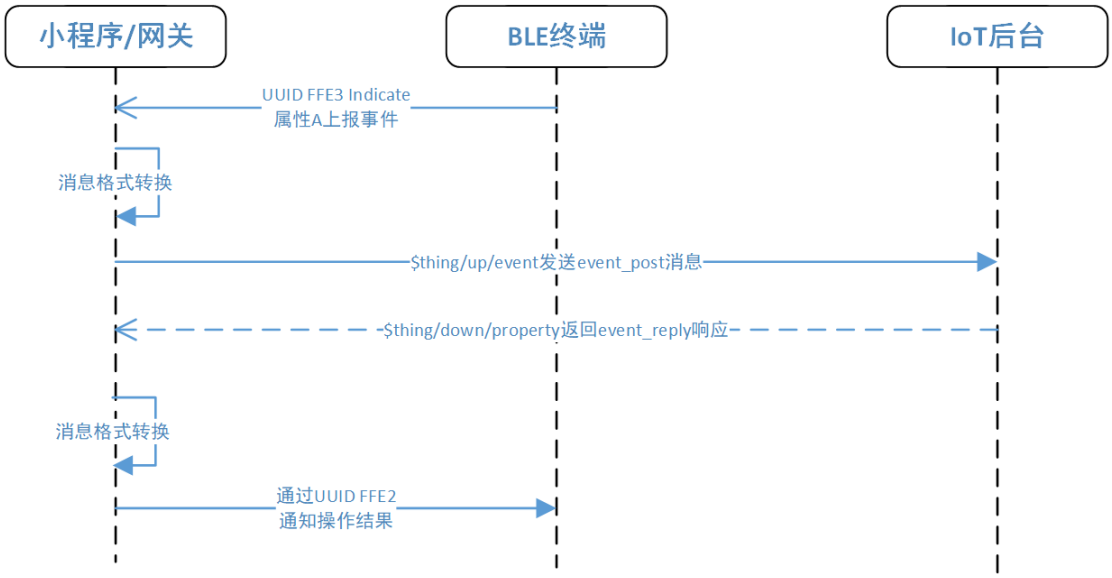
2. 通过 LLData 下发最新信息，对应数据模版的 get_status_reply 操作

header	result	length	property value
0x22	1 Byte Reply_Result	2 Bytes value length	tlv 数据

property value 中可以包含多个 property 的数据。示例

数值	描述
22	header
00	Reply_Result
00，0F	property value length
00，01	property power switch tlv data
81，00，01	property color tlv data
22，00，00，00，23	property brightness tlv data
43，00，02，31，32	property name tlv data

6.4.4 设备事件上报



1. 设备通过 LLEvent 上报事件，对应数据模版中的 event_post 操作

type	length	event id	event value
03	2 Bytes value length	1 Byte event id	tlv 数据

event value 中可以包含多个 event 参数。示例

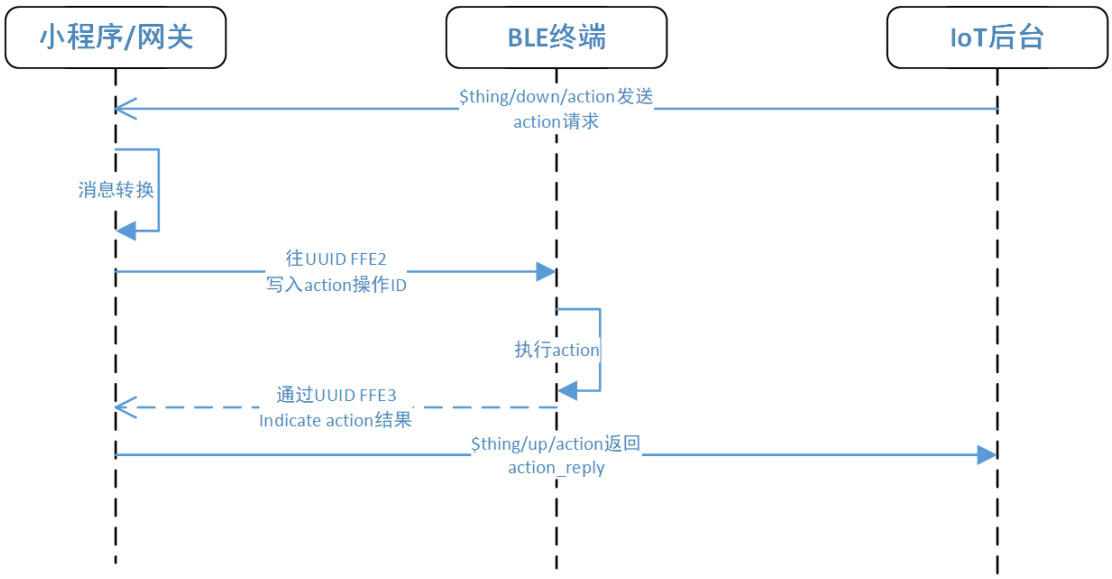
数值	描述
03	type
00，11	event value length
02	event id，对应 hardware fault
40， 00，08， 31，32，33，34，35，36，37，38	param name tlv data
21，00，00，04，00	param error code tlv data

2. 通过 LLData 返回操作结果，对应数据模版中的 event_reply 操作

header	value
见章节 4.3	1 Byte Reply_Result

假设 event id = 0，那么 header 字段应该是 0x60。

6.4.5 设备行为调用



1. 通过 LLData 向设备发起行为调用请求，对应数据模版中的 action 操作

header	length	action value
见章节 4.2	2 Bytes value length	tlv 数据

假设 action id = 0，那么 header 字段应该是 0x80。

action value 中可以包含多个 input 参数。示例

数值	描述
80	header
00, 0B	action value length
20, 00, 00, 00, 04	input interval tlv data
41, 00, 04, 31, 32, 33, 34	input message tlv data

2. 设备通过 LLEvent 上报行为调用结果，对应数据模版中的 action_reply 操作

type	length	value		
		result	action id	response params
4	2 Bytes value length	1 Byte Reply_Result	1 Byte action_id	tlv 数据

a. result 结果失败时，没有 length 等后续字段

b. response param 中可以包含多个 response 参数。示例

数值	**描述**
04	type
00, 0F	length
00	reply result
00	action id，对应 loop
00, 01	response result tlv data
41, 00, 08, 31, 32, 33, 34, 35, 36, 37, 38	response message tlv data

6.5 设备信息上报

设备连接成功后主动向小程序/网关上报设备信息，包括协议版本和设备 MTU 大小，其中版本号与 LLSync Advertisement 中必须一致。

type	length	value	
		version	mtu size
8	2 Bytes	1 Byte	2 Bytes

支持 OTA 功能后，设备信息上报中增加了设备版本号信息。

type	length	value			
		LLSync version	mtu size	firmware version	
				length	payload
8	2 Bytes	1 Byte	2 Bytes	1 Byte	N (<=32) Bytes

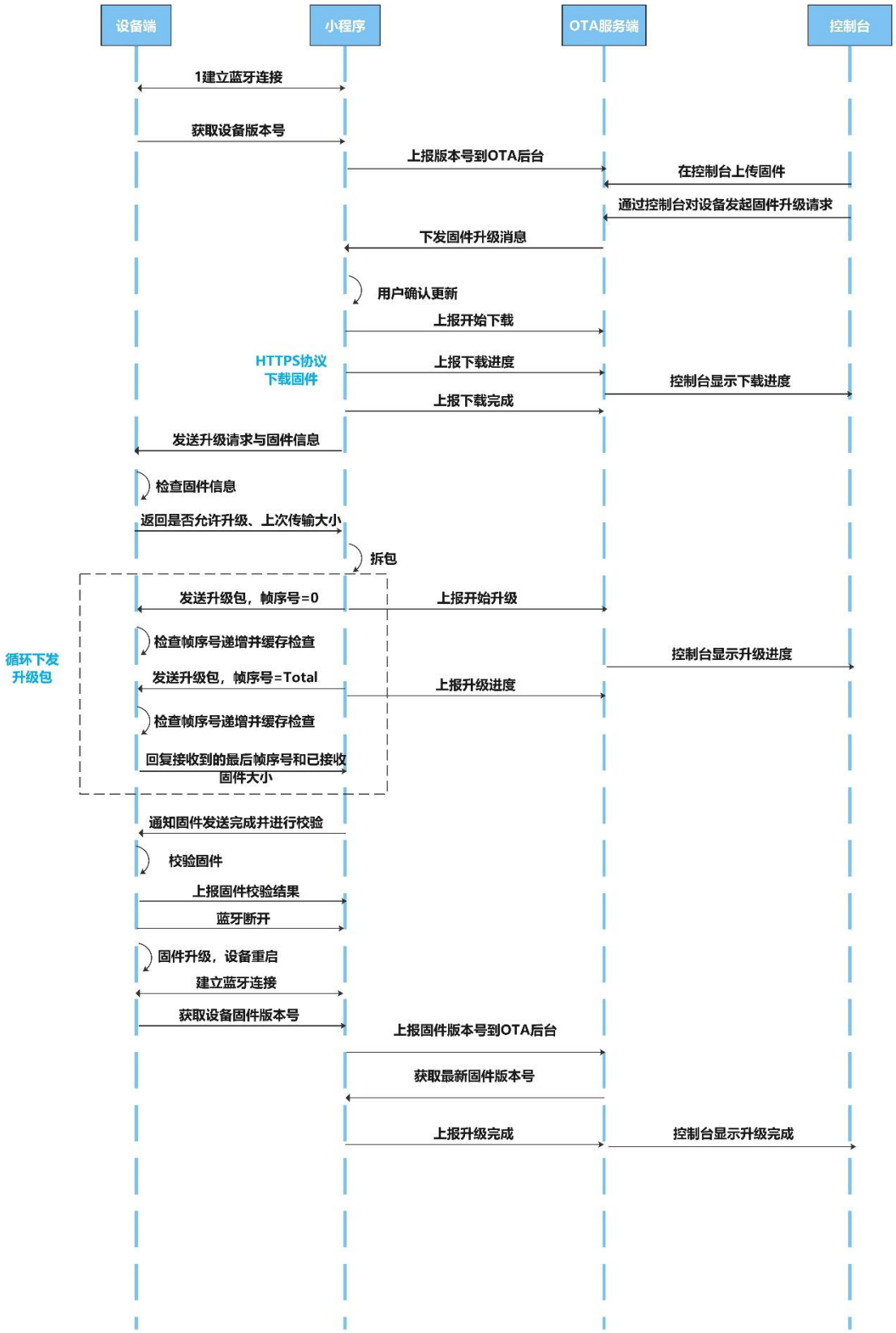
示例：

08 00 09 02 00 14 05 30 2e 30 2e 31，LLSync 版本号为 2，mtu 大小设置为 0x14，固件版本号长度 5 字节，固件版本号为”0.0.1”。

6.6 设备 OTA

设备 OTA 流程图如下，设备端只关心和小程序的数据交互。包括：

- 设备端主动上报版本号
- 小程序下发升级请求
- 设备端应答升级请求
- 小程序下发升级数据包
- 设备端应答升级数据包
- 小程序通知下发结束
- 设备端上报文件校验结果


设备端
小程序
OTA服务端
控制台
1建立蓝牙连接
获取设备版本号
上报版本号到OTA后台
在控制台上传固件
通过控制台对设备发起固件升级请求
下发固件升级消息
用户确认更新
上报开始下载
HTTPS协议
下载固件
上报下载进度
控制台显示下载进度
上报下载完成
发送升级请求与固件信息
检查固件信息
返回是否允许升级、上次传输大小
拆包
循环下发
升级包
发送升级包，帧序号=0
上报开始升级
检查帧序号递增并缓存检查
控制台显示升级进度
发送升级包，帧序号=Total
上报升级进度
检查帧序号递增并缓存检查
回复接收到的最后帧序号和已接收
固件大小
通知固件发送完成并进行校验
校验固件
上报固件校验结果
蓝牙断开
固件升级，设备重启
建立蓝牙连接
获取设备固件版本号
上报固件版本号到OTA后台
获取最新固件版本号
上报升级完成
控制台显示升级完成

6.6.1 固件版本上报

设备通过 LLEvent 进行固件版本号上报，在 6.5 中一并上报。

6.6.2 升级请求包

小程序通过 LLOTA 下发升级请求包到设备。

type	length	value			
		file size	file crc	file version len	file version
0x00	2 Bytes	4 Bytes	4 Bytes	1 Byte	1 ~ 32 Bytes

说明：

1. *约定使用 CRC32 进行文件校验。*
2. *升级请求包分片规则请参见 LLEvent 分片规则。*

示例：

00 00 0e 00 00 00 ff 18 70 16 3c 05 30 2e 30 2e 31，文件大小为 0xFF，文件 CRC 为 0x1870163C，文件版本为 0.0.1

对上述数据应用分片规则，可以分为三包：

00 40 04 00 00 00 ff

00 80 04 18 70 16 3c

00 c0 06 05 30 2e 30 2e 31

也可以分为两包：

00 40 08 00 00 00 ff 18 70 16 3c

00 c0 06 05 30 2e 30 2e 31

分包数量也可以大于三包，大于三包时会有多个 0x00,0x80 开头的数据包。

分包数量取决于数据长度和 ATT MTU 大小，LLSync 会自动处理分包和组包，用户无需关心。

6.6.3 升级请求应答包

设备通过 LLEvent 对升级请求作出应答。

type	length	value	
		indicate	payload
9	2 Bytes	1 byte	N Bytes

升级请求应答包中 value 由 1 字节 indicate 和 N 字节的 payload 构成。

- indicate 表示升级请求的请求结果。
- payload 是请求结果的延伸字段。

indicate 定义：

Bit	说明
0	0: 禁止升级 1: 允许升级
1	0: 不支持断点续传 1: 支持断点续传
2 ~ 7	Reserved

不同的 indicate 字段会有不同的 payload。

当允许升级时 payload 定义如下:

字段	说明
1 byte total package numbers	单次循环中可以连续传输的数据包个数，取值范围 0x00 ~ 0xFF。
1 byte package length	单个数据包大小，取值范围 0x00 ~ 0xF0。
1 byte data retry time	数据包的超时重传周期，单位：秒
1 byte device reboot time	设备重启最大时间，单位：秒
4 bytes last received file size	断点续传前已接收文件大小
1 byte package send interval	小程序连续两个数据包的发包间隔

说明：

1. *不支持断点续传时，已接收文件大小恒为 0。*
2. *小程序连续 5 个超时重传周期内没有收到设备端回应，认为升级失败。*
3. *设备重启最大时间是设备下载成功后重启设备，小程序等待设备上报新版本号的最大时间，超出此时间小程序认为升级失败。*
4. *升级请求应答包分片规则请参见 LLEvent 分片规则。*

示例：

0a 00 09 03 10 0f 05 14 00 00 00 00，表示设备端允许升级且支持断点续传，单次循环传输 0x10 个数据包，每个数据包数据长度为 0x0F，数据包超时设置为 5 秒，设备重启时间最大为 20 秒，断点续传前文件大小为 0

当禁止升级时 payload 表示禁止升级的原因:

错误码	说明
2	设备电量不足
3	版本号错误

示例：

0a 00 02 00 02,表示设备端禁止升级，因为设备电量过低

6.6.4 升级数据包

小程序通过 LLOTA 下发升级数据包到设备。

type	length	value	
		seq	payload
0x01	1 Byte	1 Byte	N Bytes

说明：

1. *length 字段表示 seq 和 payload 的长度之和。*
2. *seq 表示数据包在单次循环中的序列号，从 0 开始，每一包数据增加 1，直到 total package numbers - 1 结束，单次循环结束后重新从 0 开始。*

示例：

01 10 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01，表示 seq 为 0x01 的数据包，数据总长度为 0x10，有效数据长度为 0x0F，即 0x0F 个 0x01。

6.6.5 升级数据应答包

设备通过 LLEvent 对升级数据包作出应答。

type	length	value	
		next seq	file size
0x0A	2 Bytes	1 byte	4 Bytes

说明：

1. *next seq 是设备收到的数据包的 seq 的下一个 seq，file size 是设备已接收的正确文件的大小。*
2. *设备收到单个循环的所有数据包后，使用 next seq 和 file size 对此次循环作出应答，小程序收到应答后再发送下一循环的数据包数据包。*
3. *设备收到错误的 seq 时，发送应答包给小程序请求重传，小程序根据设备上报的 next seq 和 file size 重新传输数据，小程序应该从 file size 处开始传输，seq 等于 next seq。*
4. *当传输出错时，在一个数据重传周期内，设备端只会上报一次数据应答包。*
5. *连续 5 个数据重传周期内没有收到正确的数据包，设备端认为升级失败，用户可以控制断开连接。*
6. *升级数据包最后一个循环中数据包可能不足 total package numbers，设备会根据文件大小计算，以便在收到最后一个数据包时仍然可以发送数据应答包。*

示例：

0b 00 05 0f 00 00 00 f0，表示设备端收到的最后一个数据包的 seq 为 0x0F，设备当前接收的正确文件的大小为 0xF0

6.6.6 升级数据结束通知包

小程序通过 LLOTA 通知设备升级数据包下发结束。

type
0x02

说明：

1. *小程序文件下发结束后通知设备端进行固件检查并上报结果。*

6.6.7 上报固件检查结果

设备通过 LLEvent 上报升级文件的校验结果。

type	length	value
0x0B	2 Bytes	校验结果定义

校验结果定义：

Bit	说明
7	1 ： 校验通过 0 ： 校验失败
6 ～ 0	0 ： 文件 CRC 错误 1 ： flash 操作失败 2 ： 文件内容错误

说明：

1. *使用 1 字节表示校验结果，Bit 7 表示校验是否通过，如果文件校验错误，Bit 6 ～ 0 表示具体的错误原因。*

示例：

0c 00 01 80，表示文件校验通过

7. 蓝牙辅助配网

7.1 概述

蓝牙辅助方式配网，每个厂商编码方式和报文选择上有自己的协议，本文介绍腾讯蓝牙辅助方式配网基础规范，包括蓝牙接入规范和蓝牙交互服务规范等。

- 蓝牙辅助方式配网是一款基于蓝牙通道的 Wi-Fi 网络配置功能。它通过蓝牙辅助方式配网协议将 Wi-Fi 配置传输到 BLE 设备，然后 BLE 设备可基于这些信息连接到 WIFI 热点。
- 此时腾讯连连小程序可以通过 GATT 连接，例如，GATT 通讯将后台提供的配网 Token 发送给设备，并由设备转发至物联网后台，依据 Token 可以进行设备绑定。

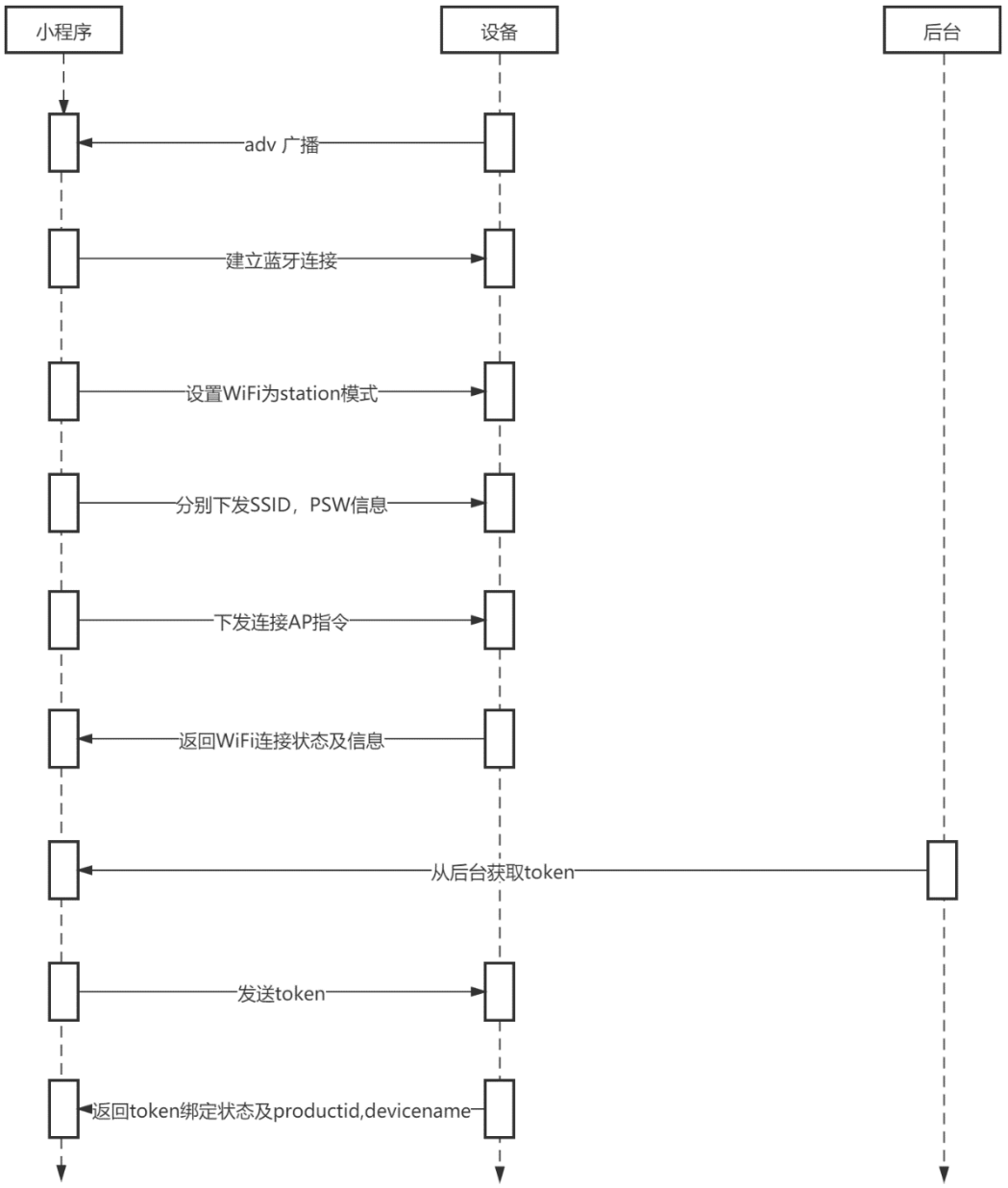
目前腾讯连连小程序已支持采用蓝牙辅助方式配网基础规范议进行蓝牙辅助配网

7.2 蓝牙辅助配网流程

配网流程

1. BLE 设备开启 GATT Server 功能，发送带有特定 adv data 的广播。
2. 使用腾讯连连小程序搜索到该特定广播，手机作为 GATT Client 连接 BLE 设备。
3. GATT 连接建立成功后，腾讯连连小程序向 BLE 设备发送“Wi-Fi mode 设置为 station 模组”控制帧
4. 腾讯连连小程序向 BLE 设备发送定义的 SSID、Password 用于 Wi-Fi 连接的必要信息。
5. 腾讯连连小程序向 BLE 设备发送“Wi-Fi 连接请求”控制帧，BLE 设备收到之后，识别为腾讯连连小程序已将必要的信息传输完毕，准备连接 Wi-Fi。
6. BLE 设备连接到 Wi-Fi 后，发送“Wi-Fi 连接状态报告”控制帧到腾讯连连小程序，以报告连接状态。
7. 腾讯连连小程序收到 BLE 设备报告的链接状态后，发送“服务端获取的 token”用于设备与后台绑定。
8. BLE 设备与后台绑定完成后，发送“设备端绑定状态报告”控制帧到腾讯连连小程序，以报告绑定状态。至此配网结束。

配网流程图如下



7.3 传输格式

腾讯连连小程序与 BLE 设备之间的通信格式定义如下

帧不分片时的标准格式（8 bit）

描述	type	Frame control	Sequence number	Data length	data
数值	1	1	1	1	${ Data length }

帧分片格式（8 bit）

描述	type	Frame control	Sequence number	Data length	data	
数值	1	1	1	1	Total Content Length	Content
					2	${Data Length} - 2

Ack 帧格式（8 bit）

描述	Type(Ack)	Frame control	Sequence number	Data length	data
数值	1	1	1	1	Acked Sequence Number
					2

1.Type

类型域，占 1 byte。分为 Type 和 Subtype（子类型域）两部分，Type 占低 2 bit，Subtype 占高 6 bit。

分为控制帧，数据帧，控制帧（0x0b’00）定义见下表

控制帧（二进制）	含义	释义	备注
0x0(b’000000)	Ack	用来回复对方发的帧， Ack 帧的 Data 域使用回复对象帧的 Sequence 值。	Data 域使用 1 byte Sequence 值， 与恢复对象帧的 Sequence 值相同。
0x2(b’000010)	设置 WIFI 工作模式	设置 BLE 设备的 Wi-Fi 模式，帧包含 opmode 信息。	Data[0]用于表示 wifi mode 类型，包括：0x00:NULL;0x01:STA
0x3(b’000011)	连接 BLE 设备到 AP	通知 BLE 设备，必要的信息已经发送完毕，可以连接 AP。	不包含 data 域
0x5(b’000101)	获取 WiFi 状态	获取 BLE 设备的 Wi-Fi 模式和状态等信息。	会通过 Wi-Fi 连接状态 报告 (Wi-Fi Connection State Report) 数据帧来回

			复小程序当前所处的 opmode、连接状态、SSID。

数据帧（0x1 b’01）定义见下表

数据帧（二进制）	含义	释义	备注
0x2(b’000010)	Wifi station 的 ssid 信息	STA 将要连接的 AP 的 SSID。	NULL
0x3(b’000011)	Wifi station 的 password 信息	STA 将要连接的 AP 的密码	NULL
0xf (b’001111)	Wi-Fi connection state report.	通知手机 BLE 设备的 Wi-Fi 状态， 包括 STA 状态， 用于小程序配置 STA 连接时的通知， 或有 STA 连接上 SoftAP 时的通知。	回复的 ACK 中 data[0]表示：opmode，包括 0x00:NULL;0x01:STA;data[1]表示：STA 的连接状态，0x0 表示处于连接状态，其他表示处于非连接状态；data[2]表示：softap 的连接状态，即表示有多少 STA 已经连接；data[3]及后面表示：为按照协议格式的 SSID 信息；
0x13 (b’010011)	Token data	用户发送 token 或者接收 token 绑定状态信息	数据较长时可分片发送。

2.FrameControl

帧控制域，占 1 byte

3.SequenceControl

序列控制域。帧发送时，无论帧的类型是什么，序列 (Sequence)都会自动加 1，用来防止重放攻击 (ReplayAttack)。每次重现连接后，序列清零。

4.Length

Data 域的长度，不包含 CheckSum。

5.Data

Data 表示用户传输的数据，以下为示例数据

实例 1、下发设置切换 WIFI mode 到 STA

命令：08 08 00 01 01

数值	描述
(0x02<<2) | 0x00=08	加载控制帧，0x00 为控制命令，0x02 设置切换 WIFI mode 到 STA
08	加载帧控制域
00	sequence 序列控制域，每发送一次加 1
01	数据长度
01	data0

实例 2、下发 WiFi 的 SSID 信息

命令：09 00 01 07 74 65 6E 63 65 6E 74

数值	描述
(0x02<<2) | 0x01=0x09	加载数据帧，0x01 为数据命令，0x02 表示发送 AP 的 SSID
00	加载帧控制域，无检验，无加密
01	sequence 序列控制域，每发送一次加 1
07	数据长度
74	data0
65	data1
6E	data2
63	data3
65	data4
6E	data5
74	data6