腾讯小微蓝牙设备通信协议

版本	修改内容	修改人	日期
2	♦ 重写文档，补充缺失，优化结构。 ♦ 更新 103 命令：设备需在 response 中包含特征值及连接状态 ♦ 更新 105 命令：特征值的首字节包含了系统及 App 的信息，增加 extra_config 字段。 ♦ 更新协议版本号到 2 ♦ 新增章节“打开 App”	changtang	2019-10-12
	♦ 修改 Android spp 连接的 uuid ♦ 修改 Android 判断腾讯小微 app 是否安装的方式	jasonsgao	2019-10-14
	♦ 删除 104，107 命令	jasonsgao	2019-10-25
	♦ 增加了命令超时重传 ♦ 重新添加 107 命令，支持取消 TTS 播放或暂停内容播放 ♦ 110 命令添加 payload 说明 ♦ 由 App 发出的指令，增加了超时重传机制(详见指令协议)。	changtang	2019-11-01
	♦ 105 命令的 deviceColor 更换为 skuid ♦ 重新添加 response code	changtang	2019-11-07
	♦ 增加 111 命令，用于 iOS 系统上控制其他 App 的音频播放	changtang	2019-11-22
	♦ 更新 111 命令，支持发送 tts end 事件 ♦ 更新了 103 命令，扩展了 connect_state 字段的定义	changtang	2019-11-27
	♦ 增加对话控制时序图，说明了 111 命令的具体使用时机	changtang	2019-12-16
	♦ 增加 112 heartBeat 命令	changtang	2019-12-18
	♦ 112 heartBeat 由 App 定时发出	changtang	2019-12-20
	♦ 更新 105 命令：extra_config 中增加设备序列号	changtang	2020-01-10
	♦ 更新 iap 的 protocol string	changtang	2020-01-14
	♦ ble 的广播包支持 2-8 个字节	changtang	2020-03-05
3	♦ 增加 113 命令	changtang	2020-03-13
	♦ 更新 113 命令：定义了更多 connect_control 的行为 ♦ 更新 103 命令：增加 keep_conn 字段	changtang	2020-04-09
	♦ 添加 response_code: 50，表示 vpa 尚未连接时，收到了只能在已连接的状态下发送的 command。	changtang	2020-05-08
4	♦ 增加 114 命令 generalConfig，通用的设置命令。	changtang	2020-07-03
5	♦ 增加 115 命令用于想耳机发送自定义技能的 query 结果 ♦ 增加 extra config 的包结构	changtang	2020-11-24
	♦		

目　录

概要 4
发现与连接 5
IAP 5
SPP 5
BLE 5
连接流程 5
录音流程 7
长按 7
短按 8
对话控制 8
采样和编码 10
采样 10
Opus 10
SBC 10
指令协议 12
包结构 12
指令列表 13
101 wakeUp 13
102 silence 13
103 getInfo 13
105 getConfig 14
106 getSig 15
107 stopPlay 15
108 requestStopRceord 15
109 setKeyFunction 15
110 clearFValue 16
111 playControl 16
112 heartBeat 16
113 connectControl 17
114 generalConfig 17
115 customSkill 18

255 voiceData 19
其他 19
extra config 19
打开 App 21
判断是否安装 21
下载页面 21
跳转链接 21
附录 22
App 对小微协议版本支持情况 22
Response Code 22

概要

蓝牙版本 4.2 及以上。

iOS 系统支持 10.0 及以上，连接方式支持 IAP，BLE。

Android 系统支持 6.0 及以上，连接方式支持 SPP。

音频采样参数：16000 采样率，单声道，16bit 采样位深。

音频编码方式支持 Opus，mSBC。

支持 OTA 升级（可选）。

发现与连接

IAP

iOS 支持。
protocol string: com.tencent.xiaowei

SPP

Android 支持。
UUID: 85dbf2f9-73e3-43f5-a129-971b91c72f1e

BLE

iOS 支持。
设备通过 Advertising Data 和 Scan Response 来向 App 传递设备的信息。优先通过 Advertising Data 来传递，如数据包大小超过限制，可放到 Scan Response 中。
数据中应包含：
Service UUID：0709
MANUFACTURER SPECIFIC DATA：

Offset (bytes)	Length (bytes)	Type	Description
0	2	uint16_t	compnay_id，由厂商自行在蓝牙组织申请
2	4	uint32_t	product_id,由厂商在腾讯小微硬件管理平台申请
6	6	uint8_t []	蓝牙 MAC 地址
12	8	uint8_t []	特征值，连接 App 过程中由 App 分配，没有时用 0 填充

建立 BLE 连接后，
设备向 App 发送数据的通道为 dfd4416e-e40c-47f7-8248-eb8be3dc47f9，
设备接收来自 App 的数据的通道为 9884d812-61fe-4a24-94d3-b2c11a851fac 。
设备需要在与 App 建立连接后停止广播。

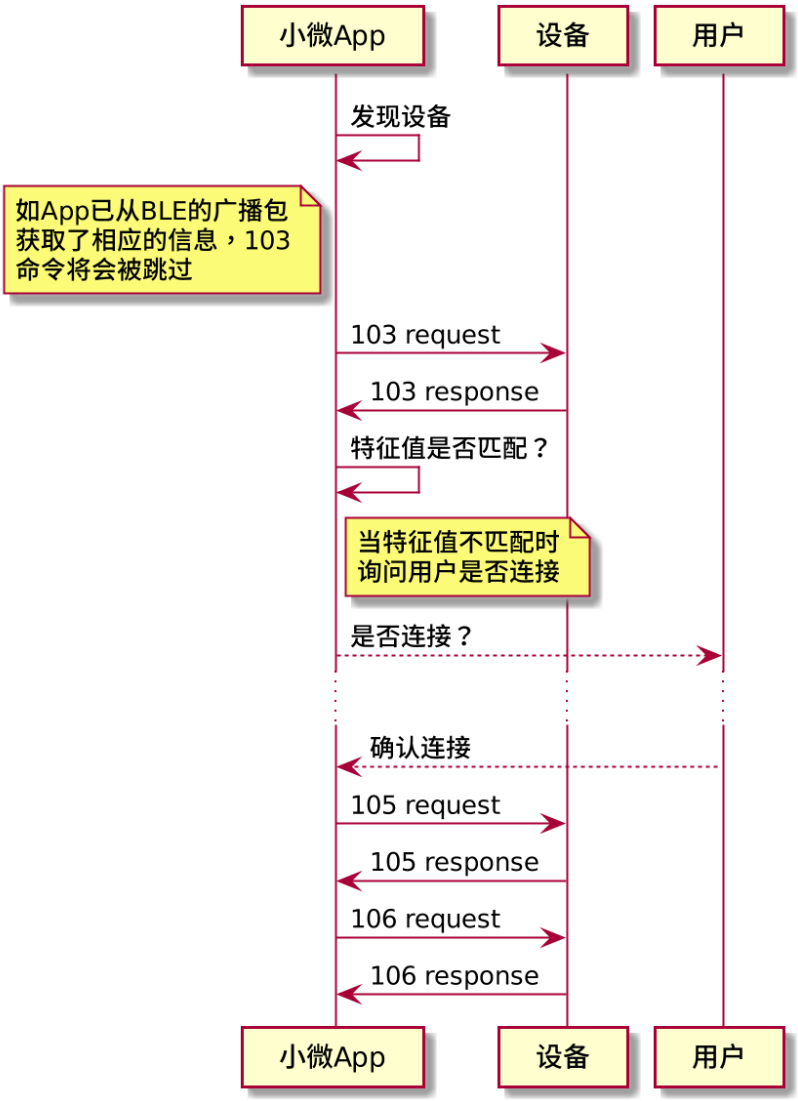
连接流程

1. App 通过 BLE，SPP，IAP 等方式查找可用设备。
2. 发现设备后，App 发送 103 命令给设备来获取 product_id，mac 地址以及特征值。如果 App 已经通过 BLE 的广播包获取到了相应信息，App 会跳过该步

骤。

3. App 检查特征值，判断是否可以自动连接。
4. 如不能自动连接，App 弹窗询问用户是否连接设备。用户确认后继续流程。
5. App 发送 105 命令，给设备分配特征值，设备设备回复配置参数。
6. App 发送 106 命令 给设备发送 phone id ,设备根据校验方式回复校验信息。
7. 连接完成，App 播放已连接的提示。
8. 连接完成后，App 会定时发送心跳（112 heartBeat）给设备。如心跳失败，App 将断开与耳机的连接。

过程如下图所示

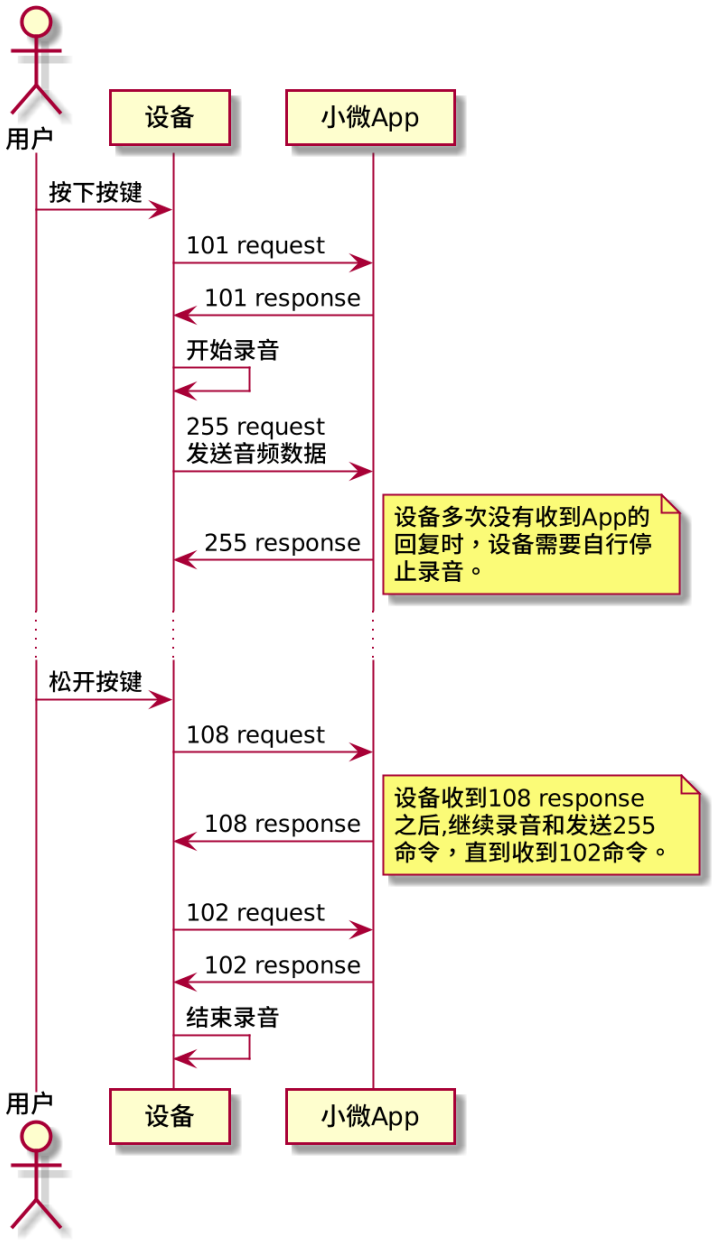


录音流程

长按

1. 用户按下按键时，设备发送 101 命令，收到回复后开始录音。
2. 设备持续通过 255 命令发送编码后的音频数据。如多次没有收到 App 的回复时，设备要停止录音并主动发送 102 命令尝试告知 App 停止录音。
3. 用户松开按键时，设备发送 108 命令，在收到回复后需要继续录音和发送音频数据。
4. 设备收到 App 发来的 102 命令时，结束录音。

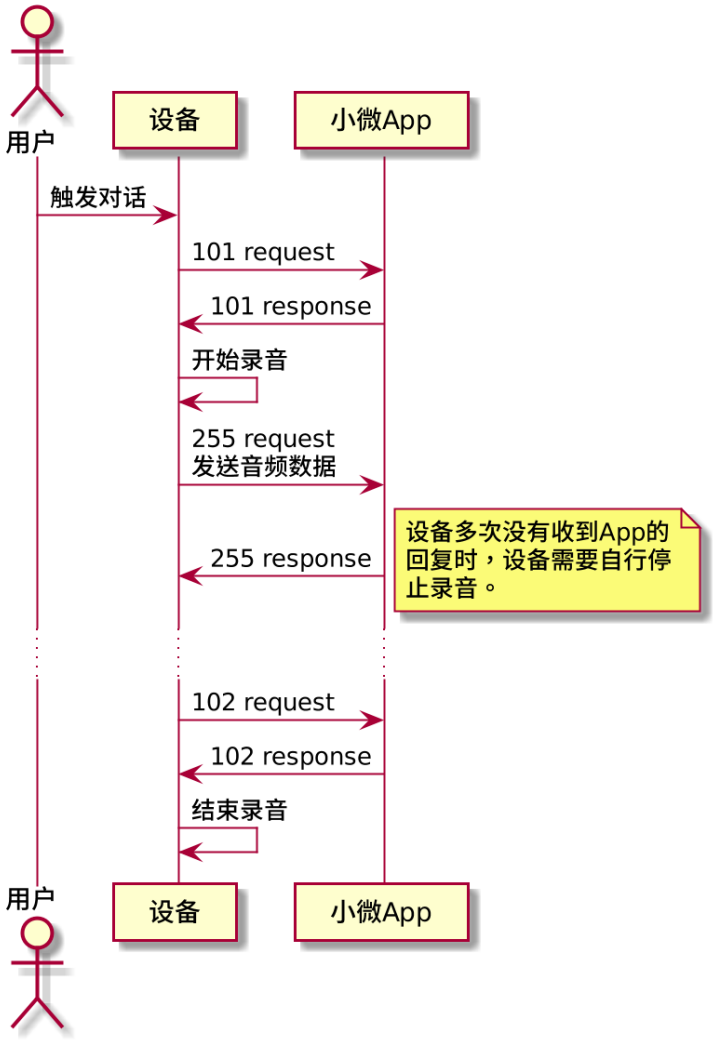
过程如下图所示



短按

1. 用户触发录音，设备发送 101 命令，收到回复后开始录音。
2. 设备持续通过 255 命令发送编码后的音频数据。如多次没有收到 App 的回复时，设备要停止录音并主动发送 102 命令尝试告知 App 停止录音。
3. 设备收到 App 发来的 102 时，结束录音。

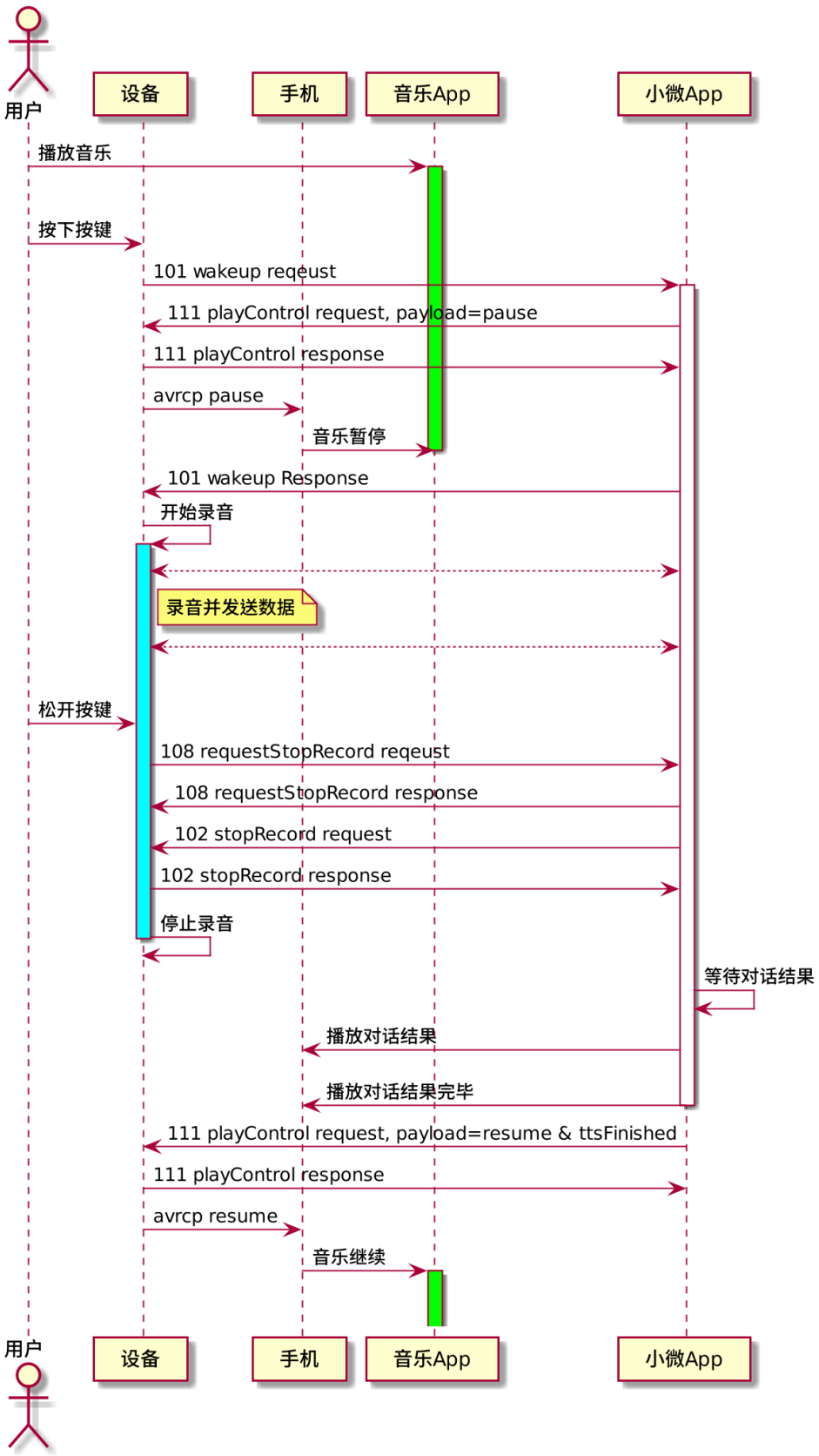
过程如下图所示



对话控制

在 iOS 平台上，如果有第三方的 App 正在播放音乐，此时小微 App 需要通过设备来发送 AVRCP 指令来控制第三方 App 的音乐播放。

以长按为例，过程如下图所示


用户
设备
手机
音乐App
小微App
播放音乐
按下按键
101 wakeup reqeust
111 playControl request, payload=pause
111 playControl response
avrcp pause
音乐暂停
101 wakeup Response
开始录音
录音并发送数据
松开按键
108 requestStopRecord reqeust
108 requestStopRecord response
102 stopRecord request
102 stopRecord response
停止录音
等待对话结果
播放对话结果
播放对话结果完毕
111 playControl request, payload=resume & ttsFinished
111 playControl response
avrcp resume
音乐继续
用户
设备
手机
音乐App
小微App

采样和编码

采样

录音采样参数：16000 采样率，单声道，16bit 采样位深。

Opus

iOS 和 Android 支持。
采用固定码率，设备每次发送 40 个字节的编码数据到 App。
支持 16 倍和 8 倍的压缩，对应的 frame_duration 分别为 20ms, 10ms。

码率（压缩前）	压缩比例	包大小	frame_duration	码率（压缩后）
256kbps	16	40Byte	20ms	16kbps
256kbps	8	40Byte	10ms	32kbps

SBC

iOS 和 Android 支持。
SBC 参数：

```
sbc->frequency = SBC_FREQ_16000;
sbc->blocks = SBC_BLK_16;
sbc->subbands = SBC_SB_8;
sbc->mode = SBC_MODE_MONO;
sbc->allocation = SBC_AM_LOUDNESS;
sbc->bitpool = 12; // 256 bytes input 32 bytes output
sbc->endian = SBC_LE;
```

mSBC 参数：

```
sbc->frequency = SBC_FREQ_16000;
sbc->blocks = MSBC_BLOCKS;
sbc->subbands = SBC_SB_8;
sbc->mode = SBC_MODE_MONO;
sbc->allocation = SBC_AM_LOUDNESS;
sbc->bitpool = 26; // 240 bytes input 57 bytes output
sbc->endian = SBC_LE;
```

指令协议

App 及设备均可主动发送 request 指令。

App 或设备收到 request 指令后，回复相应的 response 指令。如超过 0.5 秒没有收到相应的 response，耳机会重新发送 command，最多重试 2 次。

命令处理成功时，response 指令的 code 为 0，其他错误码参考 Response Code

数据序列化时采用小端字节序。

包结构

request：

Offset (bytes)	Length (bytes)	Type	Description
0	1	uin8_t	command_id
1	1	uin8_t	command_type，1: request，2: response
2	1	uin8_t	sequence number 0~255 递增循环，response 取 request 值
3	2	uint16_t	payload 长度，无数据时为 0
5	N	uint8_t []	payload，指令相关数据，无数据时为空

response：

Offset (bytes)	Length (bytes)	Type	Description
0	1	uin8_t	command_id
1	1	uin8_t	command_type，1: request，2: response
2	1	uin8_t	sequence number 0~255 递增循环，response 取 request 值
3	1	uint8_t	code，0: success，
4	2	uint16_t	payload 长度，无数据时为 0
6	N	uint8_t []	payload，指令相关数据，无数据时为空

其中 payload 数据由各指令自己定义。

指令列表

101 wakeUp

用户触发录音，发送该命令给 App，并开始录音。

request payload: NULL

response payload: NULL

102 silence

App 检测到静音时，发送该命令给设备，设备结束录音。

或设备在录音时多次没有收到 App 回复时，发送此命令结束录音。

request payload: NULL

respone payload: NULL

103 getInfo

建立连接过程中，由 App 发送给设备，设备返回 product_id 以及 MAC 地址。

request payload：NULL

response payload：

Offset (bytes)	Length (bytes)	Type	Description
0	4	uint32_t	product_id
4	6	uint8_t []	Bluetooth mac address
10	8	uint8_t []	特征值，连接 App 过程中由 App 分配，没有时用 0 填充
18	1	uint8_t	connect_state: 0 表示未连接其他 App 或者语音助手。 1 表示已连接 QQ 音乐 App。 2 表示已连接小微 App。 3 表示已连接其他语音助手（如有）。
19	1	uint8_t	keep_conn,可选字段，默认为 0 0：表示如果接下来如果不建立 VPA 连接，那么连接通道将被释放。

			1：表示保留连接（以便后续 113 命令可以重新建立连接）

105 getConfig

建立连接过程中，由 App 发送给设备，设备返回参数配置信息。

request payload:

Offset (bytes)	Length (bytes)	Type	Description
0	1	uint8_t	App 期望码率，单位 KBps，目前为 32
1	1	uint8_t	手机操作系统. 1:iOS 2:Android
2	8	uint8_t []	特征值，8 字节，其中包含了 App 的信息，参考代码：`xw_headphone_fvalue_get_host_app`

response payload:

Offset (bytes)	Length (bytes)	Type	Description
0	1	uint8_t	连接类型，1:BLE, 2:SPP, 3:IAP
1	1	uint8_t	编码类型，1:Opus, 2: SBC, 3: mSBC
2	1	uint8_t	压缩率，16， 8，编码类型为 Opus 时适用。
3	1	uint8_t	采样率，1:16000
4	1	uint8_t	录音声道，1:单声道
5	1	uint8_t	位深，1:16bit
6	1	uint8_t	录音方式，1:长按，2:短按
7	4	uint8_t []	固件版本号：1.2.3.4 对应的数据应为 0x01,0x02,0x03,0x04
11	1	uint8_t	设备型号名称长度
12	N	uint8_t []	设备型号名称，UTF8 编码
12+N	1	uint8_t	设备支持的小微协议版本号。参见附录 App 对小微协议版本支持情况。
13+N	M	uint8_t []	extra config

extra config **中包含一到多个配置项。目前支持的配置项有：**

- **功能键：默认（降噪）/小微助手**
- skuid，**该值与厂商在硬件管理平台所填保持对应。**
- **设备自定义名称**

- 设备序列号
- 更多详细信息参考代码`xw_headphone_extra_config_item`

106 getSig

建立连接过程中，由 App 发送 phoneId 给设备，设备返回校验信息。

request payload:

Offset (bytes)	Length (bytes)	Type	Description
0	N	uint8_t []	phoneId，用于生成校验参数

response payload:

Offset (bytes)	Length (bytes)	Type	Description
0	N	uint8_t []	sig，生成的校验参数

校验未启用时，sig 返回长度不为 0 的任意值。

107 stopPlay

由设备发送给 App，App 收到命令后会停止正在播放对话结果，或者暂停正在播放的内容资源（音乐、故事、新闻等）。

request payload: NULL

respone payload: NULL

108 requestStopRceord

长按录音流程中，用户松开按键时，设备需要发送此命令。

request payload: NULL

respone payload: NULL

109 setKeyFunction

设置按键功能，如设备不支持，可不实现。

request payload:

Offset	Length	Type	Description

(bytes)	(bytes)		
0	1	uint8_t	1:设备默认功能, 2:小微语音助手

response payload: NULL

110 clearFValue

清空设备的特征值。

request payload:

Offset (bytes)	Length (bytes)	Type	Description
0	1	uint8_t	1:清空,

response payload: NULL

备注：出于兼容旧设备的目的，App 会发送一个字节的 payload，其值为 1。设备可以不处理该数据。收到 110 命令后即可清除已存储的特征值。

111 playControl

播放控制对话结束通知指令，App 通过该指令告诉设备，TTS 播放结束，以及需要发送的 AVRCP 指令（pause、resume）。

在 Android 上，TTS 播放完成后会发送该命令告知设备 TTS 播放结束。

在 iOS 上，开始录音时，如有第三方 App 在播放音乐，App 会发送该命令告知设备发送 AVRCP Pause 指令 。对话结束时，发送此命令告知设备 TTS 播放完成以及是否需要发送 AVRCP Resume 指令。

request payload:

Offset (bytes)	Length (bytes)	Type	Description
0	1	uint8_t	value: 第 0 个 bit（最低位）为 1 表示是 TTS End。 第 1 个 bit 为 1 表示需要设备发送 AVRCP Pause 事件。 第 2 个 bit 为 1 表示需要设备发送 AVRCP Resume 事件。

response payload: NULL

112 heartBeat

App 和设备均可主动发送该指令，用于探测连接是否正常。

App 会定期发送心跳命令给设备，用于维持 App 在后台时不被系统挂起，并保持连接可用。

request payload: NULL

response payload: NULL

113 connectControl

设备端需要主动断开或建立连接时，发送该命令。

App 收到该命令后会根据标志位做相应处理。

request payload:

Offset (bytes)	Length (bytes)	Type	Description
0	1	uint8_t	flag，该字段可选，默认为 0。 0 表示设备希望断开连接，并释放数据通道。 1 表示设备希望断开连接，并保留数据通道，用于后续重新建立连接。 2 表示设备希望重新建立 vpa 连接，App 会发送 103 及后续命令。如果 App 已连接其他耳机会回复 code 51. 3 表示设备希望释放数据通道的连接。

response payload: NULL

114 generalConfig

该命令用于对设备进行一些设置，如降噪调节、音效调节。

命令通过 key（uint16_t）来确定具体的设置项，参数通过 length 及 payload 来传递，具体情况请参考代码中对` xw_headphone_general_config_key`的注释。

request payload:

Offset (bytes)	Length (bytes)	Type	Description
0	2	uint16_t	key
2	1	uint8_t	length(>= 0)，若 length 为 0，则 payload 不存在
3	N	int8_t []	payload

response payload：

NULL（如果设置成功）;

错误码为` xw_headphone_response_code_invalid_range`时，payload 包含对应设置的合法区间。

115 customSkill

用户 query 了耳机的自定义技能时，App 通过该命令转发技能信息给到耳机。

request payload:

Offset (bytes)	Length (bytes)	Type	Description
0	1	uint8_t	skill name length
1	N	uint8_t []	skill name
N+1	1	uint8_t	intent name length
N+2	M	uint8_t []	intent name
M+N+2	1	uint8_t	slot count
slot info			
slot info			
...			

slot info:

每个 slot info 中包含了 name 和 value。

request payload:

Offset (bytes)	Length (bytes)	Type	Description
0	1	uint8_t	slot name length
1	N	uint8_t []	slot name
N+1	1	uint8_t	slot value length
N+2	M	uint8_t []	slot value

response payload:

NULL（如果处理成功）

出错则参考错误码处理回包。

255 voiceData

设备发送语音数据给 App。

request payload:

Offset (bytes)	Length (bytes)	Type	Description
0	N	uint8_t []	压缩后的语音数据

response payload: NULL

其他

extra config

extra config 的数据由 0-N 个 config_item 拼接而成。

每个 config_item 的第一个 byte 为 config_key。

根据 config_key 的不同，config_item 接下来数据格式如下：

config_key < 200 时, config_item 长度为 2

Offset (bytes)	Length (bytes)	Type	Description
0	1	uint8_t	config_key
1	1	uint8_t	value

config_key >= 200 时，config_item 长度为 2+N，N 为对应数据的长度

Offset (bytes)	Length (bytes)	Type	Description
0	1	uint8_t	config_key
1	1	uint8_t	length
2	N	uint8_t[]	data

相关代码请参考:

` xw_headphone_extra_config_key `

`xw_headphone_extra_config_item_list_pack`

`xw_headphone_extra_config_item_list_unpack`

打开 App

支持从其他 App 跳转到小微 App。
当跳转到小微 App 时，小微 App 已连接设备，可以打开设备的设置页面。
当手机上没有安装小微 App 时，需要打开下载页面，提示用户下载安装。

判断是否安装

iOS：通过 url shceme xiaoweiapp:// ,使用系统接口-[UIApplication canOpenURL:]来判断是否安装。

Android：通过 url shceme xiaoweiapp://，使用系统接口判断是否安装。
示例代码：

```
public boolean checkXwappInstalled() {
    Intent action = new Intent(Intent.ACTION_VIEW);
            action.setData(Uri.parse("xiaoweiapp://"));
            List list = getPackageManager().queryIntentActivities(action,
PackageManager.GET_RESOLVED_FILTER);
            return list != null && list.size() > 0;
}
```

下载页面

iOS：https://apps.apple.com/cn/app/tencent-xiaowei/id1454208784?mt=8
Android：https://a.app.qq.com/o/simple.jsp?pkgname=com.tencent.xw

跳转链接

iOS & Android：xiaoweiapp://headphone/settings

附录

App 对小微协议版本支持情况

协议版本	iOS	Android
2	1.2.0	1.2.0
3	1.2.2	1.2.2
4	1.4.0	1.4.0
5	1.6.0	1.6.0

Response Code

Code	Description
0	No error
1	Invalid command code
2	Parameter length out of range
3	Parameter length too short
4	Command handling failed
5	Waiting response time-out
6	The data transfer has already been started
7	The data transfer length does not match the actually received data length
40	Invalid command
41	Wake up failed, invalid sign
50	vpa 连接尚未建立,无法处理该命令
51	App 已经连其他设备，无法响应 113_2 命令
60	设备不支持此功能
61	（114 命令）操作失败
62	设置超出范围