串口协议

硬件产品开发 > 嵌入式软件开发 > MCU 开发接入 > Zigbee 通用方案

文档版本: 20201212

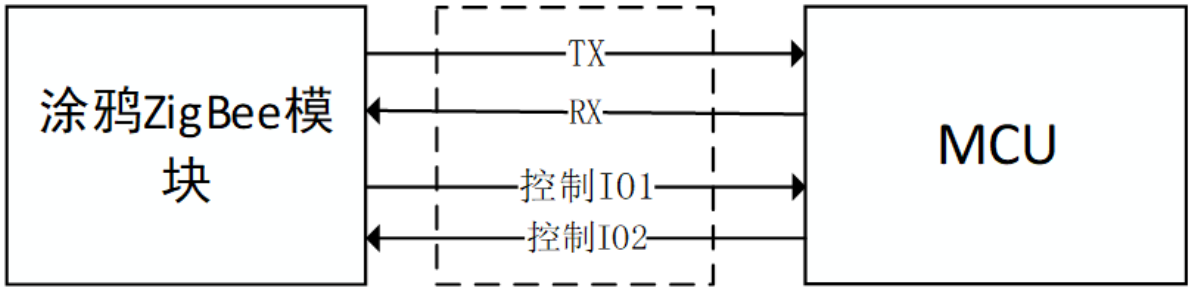
查看在线版本

目录

您可以通过涂鸦 Zigbee 串口通用协议完成涂鸦 Zigbee 模组与其它 MCU 串口的通信。

1 协议架构图

涂鸦 Zigbee 串口通用协议结构如下图所示。



{width=400px}

2 串口通信协议

- 波特率：9600/115200
- 数据位：8
- 奇偶校验：无
- 停止位：1
- 数据流控：无
- 供电电压：Zigbee 模组和 MCU 主控均采用 DC 3.3V
- 休眠模式：
  - 支持休眠功能的低功耗设备：Zigbee 模组与 MCU 之间预留两个 GPIO 口（PWM1 和 PWM2）供 MCU 和模组唤醒时使用，唤醒方式为电平触发。Zigbee 模组或 MCU 每次主动发起命令之前，都需要完成一次握手连接，具体唤醒方法参考下图。

  - 不支持休眠功能的强电设备：串口处于长监听状态，硬件不需要连接 I/O1 和 I/O2。
- MCU 唤醒模组：电平拉低之后可延时 1~5 ms 发送数据，只要保持唤醒口持续低电平，模组会一直处于唤醒状态。当电平拉高之后，模组会在约 300~550ms 之后进入休眠，减少不必要的唤醒时间，降低功耗。

> **说明**：固件支持脉冲唤醒的方式，即 MCU 端每次发送串口数据之前都需要先在唤醒口上发送一个低脉冲，时间为 1~5 ms，然后再发送串口数据。长时间拉低唤醒会导致模组功耗偏高，因此优化为脉冲方式唤醒模组。

```
set_gpio_low();
delay(1);
set_gpio_high();
uart_send_buffer();
```

3 帧格式说明

涂鸦 Zigbee 模组与 MCU 之间的 UART 通信数据帧由帧头（Front），版本（Ver），命令字（Cmd），数据长度（Length），数据（Data）和校验和（Check）组成。

字段	长度（字节）	说明
帧头（Front）	2	固定为 0x55aa
版本（Ver）	1	串口通信协议版本，升级扩展用
序列号（seq）	2	传输数据序列号，范围 0~0xfff0，到达 0xfff0 之后重新回到 0
命令字（Cmd）	1	具体帧类型
数据长度（Length）	2	传输的有效数据长度
数据（Data）	取决于具体数据	传输的有效数据
校验和（Check）	1	数据校验，从帧头开始按字节求和得出的结果对 256 求余

> **注意**：帧中的数据长度（Length）由 Zigbee 模组单个空中数据包的长度决定，涂鸦会对 Zigbee 空中数据格式重新封装，目前支持的数据上限为 62 字节。

4 命令字索引表

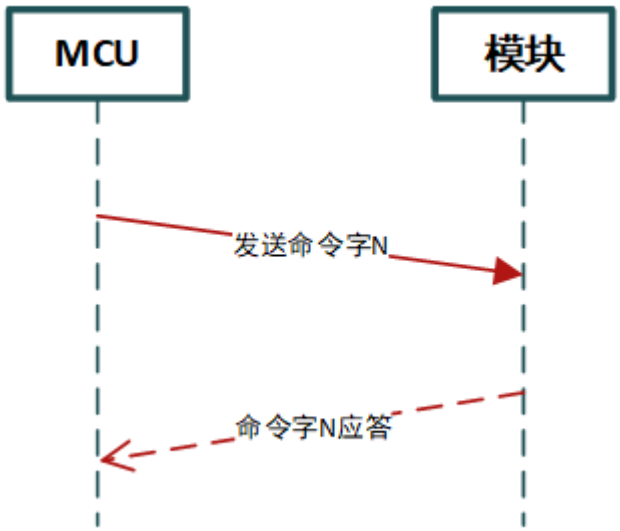
命令字	说明
0x01	上报产品信息
0x02	上报设备状态
0x03	重置设备
0x04	下发命令
0x05	上报状态
0x06	查询状态
0x07	预留命令字
0x08	检测设备功能
0x09	查询按键信息（仅场景开关类设备有效）
0x0A	唤醒场景（仅场景开关类设备有效）
0x24	同步时间
0x25	模组查询 MCU 设备类型

5 通信模式

> **说明**：所有大于 1 字节的数据均采用大端模式传输。

5.1 命令字通信模式

通常命令字采用一发一收的同步模式，即发送方发送命令，接收方应答，如下图所示。



> **说明**：具体通信方式以“协议详述”章节中为准。

5.2 模组下发命令通信模式

模组控制命令下发采用异步模式。

- 模组控制命令下发示意图假设模组控制命令下发命令字为 X，MCU 状态上报命令字为 Y。

```
1 ![image-20191205112650191](https://images.tuyacn.com/fe-static/
    tuya-
2 docs/0b46b5e1-19a7-4e0d-94de-41e0f87ccc17.png)
```

- 下发流程

1.

```
1 模组通过 0x04 指令下发命令，内容为可下发的 DP 数据。
```

2.

```
1 MCU 接收到 0x04 指令之后，进行回复，表示串口接收到该命令。
```

3.

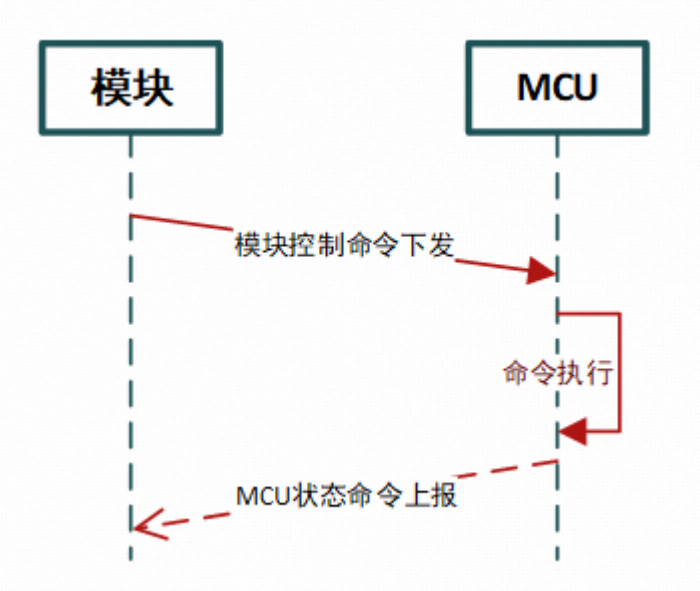
```
1 MCU 通过 0x05 指令将执行的结果上报至云端。
```

4. 验证 0x05 指令的序列号和 0x04 指令是否保持一致。

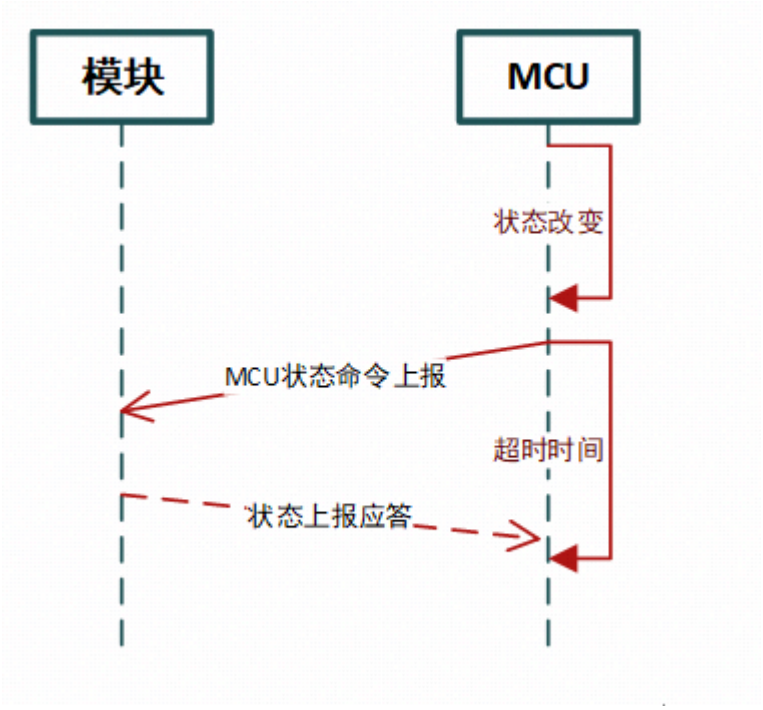
5.3 MCU 上报状态通信模式

MCU 状态上报采用异步模式。MCU 状态上报分为被动上报和主动上报两种情况；

- 被动上报：模组端发送数据命令至 MCU，MCU 执行后返回状态。



- 主动上报：MCU 端状态发生改变（物理操作或者断电重启等）时，将主动上报当前状态至模组。MCU 主动上报为异步操作，如果 MCU 未在指定时间内接收状态上报应答帧，或者接收到的应答帧中状态为不成功，MCU 端需要重新上报状态。


模块
MCU
状态改变
MCU状态命令上报
超时时间
状态上报应答

6 基础协议

6.1 模组查询 MCU 设备类型

上电后模组查询 MCU 的设备类型。查询成功后，模组会保存当前设备类型，不需要再次查询。

> **说明**：该功能为新增功能，请您测试模组中的固件是否支持该功能。

- 工作原理：接收到 MCU 的应答之后，Zigbee 模组将重新启动，载入参数后，继续和模组进行数据交互。
- 检测方法：模组上电之后先以 9600 波特率发送查询指令，如果没有收到 MCU 应答，则使用 115200 波特率进行检测。

  > **注意：**
  >
  > 检测过程中，全部按照低功耗设备处理。先在模组唤醒 MCU 的 I/O 口发送一个 50ms 的低脉冲，再发送串口数据，保证 MCU 收到数据。
  >
  > 已发布的 MCU 固件不需要进行修改，可以继续使用最新固件。新产品接入需要实现本协议。

模组发送

字段	长度（字节）	说明
帧头	2	0x55aa
版本	1	0x02
序列号	2	N
命令字	1	0x25
数据长度	2	0x0000
数据	0	0
校验和	1	从帧头开始按字节求和，将得出的结果对 256 求余

MCU 返回

字段	长度（字节）	说明
帧头	2	0x55aa
版本	1	0x02
序列号	2	N
命令字	1	0x25
数据长度	2	0x0001
数据	1	
		0x01：强电类对接设备
		0x02：低功耗设备
		0x03：强电场景面板
校验和	1	从帧头开始按字节求和，将得出的结果对 256 求余

6.2 查询产品信息

产品信息由产品 ID 和 MCU 软件版本号构成。当模组复位后，主动查询产品信息。如果 MCU 没有回复，或者回复内容有误，将会间隔 5 秒重复查询。

- 产品 ID ： 对应涂鸦开发者平台 PID (产品标识)，在创建产品时由涂鸦开发者平台自动生成，用于云端记录产品相关信息。
- MCU 软件版本号：采用点分十进制形式，格式为 `x.x.x`，x 为十进制数。

> **注意**：OTA 相关命令用单字节表示 MCU 版本时，最大版本由于字节长度限制，最大版本号为`3.3.15`。

模组发送

字段	长度（字节）	说明
帧头	2	0x55aa
版本	1	0x02

字段	长度（字节）	说明
序列号	2	N
命令字	1	0x01
数据长度	2	0x0000
数据	0	无
校验和	1	从帧头开始按字节求和，将得出的结果对 256 求余

示例：`0x55aa 02 N 01 0000 xx`

MCU 返回

字段	长度（字节）	说明
帧头	2	0x55aa
版本	1	0x02
序列号	2	N
命令字	1	0x01
数据长度	2	N
数据	N	
		数据示例： `{“p”:”AIp08kLI”, “v”:”2.0.0” }`
		参数说明：
		p：产品 ID。
		v：MCU 版本号。
校验和	1	从帧头开始按字节求和，将得出的结果对 256 求余

示例：0x55aa 02 N 01 00 1c 7b2270223a2241497031386b4c49222c2276223a22312e302e30227d xx

6.3 报告模组网络状态

网络状态是指 Zigbee 模组的网络的状态，当模组配网成功之后，即设备已加入网络，不因网关断电，父节点丢失等原因变更网络状态。当模组的网络状态发生变化，则主动下发模组网络状态至 MCU 。

模组发送

字段	长度（字节）	说明
帧头	2	0x55aa
版本	1	0x02
序列号	2	N
命令字	1	0x02
数据长度	2	0x0001
数据	1	模组网络状态：
		0x00：未入网。通常以下状态被识别为未入网。
		设备第一次上电
		设备入网失败
		设备离线
		0x01：已入网
		0x02：网络异常
		0x03：配网中
校验和	1	从帧头开始按字节求和，将得出的结果对 256 求余

示例：0x55aa 02 N 02 0001 00 xx

MCU 返回

字段	长度（字节）	说明
帧头	2	0x55aa
版本	1	0x02
序列号	2	N
命令字	1	0x02
数据长度	2	0x0000
数据	0	无
校验和	1	从帧头开始按字节求和，将得出的结果对 256 求余

示例：`0x55aa 02 N 02 0000 xx`

6.4 查询模组网络状态

支持 MCU 端查询 Zigbee 模组当前的网络状态。

> **注意**：开启本功能前，请测试当前固件是否支持。

MCU 发送

字段	长度（字节）	说明
帧头	2	0x55aa
版本	1	0x02
序列号	2	N
命令字	1	0x20
数据长度	2	0x0000
校验和	1	从帧头开始按字节求和，将得出的结果对 256 求余

示例：0x55aa 02 N 20 0000 xx

模组返回

字段	长度（字节）	说明
帧头	2	0x55aa
版本	1	0x02
序列号	2	N
命令字	1	0x20
数据长度	2	0x0001
数据	1	模组网络状态： 0x00：未入网。通常以下状态被识别为未入网。 设备第一次上电 设备入网失败 设备离线 0x01：已入网 0x02：网络异常 0x03：配网中
校验和	1	从帧头开始按字节求和，将得出的结果对 256 求余

示例：0x55aa 02 N 20 0001 xx xx

6.5 配置 Zigbee 模组

MCU 发送

字段	长度（字节）	说明
帧头	2	0x55aa
版本	1	0x02
序列号	2	N
命令字	1	0x03
数据长度	2	0x0001
数据	0	
		0x00：将模组软件复位
		0x01：将模组配置为开始配网状态（先离网再配网）
校验和	1	从帧头开始按字节求和，将得出的结果对 256 求余

示例：`0x55aa 02 N 03 0001 01 xx`

模组返回

字段	长度（字节）	说明
帧头	2	0x55aa
版本	1	0x02
序列号	2	N
命令字	1	0x03
数据长度	2	0x0000
数据	0	无
校验和	1	从帧头开始按字节求和，将得出的结果对 256 求余

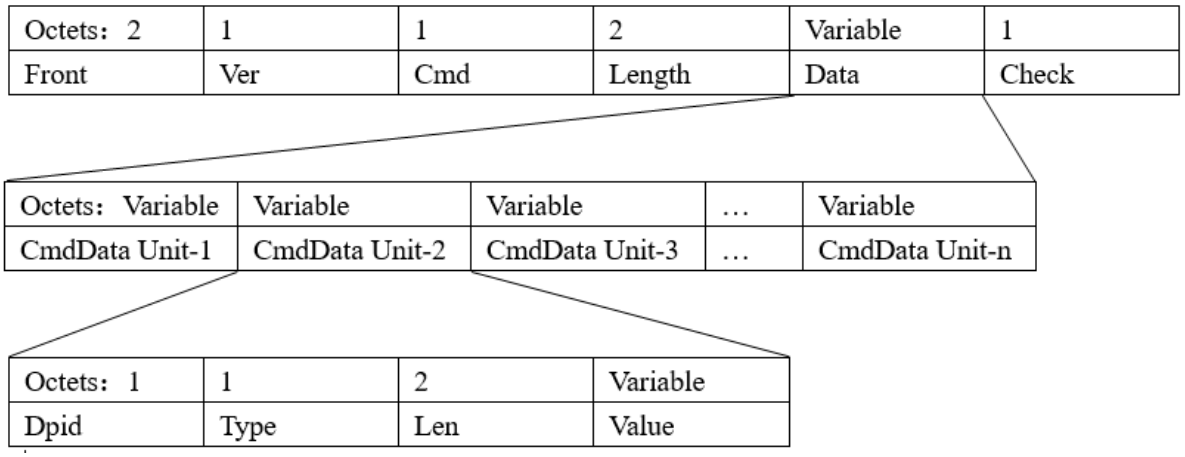
示例：`0x55aa 02 N 03 0000 xx`

6.6 命令下发

Zigbee 模组支持命令下发。

- 功能点命令或状态数据部分除 `raw` 类型外，其他类型均属于 `obj` 型。
- `obj` 型功能点支持下发多条命令。
- 命令下发为异步处理协议，对应于 MCU 的状态上报。

命令下发帧格式：



功能点格式：

数据字段属性	长度（字节）	说明
dpid	1	功能点序号
type	1	对应开放平台上功能点具体的数据类型： raw：raw 型功能点（模组输入） 表示值：0x00 长度（字节）：N

数据字段属性	长度（字节）	说明
		bool：布尔型，取值为 0x00 或 0x01
		表示值：0x01
		长度（字节）：1
		value：对应 int 类型，大端表示
		表示值：0x02
		长度（字节）：4
		string：对应字符串
		表示值：0x03
		长度（字节）：N
		enum：枚举类型，取值范围 0~255
		表示值：0x04
		长度（字节）：1

数据字段属性	长度（字节）	说明
		bitmap：故障型，长度大于 1 字节时，大端表示
		表示值：0x05
		长度（字节）：1、2 或 4
len	2	长度对应 value 的字节数（大端）
value	1/2/4/N	hex 表示，大于 1 字节采用大端传输

模组发送

字段	长度（字节）	说明
帧头	2	0x55aa
版本	1	0x02
序列号	2	N
命令字	1	0x04
数据长度	2	无
数据	取决于具体数据	参见功能点格式
校验和	1	从帧头开始按字节求和，将得出的结果对 256 求余

示例：0x55aa 02 N 04 0005 03 01 0001 01 xx

> **说明**：03 01 0001 01 即对应 3 号功能点系统开关，使用 bool 型变量，开机数值为 1。

6.7 状态上报（被动）

当 MCU 接收模组端下发的命令，并执行相应动作后，需要将新的状态上报至模组端。状态正确执行之后，仅上报执行操作的功能点状态。

- 状态上报（被动）为同步处理协议，模组端接收功能点状态后，立即返回确认字符（ACK）至 MCU。
- 状态上报（被动）可包含多个 obj 型功能点命令。
- raw 类型数据不能和obj 型数据同时上报。

MCU 发送

字段	长度（字节）	说明
帧头	2	0x55aa
版本	1	0x02
序列号	2	N
命令字	1	0x05
数据长度	2	取决于具体数据
数据	取决于具体数据	参见功能点格式
校验和	1	从帧头开始按字节求和，将得出的结果对 256 求余

示例：0x55aa 02 N 05 00 08 05 02 0004 0000001e xx

> 说明：05 02 0004 0000001e对应 5 号功能点上报湿度，使用 value 型变量，湿度为 30℃。

模组返回

字段	长度（字节）	说明
帧头	2	0x55aa

字段	长度（字节）	说明
版本	1	0x02
序列号	2	N
命令字	1	0x05
数据长度	2	0x0001
数据	0	
		0x00：状态上报失败
		0x01：状态上报成功
校验和	1	从帧头开始按字节求和得出的结果对 256 求余

示例：`0x55aa 02 N 05 0001 01 xx`

6.8 状态上报（主动）

- MCU 主动检测到功能点状态有变化，或者 MCU 重启等情况下，需要将变化后的功能点状态发送至模组。
  - 正常变化时，只上报有变化的功能点状态。
  - 重启等异常情况下，需要上报所有的功能点状态。
- **状态上报（主动）**为异步处理协议，模组端收到 Zigbee 网关的回复之后，会将状态返回给 MCU 端。如果状态返回超时，或者返回失败，MCU 需要重新上报。
- **状态上报（被动）**可包含多个 `obj` 型功能点命令。
- `raw` 类型数据不能和`obj` 型数据同时上报。> **说明**：如果需要在配网成功之后上报功能点数据以同步 App 面板，建议增加 5 秒延时上报。

MCU 发送

字段	长度（字节）	说明
帧头	2	0x55aa

字段	长度（字节）	说明
版本	1	0x02
序列号	2	N
命令字	1	0x06
数据长度	2	取决于具体数据
数据	取决于具体数据	参见功能点格式
校验和	1	从帧头开始按字节求和，将得出的结果对 256 求余

示例：`0x55aa 02 N 06 08 05 02 0004 0000001e xx`

> **说明**：`05 02 0004 0000001e`对应 5 号功能点上报湿度，使用 value 型变量，湿度为 30℃。

模组返回

字段	长度（字节）	说明
帧头	2	0x55aa
版本	1	0x02
序列号	2	N
命令字	1	0x06
数据长度	2	0x0001
数据	0	
		0x00：状态上报失败
		0x01：状态上报成功
校验和	1	从帧头开始按字节求和得出的结果对 256 求余

示例：`0x55aa 02 N 06 0001 01 xx`

6.9 Zigbee 模组功能性测试

Zigbee 模组支持扫描指定信道的 RSSI 值，返回扫描结果和信号强度百分比。本命令必须在设备未配网情况下才可正常运行，单次测试完成之后必须重启模组。

> **注意**：默认使用 11 信道，**MCU 发送**时，直接选择 11 信道。

MCU 发送

字段	长度（字节）	说明
帧头	2	0x55aa
版本	1	0x02
序列号	2	N
命令字	1	0x08
数据长度	2	0x0001
数据	1	信道值（11~26）
校验和	1	从帧头开始按字节求和得出的结果对 256 求余

示例：`0x55aa 02 N 08 0001 0b xx` 即 MCU 要求模组扫描 11 信道的 RSSI 值。

模组返回

字段	长度（字节）	说明
帧头	2	0x55aa
版本	1	0x02
序列号	2	N
命令字	1	0x08
数据长度	2	0x0002
数据	2	
		Data[0]:

字段	长度（字节）	说明
		0x00：失败
		0x01：成功
		Data[1]：
		Data[0] 为 0x00：Data[1] 表示信号强度 (0-100, 0 信号最差，100 信号最强)
		Data[0] 为 0x01：
		Data[1] 为 0x00：表示未扫描到指定的 RSSI
		Data[1] 为 0x01：表示模组未烧录授权 Key
校验和	1	从帧头开始按字节求和得出的结果对 256 求余

示例：`0x55aa 02 N 08 0002 01 64 xx`

> **说明**：无效信道取默认 11 信道。

6.10 时间同步

时间同步功能将网关的网络时间同步至 MCU。

MCU 发送

字段	长度（字节）	说明
帧头	2	0x55aa
版本	1	0x02
序列号	2	N
命令字	1	0x24

字段	长度（字节）	说明
数据长度	2	0x0000
数据	0	无
校验和	1	从帧头开始按字节求和得出的结果对 256 求余

模组返回

字段	长度（字节）	说明
帧头	2	0x55aa
版本	1	0x02
序列号	2	N
命令字	1	0x024
数据长度	2	0x0002
数据	2	数据长度为 8 字节的时间值，格式为标准时间戳 (4 byte) + 本地时间戳 (4 byte)
校验和	1	从帧头开始按字节求和得出的结果对 256 求余

- 标准时间戳为格林威治时间 1970 年 01 月 01 日 00 时 00 分 00 秒起至现在的总秒数。
- 本地时间戳为标准时间戳 + 标准时间和本地时间相差的秒数（包含时区和夏令时）。

7 场景开关协议

在场景开关设备中，MCU 只需要通过串口透传协议告知 Zigbee 模组设备按键的个数以及当前操作的按键。

7.1 查询按键信息

模组重启后会发送查询按键信息命令。

- 支持的按键上限为 10，即可创建 10 个场景。
- 仅适用于场景面板，不支持其他功能点和自定义功能点。
- 创建产品时，需要选择的功能点为场景 ID 组和场景编号（1~10）。
- 场景编号必须从小到大，从场景 1 开始。

模组发送

字段	长度（字节）	说明
帧头	2	0x55aa
版本	1	0x02
序列号	2	N
命令字	1	0x09
数据长度	2	0x0000
数据	0	无
校验和	1	从帧头开始按字节求和，将得出的结果对 256 求余

示例：`0x55aa 02 N 09 0000 xx`

MCU 返回

字段	长度（字节）	说明
帧头	2	0x55aa
版本	1	0x02

字段	长度（字节）	说明
序列号	2	N
命令字	1	0x09
数据长度	2	0x0001
数据	2	面板开关的按键总个数
校验和	1	从帧头开始按字节求和得出的结果对 256 求余

示例：`0x55aa 02 N 09 0001 02 xx`

7.2 场景唤醒命令

场景唤醒命令能够触发场景面板执行场景化操作。MCU 发送请求后，模组返回的状态如下。

- 成功：该按键已在 App 绑定场景，且对应的场景已成功执行。
- 失败：该按键未在 App 绑定场景，场景不会被执行。

当按键按下时，场景面板还会向网关发送一个按键值，用于联动云端场景。当 MCU 有按键上报时，即会上报按键给网关，即如果该场景面板仅使用云端场景功能，模组无论回复成功和失败都可以认为 MCU 上报按键成功。

云端场景和本地场景的区别

- 本地场景：即标准的 Zigbee 场景，满足 Zigbee 协议。注意，目前设备端保存的场景的数据为指定属性值，部分命令不支持的。这些不支持的功能需要通过云端场景实现。
- 云端场景：其本质是云端联动控制，是指通过云端功能实现的场景。

MCU 发送

字段	长度（字节）	说明
帧头	2	0x55aa
版本	1	0x02
序列号	2	N

字段	长度（字节）	说明
命令字	1	0x0A
数据长度	2	0x0001
数据	取决于具体数据	按键 ID。说明：例如按键总个数为 4，则数据为 01 02 03 04。
校验和	1	从帧头开始按字节求和得出的结果对 256 求余

示例：0x55aa 02 N 0A 0001 01 xx 即 MCU 要求模组执行按键 1 对应的场景。

模组返回

字段	长度（字节）	说明
帧头	2	0x55aa
版本	1	0x02
序列号	2	N
命令字	1	0x0A
数据长度	2	0x0001
数据	1	
		0：场景唤醒失败
		1：场景唤醒成功
校验和	1	从帧头开始按字节求和得出的结果对 256 求余

示例：0x55aa 02 N 0A 0001 01 xx

8 MCU OTA 协议

OTA 的流程如下：

1. 云端发出 OTA 通知。
2. MCU 接收到通知后回复通知。
3. MCU 开始发起数据请求，数据请求的包大小最大为 50 字节，
4. 模组会将该请求转发给网关。
5. 网关根据当前偏移量和数据包大小回复数据。

为了完善 MCU 端数据请求逻辑需要增加超时机制。即当发出数据请求在一段时间内没有回复时，需要重新发送该请求。

> **说明**：建议设置超时时长为 3~5 秒，超时次数为 5 次。即当连续 5 次及以上的响应时长超过 3~5 秒，则认为 OTA 升级异常，取消 OTA 升级。

8.1 OTA 版本请求的数据格式

若支持 MCU 升级必须实现本命令。支持网关会主动查询和 MCU 主动上报两种方式。

- 网关查询场景：
  - 配网成功
  - MCU 升级过程异常
- MCU 上报场景：
  - 配网成功后（必须添加）
  - 升级结束

模组发送

字段	长度（字节）	说明
帧头	2	0x55aa
版本	1	0x02
业务序列号（Seq）	2	模组产生
命令字	1	0x0B
数据长度	2	0x0000

字段	长度（字节）	说明
数据	0	无
校验和	1	从帧头开始按字节求和得出的结果对 256 求余

示例：`0x55 AA 02 00 f0 0B 00 00 XX`

MCU 返回

字段	长度（字节）	说明
帧头	2	0x55aa
版本	1	0x02
业务序列号（Seq）	2	MCU 下发的序号
命令字	1	0x0B
数据长度	2	0x0001
数据	1	当前版本版本号。例如（Bits）`01.00.0001` 表示 `1.0.1` 。
校验和	1	从帧头开始按字节求和得出的结果对 256 求余

示例：`55 AA 02 00 39 0B 00 01 40 XX`

> **注意**：OTA 相关命令用单字节表示 MCU 版本时，由于字节长度限制，最大版本号为 `3.3.15`。

8.2 OTA 升级通知

模组发送

字段	长度（字节）	说明
帧头	2	0x55aa
版本	1	0x02
业务序列号（Seq）	2	模组产生
命令字	1	0x0C
数据长度	2	0x0011
数据	8	PID。Data[0]~ Data[7]。
数据	1	当前版本版本号。例如（Bits）01.00.0001 表示 1.0.1 。
数据	4	固件大小。最大为 256 K。
数据	4	固件校验和：从固件第一个字节按字节求和，得出的结果对 256 求余。
校验和	1	从帧头开始按字节求和，将得出的结果对 256 求余。

示例：0x55 AA 02 00 1C 0C 00 0F 30 31 32 33 34 35 36 37 40 00 01 00 00 30 31 32 33 XX

MCU 返回

字段	长度（字节）	说明
帧头	2	0x55aa
版本	1	0x02
业务序列号（Seq）	2	模组下发的序列号
命令字	1	0x0C
数据长度	2	0x0001
数据	1	

字段	长度（字节）	说明
		0x00：OK
		0x01：error
校验和	1	从帧头开始按字节求和得出的结果对 256 求余

示例：`0x55 AA 02 00 1C 0C 00 01 00 XX`

8.3 OTA 固件内容请求

MCU 发送

字段	长度（字节）	说明
帧头	2	0x55aa
版本	1	0x02
业务序列号（Seq）	2	0x0000
命令字	1	0x0D
数据长度	2	0x000E
数据	8	PID
数据	1	当前版本版本号。例如（Bits）`01.00.0001` 表示 `1.0.1` 。
数据	4	数据包的偏移量（固件的位置）
数据	1	数据包的大小（最大 50 字节）
校验和	1	从帧头开始按字节求和得出的结果对 256 求余

示例：0x55 AA 02 00 00 0D 00 0E 30 31 32 33 34 35 36 37 40 00 00 00 01 32 XX

模组响应

字段	长度（字节）	说明
帧头	2	0x55aa
版本	1	0x02
业务序列号（Seq）	2	0x0000
命令字	1	0x0D
数据长度	2	0x0006+N
数据	1	
		0x00：成功
		0x01：失败
数据	8	PID
数据	1	当前版本版本号。例如（Bits）01.00.0001 表示 1.0.1 。
数据	4	数据包的偏移量（固件的位置）
数据	N	数据
校验和	1	从帧头开始按字节求和得出的结果对 256 求余

示例：55 AA 02 00 39 0D XXXX 00 30 31 32 33 34 35 36 37 40 00 00 00 01 …. XX

8.4 OTA 固件升级结果上报

MCU 发送

字段	长度（字节）	说明
帧头	2	0x55aa
版本	1	0x02
业务序列号（Seq）	2	MCU 下发的序列号
命令字	1	0x0E
数据长度	2	0x000A
数据	1	
		0x00：成功
		0x01：失败
数据	8	PID
数据	1	当前版本版本号。例如（Bits）01.00.0001 表示 1.0.1 。
校验和	1	从帧头开始按字节求和得出的结果对 256 求余

示例：0x55 AA 03 00 f0 0E 00 0A 00 30 31 32 33 34 35 36 37 40 26

模组响应

字段	长度（字节）	说明
帧头	2	0x55aa
版本	1	0x02
业务序列号（Seq）	2	MCU 下发的序列号
命令字	1	0x0E
数据长度	2	0x0001
数据	1	
		0x00：OK

字段	长度（字节）	说明
		0x01：error
校验和	1	从帧头开始按字节求和得出的结果对 256 求余

示例：0x55 AA 02 00 1C 0E 00 01 00 XX

8.5 MCU 广播数据

MCU 端需要将数据进行全网通知时，使用该帧数据。

> **说明**：广播之间需要有一定的时间间隔，间隔由网络的规模决定。

MCU 广播数据可以让全网络中的设备接收到。如果存在低功耗设备，该设备需要处于周期唤醒的状态，且唤醒周期需要小于广播周期。否认在唤醒前，下一条广播数据就会将之前的广播数据覆盖。

MCU 发送

字段	长度（字节）	说明
帧头	2	0x55aa
版本	1	0x02
序列号	2	N
命令字	1	0x27
数据长度	2	N
数据	取决于具体数据	具体功能点格式
校验和	1	从帧头开始按字节求和得出的结果对 256 求余

模组返回

字段	长度（字节）	说明
帧头	2	0x55aa
版本	1	0x02
序列号	2	N
命令字	1	0x027
数据长度	2	0x0001
数据	1	00 上报失败 01 上报成功
校验和	1	从帧头开始按字节求和得出的结果对 256 求余

8.6 MCU 配置 Zigbee 网络策略参数

本指令可以在接收到模组发送的查询 PID 帧后，进行毫秒级延时，然后发送。

> **注意**：该命令接收完成并成功应答后，模组将执行重启。

- 心跳时间

  心跳时间是用来维护设备和网关之间的数据链路是否正常的手段，强电设备的心跳时间默认为 150+random（30）秒，低功耗设备的心跳时间默认为 4 小时，且网关判定 12 小时内没有收到心跳则认为设备离线。仅支持低功耗设备的心跳时间支持修改。

- 超时时间

  当 **MCU 发送**配网指令之后，模组会执行一段时间的配网操作，并发送当前网络状态为配网状态。在一段时间内由于某些原因（例如附近没有开启配网的网络或者距离较远）导致模组没有加入到合适网络，则配网超时。配网超时之后，模组将处于未配网状态，同时也会将此状态发送给 MCU。

- 轮询（Poll）

  Poll 周期是指已经加入到网络的低功耗模组会在周期内唤醒。唤醒之后，低功耗模组会发送数据请求（Data request）至其父节点，用于告知父节点：其当前处于唤醒状态，父节点是否为其缓存数据。如果存在缓存数据，则父节点可以将数据发送给低功耗模组。

  - 不同产品 Poll 设置

    * 对实时性要求很高的产品：例如单火开关，可以将 Poll 值设置为 250ms。

    * 其他产品：例如传感器，只有当状态发生变化或者执行周期上报。即只需要上报数据时，就可以把 Poll 关闭，模组将收不到网关下发的控制指令。

  - 上电后 Poll 设置通常上电之后，设置一段时间的快速 Poll，可以在这个时间窗内将网关的配置命令下发。上电之后的快速 Poll 的时间默认为 30 秒，支持 MCU 设置。如果设备需要关闭 Poll，且有网关的配置需要下发，建议将快速 Poll 的时间窗增加。
  - 关闭 Poll 网关会缓存数据。当模组上报数据时，会携带数据请求（Data request），此时网关会将数据发下给模组。

> **注意**：Poll 值主要是影响功耗。唤醒周期越短，功耗越大。Poll 最小值为 200ms，小于最小值按照最小值处理。建议取值小于等于 8s。如果设置为 0 则关闭 Poll。

- 重连（Rejoin）

  Rejion 即重新加入到网络，是一种专门用于低功耗设备在父节点丢失时，重新加入的网络的一种机制。

> **说明**：Rejion 这里不是指配网，无须网关开启配网模式，

  模组和父节点的交互流程如下：

  1. 模组发送数据请求（Data request）。
  2. 父节点回复。
     - 有缓存数据：将数据发送给模组。
     - 无缓存数据：仅回复 ACK。

  如果模组发送数据请求（Data request），但由于环境、距离、父节点断电等因素导致模组没有收到确认字符（ACK），则认为模组的轮询（Poll）失败。当累加到的一定的值时（Poll 失败次数），则认为模组丢失父节点，需要触发重连（Rejion）。

> **说明**：如果在累加的过程中重新收到父节点的 ACK，则累加清零。

  - Rejion 触发目前提供 2 种独立的 Rejion 触发方式：

```
- 应用层发送数据
- 定时触发：模组处于离线状态，周期性触发 Rejion，直到模组重新加入到之前网络。
> **说明**：Rejion 成功只能表示其和父节点能够正常通信，数据能否到达网关需要根据网关和父节点的路由情况。目前这两个参数都可以
由 MCU 灵活配置。
```

```
- Rejion 间隔时间
即周期触发 Rejion 的时间间隔。针对对数据要求严格或者低功耗要求的场景，可以减小 Rejion 间隔，例如 3~5 秒。针对传感设
备或者通过数据上报触发的场景，可以增加时间间隔，例如 1 小时。
- Rejoin 尝试次数
指设备触发 Rejion 之、后，模组可以发送 Rejion 的次数。对于 Poll 时间较短的场合，以及 Rejion 间隔短的应用，
可以减少尝试次数，例如 1~2 次。对于 Rejion 间隔较长的场合，可以将 Rejion 尝试次数稍微增加，例如 3~4 次。
>**注意**：如果设置的参数值不在取值范围内的，参数值不生效。
```

MCU 发送

字段	长度（字节）	说明
帧头	2	0x55aa
版本	1	0x02
序列号	2	N
命令字	1	0x26
数据长度	2	0x0e
数据	2	心跳时间（秒） 仅低功耗设备支持心跳修改。低功耗设备的心跳默认为 4 小时，设置范围为 10~5*3600 秒。 0xffff：当前值 0xfffe：默认值
	2	配网超时时间（秒） 配网超时时间默认为 180 秒，取值范围为 30~600。 0xffff：当前值 0xfffe：默认值

字段	长度（字节）	说明
	2	Rejoin 间隔时间（秒）
		取值范围 3~3600，默认值为 180 秒。即当设备丢失父节点时，会间隔 180 秒尝试 Rejion。
		0xffff：当前值
		0xfffe：默认值
	2	Poll 时间（毫秒）
		默认为 5000ms ，取值范围 200~10000 。模组会间隔 Poll 周期唤醒一次，用于确认父节点是否有数据发送。如果产品为传感类型，仅有数据上传，则可以将其设置为 0。
		0：关闭 Poll
		0xffff：当前值
		0xfffe：默认值
	2	持续快速 Poll 的时间段(秒)
		取值范围 10~3000。Poll 上电之后持续快速 Poll 的时长。快速 Poll 的时长为 250ms，到达后按照 Poll 设置的时间运行, 默认为 30 秒。
		0xffff：当前值
		0xfffe：默认值
	1	Poll 失败次数

字段	长度（字节）	说明
		取值范围 3 ~ 40。当达到最大值时且配置了 Rejion 触发时间，则到达时间时会触发设备 Rejion。
		0xff：当前值
		0xfe：默认值
	1	应用数据发送是否触发 Rejion
		0 表示不触发，1 表示触发。默认值为 1。
		0xff：当前值
		0xfe：默认值
	1	Rejoin 尝试次数
		默认值为 1。取值范围为 1~10。
		0xff：当前值
		0xfe：默认值
	1	发射功率
		默认值为 11。取值范围 3~19dB。
		0xff：当前值
		0xfe：默认值
校验和	1	从帧头开始按字节求和，将得出的结果对 256 求余

模组返回

字段	长度（字节）	说明
帧头	2	0x55aa
版本	1	0x02
序列号	2	N
命令字	1	0x026
数据长度	2	0x0001
数据	1	
		0x00：失败
		0x01：成功
校验和	1	从帧头开始按字节求和得出的结果对 256 求余