**移远串口16进制协议V1.3.1**

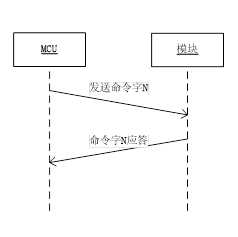
**串口通信约定**

* 波特率：115200
* 数据位：8
* 奇偶校验：无
* 停止位：1
* 数据流控：无
* MCU：用户控制板控制芯片，与移远模组通过串口进行通信

**帧格式**

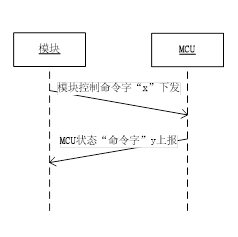
| **字段** | **字节数** | **说明** |
| --- | --- | --- |
| 帧头 | 2 | 固定为 0x55aa |
| 版本 | 1 | 升级扩展用 |
| 命令字 | 1 | 具体帧类型 |
| 数据长度 | 2 | 大端 |
| 数据 | N | 实体数据 |
| 校验和 | 1 | 从帧头开始，按字节求和，得出的结果对 256 求余 |

* 所有大于 1 个字节的数据均采用大端模式传输。
* 一般情况下，采用同命令字一发一收同步机制。  
  即一方发出命令，另一方应答，若发送方超时未收到正确的响应包，则超时传输，如下图所示：

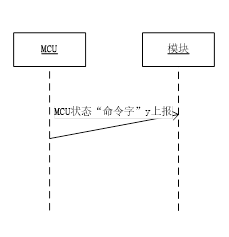


具体通信方式以 **协议列表** 章节中为准。

* 模组控制命令下发及 MCU 状态上报则采用异步模式，假设模组控制命令下发的命令字为 x，MCU 状态上报的命令字为 y，如下所示：
* 模组控制命令下发：



* MCU 状态上报：



* 版本字段

版本用于拓展功能使用，为兼容新老版本协议，模组发送版本号均为 0x00，根据 MCU 回复版本号确定使用协议的版本。新用户默认使用协议版本为 0x03 即可。

**联网模组串口缓冲区大小**

| **芯片平台** | **接收缓冲区大小** | **发送缓冲区大小** | **备注** |
| --- | --- | --- | --- |
| FC41D | 最小为 1024 字节 | 最小为 1024 字节 |  |

MCU 完整数据大小（即从**帧头**开始计算到**校验和**结束计算）不可超出模组接收缓冲区大小值。如果数据包长度超出，则会出现数据通信异常情况。

**状态数据单元**

命令和状态数据单元（datapoint）如下所示：

| **数据段** | **字节数** | **说明** |
| --- | --- | --- |
| dpid | 2 | datapoint 序号,大端传输 |
| type | 1 | 参考下表中 type 字段说明 |
| len | 2 | 长度对应 value 的字节数，参考下表中 type 字段说明 |
| value | 1/2/4/N/L | 用 16 进制表示，大于 1 字节采用大端传输 |

datapoint 中的 type 字段说明：

| **type** | **数据类型** | **字节数** | **说明** |
| --- | --- | --- | --- |
| 0x00 | Raw | N | 对应RAW物模型，用于上报纯二进制数据 |
| 0x01 | Bool | 1 | 对应Bool物模型，范围：0x00/0x01 |
| 0x02 | Value | 4 | 对应Int或者枚举物模型，采用大端表示 |
| 0x03 | String | N | 对应TEXT物模型，字符串数据 |
| 0x05 | FAULT | 1/2/4 | 会将数据的每一Bit作为bool型上报，如上报长度大于 1 字节时，大端表示 |
| 0x11 | Double | 8 Byte | 大端，对应Double/Float类物模型，用于传递小数（Float类物模型需要强转为Double格式），遵循格式与C语言遵循的 IEEE 754标准一致。 |
| 0x12 | Struct | L Byte | 对应结构体/数组类型物模型，该类物模型为多种物模型的组合。L为结构体所有成员依序排列后所占用的字节长度。 |

* 状态数据可含多个 datapoint 命令数据单元。

**协议列表**

**心跳检测**

* Wi-Fi 模组上电后，将会以 1 秒/次的频率持续发送心跳包，等待 MCU 回复。模组若收正确的心跳包回复，心跳包发送间隔变为 15 秒/次，继续执行其余初始化指令。模组若未收到正确回复，将保持 1 秒/次心跳包发送频率，直到收到正确回复。
* MCU 也可依据心跳定期检测模组是否正常工作。若模组无心跳下发，则 MCU 可通过模组提供的硬件复位引脚复位 Wi-Fi 模组。若模组在60秒内没有收到MCU心跳回复，则认为 MCU 离线。

**模组发送**

| **字段** | **字节数** | **说明** |
| --- | --- | --- |
| 帧头 | 2 | 0x55aa |
| 版本 | 1 | 0x00 |
| 命令字 | 1 | 0x00 |
| 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 |

发送示例：55 aa 00 00 00 00 ff

**MCU 返回**

| **字段** | **字节数** | **说明** |
| --- | --- | --- |
| 帧头 | 2 | 0x55aa |
| 版本 | 1 | 0x03 |
| 命令字 | 1 | 0x00 |
| 数据长度 | 2 | 0x0001 |
| 数据 | 1 | * 0x00： MCU 重启后第一次心跳返回值，仅发送一次，用于模组判断工作过程中 MCU 是否重启 * 0x01：除 MCU 重启后第一次返回 0 外，其余均返回此值 |
| 校验和 | 1 | 从帧头开始，按字节求和，得出的结果对 256 求余 |

* MCU 第一次返回示例：55 aa 03 00 00 01 00 03
* 除第一次外，正常返回：55 aa 03 00 00 01 01 04

**查询产品信息**

产品信息由ProductKey, ProductSecret 和 MCU 软件版本构成：

* ProductKey：即产品的唯一性标识，对应移远开发者中心对应产品 PID，由移远开发者中心平台生成，用于云端记录产品相关信息。
* ProductSecret：对应ProductKey的密钥，需要同时上报才会被移远云平台识别
* MCU 软件版本：采用点分十进制形式，格式为 x.x.x，其中 x 为十进制数，取值范围为 0-99。

**模组发送**

| **字段** | **字节数** | **说明** |
| --- | --- | --- |
| 帧头 | 2 | 0x55aa |
| 版本 | 1 | 0x00 |
| 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，得出的结果对 256 求余 |

**示例**：55 aa 00 01 00 00 00

**MCU 返回**

| **字段** | **字节数** | **说明** |
| --- | --- | --- |
| 帧头 | 2 | 0x55aa |
| 版本 | 1 | 0x03 |
| 命令字 | 1 | 0x01 |
| 数据长度 | 2 | N |
| 数据 | N | {“p”:“p11\*\*\*\_Qkh2\*\*\*\*\*\*\*\*bUpZ”, “v”:“1.0.0” } |
| 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**：{"p":"p11\*\*\*\_Qkh2\*\*\*\*\*\*\*\*bUpZ", "v":"1.0.0"，"new\_dp":1 }

产品信息示例字段说明:

| **字段** | **说明** |
| --- | --- |
| p | 表示ProductKey为 p11\*\*\*，ProductSecret为Qkh2\*\*\*\*\*\*\*\*bUpZ。两者之间通过下划线”\_”连接。 ProductKey和ProductSecret可通过开发者中心创建的产品中获取。 |
| v | 表示 MCU 版本为 1.0.0 |

**报告设备联网状态**

| **设备联网状态** | **描述** | **状态值** |
| --- | --- | --- |
| 状态 1 | Wi-Fi 配网状态（蓝牙处于配网状态） | 0x00 |
| 状态 3 | Wi-Fi 已配置但未连上路由器 | 0x02 |
| 状态 4 | Wi-Fi 已配置且连上路由器 | 0x03 |
| 状态 5 | 设备成功连接到云端 | 0x04 |

* 设备联网状态：
  + Wi-Fi 配网配置状态（蓝牙处于配网状态）
  + Wi-Fi 配置成功但未连上路由器
  + Wi-Fi 配置成功且连上路由器
  + 设备成功连接到云端。
* 当模组的联网状态发生变化，则主动下发设备联网状态至 MCU。

**模组发送**

| **字段** | **字节数** | **说明** |
| --- | --- | --- |
| 帧头 | 2 | 0x55aa |
| 版本 | 1 | 0x00 |
| 命令字 | 1 | 0x03 |
| 数据长度 | 2 | 0x0001 |
| 数据 | 1 | 指示 Wi-Fi 工作状态：   * 0x00：状态 1 * 0x02：状态 3 * 0x03：状态 4 * 0x04：状态 5 |
| 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 00 03 00 01 00 03

**MCU 返回**

| **字段** | **字节数** | **说明** |
| --- | --- | --- |
| 帧头 | 2 | 0x55aa |
| 版本 | 1 | 0x03 |
| 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，得出的结果对 256 求余 |

**示例**：55 aa 03 03 00 00 05

**重置 Wi-Fi**

配网相关命令，模组收到该指令后会重启，初始化相关命令交互完成后模组进入配网模式，可以使用 App 进行配网。

**MCU 发送**

| **字段** | **字节数** | **说明** |
| --- | --- | --- |
| 帧头 | 2 | 0x55aa |
| 版本 | 1 | 0x03 |
| 命令字 | 1 | 0x04 |
| 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 |

**示例**：55 aa 03 04 00 00 06

**模组返回**

| **字段** | **字节数** | **说明** |
| --- | --- | --- |
| 帧头 | 2 | 0x55aa |
| 版本 | 1 | 0x00 |
| 命令字 | 1 | 0x04 |
| 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 |

**示例**：55 aa 00 04 00 00 03

**命令下发**

* “命令下发可含多个 datapoint 状态数据单元。
* “命令下发为异步处理协议，通常下发数据解析完成后，MCU 会根据 datapoint 数据执行对应功能，若 datapoint 状态发生改变，MCU 还需使用状态上报命令

**模组发送**

| **字段** | **字节数** | **说明** |
| --- | --- | --- |
| 帧头 | 2 | 0x55aa |
| 版本 | 1 | 0x00 |
| 命令字 | 1 | 0x06 |
| 数据长度 | 2 | 取决于命令数据单元类型以及个数 |
| 数据 | N | [状态数据单元](https://developer.tuya.com/cn/docs/iot/tuya-cloud-universal-serial-port-access-protocol?id=K9hhi0xxtn9cb#StatusDataType) 组 |
| 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**：系统开关对应 3 号物模型，使用 bool 型变量，开机数值为 1

55 aa 00 06 00 06 00 03 01 00 01 01 11

**状态上报（异步）**

* datapoint 状态数据单元说明，请参考 [状态数据单元](https://developer.tuya.com/cn/docs/iot/tuya-cloud-universal-serial-port-access-protocol?id=K9hhi0xxtn9cb#StatusDataType)。
* 该协议为异步处理协议，状态上报触发机制有三类：
  + MCU 收到命令下发处理帧时，正确执行相应 datapoint 命令。再通过状态上报帧，将变化后的 datapoint 状态发送至模组。
  + MCU 主动检测到 datapoint 有变化，将变化后的 datapoint 状态发送至模组。
  + MCU 收到 状态查询 帧时，将所有的 datapoint 状态发送至模组。
* 状态上报可含多个 datapoint 状态数据单元。
* 建议在 datapoint 状态改变时上报即可，有利于保证数据传输稳定性。请避免在短时间内，持续循环进行数据上报。
* 在设备待机或者工作稳定状态下，请尽量控制重复 datapoint 的循环上报频率。建议最小上报间隔为 1 分钟。

**MCU 发送**

| **字段** | **字节数** | **说明** |
| --- | --- | --- |
| 帧头 | 2 | 0x55aa |
| 版本 | 1 | 0x03 |
| 命令字 | 1 | 0x07 |
| 数据长度 | 2 | 取决于 **状态数据单元**类型以及个数 |
| 数据 | N | [状态数据单元](https://developer.tuya.com/cn/docs/iot/tuya-cloud-universal-serial-port-access-protocol?id=K9hhi0xxtn9cb#StatusDataType) 组 |
| 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**：

* 湿度对应 5 号 物模型，使用 value 型变量，湿度为 30℃

55 aa 03 07 00 09 00 05 02 00 04 00 00 00 1e 3b

**状态上报（同步）**

* 此命令为同步指令，MCU 数据状态上报后，需要等待模组返回结果。
* 每次状态上报，模组都会有响应。模组未响应前，不可多次上报。
* 网络环境差、数据难以及时上报时，模组会在 5 秒后返回失败。此时，MCU 需要等待 5 秒以上。
* datapoint 状态数据单元说明，请参考 [状态数据单元](https://developer.tuya.com/cn/docs/iot/tuya-cloud-universal-serial-port-access-protocol?id=K9hhi0xxtn9cb#StatusDataType)。
* 状态上报可含多个 datapoint 命令数据单元。
* 建议在 datapoint 状态改变时上报即可，有利于保证数据传输稳定性。请避免在短时间内，持续循环进行数据上报。
* 在设备待机或者工作稳定状态下，请尽量控制重复 datapoint 的循环上报频率。建议最小上报间隔为 1 分钟。

**MCU 发送**

| **字段** | **字节数** | **说明** |
| --- | --- | --- |
| 帧头 | 2 | 0x55aa |
| 版本 | 1 | 0x03 |
| 命令字 | 1 | 0x22 |
| 数据长度 | 2 | 取决于 **状态数据单元** 类型以及个数 |
| 数据 | N | [状态数据单元](https://developer.tuya.com/cn/docs/iot/tuya-cloud-universal-serial-port-access-protocol?id=K9hhi0xxtn9cb#StatusDataType) 组 |
| 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**：上报 bool 型 dp2，状态为 true

55 aa 03 22 00 06 00 02 01 00 01 01 2f

**模组返回**

| **字段** | **字节数** | **说明** |
| --- | --- | --- |
| 帧头 | 2 | 0x55aa |
| 版本 | 1 | 0x00 |
| 命令字 | 1 | 0x23 |
| 数据长度 | 2 | 0x0001 |
| 数据 | Data | * 0x00：表示失败 * 0x01：表示成功 |
| 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**：55 aa 00 23 00 01 01 24

**状态查询**

* 状态查询为异步处理协议，主要用于模组查询 MCU 所有类型的 datapoint 状态，当 MCU 收到此帧时，通过状态上报帧上报 datapoint 状态。
* 状态查询发送时机主要有两种：
  + 模组首次上电，通过心跳与 MCU 建立连接后，查询发送。
  + 模组工作过程中检测到 MCU 重启或发生了离线再上线的过程，查询发送。

**模组发送**

| **字段** | **字节数** | **说明** |
| --- | --- | --- |
| 帧头 | 2 | 0x55aa |
| 版本 | 1 | 0x00 |
| 命令字 | 1 | 0x08 |
| 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 |

**示例**：55 aa 00 08 00 00 07

### 获取系统时间(格林时间)

* 格林时间不带有时区的因素，是一个国际标准的时间基准。
* 当模组连接上网络后，本地的时间戳校准完成才会返回成功，并携带有效的时间数据。
* 当接收到模组的 **报告设备联网连接状态** 状态值为 0x04（ 设备成功连接到云端）后，模组会执行时间戳校准功能。
* 模组时间戳校准的功能，需要一定的时间消耗，MCU 在收到模组对应状态通知后立刻获取时间，可能会出现获取失败情况，此时请重试。

**MCU 发送**

| **字段** | **字节数** | **说明** |
| --- | --- | --- |
| 帧头 | 2 | 0x55aa |
| 版本 | 1 | 0x03 |
| 命令字 | 1 | 0x0c |
| 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 |

**示例**：55 aa 03 0c 00 00 0e

**模组返回**

| **字段** | **字节数** | **说明** |
| --- | --- | --- |
| 帧头 | 2 | 0x55aa |
| 版本 | 1 | 0x00 |
| 命令字 | 1 | 0x0c |
| 数据长度 | 2 | 0x0007 |
| 数据 | 7 | 数据长度为 7 字节:   * Data[0] 为是否获取时间成功标志，为 0 表示失败，为 1 表示成功 * Data[1] 为年份， 0x00 表示 2000 年 * Data[2] 为月份，从 1 开始到 12 结束 * Data[3] 为日期，从 1 开始到 31 结束 * Data[4] 为时钟，从 0 开始到 23 结束 * Data[5] 为分钟，从 0 开始到 59 结束 * Data[6] 为秒钟，从 0 开始到 59 结束 |
| 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**：格林时间 2016 年 4 月 19 日 5 时 6 分 7 秒

55 aa 00 0c 00 07 01 10 04 13 05 06 07 4c

### 获取本地时间

* 本地时间是在格林时间的基础上加上当地(设备所在地)时区。设备所在地的时区通过当前设备的IP获取
* 当模组连接上网络后，本地的时间戳校准完成才会返回成功，并带有有效的时间数据。
* 当接收到模组的 报告**设备联网连接状态** 状态值为 0x04（ 设备成功连接到云端）后，模组会执行时间戳校准功能。
* 模组时间戳校准的功能，需要一定的时间消耗，MCU 在收到模组对应状态通知后立刻获取时间，可能会出现获取失败情况，此时请重试。

**MCU 发送**

| **字段** | **字节数** | **说明** |
| --- | --- | --- |
| 帧头 | 2 | 0x55aa |
| 版本 | 1 | 0x03 |
| 命令字 | 1 | 0x1c |
| 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 |

**示例**：55 aa 03 1c 00 00 1e

**模组返回**

| **字段** | **字节数** | **说明** |
| --- | --- | --- |
| 帧头 | 2 | 0x55aa |
| 版本 | 1 | 0x00 |
| 命令字 | 1 | 0x1c |
| 数据长度 | 2 | 0x0008 |
| 数据 | 8 | 数据长度为 8 字节:   * Data[0] 为是否获取时间成功标志，为 0 表示失败，为 1 表示成功 * Data[1] 为年份， 0x00 表示 2000 年 * Data[2] 为月份，从 1 开始到 12 结束 * Data[3] 为日期，从 1 开始到 31 结束 * Data[4] 为时钟，从 0 开始到 23 结束 * Data[5] 为分钟，从 0 开始到 59 结束 * Data[6] 为秒钟，从 0 开始到 59 结束 * Data[7] 为星期，从 1 开始到 7 结束，1 代表星期一 |
| 校验和 | 1 | 从帧头开始，按字节求和，得出的结果对 256 求余 |

* 如设备所使用的网络在国内，则当地时间为北京时间(东 8 区)。  
  例：北京时间 2016 年 4 月 19 日 5 时 6 分 7 秒  
  55 aa 00 1c 00 08 01 10 04 13 05 06 07 02 5f
* 如果设备在国外使用，则当地时间为设备所处时区时间。

**Wi-Fi 功能性测试（扫描指定路由）**

* 模组内部目前扫描指定的 SSID：quectel\_mdev\_test，返回扫描结果和信号强度百分比。
* 这里为了最大程度防止不良品这里建议客户将路由于设备距离控制在 5 米左右，信号强度大于等于 60%为合格，这里可以根据自己产线和工厂环境的情况自行调整。

**MCU 发送**

| **字段** | **字节数** | **说明** |
| --- | --- | --- |
| 帧头 | 2 | 0x55aa |
| 版本 | 1 | 0x03 |
| 命令字 | 1 | 0x0e |
| 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 |

**示例**：55 aa 03 0e 00 00 10

**模组返回**

| **字段** | **字节数** | **说明** |
| --- | --- | --- |
| 帧头 | 2 | 0x55aa |
| 版本 | 1 | 0x00 |
| 命令字 | 1 | 0x0e |
| 数据长度 | 2 | 0x0002 |
| 数据 | 2 | 数据长度为 2 字节：   * Data[0]: 0x00 失败，0x01 成功。 * 当 Data[0]为 0x01，即成功时，Data[1]表示信号强度 (0-100，0 信号最差100 信号最强)。 * 当 Data[0]为 0x00，即失败时，Data[1]为 0x00 表示未扫描到指定的 SSID，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**：未扫描到指定的 ssid：55 aa 00 0e 00 02 00 00 0f

**获取模组内存**

获取 Wi-Fi 模组剩余内存

**MCU 发送**

| **字段** | **字节数** | **说明** |
| --- | --- | --- |
| 帧头 | 2 | 0x55aa |
| 版本 | 1 | 0x03 |
| 命令字 | 1 | 0x0f |
| 数据长度 | 2 | 0x0000 |
| 数据 | Data | 无 |
| 校验和 | 1 | 从帧头开始，按字节求和，得出的结果对 256 求余 |

例：55 aa 03 0f 00 00 11

**模组返回**

| **字段** | **字节数** | **说明** |
| --- | --- | --- |
| 帧头 | 2 | 0x55aa |
| 版本 | 1 | 0x00 |
| 命令字 | 1 | 0x0f |
| 数据长度 | 2 | 0x0004 |
| 数据 | 4 | 数据长度为 4 字节，大端格式： 如 0x00 0x00 0x28 0x00 代表剩余 10240 字节内存 |
| 校验和 | 1 | 从帧头开始，按字节求和，得出的结果对 256 求余 |

例：MemorySize: 53328 Byte：55 aa 00 0f 00 04 50 d0 00 00 32

**获取当前 Wi-Fi 信号强度（可选）**

**MCU 发送**

| **字段** | **字节数** | **说明** |
| --- | --- | --- |
| 帧头 | 2 | 0x55aa |
| 版本 | 1 | 0x03 |
| 命令字 | 1 | 0x24 |
| 数据长度 | 2 | 0 |
| 数据 | N | 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**：55 aa 03 24 00 00 26

**模组返回**

| **字段** | **字节数** | **说明** |
| --- | --- | --- |
| 帧头 | 2 | 0x55aa |
| 版本 | 1 | 0x00 |
| 命令字 | 1 | 0x24 |
| 数据长度 | 2 | 0x0001 |
| 数据 | Data | * 0x00：表示失败 * 小于 0：表示信号强度，如(-60db) |
| 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**：RSSI: -20：55 aa 00 24 00 01 ec 10

**获取当前 Wi-Fi 联网状态**

| **设备联网状态** | **描述** | **状态值** |
| --- | --- | --- |
| 状态 1 | Wi-Fi 配网状态 | 0x00 |
| 状态 3 | Wi-Fi 已配置但未连上路由器 | 0x02 |
| 状态 4 | Wi-Fi 已配置且连上路由器 | 0x03 |
| 状态 5 | 已连上路由器且连接到云端 | 0x04 |

与[报告设备联网状态](https://developer.tuya.com/cn/docs/iot/tuya-cloud-universal-serial-port-access-protocol?id=K9hhi0xxtn9cb#MonitorDeviceConnection)保持一致。

**MCU 发送**

| **字段** | **字节数** | **说明** |
| --- | --- | --- |
| 帧头 | 2 | 0x55aa |
| 版本 | 1 | 0x03 |
| 命令字 | 1 | 0x2B |
| 数据长度 | 2 | 0x0000 |
| 数据 | Data | 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**： 55 aa 03 2b 00 00 2d

**模组返回**

| **字段** | **字节数** | **说明** |
| --- | --- | --- |
| 帧头 | 2 | 0x55aa |
| 版本 | 1 | 0x00 |
| 命令字 | 1 | 0x2B |
| 数据长度 | 2 | 0x0001 |
| 数据 | 1 | * 0x00 ：Wi-Fi 配网配置状态 * 0x02 ：Wi-Fi 已配置但未连上路由器 * 0x03 ：Wi-Fi 已配置且连上路由器 * 0x04 ：已连上路由器且连接到云端 |
| 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**：状态 4(已连上路由器且连接到云端)： 55 aa 00 2b 00 01 04 2f

**获取模组 MAC**

**MCU 发送**

| **字段** | **字节数** | **说明** |
| --- | --- | --- |
| 帧头 | 2 | 0x55aa |
| 版本 | 1 | 0x03 |
| 命令字 | 1 | 0x2d |
| 数据长度 | 2 | 0x0000 |
| 数据 | Data | 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**：55 aa 03 2d 00 00 2f

**模组返回**

| **字段** | **字节数** | **说明** |
| --- | --- | --- |
| 帧头 | 2 | 0x55aa |
| 版本 | 1 | 0x00 |
| 命令字 | 1 | 0x2d |
| 数据长度 | 2 | 0x0007 |
| 数据 | Data | * data[0]：   + 获取 MAC 地址是是否成功的标志：   + 0x00 表示成功，表示后面 6 字节的 MAC 地址有效   + 0x01 表示获取 MAC 失败，表示后面 6 字节的 MAC 无效 * data[1]~data[6]：当获取 MAC 地址标志位 data[0]表示成功则表示模组有效的 MAC 地址 |
| 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**：MAC: 508A0E3A2D9

55 aa 00 2d 00 07 00 50 8a 06 e3 a2 d9 71