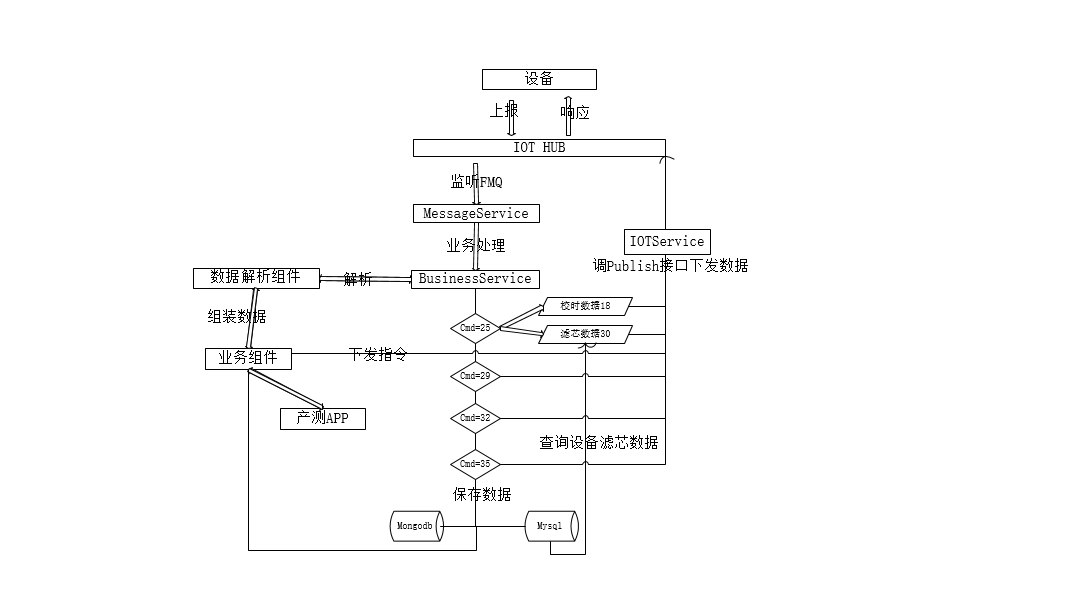
# 对接IOT Hub模块

1. **设备指令模块处理**
2. 流程图



1. 设备主动上报数据处理流程：
2. MsgConsumer类监听fmq消息,由MessageService类的receive处理消息,消息组件只接受消息不处理，具体业务由BusinessService类的process()处理;其中根据msgType=1时表示为设备上下线消息，更新mysql t\_device\_info 表 is\_online字段，msgType=0表示普通数据上报，需要根据下面的流程继续处理.

入参:

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 类型 | 必传 | 说明 |
| message | String | Y | 消息数据 |

出参:

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 类型 | 必传 | 说明 |
|  |  |  |  |

1. 调用解析模块AnalysisData类处理message数据解析，返回解析后的数据

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 类型 | 必传 | 说明 |
| message | String | Y | 消息数据 |

出参:

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 类型 | 必传 |  |
| result | String | Y | Json字符串 |

result 说明:

{

“code”:”命令中的具体操作，比如校准时间，下发滤芯数据”,

“cmd”:“指令编号”，

“msg”:”处理的结果内容”，

“data”:{ ---对应的数据实体

}

}

1. 根据cmd执行具体的业务处理，下列指令需要返回设备信息，调用IotHubService的publish()处理，具体接口见第二部分IOT对接部分

cmd=25

当d1 = 1 触发校时指令，封装系统当前时间 由数据解析模块解析处理后，调用IotHubService的publish()方法反馈设备

|  |  |  |
| --- | --- | --- |
| 服务器发送 | {  "flag": 242,  "version": 1,  "cmd": 18,  "gprs": 1,  "addr": "192.168.0.1",  "barcodeid": "12345678",  "len": 6,  "d1":2019,  "d2":4,  “d3”:2  "d4":00,  “d5”:00,  “d6”:00  } | d1-d6表示年月日时分秒 |
| 设备返回 | {  "flag": 242,  "version": 1,  "cmd": 18,  "gprs": 1,  "addr": "经纬度",  "barcodeid": "12345678",  "len": 1,  "d1":“0x00”  } | d1:校时结果  0xff 设置成功  0x00 设置失败 |

d1=2 触发滤芯数据下发指令,查询mysql 中表 t\_device\_status\_info; 获取当前设备滤芯最新一条数据返回设备;

|  |  |  |
| --- | --- | --- |
| 服务器发送 | {  "flag": 242,  "version": 1,  "cmd": 30,  "gprs": 1,  "addr": "192.168.0.1",  "barcodeid": "12345678",  "len": 20,  "d1":滤芯1最大寿命，  "d2":滤芯1最大流量，  "d3":滤芯1剩余寿命，  "d4":滤芯1剩余流量，  "d5":滤芯2最大寿命，  "d6":滤芯2最大流量，  "d7":滤芯2剩余寿命，  "d8":滤芯2剩余流量，  "d9":滤芯3最大寿命，  "d10":滤芯3最大流量，  "d11":滤芯3剩余寿命，  "d12":滤芯3剩余流量，  "d13":滤芯4最大寿命，  "d14":滤芯4最大流量，  "d15":滤芯4剩余寿命，  "d16":滤芯4剩余流量，  "d17":滤芯5最大寿命，  "d18":滤芯5最大流量，  "d19":滤芯5剩余寿命，  "d20":滤芯5剩余流量  } |  |
| 设备返回 | {  "flag": 242,  "version": 1,  "cmd": 30,  "gprs": 1,  "addr": "经纬度",  "barcodeid": "12345678",  "len": 1,  "d1":” 0x00”  } | d1:接收结果  0x00—失败  0xff—成功 |

Cmd=29 当机器状态转换的时候，例如从制水状态转入满水状态的时候，设备主动上传此条数据指令,服务器需要返回信息确认，如果服务器没有返回数据，会重复发送一次。

|  |  |  |
| --- | --- | --- |
| 服务器返回 | {  "flag": 242,  "version": 1,  "cmd": 29,  "gprs": 1,  "addr": "192.168.0.1",  "barcodeid": "12345678",  "len": 1,  "d1":设备发送的d1  } | d1:  0x01： 制水  0x02: 满水  其他： 保留 |

Cmd=32 表示请求升级文件 如果连续三次请求数据不成功，返回升级失败。

|  |  |  |
| --- | --- | --- |
| 服务器返回 | {  "flag": 242,  "version": 1,  "cmd": 32,  "gprs": 1,  "addr": "192.168.0.1",  "barcodeid": "12345678",  "len": 5,  "d1":主板0/小板1，  "d2":供应商，  "d3":版本，  "d4":文件片序号，  "d5":文件片字节数，  "d6":文件数据  } |  |

cmd=35 滤芯复位完成之后，退出滤芯复位模式之后，设备主动上报此条指令(只针对复位按键复位滤芯的情况)。如果设备收到平台接收此条指令成功之后，就不再发送此条指令。如果设备收到平台接收此条指令失败信息，会再推送一次。如果设备连续发送三次此条指令平台都接收失败，则不再发送此条指令。

|  |  |  |
| --- | --- | --- |
| 服务器返回 | {  "flag": 242,  "version": 1,  "cmd": 35,  "gprs": 1,  "addr": "192.168.0.1",  "barcodeid": "12345678"，  "len": 1,  "d1":成功/失败标志  } | d1:  0x00—失败  0xff—成功 |

1. 数据指令统一保存在 mongodb 日志表中， 根据设备id查询mongodb 中设备与整机码，物流码绑定的表中是否有数据，有数据认为产测完成了，没有数据认为是需要产测的设备

产测设备需要保存信息到device\_qc\_post\_log\_mongodb中

|  |  |  |
| --- | --- | --- |
| 字段 | 类型 | 说明 |
| data | Map | 数据 |
| deviceId | String | 设备Id |
| version | String | 协议版本 |
| len | int | 数据长度 |
| cmdText | String | 指令的说明 |
| createTime | Date | 创建时间 |
| flag | int | 起始标志 |
| json | String | 指令原文 |
| gprs | String | gprs状态信息。CMD=25的时候为gprs信号强度 |
| cmd | int | 命令字 |
| id | String | id |
| addr | String | 默认为经纬度，CMD=28时候，为IMEI号 |
| barcodeId | String | 配件码 |
| createUser | String | 创建人 |
| updateTime | Date | 修改时间 |
| updateUser | String | 修改人 |

非产测设备上报信息保存到 device\_post\_log\_mongodb表中

|  |  |  |
| --- | --- | --- |
| 字段 | 类型 | 说明 |
| data | Map | 数据 |
| deviceId | String | 设备Id |
| version | String | 协议版本 |
| len | int | 数据长度 |
| cmdText | String | 指令的说明 |
| createTime | Date | 创建时间 |
| flag | int | 起始标志 |
| json | String | 指令原文 |
| gprs | String | gprs状态信息。CMD=25的时候为gprs信号强度 |
| cmd | int | 命令字 |
| id | String | id |
| addr | String | 默认为经纬度，CMD=28时候，为IMEI号 |
| barcodeId | String | 配件码 |
| createUser | String | 创建人 |
| updateTime | Date | 修改时间 |
| updateUser | String | 修改人 |

其中 cmd=21 的数据如果指令显示若存在故障，需要保存数据到device\_fault\_log\_mongodb

|  |  |  |
| --- | --- | --- |
| 字段 | 类型 | 说明 |
| data | map | 数据 |
| deviceId | String | 设备Id |
| version | String | 协议版本 |
| len | int | 数据长度 |
| cmdText | String | 指令的说明 |
| createTime | Date | 创建时间 |
| flag | int | 起始标志 |
| json | String | 指令原文 |
| gprs | String | gprs状态信息。CMD=25的时候为gprs信号强度 |
| cmd | int | 命令字 |
| id | String | id |
| addr | String | 默认为经纬度，CMD=28时候，为IMEI号 |
| barcodeId | String | 配件码 |
| createUser | String | 创建人 |
| updateTime | Date | 修改时间 |
| updateUser | String | 修改人 |

对于产测设备服务器下发的指令数据 需要保存到 device\_qc\_control\_log\_mongodb表

|  |  |  |
| --- | --- | --- |
| 字段 | 类型 | 说明 |
| data | List | 数据 |
| deviceId | String | 设备Id |
| version | String | 协议版本 |
| len | int | 数据长度 |
| cmdText | String | 指令的说明 |
| createTime | Date | 创建时间 |
| flag | int | 起始标志 |
| json | String | 指令原文 |
| gprs | String | gprs状态信息。CMD=25的时候为gprs信号强度 |
| cmd | int | 命令字 |
| id | String | id |
| addr | String | 默认为经纬度，CMD=28时候，为IMEI号 |
| barcodeId | String | 配件码 |
| createUser | String | 创建人 |
| updateTime | Date | 修改时间 |
| updateUser | String | 修改人 |

非产测设备服务器下发的指令数据 需要保存到 device\_control\_log\_mongodb表

|  |  |  |
| --- | --- | --- |
| 字段 | 类型 | 说明 |
| data | List | 数据 |
| deviceId | String | 设备Id |
| version | String | 协议版本 |
| len | int | 数据长度 |
| cmdText | String | 指令的说明 |
| createTime | Date | 创建时间 |
| flag | int | 起始标志 |
| json | String | 指令原文 |
| gprs | String | gprs状态信息。CMD=25的时候为gprs信号强度 |
| cmd | int | 命令字 |
| id | String | id |
| addr | String | 默认为经纬度，CMD=28时候，为IMEI号 |
| barcodeId | String | 配件码 |
| createUser | String | 创建人 |
| updateTime | Date | 修改时间 |
| updateUser | String | 修改人 |

其中 cmd=22 设备定时主动上报的数据需要保存到mysql 中 t\_device\_status表中，记录当前设备的一些核心指标数据;

1. **设备调用平台接口**
2. 设备获取连接字符串

描述：设备配网成功后，需要从管理平台获取到连接IOT的一些信息，app根据

接口地址：/device/getConnectionStr 类型：get

入参：

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 类型 | 说明 | 是否必传 |
| deviceId | String | 设备Id | Y |

出参：

|  |  |  |
| --- | --- | --- |
| 参数 | 类型 | 说明 |
| code | String | 0-成功 1-失败 |
| msg | String | 请求结果 |
| data | Map |  |
|  |  |  |

Map:

|  |  |  |
| --- | --- | --- |
| deviceId | String | 设备ID |
| iotHost | String | IOT连接地址 |
| sharedAccessKey | String | productKey&deviceName |

{

“code”:”0”,

“msg”:”请求成功”,

“data”:{

“deviceId”:”00001”,

“iotHost”:”7299229929”,

“sharedAccessKey”:”demo@0001”,

}

}

1. 查询设备上下线情况

描述：用于APP连接IOT平台成功后检查设备是否已经上线,根据设备id查询 设备表 t\_device\_info 检查该设备的is\_online字段

请求地址：/device/isOnline /{deviceId} 类型: get

入参：

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 类型 | 说明 | 是否必传 |
| deviceId | String | 设备Id | Y |

出参：

|  |  |  |
| --- | --- | --- |
| 参数 | 类型 | 说明 |
| code | String | 0-成功 1-失败 |
| Msg | String | 请求结果 |
| data | Object | 结果为DeviceEntity |

DeviceEntity:

|  |  |  |
| --- | --- | --- |
| deviceId | String | 设备ID |
| status | String | 100-上线 101-下线 |
| id | String | Mongdb主键 |
| ip | String | 设备IP |
| cleanSession | String | 是否清除会话 |
| orderTime | Long | 指令时间戳 |

{

“code”:”0”,

“msg”:”请求成功”,

“data”:{

“clientID”:”00001”,

“iotHost”:”7299229929”,

“status”:”100”,

“ip”:”192.168.6.10”,

“id”:”0001”,

“cleanSession”:false,

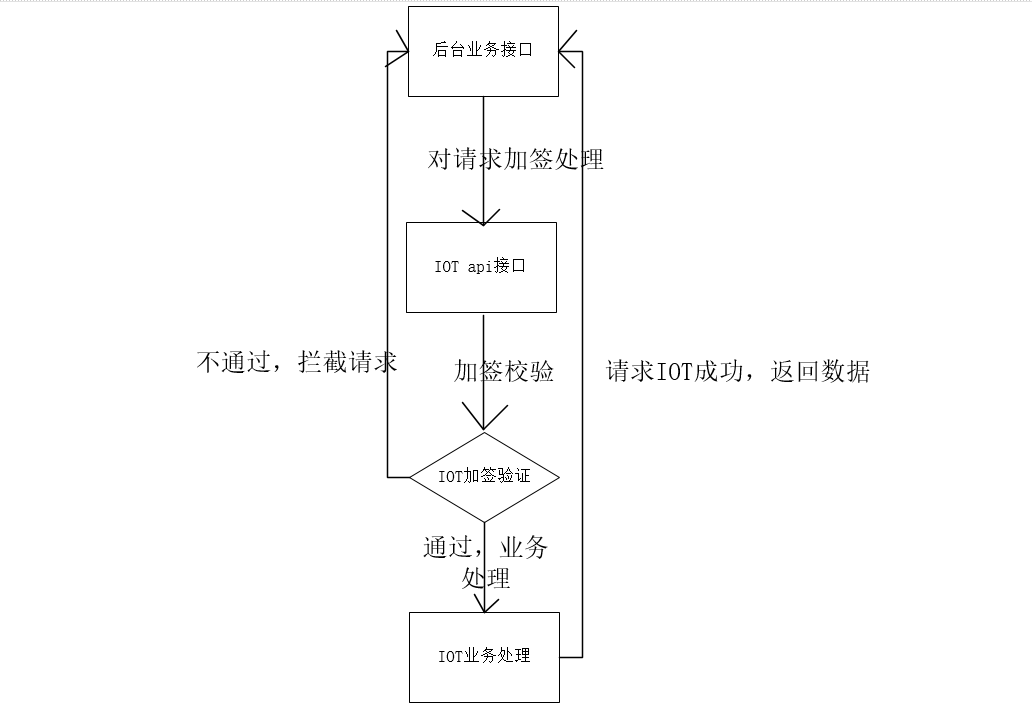
“orderTime”:1020202039

}

}

1. **设备平台请求IOT接口**

流程图



公共入参

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名称 | 类型 | 是否必须 | 描述 |
| accessKeyId | String | Y | Access key id |
| signatureMethod | String | Y | 签名方法 |
| signatureNonce | String | Y | 请求随机数 |
| signatureVersion | String | Y | 签名版本 |
| timestamp | String | Y | 请求时间戳 |
| version | String | Y | Api版本 |
| signature | String | Y | 请求签名 |

公共出参

|  |  |  |  |
| --- | --- | --- | --- |
| requestId | String | Y | Trace id，每次请求新生成 |
| code | String | Y | 请求结果状态 |
| message | String | Y | 请求结果国际化的消息 |
| resultCode | String | Y | 请求结果状态码 A09000-Z99999 |

描述: IotHubService类中定义了模拟HTTP请求的方法，具体业务方法同Iot API实现交互，app操控设备，服务器下发指令，创建产品等一些都需要通过这个类调用

公共入参

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名称 | 类型 | 是否必须 | 描述 |
| accessKeyId | String | Y | Access key id |
| signatureMethod | String | Y | 签名方法 |
| signatureNonce | String | Y | 请求随机数 |
| signatureVersion | String | Y | 签名版本 |
| timestamp | String | Y | 请求时间戳 |
| version | String | Y | Api版本 |
| signature | String | Y | 请求签名 |

1. 服务下发指令接口：

1) 发送普通消息 方法: publish()

请求地址:https://api.yun.pingan.com/v1?Action=Publish&<公共参数>

入参：

|  |  |  |  |
| --- | --- | --- | --- |
| productKey | String | Y | 产品key |
| deviceName | String | Y | 设备名 |
| topicName | String | Y | 消息主题 |
| content | String | Y | 消息内容 |
| gos | String | Y | 发送频次 0-最多送达一次 1-至少送达一次 |

出参：

|  |  |  |  |
| --- | --- | --- | --- |
| code | String | Y | 0-成功 1-失败 |
| msg | String | Y | 请求成功 |
| data | Object |  |  |
| status | String |  | Success-成功 fail-失败 |

2) 发送广播消息 方法 ：publishBroadcast()

请求地址:https://api.yun.pingan.com/v1?Action=PublishBroadcast&<公共参数>

入参：

|  |  |  |  |
| --- | --- | --- | --- |
| productKey | String | Y | 产品key |
| content | String | Y | 消息内容 |
| gos | String | Y | 发送频次 0-最多送达一次 1-至少送达一次 |

出参：

|  |  |  |  |
| --- | --- | --- | --- |
| code | String | Y | 0-成功 1-失败 |
| msg | String | Y | 请求成功 |
| data | Object |  |  |
| status | String |  | Success-成功 fail-失败 |

3) 创建产品 方法 createProduct（）

请求地址:https://api.yun.pingan.com/v1?Action=CreateProduct&<公共参数>

入参：

|  |  |  |  |
| --- | --- | --- | --- |
| productName | String | Y | 产品名称 |
| productDesc | String | N | 产品描述 |
| authId | String | N | 1. 证书 2.密钥   默认1 |
| region | String | N | 区域id |

出参：

|  |  |  |  |
| --- | --- | --- | --- |
| code | String | Y | 0-成功 1-失败 |
| msg | String | Y | 请求成功 |
| data | Object |  |  |
| id | String |  | 产品id |

4）查询产品详情 方法：queryProduct()

请求地址:https://api.yun.pingan.com/v1?Action=QueryProduct&<公共参数>

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 入参 |  |  |  |  |
| 参数名 | 类型 | 必传 | 描述 | 备注 |
| id | String | Y | 产品id |  |
|  |  |  |  |  |
| 出参 |  |  |  |  |
| 参数名 | 类型 | 必传 | 描述 | 备注 |
| code | String | Y | 状态 | 0-成功 1-失败 |
| msg | String | Y | 请求描述 |  |
| data | Object | Y |  |  |
| id | String |  | 产品id |  |
| productName | String |  | 产品名称 |  |
| productKey | String |  | 产品key |  |
| productSecret | String |  | 产品secret |  |
| region | String |  | 区域id |  |
| authId | String |  | 认证方式id |  |
| authName | String |  | 认证方式名称 |  |
| productDesc | String |  | 产品描述 |  |
| productStatus | String |  | 状态 |  |
| createDate | String |  | 创建时间 |  |
| updateDate | String |  | 修改时间 |  |
| userId | String |  | 用户id |  |
| productVersion | int |  | 版本 |  |
| nodeType | int |  | 节点类型 | 设备网关 |
| dataType | int |  | 数据类型 | json |
| productType | int |  | 产品类型 |  |
| deviceCount | int |  | 产品下设备数量 |  |
| accountName | String |  | 账户名 |  |
| dynRegistration | int |  | 是否启用动态注册 |  |

5）动态注册设备 方法：registDevice()

请求地址:https://api.yun.pingan.com/v1?Action=RegisterDevice&<公共参数>

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 入参 |  |  |  |  |
| 参数名 | 类型 | 必传 | 描述 | 备注 |
| productKey | String | Y | 产品ke'y |  |
| productSecret | String | Y | 产品secret |  |
| deviceName | String | Y | 设备名称 |  |
|  |  |  |  |  |
| 出参 |  |  |  |  |
| 参数名 | 类型 | 必传 | 描述 | 备注 |
| code | String | Y | 状态 | 0-成功 1-失败 |
| msg | String | Y | 请求描述 |  |
| data | Object | Y | 返回具体数据 | rule对象 |
| id | String |  | 设备唯一id |  |
| productKey | String |  | 产品key |  |
| deviceName | String |  | 设备名称 |  |
| deviceSecret | String |  | 设备secret |  |
| createDate | String |  | 创建时间 |  |
| updateDate | Boolean |  | 修改时间 |  |

6）批量创建设备 方法: batchCreateDeviceList

请求地址:https://api.yun.pingan.com/v1?Action=BatchCreateDeviceWithNames&<公共参数>

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 入参 |  |  |  |  |
| 参数名 | 类型 | 必传 | 描述 | 备注 |
| productKey | String | Y | 产品key |  |
| deviceName | String | Y | 设备名称 |  |
|  |  |  |  |  |
| 出参 |  |  |  |  |
| 参数名 | 类型 | 必传 | 描述 | 备注 |
| code | String | Y | 状态 | 0-成功 1-失败 |
| msg | String | Y | 请求描述 |  |
| data | Object | Y |  | 空 |
| applyId | String | Y | 申请批次号 |  |

7）批量查询设备 方法：queryDeviceListByApplyId

请求地址:https://api.yun.pingan.com/v1?Action=QueryDeviceListWithApplyId&<公共参数>

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 入参 |  |  |  |  |
| 参数名 | 类型 | 必传 | 描述 | 备注 |
| productKey | String | Y | 产品key |  |
| applyId | String | Y | 申请批次号 |  |
| pageIndex | int | N | 页码 |  |
| pageSize | int | N | 页数 |  |
|  |  |  |  |  |
|  |  |  |  |  |
| 出参 |  |  |  |  |
| 参数名 | 类型 | 必传 | 描述 | 备注 |
| code | String | Y | 状态 | 0-成功 1-失败 |
| msg | String | Y | 请求描述 |  |
| data | List | Y |  | 空 |
| id | String |  | 设备id |  |
| productKey | String |  | 产品key |  |
| deviceName | String |  | 设备名称 |  |
| deviceSecret | String |  | 设备secret |  |

8) 查询设备详情 方法: queryDevice()

请求地址:https://api.yun.pingan.com/v1?Action=QueryDevice&<公共参数>

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 入参 |  |  |  |  |
| 参数名 | 类型 | 必传 | 描述 | 备注 |
| id | String | Y | 设备id |  |
|  |  |  |  |  |
|  |  |  |  |  |
| 出参 |  |  |  |  |
| 参数名 | 类型 | 必传 | 描述 | 备注 |
| code | String | Y | 状态 | 0-成功 1-失败 |
| msg | String | Y | 请求描述 |  |
| data | Object | Y |  | 设备实体 |
| id | String |  | 主键id |  |
| productKey | String |  | 产品key |  |
| deviceName | String |  | 设备名称 |  |
| createDate | Date |  | 创建时间 |  |
| updateDate | Date |  | 修改时间 |  |
| deviceStatus | int |  | 设备状态 | 0-禁用 1-启用 |
| userId | String |  | 用户id |  |
| accessStatus | int |  | 接入状态 | 默认 1-未激活 2离线 3-在线 |
| lastAccessDate | Date |  | 最后接入时间 |  |
| deviceSecret | String |  | 设备secret |  |
| deviceIp | String |  | 最后上线ip |  |
| productName | String |  | 产品名称 |  |
| nodeType | String |  | 节点类型 |  |
| activateDate | Date |  | 激活时间 |  |
| region | String |  | 区域id |  |
| productId | String |  | 产品id |  |