DTU通信数据协议

# 概述

查询DTU，复位DTU，设置参数，数据透传等指令

# 帧格式

|  |  |  |
| --- | --- | --- |
| 字段名称 | 长度 | 说明 |
| 帧头 | 3B | 固定为0x6A 0x6A 0x6A |
| 功能码 | 2B | 标识帧的具体功能 |
| 帧长度 | 2B | 整个帧的长度，包括帧头帧尾，高字节在前(网络字节序) |
| 数据内容 | 0~1056B | 数据内容部分, 1024（业务数据）+32（为电话号码预留） |
| 校验和 | 1B | 除本字节外其他所有字节之和 |
| 帧尾 | 3B | 固定为0x6F 0X6F 0x6F |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 帧头 | | | 功能码 | 帧长度 | 数据内容 | 校验和 | 帧尾 | | |
| 0x6a | 0x6a | 0x6a | 0x16 | 2B | 2B | 1B | 0x6f | 0x6f | 0x6f |

# 指令说明

**协议功能码说明：**

**1．查询DTU，复位DTU，设置参数，数据透传的功能码和返回数据的功能码一致**

**2. 返回数据的状态码可查询对应的状态码表**

## 功能码表

|  |  |
| --- | --- |
| 功能码 | 功能 |
| 0x00-0x3f | 查询指令 |
| 0x00 | 查询IMEI |
| 0x01 | 查询本机号码 |
| 0x02 | 查询固件版本号 |
| 0x03 | 查询信号强度 |
| 0x04 | 查询当前配置参数 |
| 0x05 | 诊断查询 |
| 0x41~0x4f | 通道透传指令 |
| 0x41 | 通道1 |
| 0x42 | 通道2 |
| 0x43 | 通道3 |
| 0x44 | 通道4 |
| 0x45 | 通道5 |
| 0x46 | 通道6 |
| 0x47 | 通道7 |
| 0x50~0x8f | 设置指令 |
| 0x50 | 协议短信透传 |
| 0x51 | 配置密码 |
| 0x52 | 添加设备识别码IMEI |
| 0x53 | 登录服务器发送注册信息 |
| 0x54 | 固件版本号 |
| 0x55 | 是否启用自动更新 |
| 0x56 | 日志输出 |
| 0x57 | 服务器获取配置参数 |
| 0x58 | 串口参数 |
| 0x59 | 通道配置参数 |
| 0x60 | Apn设置 |
| 0x61 | GPIO设置 |
| 0x62 | GPS |
| 0x63 | 数据流 |
| 0x64 | 预警 |
| 0x65 | 任务 |
| 0xfd | 协议终止指令 |
| 0xfe | DTU启动中，无法接收指令 |
| 0xff | 复位指令 |

# 查询指令

### 查询IMEI

**说明：**

DTU的IMEI号

功能码: 0x00

返回的数据内容：

**{“code”: ”20000” , ”data”: ” 123456789012345 ” , ”success”:1}**

字段说明:

| **字段** | **类型** | **含义** |
| --- | --- | --- |
| code | str | 状态码（如果查询IMEI失败，可查询状态码表来定位具体错误信息） |
| data | str | 返回IMEI |
| success | int | 0 失败 1成功 |

### 查询本机号码

**说明：**

查询SIM卡的号码

功能码: 0x01

返回的数据内容：

**{“code”: ”20000” , ”data”: ” 17201593988 ” , ”success”:1}**

| **字段** | **类型** | **含义** |
| --- | --- | --- |
| code | Str | 状态码 |
| data | str | SIM卡的手机号码 |
| success | int | 0 失败 1成功 |

### 查询固件版本号

**说明：**

查询当前的固件版本号（当开启fota升级，版本号小于服务器端的固件版本号会进行fota升级）

固件版本号格式为: v 1

功能码: 0x02

返回的数据内容：

**{“code”: ”20000” , ”data”: ” v 1 ” , ”success”:1}**

| **字段** | **类型** | **含义** |
| --- | --- | --- |
| code | str | 状态码 |
| data | str | 固件版本号 |
| success | int | 0 失败 1成功 |

### 查询信号强度

**说明：**

网络信号强度值范围0~31，值越大表示信号强度越好。

功能码: 0x03

返回的数据内容：

**{“code”: ”20000” , ”data”: ” CSQ17 ” , ”success”:1}**

| **字段** | **字符串** | **含义** |
| --- | --- | --- |
| code | str | 状态码 |
| data | str | CSQ1~CSQ31 |
| success | int | 0 失败 1成功 |

### 查询当前配置参数

功能码: 0x04

数据内容：

{ “password”: ” **012345**”,

“data”:{}

}

返回的数据内容：

**{“code”: ”20000” , ”data”: ” req config ” , ”success”:1}**

| **字段** | **字符串** | **含义** |
| --- | --- | --- |
| code | Str | 状态码 |
| data | str | req config |
| success | int | 0 失败 1成功 |

### 诊断查询

说明: 查询当前DTU运行的错误上报信息

功能码: 0x05

返回的数据内容：

**{“code”:”20000”,**

**”data”:[ {“func\_code”: ”0x01” , ”error\_code”: ” 6001”} ],**

**”success”:1}**

| **字段** | **类型** | **含义** |
| --- | --- | --- |
| code | str | 状态码 |
| func\_code | str | 功能码 |
| error\_code | str | 错误码 |
| success | int | 0 失败 1成功 |

# 复位指令

功能码: 0xff

数据内容：

{ Password: ” **012345**”,

“data”:{}

}

返回的数据内容：

**{“code”: ”20000” , ”data”: ” reset dtu ” , ”success”:1}**

| **字段** | **类型** | **含义** |
| --- | --- | --- |
| code | Str | 状态码 |
| data | str | reset dtu |
| success | int | 0 失败 1成功 |

# 设置指令

## 基础设置

### 协议短信(SMS)透传 message

功能码: 0x50

数据内容：

* 号码：32个字节，标识目的手机号码,如果没有32字节，必须补0x00
* 内容：最大1024字节，为短信数据内容

{

“password”:"",

“data”:{

message: {"number":"12123123", -- 目标号码

"data:" " -- 发送短信

}

}

}

返回的数据内容：

**{“code”: ”20000” , ”data”: ” ” , ”success”:1}**

| **字段** | **类型** | **含义** |
| --- | --- | --- |
| code | str | 状态码 |
| data | str | 接收的短信 |
| success | int | 0 失败 1成功 |

### 配置密码 password

**说明：**

查询IMEI,查询本机号码，查询固件版本号，查询信号强度不需要密码

查询当前配置参数和修改透传绑定的通道需要密码

是否开启自动更新需要密码

功能码: 0x51

数据内容：

{

“password”:"",

“data”:{ “password”: ” **012345**”}

}

说明：初始密码为固件IMEI的后六位

**如IMEI : 123456789012345 初始密码为012345**

| **字段** | **含义** |
| --- | --- |
| password | 当前密码 |
| data | password : 修改后的密码 |

返回的数据内容：

**{“code”: ”20000” , ”data”: ” ” , ”success”:1}**

| **字段** | **含义** |
| --- | --- |
| code | 状态码 |
| data |  |
| success | 0 失败 1成功 |

### 添加设备识别码IMEI

功能码: 0x52

数据内容：

{

**“password”:"",**

**“data”:{"plate": 1}**

}

| **数据** | **类型** | **含义** |
| --- | --- | --- |
| plate | int | 0关闭/1打开 |

**说明：**

设备识别码IMEI,添加设备识别码成功之后，串口接收的数据前面会加IMEI

如果plate置为1则返回的数据如下所示

866327040830317**{“code”: ”20000” , ”data”: ” set password ” , ”success”:1}**

如果plate置为0则返回的数据如下所示

**{“code”: ”20000” , ”data”: ” set password ” , ”success”:1}**

返回的数据内容：

**{“code”: ”20000” , ”data”: ” ” , ”success”:1}**

| **字段** | **类型** | **含义** |
| --- | --- | --- |
| code | Str | 状态码 |
| data | str | add IMEI |
| success | int | 0 失败 1成功 |

### 登录服务器发送注册信息 reg

**说明：**

首次登陆服务器发送注册信息

功能码: 0x53

数据内容：

{

**“password”**:"",

**“data”**:{"reg": 1}

}

| **Reg** | **值** |
| --- | --- |
| 0 | 不发送  { "reg": 0} |
| 1 | { "reg": 1}  则首次登录服务器发送下面的json数据：  {"csq":rssi,"imei":imei,"iccid":iccid,"ver":version}  csq 信号强度  imei 固件的imei  iccid SIM卡的iccid  ver 固件的版本号 |
| 自定义 | { "reg": "自定义的注册信息"} |

返回的数据内容：

**{“code”: ”20000” , ”data”: ” ” , ”success”:1}**

| **字段** | **类型** | **含义** |
| --- | --- | --- |
| code | str | 状态码 |
| data | str | send reg |
| success | int | 0 失败 1成功 |

### 固件版本号 version

**说明：**

修改固件版本号，用于fota升级（当开启fota升级，版本号小于服务器端的固件版本号就会进行fota升级）

功能码: 0x54

数据内容：

{

"password":"",

"data":{

" version ": "100" --- 版本号（使用数字字符串）

}

}

返回的数据内容：

**{“code”: ”20000” , ”data”: ” ” , ”success”:1}**

| **字段** | **类型** | **含义** |
| --- | --- | --- |
| code | Str | 状态码 |
| data | str | 固件版本号 |
| success | int | 0 失败 1成功 |

### 是否启用自动更新 fota

**说明：**

Fota升级开关

功能码: 0x55

数据内容：

{

"password":"",

"data":{

"fota": 1 -- 0关闭/ 1 开启（int类型）

}

}

返回的数据内容：

**{“code”: ”20000” , ”data”: ” fota” , ”success”:1}**

| **字段** | **类型** | **含义** |
| --- | --- | --- |
| code | str | 状态码 |
| data | str | fota |
| success | int | 0 失败 1成功 |

### 日志输出 nolog

**说明：**

串口打印日志记录,目前不支持。日志输出连接Debug口

功能码: 0x56

数据内容：

{

"password":"",

"data":{

{"nolog": 1} 是否开启日志 0关闭/1打开（int类型）

}

}

返回的数据内容：

**{“code”: ”20000” , ”data”: ”log ” , ”success”:1}**

| **字段** | **类型** | **含义** |
| --- | --- | --- |
| code | Str | 状态码 |
| data | str | log |
| success | int | 0 失败 1成功 |

### 服务器获取配置参数

功能码: 0x57

数据内容：

{

"password":"",

"data":{

{" service\_acquire"：0}

}

}

注:本地配置默认开启向服务器获取参数配置

| **字段** | **类型** | **含义** |
| --- | --- | --- |
| service\_acquire | Int | 是否开启服务器获取参数0关闭/1打开 |

返回的数据内容：

**{“code”: ”20000” , ”data”: ”service acquire ” , ”success”:1}**

| **字段** | **类型** | **含义** |
| --- | --- | --- |
| code | Str | 状态码 |
| data | str | service acquire |
| success | int | 0 失败 1成功 |

### 串口参数 uconf

功能码: 0x58

数据内容：

{"password": ""

"data":{

"uconf": [

[1, -- 串口ID号

"115200", -- 串口波特率

8, -- 串口数据位

2, -- 串口parity

0], -- 串口stopbits

[],

[],

[]

]}

}

返回的数据内容：

**{“code”: ”20000” , ”data”: ” ” , ”success”:1}**

| **字段** | **类型** | **含义** |
| --- | --- | --- |
| code | Str | 状态码 |
| data | str | uconf |
| success | int | 0 失败 1成功 |

### 通道配置参数 conf

功能码: 0x59

数据内容：

**{“password”:””,**

**“data”:{**

**“conf”:[**

[**[http,method,url,timeout,serialD]**](#_通道1_HTTP参数)**, -- 通道1**

[**[tcp,ping,time,url,port,keepAlive, serialD]**](#_通道2_SOCKET_tcp参数)**, -- 通道2**

[**[udp,ping,time,url,port,keepAlive,seriaID],**](#_通道3_SOCKET_udp) **-- 通道3**

[**[mqtt, clentID ,keepAlive, address, port, cleanSession,sub,pub, qos, retain,seriaID],**](#_通道4_MQTT参数) **-- 通道4**

[**[aliyun,type,keepAlive,clientID,ProductKey,DeviceSecret,ProductSecret,Devicename,cleanSession,QOS, subTopic ,pubTopic,seriaID]**](#_通道5_阿里云参数)**, -- 通道5**

[**[tengxun,type,keepAlive,clientID,ProductKey,DeviceSecret,ProductSecret,Devicename,cleanSession,QOS, subTopic ,pubTopic,seriaID],**](#_通道6腾讯云通道配置) **-- 通道6**

**[ ] ] -- 通道7**

**}}**

**对应通道的配置参数详见6.1.10.1的通道配置详解 ：**

返回的数据内容：

**{“code”: ”20000” , ”data”: ” ” , ”success”:1}**

| **字段** | **类型** | **含义** |
| --- | --- | --- |
| code | str | 状态码 |
| data | str | conf |
| success | int | 0 失败 1成功 |

#### 通道配置详解

##### HTTP参数

**[http,method,url,timeout,serialD]**

| **字段** | **类型** | **含义** |
| --- | --- | --- |
| http | str | 通信方式http标识 |
| method | str | 提交请求的方法 |
| url | str | HTTP请求的地址和参数 |
| timeout | int | HTTP请求最长等待时间 |
| serialD | int | HTTP绑定的串口号（1~2） |

##### SOCKET tcp参数

**[tcp,ping,time,url,port,keepAlive, serialD]**

| **字段** | **类型** | **含义** |
| --- | --- | --- |
| tcp | str | Socket的tcp协议标识 |
| ping | str | 用户自定义的心跳包,只支持数字和字母,建议2-4个字节 |
| time | int | 0为关闭心跳包，建议60s-300s |
| url | str | socket的地址或域名 |
| port | int | socket服务器的端口号 |
| KeepAlive | int | 链接超时最大时间单位秒,默认300秒 |
| serialD | int | tcp/udp绑定的串口号(1~2) |

##### SOCKET udp 参数

**[udp,ping,time,url,port,keepAlive,seriaID]**

| **字段** | **类型** | **含义** |
| --- | --- | --- |
| udp | str | Socket的udp协议标识 |
| ping | str | 用户自定义的心跳包,只支持数字和字母,建议2-4个字节 |
| time | int | 0为关闭心跳包，建议60s-300s |
| url | str | socket的地址或域名 |
| port | int | socket服务器的端口号 |
| KeepAlive | int | 链接超时最大时间单位秒,默认300秒 (60~600) |
| serialD | int | tcp/udp绑定的串口号(1~2) |

##### MQTT参数

**[mqtt, clentID ,keepAlive, address, port, cleanSession,sub,pub, qos, retain,seriaID]**

| **字段** | **类型** | **含义** |
| --- | --- | --- |
| mqtt | str | 表示MQTT协议 |
| clentID | str | 自定义客户端ID，使用IMEI做客户端ID此处留空 |
| keepAlive | int | 客户端的keepalive超时值。 默认为60秒 |
| address | str | MQTT的地址或域名 |
| port | int | socket服务器的端口号 |
| cleanSession | int | MQTT是否保存会话标志位,0持久会话,1离线自动销毁 |
| Sub | str | 订阅主题 |
| pub | str | 发布主题 |
| qos | int | MQTT的QOS级别,默认0 |
| retain | int | MQTT的publish参数retain，默认0 |
| serialD | int | MQTT通道捆绑的串口ID (1~3) |

##### 阿里云参数

**[aliyun,type,keepAlive,clientID,ProductKey,DeviceSecret,ProductSecret,Devicename,cleanSession,QOS, subTopic ,pubTopic,seriaID]**

| **字段** | **类型** | **含义** |
| --- | --- | --- |
| aliyun | Str | 阿里云IOT的标识 |
| type | str | 一型一密tas/一机一密mos |
| keepAlive | int | 通信之间允许的最长时间段（以秒为单位）,默认为300，范围（60-1200）使用默认值就填””或者” ” |
| clientID | str | clientID ,自定义字符（不超过64） |
| Devicename | str | 设备名称 |
| ProductKey | str | 产品密钥 |
| DeviceSecret | str | 设备密钥（使用一型一密认证此参数传入””) |
| ProductSecret | str | 产品密钥（使用一机一密认证时此参数传入””) |
| cleanSession | int | MQTT 保存会话标志位( 0则客户端是持久客户端，当客户端断开连接时，订阅信息和排队消息将被保留, 1代理将在其断开连接时删除有关此客户端的所有信息 ) |
| QOS | int | MQTT消息服务质量（默认0，可选择0或1）0：发送者只发送一次消息，不进行重试 1：发送者最少发送一次消息，确保消息到达Broker |
| subTopic | str | 订阅主题 |
| pubTopic | str | 发布主题 |
| serialD | int | MQTT通道捆绑的串口ID (1~3) |

##### 腾讯云参数

**[txyun,type,keepAlive,clientID,ProductKey,DeviceSecret,ProductSecret,Devicename,cleanSession,QOS, subTopic ,pubTopic,seriaID]**

| **字段** | **类型** | **含义** |
| --- | --- | --- |
| txyun | str | 腾讯云IOT的标识 |
| type | str | 一型一密tas/一机一密mos |
| keepAlive | int | 通信之间允许的最长时间段（以秒为单位）,默认为300，范围（60-1200）使用默认值就填””或者” ”。 |
| clientID | str | clientID ,自定义字符（不超过64） |
| Devicename | str | 设备名称 |
| ProductKey | str |  |
| DeviceSecret | str | 设备密钥（使用一型一密认证此参数传入””) |
| ProductSecret | str | 产品密钥（使用一机一密认证时此参数传入””) |
| cleanSession | int | MQTT 保存会话标志位( 0则客户端是持久客户端，当客户端断开连接时，订阅信息和排队消息将被保留, 1代理将在其断开连接时删除有关此客户端的所有信息 ) |
| QOS | int | MQTT消息服务质量（默认0，可选择0或1）0：发送者只发送一次消息，不进行重试 1：发送者最少发送一次消息，确保消息到达Broker |
| subTopic | str | 订阅主题 |
| pubTopic | str | 发布主题 |
| serialD | int | MQTT通道捆绑的串口ID (1~3) |

# 通道数据透传指令

**说明：**

通道数据透传协议：

0x41 代表通道1

0x42 代表通道2

0x43 代表通道3

0x44 代表通道4

0x45 代表通道5

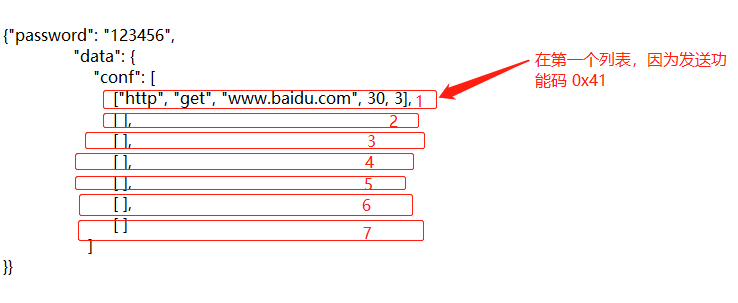
0x46 代表通道6

0x47 代表通道7

**功能码:**

0x4 + (列表下标+1)

例如：



数据内容格式：

**{**

**“password”:"",**

**“data”:**

**{**

**“method”:””**

**“send\_data”:””**

**}**

| **字段** | **类型** | **含义** |
| --- | --- | --- |
| method | str | **仅HTTP需要加 method字段，其他透传不需要加** |
| send\_data | json | 需要透传的数据 |

返回的数据内容：

**{“code”: ”20000” , ”data”: ” ” , ”success”:1}**

字段说明:

| **字段** | **类型** | **含义** |
| --- | --- | --- |
| code | str | 状态码 |
| data | str | 服务器返回的数据 |
| success | int | 0 失败 1成功 |

# APN设置 apn

说明：这个指令只适合配置和使用不是同一张卡的场景

功能码: 0x60

数据内容：

{

“password”:” ”,

“data”:{"apn": ["", "", ""]}

}

apn对应列表说明:

列表第一个参数: apn 的名称

列表第二个参数: apn 的用户名

列表第三个参数: apn 的密码

返回的数据内容：

**{“code”: ”20000” , ”data”: ”set apn” , ”success”:1}**

| **字段** | **类型** | **含义** |
| --- | --- | --- |
| code | Str | 状态码 |
| data | str | set apn |
| success | int | 0 失败 1成功 |

# GPIO pins

功能码: 0x61

数据内容：

{"password": " ",

“data”:{"pins":[

"pio2", -- 网路指示灯的GPIO (pio1~pio128)

"pio4", -- 与服务器连上后通知GPIO (pio1~pio128)

"pio4" -- 重置DTU参数的GPIO (pio1~pio128)

]}}

返回的数据内容：

**{“code”: ”20000” , ”data”: ”set gpio pins” , ”success”:1}**

| **字段** | **类型** | **含义** |
| --- | --- | --- |
| code | Str | 状态码 |
| data | str | set gpio pins |
| success | int | 0 失败 1成功 |

# GPS

功能码: 0x62

数据内容：

{"password": " ",

“data”:{"gps": {

"pio": ["pio3", "pio6", "pio7", "pio13", 0, "1"],  
 "fun": [1, "115200", 0, "10", 0, "json", "0", "|", "10", "1"]  
}}}

说明:字段pio、fun对应的列表里面的值的含义如下表所示：

| **字段** | **含义** |
| --- | --- |
| GPSLED | GPIO引脚（pio1~pio128） |
| 震动检测: | GPIO引脚（pio1~pio128） |
| 开锁检测 | GPIO引脚（pio1~pio128） |
| 充电状态检测 | GPIO引脚（pio1~pio128） |
| 电池电压检测 | 0/1 ( 0 ADC0 /1 ADC 1 ) |
| 分压比（VCC/1.8V+1） | 1~50 |
| 串口ID | 1~3 |
| 波特率 | 115200 |
| 工作模式 | 0正常/1低功耗/2自动跟踪 |
| 采集间隔 | 自动任务采集时间（单位s） |
| 采集方式: | 0触发采集/1连续采集 |
| 报文格式 | json/hex |
| 报文缓存 | 1~100s 缓存后一次发送 |
| 分隔标记 | 不支持 逗号(,) |
| 状态更新 | 单位min GPS设备状态报文的上报间隔 |
| 上报通道 | 1~7 默认空 为自动捆绑GPS串口对应的网络通道 |

返回的数据内容：

**{“code”: ”20000” , ”data”: ”set gps” , ”success”:1}**

| **字段** | **类型** | **含义** |
| --- | --- | --- |
| code | Str | 状态码 |
| data | str | set gps |
| success | int | 0 失败 1成功 |

# 数据流

功能码: 0x63

数据内容：

{

"password": "830317",

“data:{”"upprot": ["", "", "", "", "", "", ""],  
"dwprot": ["", "", "", "", "", "", ""]}}

| **字段** | **类型** | **含义** |
| --- | --- | --- |
| upprot | json | 发送数据流模板 |
| dwprot | json | 接收数据流模板 |

返回的数据内容：

**{“code”: ”20000” , ”data”: ”set dataflow” , ”success”:1}**

| **字段** | **类型** | **含义** |
| --- | --- | --- |
| code | str | 状态码 |
| data | str | set dataflow |
| success | int | 0 失败 1成功 |

# 预警

功能码: 0x64

数据内容：

{

"password": "830317", -- 预警配置修改的密码

“data”:{"warn": {  
 "adc0": [   
 1, -- 启用欠压报警  
 "1000", -- 欠压下限 1000mV  
 1, -- 启用过压报警   
 "3000", -- 过压报警上限 3000mV   
 "300", -- 报警电压回差（防止临界值反复报警）  
 "ADC0 waring!", -- 报警内容  
 "1", -- 网路报警上报通道ID  
 "10", -- 检测频率  
 "100", -- 上报频率（防止报警没接触一直上报）  
 1, -- 启用互联网上报  
 1, -- 启用短信上报  
 0 -- 不启用电话通知  
 ],  
 "adc1": [],  
 "vbatt": [],  
 "gpio": [  
 [  
 "pio7", -- 按键GPIO  
 1, -- 启用按下报警（下降沿触发）  
 0, -- 不启用松开报警（上升沿触发）  
 "GPIO 7 is keyprees!", -- 报警内容  
 "1", -- 网路报警上报通道ID   
 1, -- 启用网路报警  
 0, -- 不启用短信报警   
 0 -- 不启用电话通知  
 ]  
 ]  
}

}}

返回的数据内容：

**{“code”: ”20000” , ”data”: ”set warn” , ”success”:1}**

| **字段** | **类型** | **含义** |
| --- | --- | --- |
| code | Str | 状态码 |
| data | str | set warn |
| success | int | 0 失败 1成功 |

# 任务

功能码: 0x65

数据内容：

{ "password": " ",

“data”:{"cmds": [  
 [ -- 串口1的自动采集任务  
 "100", -- 一次下发采集指令之间的间隔(主意和通道配置的采集间隔不是一个意思)  
 "11 22 33 44 55 66 77 88 ", -- 下发给串口1的采集指令（会自动转成数字bin）  
 "11 22 33 44 55 66 77 88 " -- 同上  
 ],  
 [] -- 串口2的自动采集任务关闭  
]}}

返回的数据内容：

**{“code”: ”20000” , ”data”: ”set cmds” , ”success”:1}**

| **字段** | **类型** | **含义** |
| --- | --- | --- |
| code | Str | 状态码 |
| data | str | set cmds |
| success | int | 0 失败 1成功 |