



# DTU4G 模块

(MT-D001-LADY)

使用手册

成都美特智控科技有限公司



## 目录

1. 概述 .....	1
1.1 本文概要 .....	1
1.2 产品参数 .....	1
2. 外观及接口 .....	2
2.1 产品外观 .....	2
2.2 接口说明 .....	2
2.3 指示灯说明 .....	2
3. 使用注意事项 .....	3
4. 功能介绍 .....	4
4.1 配置平台介绍 .....	4
4.2 配置参数说明 .....	5
4.2.1 基础配置 .....	5
4.2.2 网络通道配置 .....	6
5. 编码/解码函数 .....	12
5.1 编码函数 .....	12
5.2 解码函数 .....	13
5.3 内部可调用函数 .....	13
6. 使用示例 .....	15
6.1 设备连接使用前准备 .....	15
6.2 基础配置 .....	15
6.3 网络平台使用示例 .....	15
6.3.1 OneNet .....	15
6.3.2 阿里云 .....	25
6.3.3 腾讯云 .....	38
6.3.4 百度云 .....	49
6.3.5 MQTT 透传 .....	55
6.3.6 TCP 透传 .....	56
6.3.7 UDP 透传 .....	57
6.3.8 HTTP 透传 .....	58
7. 应用实例 .....	59
7.1 创建腾讯云产品（物模型） .....	60
7.2 创建阿里云产品（自定义透传） .....	61
7.3 重新编写编码与解码函数 .....	61
7.4 物联网管理平台配置设备参数 .....	62
7.5 数据上报 .....	64
7.6 数据下发 .....	65

## 修订记录

修订日期	修订记录	备注
2024/1/16	首次发布	~

# 1. 概述

## 1.1 本文概要

本文档介绍美特智控旗下 DTU4G 模块（CAT1 透传 TTL/232/48 物联网）产品使用方法。

## 1.2 产品参数

- 接口类型：通信接口支持 TTL、RS485、RS232 三种模式  
TTL 模式下支持波特率范围 2400~921600bps  
RS232 模式下支持波特率范围 2400~115200  
RS485 模式下支持波特率范围 2400~230400bps
- 供电范围：9V~36VDC
- 功耗：≤20mA
- 工作温度：-35℃ ~ +75℃
- 支持协议：TCP、UDP、HTTP、MQTT
- 支持物联网平台：Onenet、阿里云、腾讯云、百度云
- 频段：TDD-LTE B34/B38/B39/B40/B41，FDD-LTE B1/B3/B5/B8
- 网络制式：LTE Cat.1 全网通
- SIM 卡：4G 全网通
- 体积尺寸：80\*23\*16mm（不含端子），95.3\*23\*16mm（含端子）
- 安装方式：挂耳/导轨（导轨标准 C45 国际，35mm）

## 2. 外观及接口

### 2.1 产品外观



### 2.2 接口说明

设备下方有五个接线口根据分别为 VCC、GND、A/TX、B/RX 以及 GND，其中 VCC 接入电源正极（电压范围 9~36VDC）。A/TX 和 B/RX 为串口的发送和接收口，使用 RS484 功能时为 A 相、B 相。

### 2.3 指示灯说明

- PWR: 设备通电时 PWR 灯将会亮起。
- NET: 设备网络正常时 NET 灯常亮，正在搜索网络时 100 毫秒闪烁一次，网络未注册时 500 毫秒闪烁一次，网络注册失败时亮 100 毫秒熄灭 400 毫秒。
- STA: 设备运行正常时 STA 灯间隔 500 毫秒闪烁一次，设备配置参数存在错误导致设备异常时间间隔 100 毫秒闪烁一次。
- 485: 设备配置为 RS485 通信接口时 485 灯亮起。
- 232: 设备配置为 RS232 通信接口时 232 灯亮起
- TTL: 设备配置为 TTL 通信接口时 TTL 灯亮起

### 3. 使用注意事项

- 供电电压为 9V~36VDC，若超出该范围可能导致设备损坏。
- 本设备为 LTE Cat.1 产品，支持 4G 全网通手机卡、4G 物联网卡。
- 设备上电前需要先 SIM 卡正确插入。
- 设备上电后不要触摸电源接口，不要在上电期间拔出 SIM 卡。
- 本设备不支持私网 SIM 卡
- 使用时请注意天线安装是否正确，安装位置是否能够良好通信，若天线安装松动或安装位置不正确会导致设备信号差，从而影响网络通信。
- 使用前保证 SIM 卡能够正常使用，有流量，SIM 卡状态正常。
- 配置平台参数配置错误可能导致设备运行异常，当设备 STA 灯闪烁方式为 100 毫秒闪烁一次时请检测配置参数是否正确、所配置的网络通道是否可以正常连接。
- 通配符使用：在参数配置时，部分参数可使用通配符进行命名，通配符包括  $\${imei}$ 、 $\${productId}$ 、 $\${coreId}$  以及  $\${clientId}$ ，当参数中包含  $\${imei}$  时，通配符  $\${imei}$  将会被替换为 DTU 设备的 IMEI（每台 DTU 设备都有其唯一的 IMEI 码），假设 DTU 设备 IMEI 码为 845729846709134 客户端 ID 填写为  $MT\${imei}$ ，那么实际客户端 ID 将为 MT845729846709134。假设客户端 ID 为  $MT\${imei}$ ，发布主题为  $MT/\${clientId}/\${imei}$ ，则实际发布主题将为  $MT/MT845729846709134/845729846709134$ 。 $\${productId}$ 、 $\${coreId}$  和  $\${clientId}$  也是如此，但其替换的参数根据网络通道的选择而决定，具体替换参数在下列参数说明中查看。

## 4. 功能介绍

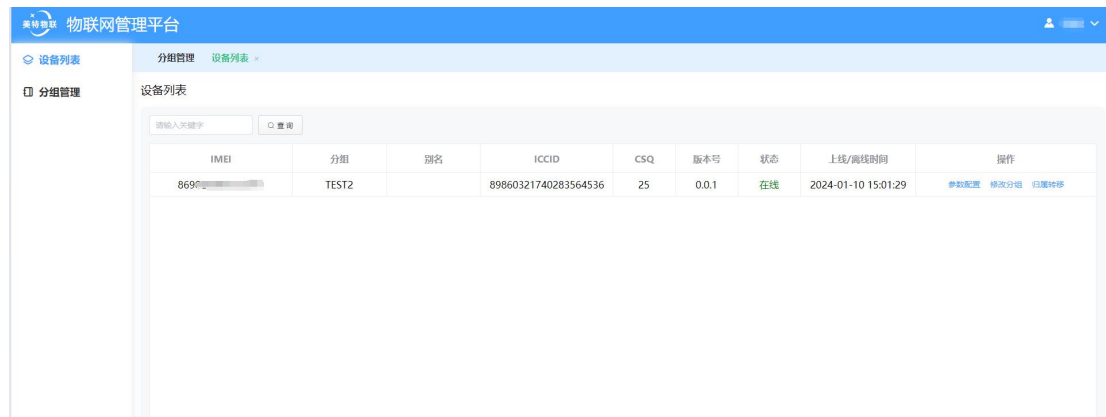
本设备可通过 TLL、RS232、RS485 三种接口实现与下位机的数据通信，可通过 MQTT、TCP、UDP、HTTP 四种网络协议实现云端数据通信，且可通过配置直接接入 ONENET、阿里云、腾讯云、百度云物联网平台，能够通过透传或自定义数据解析/打包的方式实现数据采集与传输，且最多可同时支持四个网络通道连接。

### 4.1 配置平台介绍

配置平台网址为 <http://iot.matezk.cn>

在购买设备后，请联系供应商将购买设备绑定至你的配置平台账号下（若无平台账号可通过 <http://iot.matezk.cn> 注册）

登录配置平台后，点击左侧设备列表，可查看你账号下所存在的设备，点击设备对应的参数配置可查看当前该设备的配置情况，点击修改分组可更改设备配置组（此处只是更改配置分组，实际配置需要通过分组管理中参数配置更改），点击转移归属，可将设备添加到指定账号下。



点击页面左侧分组管理，可查看存在分组，点击右上角 **新增** 可添加新分组。



点击对应分组下的参数配置可更改该分组下所有设备的配置信息。

## 参数配置(TEST2)

基础配置	编码函数	解码函数	网络通道1	网络通道2	网络通道3	网络通道4
<b>串口配置</b> 波特率(bps) <input type="text" value="9600"/>						
数据位	<input checked="" type="radio"/> 8位	<input type="radio"/> 7位				
校验位	<input checked="" type="radio"/> 无校验	<input type="radio"/> 奇校验	<input type="radio"/> 偶校验			
停止位	<input checked="" type="radio"/> 1位	<input type="radio"/> 2位				
通讯方式	<input checked="" type="radio"/> TTL	<input type="radio"/> 232	<input type="radio"/> 485			
<b>采集方式</b> <input checked="" type="radio"/> 主动上报 <input type="radio"/> 轮询						

点击设备管理可管理该分组下存在的设备。

### 设备管理

请输入关键字

IMEI	别名	ICCID	版本号	状态	操作
869020066533371		89860321740283564536	0.0.1	在线	<a href="#">修改别名</a> <a href="#">移除分组</a>

## 4.2 配置参数说明

配置参数使用方法：登录物联网管理平台>分组管理>参数配置，以分组的形式修改参数配置，在此修改配置后该分组下所有设备参数都将被修改。

分组名称	设备数量	创建时间	操作
test3	0	2024-01-08 14:09:29	<a href="#">修改配置</a> <a href="#">设备管理</a> <a href="#">删除</a>
TEST2	1	2024-01-06 10:03:34	<a href="#">修改配置</a> <a href="#">设备管理</a> <a href="#">删除</a>
TEST1	0	2023-12-01 09:54:38	<a href="#">修改配置</a> <a href="#">设备管理</a> <a href="#">删除</a>

### 4.2.1 基础配置

基础配置条目下可配置设备串口通信波特率、数据位、校验位、停止位及通信接口类型（485/232/TTL）。采集方式可选择主动上报/轮询，选择主动上报时，设备只负责上下行数据通信，选择轮询时，设备将根据轮询周期定时通过串口下发轮询指令，轮询指令可以有几条，轮询指令为 HEX 类型数据。



基础配置
编码函数
解码函数
网络通道1
网络通道2
网络通道3
网络通道4

**串口配置**

波特率(bps)

数据位  8位  7位

校验位  无校验  奇校验  偶校验

停止位  1位  2位

通讯方式  TTL  232  485

**采集方式**

主动上报  轮询

轮询周期  秒

轮询指令  (+)

## 4.2.2 网络通道配置

网络通道 1 ~ 网络通道 4 配置参数都是相同的，在使用网络通道前需要先使能对应通道。

基础配置
编码函数
解码函数
网络通道1
网络通道2
网络通道3
网络通道4

禁用  使能

### 1) MQTT 参数说明

在网络通道下，通道类型选择 MQTT。

参数	描述	备注
用户名	MQTT 连接时使用的用户名	未使用时可不填写
密码	MQTT 连接时使用的密码	未使用时可不填写
客户端 ID	MQTT 连接时的客户端标识符	可使用通配符 $\${imeid}$ （通配符使用方法见第 3 章）
服务器 IP 或域名	连接服务器的 IP 地址或者域名	~
服务器端口	连接服务器的端口号	~
心跳包间隔	心跳包间隔	范围 15~600 秒
发布主题	设备发布时使用的 Topic	可使用通配符 $\${imeid}$ 、 $\${clientId}$ （通配符使用方法见第 3 章）， $\${clientId}$ 将被替换为客户 ID
订阅主题	设备订阅的 Topic	可使用通配符 $\${imeid}$ 、 $\${clientId}$ （通配符使用方法见第 3 章）， $\${clientId}$ 将被替换为客户 ID
是否开启 SSL	是否使用 SSL 加密	~

发布 QOS 级别	QOS 级别	~
发布是否为保留消息	发布消息是否为保留消息	如果将 RETAIN 标志位设置为 true, 那么 MQTT 服务器会将最近收到的一条 RETAIN 标志位为 true 的消息保存在服务器端 (内存或文件)

## 2) OneNet

在网络通道下, 通道类型选择 OneNet。可通过 OneNet 文档中心物联网平台文档, 查阅相关资料, 网址 [https://open.iot.10086.cn/doc/v5/fuse/detail/lot\\_platform](https://open.iot.10086.cn/doc/v5/fuse/detail/lot_platform)。

参数	描述	备注
注册类型	<p>自动注册: 填写相关参数后, 自动在对应产品下创建设备, 且自动接入 OneNet。</p> <p>一型一密: 需要在对应产品下先创建设备, 再正确填写相关参数, 设备才可接入 OneNet。</p> <p>一机一密: 需要在对应产品下先创建设备, 再正确填写相关参数, 设备密钥需要与所创建设备的密钥一一对应, 才可接入 OneNet。</p>	~
数据协议	<p>自定义透传: 上下行数据直接透传。</p> <p>物模型: 使用 OneNet 平台物模型功能, 需要用户编写编码和解码函数</p>	选择物模型需要用户了解 OneNet 平台物模型功能, 且具备一定 Lua 语言编程能力
用户 ID	OneNet 账号对应的用户 ID 可在用户中心>访问权限查看	只在选择自动注册/一型一密时需要使用
用户 Accesskey	OneNet 账号对应的 Accesskey 可在用户中心>访问权限查看	只在选择自动注册/一型一密时需要使用
产品 ID	需要接入的产品 ID	~
设备名称	要创建的设备名称	可使用通配符 $\${imei}$ $\${productId}$ , $\${productId}$ 将被替换为产品 ID (通配符使用方法见第 3 章)
设备密钥	创建设备后, 设备对应的密钥	只在一机一密时需要使用
心跳包间隔	心跳包间隔	范围 15~600 秒
是否开启 SSL	是否使用 SSL 加密	~
发布 QOS 级别	QOS 级别	~

## 3) 阿里云

在网络通道下，通道类型选择阿里云。可通过阿里云文档中心物联网平台文档，查阅相关资料，网址 <https://help.aliyun.com/zh/iot/>。

参数	描述	备注
注册类型	<p>自动注册：填写相关参数后，自动在对应产品下创建设备，且自动接入阿里云。</p> <p>一型一密：需要在对应产品下先创建设备，再正确填写相关参数，设备才可接入阿里云。</p> <p>一机一密：需要在对应产品下先创建设备，再正确填写相关参数，设备密钥需要与所创建设备的密钥一一对应，才可接入阿里云。</p>	~
数据协议	<p>自定义透传：上下行数据直接透传。</p> <p>物模型：使用阿里云平台物模型功能，需要用户编写编码和解码函数</p>	选择物模型需要用户了解阿里云平台物模型功能，且具备一定 Lua 语言编程能力
实例 ID	使用的物联网平台实例 ID	只在选择自动注册且使用新版物联网平台时需要填写，使用旧版时不要填写
地域(regionId)	物联网平台选择的地区决定	
ProductKey	创建的产品 ProductKey	~
AccessKey ID	在账号 AccessKey 下可查看	只在自动注册时使用
AccessKey Secret	在账号 AccessKey 下可查看	只在自动注册时使用
DeviceName	要创建的设备名称	可使用通配符 $\${imei}$ $\${productId}$ ， $\${productId}$ 将被替换为产品 ID (通配符使用方法见第 3 章)
ProductSecret	产品密钥，在对应产品下查看	只在一型一密时使用
DeviceSecret	设备密钥，在对应设备下查看	只在一机一密时使用
心跳包间隔	心跳包间隔	范围 15~600 秒
是否开启 SSL	是否使用 SSL 加密	~
发布 QOS 级别	QOS 级别	~

## 4) 百度云

在网络通道下，通道类型选择百度云。可通过百度云物联网核心套件文档，查阅相关资料，网址 <https://cloud.baidu.com/doc/IoTCore/index.html>。

参数	描述	备注
注册类型	自动注册：填写相关参数后，自动在对应产品下创建设备，且自动接入百度云。  一机一密：需要在对应产品下先创建设备，再正确填写相关参数，设备密钥需要与所创建设备的密钥一一对应，才可接入百度云。	~
地域(regionId)	根据账号地区选择	中国选择广州（gz）
IOTCoreId	实例 ID	~
Access Key	百度云安全认证 Access Key	只在自动注册时需要使用
Secret Key	百度云安全认证 Secret Key	只在自动注册时需要使用
模板 ID	设备需要使用的模板 ID	只在自动注册时需要使用
DeviceKey	要创建的设备名称	可使用通配符 $\${imei}$ $\${coreid}$ ， $\${coreid}$ 将被替换为产品 IOTCoreId（通配符使用方法见第 3 章）
DeviceSecret	设备密钥，在对应设备下查看	只在一机一密时使用
心跳包间隔	心跳包间隔	范围 15~600 秒
是否开启 SSL	是否使用 SSL 加密	~
发布 QOS 级别	QOS 级别	~

## 5) 腾讯云

在网络通道下，通道类型选择腾讯云。可通过腾讯云文档中心物联网开发平台，查阅相关资料，网址 <https://cloud.tencent.com/document/product/1081>。

参数	描述	备注
注册类型	<p>自动注册：填写相关参数后，自动在对应产品下创建设备，且自动接入腾讯云。</p> <p>一机一密：需要在对应产品下先创建设备，再正确填写相关参数，设备密钥需要与所创建设备的密钥一一对应，才可接入腾讯云。</p>	使用自动注册时需要在产品页打开动态注册配置，和自动创建设备开关。
数据协议	<p>自定义透传：上下行数据直接透传。</p> <p>物模型：使用腾讯云平台物模型功能，需要用户编写编码和解码函数</p>	选择物模型需要用户了解阿里云平台物模型功能，且具备一定 Lua 语言编程能力
产品 ID	产品 ID	~
ProductSecret	产品对应的产品密钥	打开动态注册后才可查看
设备名称	要创建的设备名称	可使用通配符 <code>\${imei}</code> <code>\${productId}</code> ， <code>\${productId}</code> 将被替换为产品 ID (通配符使用方法见第 3 章)
DeviceSecret	设备密钥，在对应设备下查看	只在一机一密时使用
心跳包间隔	心跳包间隔	范围 15~600 秒
是否开启 SSL	是否使用 SSL 加密	~
发布 QOS 级别	QOS 级别	~

### 6) TCP

在网络通道下，通道类型选择 TCP。

参数	描述	备注
服务器 IP 或域名	要连接的 TCP 服务器 IP/域名	~
服务器端口号	要连接的 TCP 服务器端口号	~
使能注册包	是否需要在连接成功后发送一个注册包到 TCP 服务器	~
注册包	连接成功后发送的注册包, 可选为字符串/HEX 类型数据	使能注册包选择使能时需要填写。字符串类型下可使用通配符 $\${imei}$ (通配符使用方法见第 3 章)
使能心跳包	是否需要周期性发送心跳包到服务器	~
心跳包间隔	心跳包发送周期	使能心跳包选择使能时需要填写
心跳包	心跳包内容, 可选为字符串/HEX 类型数据	使能心跳包选择使能时需要填写。字符串类型下可使用通配符 $\${imei}$ (通配符使用方法见第 3 章)
是否开启 SSL	是否使用 SSL 加密	~

### 7) UDP

在网络通道下，通道类型选择 UDP。

参数	描述	备注
服务器 IP 或域名	要连接的 UDP 服务器 IP/域名	~
服务器端口号	要连接的 UDP 服务器端口号	~

### 8) HTTP

在网络通道下，通道类型选择 HTTP。

参数	描述	备注
Url	资源路径	~
Method	固定为 POST 模式	~
请求类型	固定为 JSON	~
超时时间	POST 请求后, 最长等待时间	为 0 时表示永久等待
Basic	header 中 Authorization	可以不填写

## 5. 编码/解码函数

编码与解码函数使用 Lua 语言编写，Lua 版本为 **Lua5.3**。在编码/解码函数中可以使用 Lua5.3 相关库函数以及 LuatOS 相关 API。

Lua5.3 参考手册链接 <https://www.lua.org/manual/5.3/>

LuatOS 接口文档链接 <https://wiki.luatos.com/api/index.html>

**注：编码/解码函数中只能含有英文及英文字符，编码/解码函数出错可能导致设备异常，在修改编码/解码函数时，请先验证通过后在实际应用。**

编码/解码函数是上下行数据处理的重要环节，所有上行数据（传感器到云）都要经过编码函数处理，所有下行数据（云到传感器）都要经过解码函数处理。用户可在编码/解码函数中对数据进行处理、解析或打包，亦可不处理数据直接返回原始数据，实现云端到传感器/传感器到云端的数据透传。



### 5.1 编码函数

编码函数实现对上行数据（设备到云端）的处理和打包。DTU 设备从串口接收到数据后，会自动调用编码函数对数据进行处理，并将处理后的数据上传至配置连接的各个网络通道，串口接收到的所有数据都必须通过编码函数。

第一行代码固定 `return function(ch, buf, len)`，其定义了编码函数的传参，最后一行代码为 `end` 表示编码函数结束。

参数	类型	描述	备注
ch	字符串	表示该条数据要上传到的网络通道，其值可为以下内容： <b>1、2、3、4</b>	处理打包网络通道 1 的内容时， <code>ch == 1</code> 。 处理打包网络通道 2 的内容时， <code>ch == 2</code> 。 ...以此类推。
buf	字符串	DTU 设备从串口中接收到的数据	~
len	int	DTU 设备从串口中接收到的数据长度	~

## 5.2 解码函数

解码函数实现对下行数据（云端到设备）的处理和打包。DTU 设备从各个网络通道接收到数据后，会自动调用解码函数对数据进行处理，并将处理后的数据通过串口下发至连接的设备，网络通道 1~4 下行数据都必须通过解码函数。

第一行代码固定 `return function(ch, buf, len, p1, p2, p3, p4, p5)`，其定义了解码函数的传参，最后一行代码为 `end` 表示解码函数结束。

参数	类型	描述	备注
ch	字符串	表示该条数据是从那一个网络通道下发的，其值可为以下内容： <b>1、2、3、4</b>	处理打包网络通道 1 的内容时， <code>ch == 1</code> 。 处理打包网络通道 2 的内容时， <code>ch == 2</code> 。 ...以此类推。
buf	字符串	DTU 设备从该网络通道(ch)中接收到的数据	~
len	int	DTU 设备从该网络通道(ch)中接收到的数据长度	~
p1	~	保留	可省略
p2	~	保留	可省略
p3	~	保留	可省略
p4	~	保留	可省略
p5	~	保留	可省略

## 5.3 内部可调用函数

设备内部自定义了相关函数，用户可在编码/解码函数中直接调用。

函数声明 `tools.ticks()`

说明：获取启动后的 tick 数,注意会出现溢出会出现负数。

返回值类型	解释
int	当前 tick 值

函数声明 `tools.imei()`

说明：获取当前的 IMEI 值

返回值类型	解释
string	当前的 IMEI 值,若失败返回 nil

函数声明 `tools.timestamp()`

说明：时间戳函数

返回值类型	解释
-------	----



int	时间戳
-----	-----

函数声明 `tools.date(format)`

说明：日期函数

传入值类型	解释
格式化字符串,可以是 nil	-- 值得注意的几点: -- 1. 若需要 UTC 时间, <code>fmt</code> 的第一个字符写"!" -- 2. <code>fmt</code> 的格式化遵循 C 函数 <code>strftime</code> , 可以查阅 <a href="https://developer.aliyun.com/article/320480">https://developer.aliyun.com/article/320480</a>

返回值类型	解释
table/string	根据 <code>fmt</code> 的不同,返回值不同

函数声明 `tools.crc16(data)`

说明：直接计算 modbus 的 crc16 值

传入值类型	解释
string	数据

返回值类型	解释
int	对应的 CRC16 值

## 6. 使用示例

本章节介绍如何使用本设备实现数据采集并上传数据到 OneNet、阿里云、腾讯云、百度云或其他网络平台。

### 6.1 设备连接使用前准备

- 将天线连接好
- 将 SIM 卡插入设备
- 将串口引脚与串口调试助手连接好
- 设备上电

### 6.2 基础配置

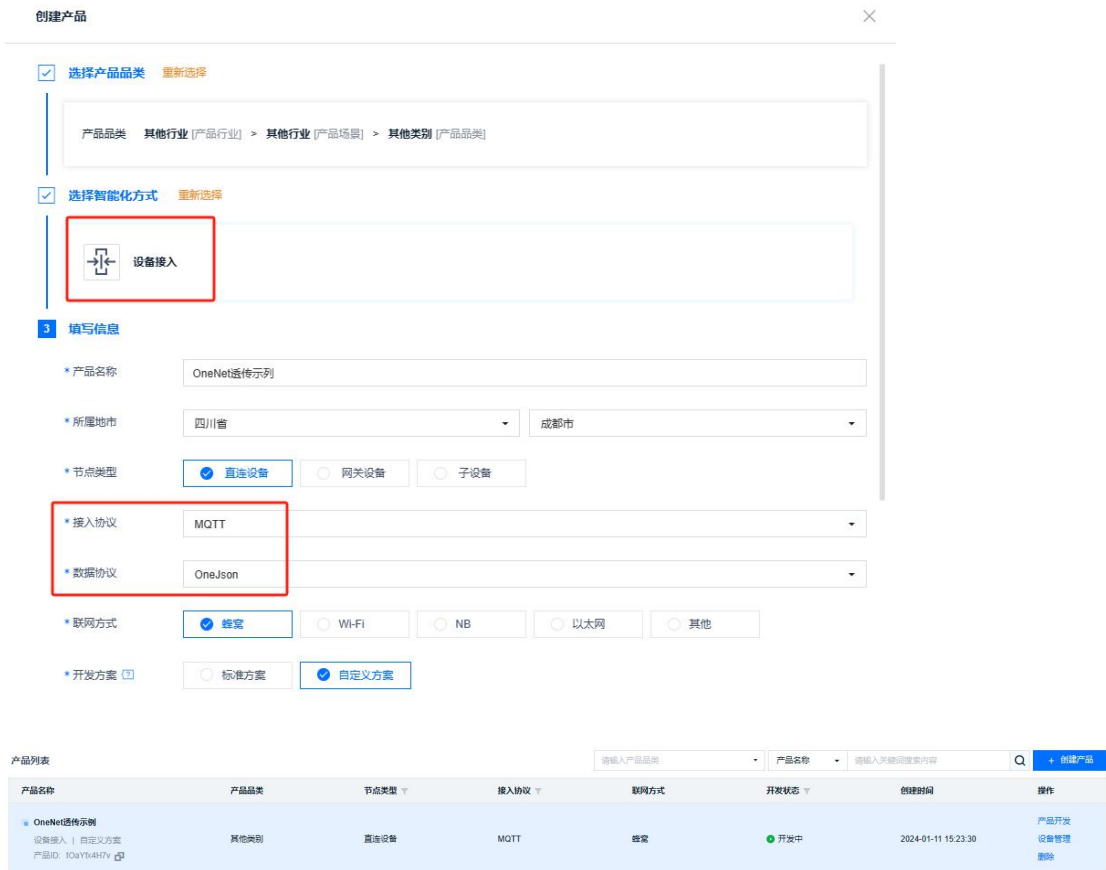
本章节使用的基础配置都为波特率 115200、数据位 8、无校验位、1 位停止位、TTL 接口、采集方式为主动上报。

基础配置	编码函数	解码函数	网络通道1	网络通道2	网络通道3	网络通道4
<b>串口配置</b>						
波特率(bps)	115200					
数据位	<input checked="" type="radio"/> 8位 <input type="radio"/> 7位					
校验位	<input checked="" type="radio"/> 无校验 <input type="radio"/> 奇校验 <input type="radio"/> 偶校验					
停止位	<input checked="" type="radio"/> 1位 <input type="radio"/> 2位					
通讯方式	<input checked="" type="radio"/> TTL <input type="radio"/> 232 <input type="radio"/> 485					
<b>采集方式</b>						
	<input checked="" type="radio"/> 主动上报 <input type="radio"/> 轮询					

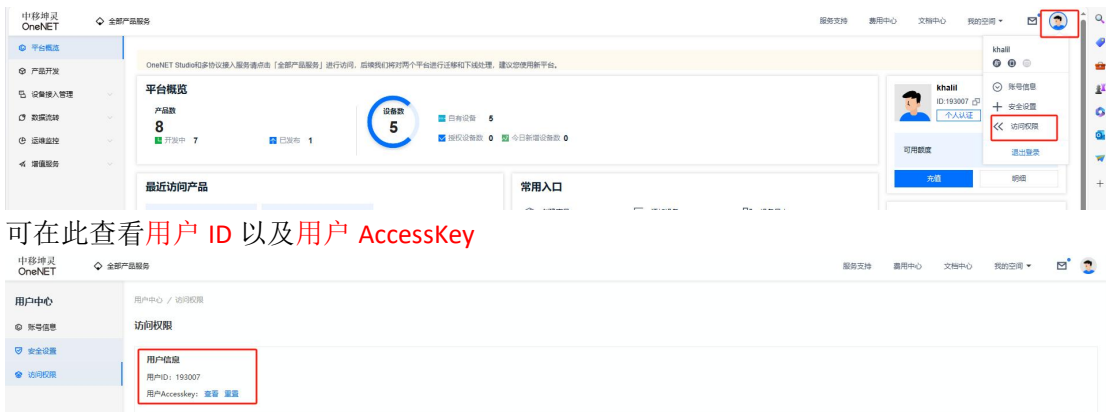
### 6.3 网络平台使用示例

#### 6.3.1 OneNet

登录 OneNet(中国移动物联网开放平台), 点击右上角 开发者中心>产品开发, 进入产品列表页面, 点击创建产品, 产品创建时智能化方式选择设备接入, 接入协议选择 MQTT, 数据协议 OneJson, 产品创建完成后可查看产品 ID, 本例中为 tOaYfx4H7v。



鼠标置于右上角头像处点击访问权限，进入访问权限页



可在此查看用户 ID 以及用户 AccessKey

### 1) 配置平台参数配置

#### ➤ 自动注册

使用物模型时，数据协议处勾选为物模型，使用自定义透传时，数据协议处勾选为自定义透传，其他参数相同

基础配置	编码函数	解码函数	网络通道1	网络通道2	网络通道3	网络通道4
禁用 <input checked="" type="checkbox"/> 使能 <input type="checkbox"/>						
通道类型	<input type="radio"/> MQTT <input checked="" type="radio"/> OneNet <input type="radio"/> 阿里云 <input type="radio"/> 百度云 <input type="radio"/> 腾讯云 <input type="radio"/> TCP <input type="radio"/> UDP <input type="radio"/> HTTP					
注册类型	<input checked="" type="radio"/> 自动注册 <input type="radio"/> 一型一密 <input type="radio"/> 一机一密					
数据协议	<input checked="" type="radio"/> 自定义透传 <input type="radio"/> 物模型					
用户ID	<input type="text" value="193007"/>					
用户Accesskey	<input type="text" value="ZS...pdhF60NA"/>					
产品ID	<input type="text" value="tOaYfx4H7v"/>					
设备名称	<input type="text" value="MT\${imei}"/>					
心跳包间隔	<input type="text" value="300"/> 秒					
是否开启SSL	<input checked="" type="radio"/> 否 <input type="radio"/> 是					
发布QOS级别	<input checked="" type="radio"/> 0 <input type="radio"/> 1					

### ➤ 一型一密

使用物模型时，数据协议处勾选为物模型，使用自定义透传时，数据协议处勾选为自定义透传，其他参数相同，为一型一密时需要用户先在对应产品下创建设备，创建流程设备管理>添加设备。

基础配置	编码函数	解码函数	网络通道1	网络通道2	网络通道3	网络通道4
禁用 <input checked="" type="checkbox"/> 使能 <input type="checkbox"/>						
通道类型	<input type="radio"/> MQTT <input checked="" type="radio"/> OneNet <input type="radio"/> 阿里云 <input type="radio"/> 百度云 <input type="radio"/> 腾讯云 <input type="radio"/> TCP <input type="radio"/> UDP <input type="radio"/> HTTP					
注册类型	<input type="radio"/> 自动注册 <input checked="" type="radio"/> 一型一密 <input type="radio"/> 一机一密					
数据协议	<input checked="" type="radio"/> 自定义透传 <input type="radio"/> 物模型					
用户ID	<input type="text" value="193007"/>					
用户Accesskey	<input type="text" value="ZS...pdhF60NA"/>					
产品ID	<input type="text" value="tOaYfx4H7v"/>					
设备名称	<input type="text" value="MT\${imei}"/>					
心跳包间隔	<input type="text" value="300"/> 秒					
是否开启SSL	<input checked="" type="radio"/> 否 <input type="radio"/> 是					
发布QOS级别	<input checked="" type="radio"/> 0 <input type="radio"/> 1					

### ➤ 一机一密

使用物模型时，数据协议处勾选为物模型，使用自定义透传时，数据协议处勾选为自定义透传，其他参数相同，为一机一密时需要用户先在对应产品下创建设备，创建流程设备管理>添加设备，创建设备后可在设备详情获取设备密钥。

设备接入管理 > 设备详情

← **MT869020066533371**

设备详情 属性 事件记录 服务记录 操作记录 文件管理 设备调试 SIM卡查询

**设备信息**

设备密钥: ZFNyVUFveVIT0hVkhQS202czFQb1VDQ... Auth\_Code: -

激活时间: 2024-01-11 16:04:01 最近在线时间: 2024-01-11 16:04:01

设备位置: - 设备描述: -

基础配置
编码函数
解码函数
网络通道1
网络通道2
网络通道3
网络通道4

禁用  使能

通道类型  MQTT  OneNet  阿里云  百度云  腾讯云  TCP  UDP  HTTP

注册类型  自动注册  一型一密  一机一密

数据协议  自定义透传  物模型

产品ID

设备名称

设备密钥

心跳包间隔  秒

是否开启SSL  否  是

发布QOS级别  0  1

添加设备 ×

单个设备
批量添加

\* 所属产品

\* 设备名称

## 2) OneNet 透传

### ➤ 创建物模型

从产品列表点击对应产品右侧的产品开发，进入产品页面，点击功能定义>物模型>设置物模型>添加自定义功能点，在此添加 2 个物模型功能点分别为 **downlink** 和 **uplink**，添加完成后点击保存。

添加自定义功能点 ×

\* 功能类型

\* 功能名称

\* 标识符

\* 数据类型

\* 数据长度  字符

\* 读写类型

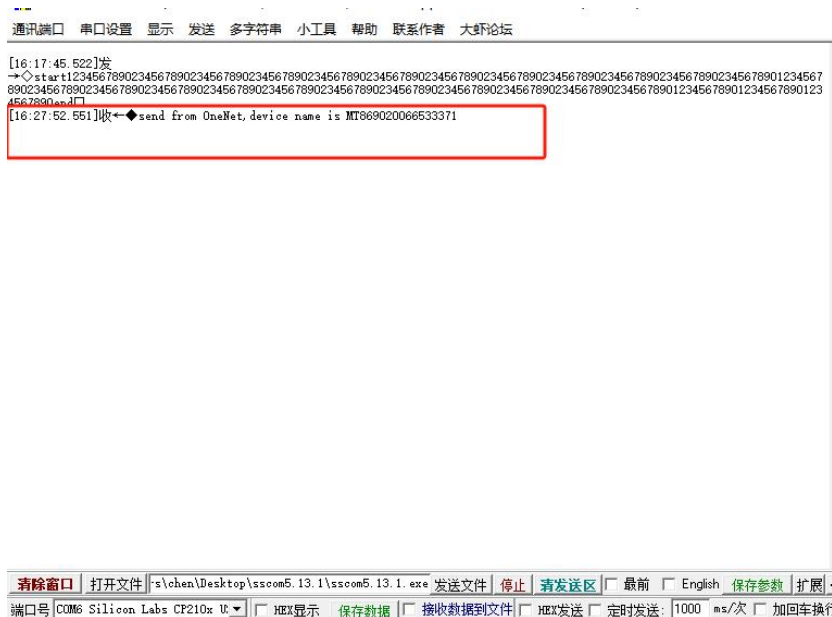


➤ 数据下发

设备详情>设备调试>应用模拟器>属性设置，勾选下行数据，在输入框内输入要下发的数据，输入完成后点击下发属性设置。



在串口调试助手可看到云端下发数据。



➤ 编码函数

透传时编码函数固定为，若用户没有 LUA 语言编程基础不建议修改编码/解码函数。源

码可在第 5 章中查看。

基础配置 编码函数 解码函数 网络通道1 网络通道2 网络通道3 网络通道4

```
return function(ch, buf, len)
    return buf, len
end
```

### ➤ 解码函数

透传时解码函数固定为以下内容，若用户没有 LUA 语言编程基础不建议修改编码/解码函数。源码可在第 5 章中查看。

参数配置(test3)

基础配置 编码函数 解码函数 网络通道1 网络通道2 网络通道3 网络通道4

```
return function(ch, buf, len, p1, p2, p3, p4, p5)
    return buf, len
end
```

### 3) OneNet 物模型

在参数配置需中，数据协议选择物模型，选用物模型时用户必须根据定义物模型编写相关编码/解码函数，注意根据编码/解码函数编写可能会导致设备死机，若用户没有 LUA 语言编程基础不建议使用物模型通信。



本示例中，将上行数据（温度值）通过编码函数，将数据编码为 OneJson 格式，再将其上传至 OneNet。通过解码函数将下行数据（温度系数），从 OneJson 格式数据中解析出来，传递给串口助手。

### ➤ 物模型创建

创建上行数据功能点

添加自定义功能点 ×

---

\* 功能类型

\* 功能名称

\* 标识符

\* 数据类型

\* 取值范围  -

步长

单位

\* 读写类型



## 创建下行数据功能点

添加自定义功能点×

---

\* 功能类型

\* 功能名称

\* 标识符

\* 数据类型

\* 取值范围  -

步长

单位

\* 读写类型

## ➤ 编码函数

此处 `ch == 1`，是因为本例中在物联网管理平台将 OneNet 配置为网络通道 1

源码：

```
return function(ch, buf, len)
  if ch == 1 then
    if string.find(buf, '=') ~= nil then
      local offset = string.find(buf, '=')
      local temp_value = string.sub(buf, tonumber(offset) + 1)
      buf = string.format('{ "id": "%d", "params": { "temp":
{"value": %s} } }', os.time() % 1000000, temp_value)
      return buf, #buf
    end
  end
end
```

## ➤ 解码函数

此处 `ch == 1`，是因为本例中在物联网管理平台将 OneNet 配置为网络通道 1

源码：

```
return function (ch, buf, len, p1, p2, p3, p4, p5)
  if ch == 1 then
    local json_obj, result, err = json.decode(buf)
    if result ~= 1 or json_obj == nil then
      return
    end
    if json_obj['params'] == nil or json_obj['id'] == nil then
      return
    end
  end
end
```

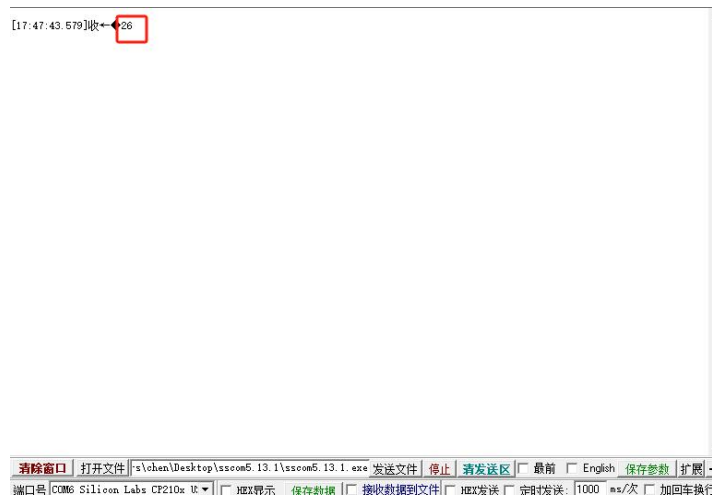
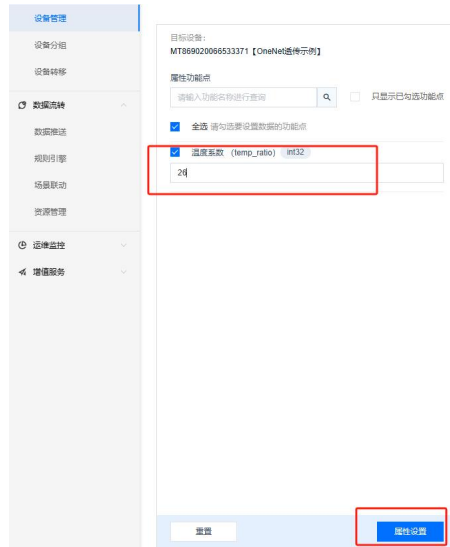
```

return json_obj["params"]["temp_ratio"],
string.len(json_obj["params"]["temp_ratio"])
end
end

```

➤ 数据下发

设备详情>设备调试>应用模拟器>属性设置，勾选温度系数，在输入框内输入要下发的数据，输入完成后点击下发属性设置。

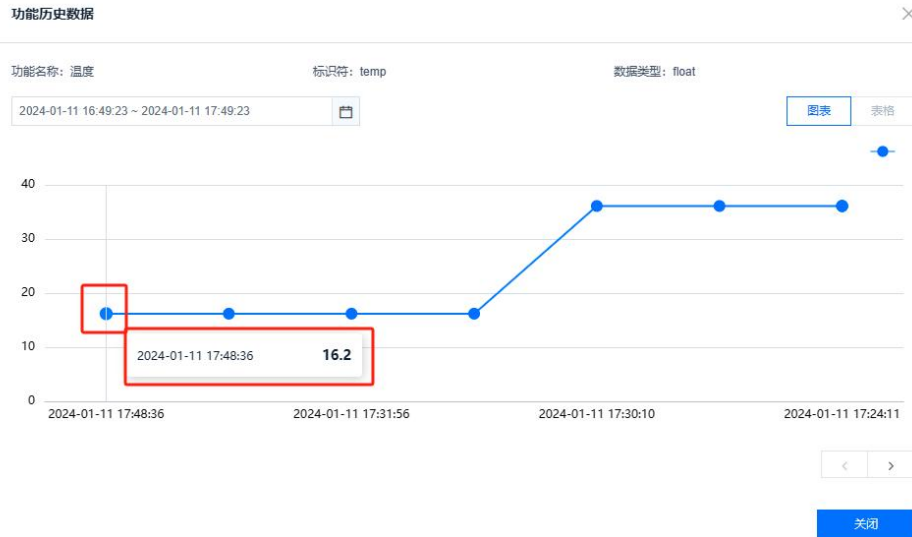


➤ 数据上传

通过串口助手发送 **temperature=16.2**，设备详情>属性>下行数据>右上角历史记录，可查看接收到的数据。

[17:48:35.204]发->temperature=16.2

清除窗口 打开文件: s:\chen\Desktop\sscom5.13.1\sscom5.13.1.exe 发送文件 停止 清除发送区 最前 English 保存参数 扩展  
端口号: COM6 Silicon Labs CP210x UART HEX显示 保存数据 接收数据到文件 HEX发送 定时发送: 1000 ms/次 加回车换行  
关闭串口 更多串口设置 加时间戳和分包显示 超时时间: 20 ms 第1字节至末尾 加校验: None  
RTS DTR 波特率: 115200 temperature=16.2



## 6.3.2 阿里云

登录阿里云点击右上角控制台，从搜索框中搜索物联网平台，进入物联网平台，在此页面根据需求选择相关实例（公共实例/企业版实例），此处可查看实例 ID（新版阿里云物联网平台才有，自动注册时需要使用）。



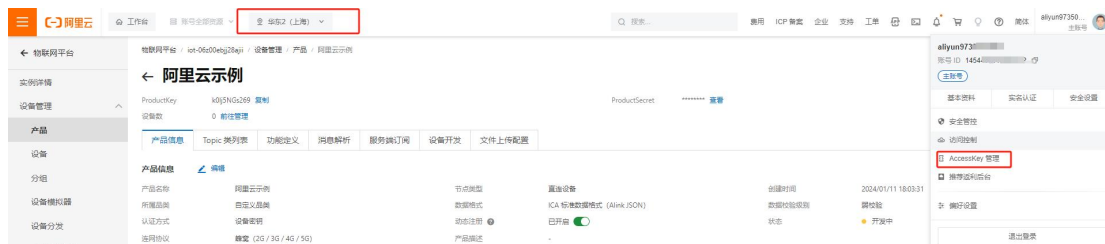
点击使用的实例>点击设备管理>产品>创建产品，进入产品创建界面。注意数据格式条目，配置参数的数据协议中使用的是自定义透传时请选择透传/自定义，使用的物模型时请选择 ICA 标准数据格式（Alink JSON）。

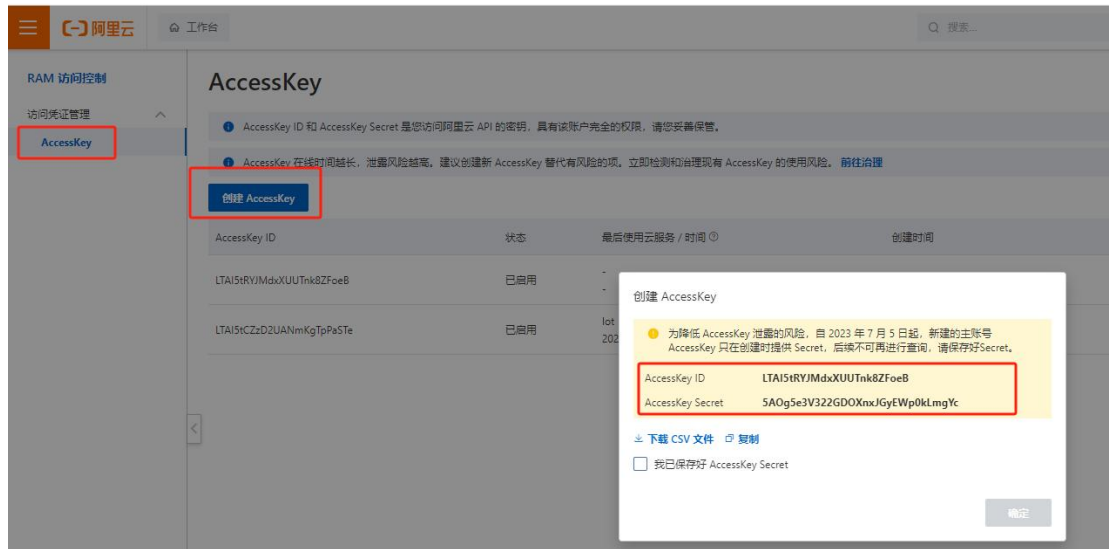


创建完成后点击查看产品，可在此处获取 **ProductKey** 和 **ProductSecret**，使用到一型一密或自动注册时需要开启动态注册开关。



进入实例后可从上方工作栏中查看地域信息，将鼠标置于右上角头像处，点击 **AccessKey 管理**>创建 **AccessKey**，创建完成后可获取 **AccessKey ID** 和 **AccessKey Secret**。





### 1) 配置平台参数配置

通道类型选择阿里云，下列数据协议栏（自定义透传/物模型）根据用户需要自行选择。

#### ➤ 自动注册

首次通过自动注册创建设备时请注意对应产品下是否已存在设备名与本次创建设备名相同的设备，若有可能导致自动注册失败。



#### ➤ 一型一密

使用一型一密时，需要现在阿里云对应产品下创建设备，DeviceName 要与创建的设备名称相同。

基础配置	编码函数	解码函数	网络通道1	网络通道2	网络通道3	网络通道4
禁用 <input type="checkbox"/> 使能 <input checked="" type="checkbox"/>						
通道类型	<input type="radio"/> MQTT	<input type="radio"/> OneNet	<input checked="" type="radio"/> 阿里云	<input type="radio"/> 百度云	<input type="radio"/> 腾讯云	<input type="radio"/> TCP <input type="radio"/> UDP <input type="radio"/> HTTP
注册类型	<input type="radio"/> 自动注册	<input checked="" type="radio"/> 一型一密	<input type="radio"/> 一机一密			
数据协议	<input checked="" type="radio"/> 自定义透传	<input type="radio"/> 物模型				
地域(regionId)	华东2(cn-shanghai) ▼					
ProductKey	k0lj5NGs269					
ProductSecret	6QeghBLKmF81Hb60					
DeviceName	MT\${imei}\${productId}					
心跳包间隔	120	秒				
是否开启SSL	<input checked="" type="radio"/> 否	<input type="radio"/> 是				
发布QOS级别	<input checked="" type="radio"/> 0	<input type="radio"/> 1	<input type="radio"/> 2			
发布是否为保留消息	<input checked="" type="radio"/> 否	<input type="radio"/> 是				

### ➤ 一机一密

使用一机一密时，需要现在阿里云对应产品下创建设备，DeviceName 要与创建的设备名称相同。设备创建完成后点击设备右侧查看进入设备信息页，可获取 DeviceSecret

物联网平台 / iot-06z00ebj28aji / 设备管理 / 设备 / MT869020066533371k0lj5N...

← **MT869020066533371k0lj5NGs269** 在线

产品 [阿里云示例](#) [查看](#)

ProductKey [k0lj5NGs269](#) [复制](#)

DeviceSecret \*\*\*\*\* [查看](#)

设备信息 | [Topic 列表](#) | [物模型数据](#) | [设备影子](#) | [文件管理](#) | [日志服务](#) | [在线调试](#) | [分组](#) | [任务](#)

已订阅 Topic 列表

基础配置	编码函数	解码函数	网络通道1	网络通道2	网络通道3	网络通道4
禁用 <input type="checkbox"/> 使能 <input checked="" type="checkbox"/>						
通道类型	<input type="radio"/> MQTT	<input type="radio"/> OneNet	<input checked="" type="radio"/> 阿里云	<input type="radio"/> 百度云	<input type="radio"/> 腾讯云	<input type="radio"/> TCP <input type="radio"/> UDP <input type="radio"/> HTTP
注册类型	<input type="radio"/> 自动注册	<input type="radio"/> 一型一密	<input checked="" type="radio"/> 一机一密			
数据协议	<input checked="" type="radio"/> 自定义透传	<input type="radio"/> 物模型				
地域(regionId)	华东2(cn-shanghai) ▼					
ProductKey	k0lj5NGs269					
DeviceSecret	9c02e3de1a27679a2509d923bc47ea29					
DeviceName	MT\${imei}\${productId}					
心跳包间隔	120	秒				
是否开启SSL	<input checked="" type="radio"/> 否	<input type="radio"/> 是				
发布QOS级别	<input checked="" type="radio"/> 0	<input type="radio"/> 1	<input type="radio"/> 2			
发布是否为保留消息	<input checked="" type="radio"/> 否	<input type="radio"/> 是				

### 2) 阿里云自定义透传

使用自定义透传时首先需要创建产品，产品创建如上所述，注意在创建产品时数据格式选择透传/自定义。

连网与数据

\* 连网方式

蜂窝 (2G / 3G / 4G / 5G) ▼

数据格式

透传/自定义 ▼

需要设置透传 Topic，点击查看产品>Topic 类列表>自定义 Topic>定义 Topic 类，创建透传上下行 Topic，**up** 和 **down**。创建好产品且定义好相关 Topic 后，在物联网管理平台配置好对应参数，等待设备连接阿里云即可。

定义 Topic 类
✕

**1** Topic 格式必须以"/"进行分层，区分每个类目。其中前三个类目已经规定好，第一个代表产品标识 ProductKey，第二个 \${deviceName} 适配 DeviceName，第三个 user 用来标识产品的自定义 Topic 类。简单来说，Topic 类：/a15T\*\*\*\*dhK/\${deviceName}/user/update 是具体 Topic：/a15T\*\*\*\*dhK/mydevice1/user/update 和 /a15T\*\*\*\*dhK/mydevice2/user/update 等的集合。

\* 设备操作权限

发布
▼

\* Topic 类

up

**1** 这里的压缩解压缩是指物联网平台会对消息做压缩解压缩

\* 是否开启压缩/解压缩

定义 Topic 类
✕

**1** Topic 格式必须以"/"进行分层，区分每个类目。其中前三个类目已经规定好，第一个代表产品标识 ProductKey，第二个 \${deviceName} 适配 DeviceName，第三个 user 用来标识产品的自定义 Topic 类。简单来说，Topic 类：/a15T\*\*\*\*dhK/\${deviceName}/user/update 是具体 Topic：/a15T\*\*\*\*dhK/mydevice1/user/update 和 /a15T\*\*\*\*dhK/mydevice2/user/update 等的集合。

\* 设备操作权限

订阅
▼

\* Topic 类

down

**1** 这里的压缩解压缩是指物联网平台会对消息做压缩解压缩

\* 是否开启代理订阅

**1** 这里的压缩解压缩是指物联网平台会对消息做压缩解压缩

\* 是否开启压缩/解压缩

创建完成后在自定义 Topic 下多出如下 Topic。

物联网平台 / iot-06e00ebj28aji / 设备管理 / 产品 / 阿里云示例
物联网发布

← 阿里云示例
ProductSecret \*\*\*\*\* 查看

ProductKey k0lj5E0vKkX 复制
设备数 1 前往管理

产品信息
Topic 类列表
功能定义
消息解析
服务端订阅
设备开发
文件上传配置

基础通信 Topic
物模型通信 Topic
自定义 Topic

定义 Topic 类
请输入 Topic
?

自定义 Topic	操作权限	是否开启代理订阅	压缩/解压缩	描述	操作
/k0lj5E0vKkX/\${deviceName}/down	订阅	<input type="checkbox"/>	未开启	-	
/k0lj5E0vKkX/\${deviceName}/up	发布	-	未开启	-	



创建完成 Topic 后，点击设备>查看设备>Topic 列表，可见以订阅 Topic 列表中多出一个.../down。**注：**若未出现该 Topic 且设备显示已经上线，可重启设备，重新连接阿里云即可解决，该问题是由于产品创建后，通过配置平台设置参数后设备已经连接阿里云，但这时还未在产品下创建 up 和 down 两个 Topic 故导致设备无法通过这两个 Topic 通信。



### ➤ 数据下发

在设备 Topic 列表中，点击.../down 后的发布信息，可弹出消息发布窗口，在该窗口输入需要发布的信息，点击发送即可。



通过串口助手查看接收数据正确。

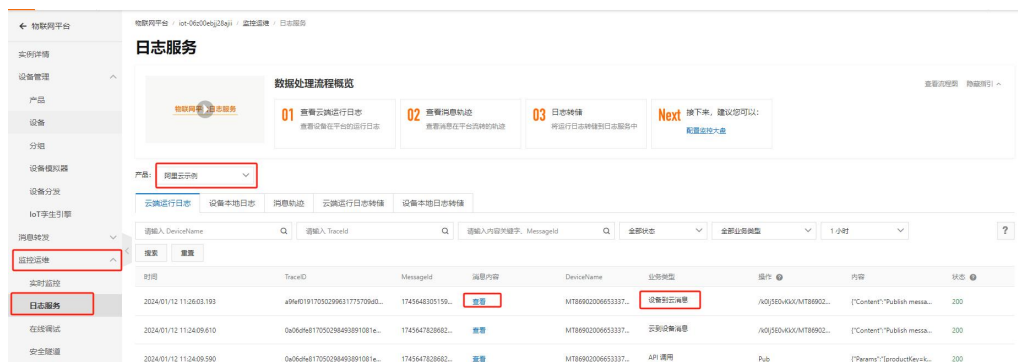
[11:24:09.987]收←◆hello, it's aliyun !!!

- 数据上传  
通过串口助手发送数据。

[11:26:03.154]发→hello, it's sscom !!!□



在阿里云平台中，点击监控运维>日志服务>产品选择对应创建的产品（本例为阿里云示例），在此可查看设备日志，业务类型为设备到云消息，则表示设备上传的数据，点击查看，可得到透传数据。可在内容栏选择数据显示方式，此处为 Text(UTF-8)。



- 编码函数  
透传时编码函数固定为，若用户没有 LUA 语言编程基础不建议修改变码/解码函数。可在第 5 节中查看。

基础配置   编码函数   解码函数   网络通道1   网络通道2   网络通道3   网络通道4

```
return function(ch, buf, len)
    return buf, len
end
```

➤ 解码函数

透传时解码函数固定为以下内容，若用户没有 LUA 语言编程基础不建议修改编码/解码函数。可在第 5 节中查看。

参数配置(test3)

基础配置   编码函数   解码函数   网络通道1   网络通道2   网络通道3   网络通道4

```
return function(ch, buf, len, p1, p2, p3, p4, p5)
    return buf, len
end
```

3) 阿里云物模型

参数配置需中，数据协议选择物模型，在阿里云创建产品时数据格式选择 ICA 标准数据格式（Alink JSON），选用物模型时用户必须根据定义的物模型编写相关编码/解码函数，注意根据编码/解码函数编写可能会导致设备死机，若用户没有 LUA 语言编程基础不建议使用物模型通信。在阿里云物联网平台创建好产品后，通过物联网平台配置对应参数，等待设备连接阿里云即可。

数据协议    自定义透传    物模型

### ← 新建产品 (设备模型)

\* 产品名称

阿里云示例

\* 所属品类

标准品类
  自定义品类

\* 节点类型

---

#### 连网与数据

\* 连网方式

蜂窝 (2G / 3G / 4G / 5G)

\* 数据格式

ICA 标准数据格式 (Alink JSON)

[校验类型](#)  
[认证方式](#)

更多信息

[产品描述](#)

本物模型使用示例中，用串口向 DTU 模块发送数据，数据内容为 `temperature=17.5`，通过编码函数将数据提前并打包为 Alink JSON 格式。最终上传数据为。。。。。

通过阿里云物联网平台向 DTU 设备发送温度系数，温度系数值为 13，实际下发数据为 `{"method":"thing.service.property.set","id":"1946882363","params":{"temperature_ratio":13},"version":"1.0.0"}`，通过解码函数提取数据 13，并向串口发送提前到的数据 13。

使用阿里云物模型通信时，需要先定义相关物模型，创建方法为设备管理>产品>查看对应产品>功能定义>前往编辑草稿/编辑草稿>添加自定义功能，添加温度和温度系数属性。添加完成后点击发布上线

物联网平台 / iot-06z00ebjj28ajii / 设备管理 / 产品 / 阿里云示例

### ← 阿里云示例

ProductKey: k0lj5slgdTG [复制](#) ProductSecret: \*\*\*\*

设备数: 0 [前往管理](#)

[产品信息](#)
[Topic 类别表](#)

[消息解析](#)
[服务端订阅](#)
[设备开发](#)
[文件上传配置](#)

ⓘ 当前展示的物模型是已发布到线上的版本: 1705046965640 (2024/01/12 16:09:25), 如需修改, 请点击 [编辑草稿](#)

[前往编辑草稿](#)

请输入模块名称

默认模块	功能类型	功能名称 (全部)	标识符	数据类型
默认模块				

添加温度属性，标识符为 **temperature**

The screenshot shows a dialog box titled "添加自定义功能" (Add Custom Function). It contains the following fields and options:

- 功能类型 (Function Type):** 属性 (Attribute) is selected.
- 功能名称 (Function Name):** 温度 (Temperature)
- 标识符 (Identifier):** temperature
- 数据类型 (Data Type):** float (单精度浮点型)
- 取值范围 (Value Range):** -100 ~ 100
- 步长 (Step):** 0.1
- 单位 (Unit):** 摄氏度 / °C
- 读写类型 (Read/Write Type):** 只读 (Read-only) is selected.

添加温度系数属性，标识符为 **temperature\_ratio**

The screenshot shows a dialog box titled "添加自定义功能" (Add Custom Function). It contains the following fields and options:

- 功能类型 (Function Type):** 属性 (Attribute) is selected.
- 功能名称 (Function Name):** 温度系数 (Temperature Coefficient)
- 标识符 (Identifier):** temperature\_ratio
- 数据类型 (Data Type):** int32 (整型)
- 取值范围 (Value Range):** 0 ~ 100
- 步长 (Step):** 1
- 单位 (Unit):** 请选择单位 (Please select a unit)
- 读写类型 (Read/Write Type):** 读写 (Read/Write) is selected.

## ← 编辑草稿

产品名称 阿里云示例

ProductKey

K0j5slgdTG 复制

快速导入 物模型 TSL 历史版本

+ 请输入模块名称

默认模块

默认模块

添加标准功能 添加自定义功能

功能类型	功能名称 (全部)	标识符 作	数据类型	数据定义	操作
属性	温度系数 <a href="#">自定义</a>	temperature_ratio	int32 (整数型)	取值范围: 0 ~ 100	<a href="#">编辑</a>   <a href="#">删除</a>
属性	温度 <a href="#">自定义</a>	temperature	float (单精度浮点型)	取值范围: -100 ~ 100	<a href="#">编辑</a>   <a href="#">删除</a>

发布上站

返回

## ➤ 解码函数示例

 此处 `ch == 2`，是因为本例中在物联网管理平台将阿里云配置为网络通道 2

源码：

```
return function(ch, buf, len, p1, p2, p3, p4, p5)
    if ch == 2 then
        local json_obj, result, err = json.decode(buf)
        if result ~= 1 or json_obj == nil then
            return
        end
        if json_obj['method'] ~= "thing.service.property.set" or
json_obj['id'] == nil or json_obj['params'] == nil then
            return
        end
        return json_obj['params']['temperature_ratio'],
string.len(json_obj['params']['temperature_ratio'])
    end
end
```

## ➤ 编码函数示例

 此处 `ch == 2`，是因为本例中在物联网管理平台将阿里云配置为网络通道 2

源码：

```
return function(ch, buf, len)
    if ch == 2 then
        if string.find(buf, '=') ~= nil then
            local offset = string.find(buf, '=')
            local temperature_value = string.sub(buf, tonumber(offset) + 1)
            buf = string.format('{ "id":
"%d", "version": "1.0", "params": { "temperature":
```

```

{"value": %s}}, "method": "thing.event.property.post"}', math.random(1,
10000), temperature_value)
        return buf, #buf
    end
end
end
end
    
```

### ➤ 数据下发

通过物联网平台下监控运维>在线调试>选择对应产品>选择对应设备>选择属性调试, 在温度系数栏输入数据 **13**, 点击设置, 则数据下发完成。



通过串口调试助手查看数据, 得到解析后的数据 **13**。

[16:58:49.249]收←◆13



### ➤ 数据上传

通过串口助手发送数据 **temperature=17.5**, 经过编码函数解析数据得到实际温度值 17.5, 再将其打包为符合 Alink JSON 要求的数据包:

```

{"id":"8407","version":"1.0","params":{"temperature":{"value":17.5}}, "method": "thing.event.pro
perty.post"}
    
```

再通过 DTU 设备上传到阿里云, 在阿里云物联网平台, 设备管理>设备>选择对应设备>物模型数据>运行状态>默认模块, 可查看到上报温度为 17.5。





### 6.3.3 腾讯云

登录腾讯云点击右上角控制台，从搜索框中搜索物联网开发平台，进入物联网平台，在此页面根据需要选择相关实例（公共实例/企业版实例）。

进入实例后从项目列表>新建项目，创建项目，项目创建完成后进入项目，点击产品开发>新建产品，新建好产品后，可在产品开发页查看产品 ID。新建产品时需要注意，若使用自定义透传则在数据协议栏选择自定义透传、若使用物模型则选择物模型，在物联网管理平台设置参数时也要与此处对应。认证方式选择密钥认证。



#### 产品开发 / 新建产品

**新建产品**

产品名称 \* 腾讯云透传示例  
支持中文、英文、数字、下划线、空格（非首尾字符）、中英文括号、-、@、\、/的组合，最多不超过40个字符

产品品类 标准品类 自定义品类

设备类型 设备 网关 子设备

通信方式 \* 2G/3G/4G  
请根据业务场景正确选择产品的通信方式，否则会影响后续产品开发

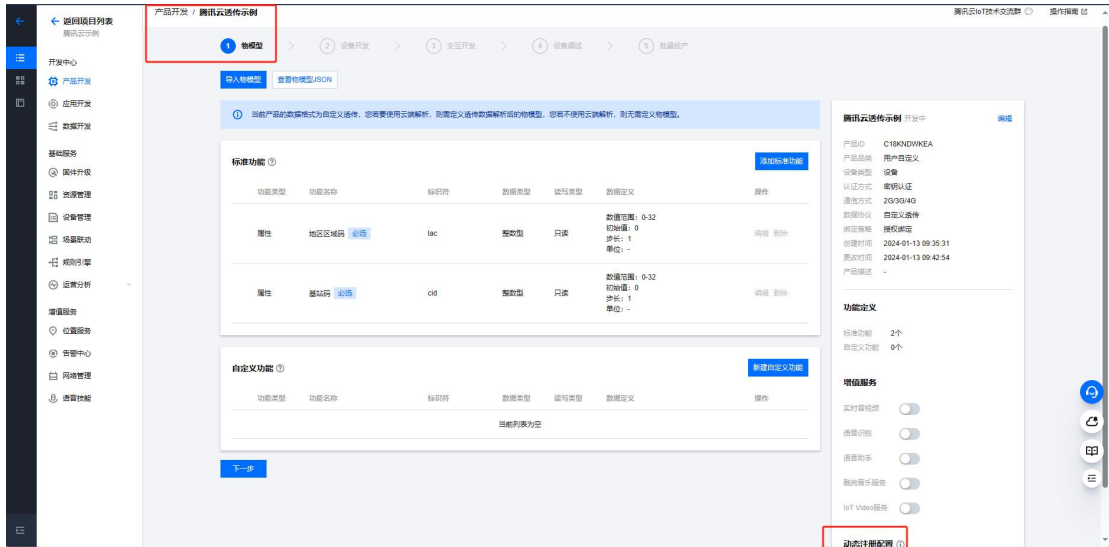
认证方式 密钥认证 证书认证

数据协议 物模型 自定义透传 ⓘ

描述 选填  
最多不超过80个字符

确定 取消

进入对应产品，在页面右下角动态注册配置栏，打开动态注册及自动创建上报开关，打开后可获取 **ProductSecret**。



### 1) 配置平台参数配置

通道类型选择腾讯云，下列数据协议栏（自定义透传/物模型）根据用户需要自行选择，需要与腾讯云中创建产品选择的数据协议对应。

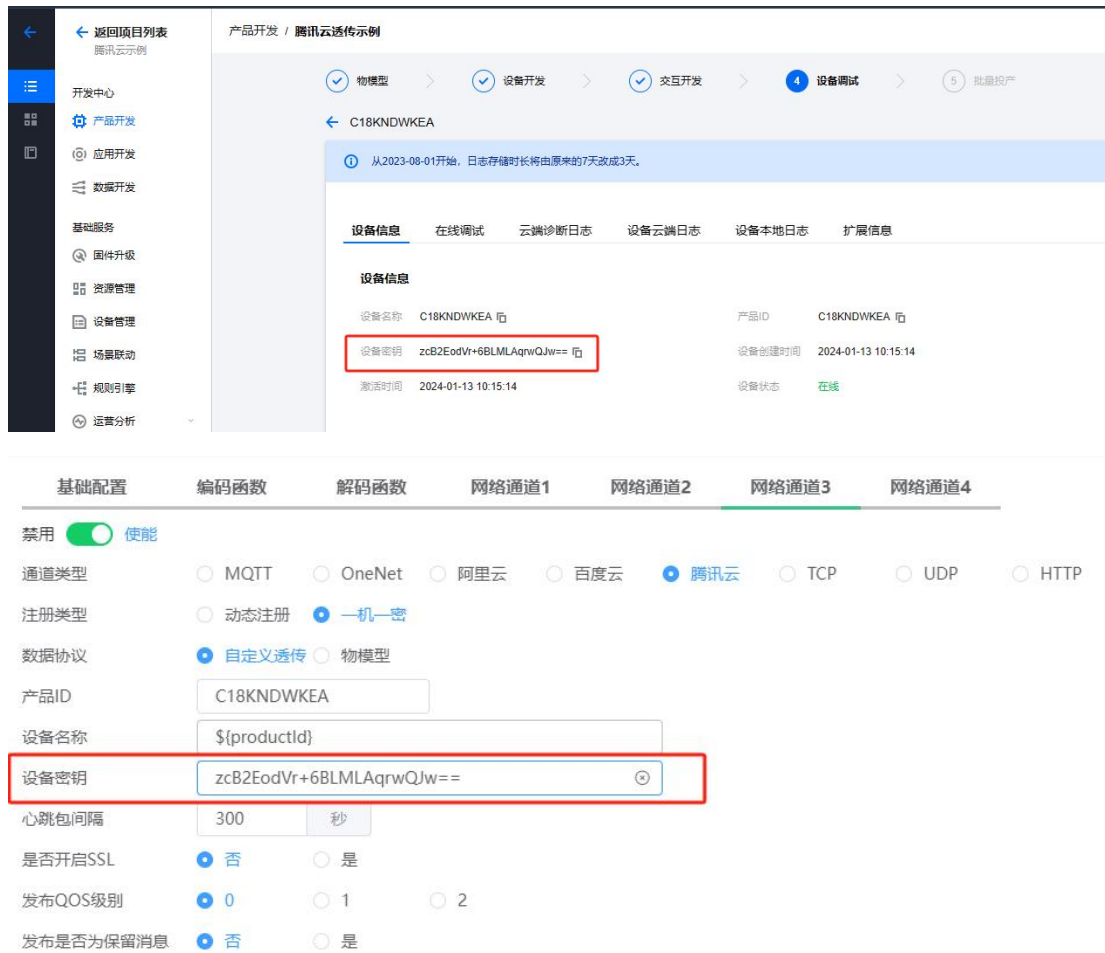
#### ➤ 动态注册

此处设备名称使用通配符  $\${productId}$ ，实际设备名将被替换为产品 ID，本例中为 **C18KNDWKEA**。



### ➤ 一机一密

使用一机一密时，需要先在腾讯云对应产品下先创建设备，配置参数中设备名称要与腾讯云平台创建的设备名称相同。创建完成后点击设备名称进入设备信息页，可获取**设备密钥**。



### 2) 腾讯云自定义透传

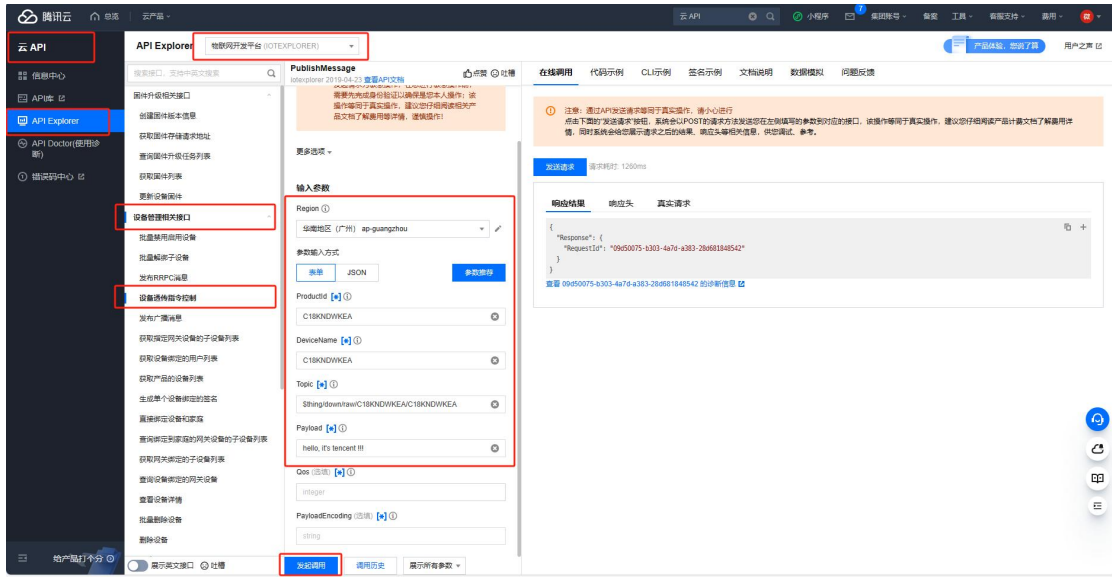
使用自定义透传时首先需要创建产品，产品创建如上所述，注意在创建产品时数据协议栏选择自定义透传，物联网管理平台配置参数时也需要选择为自定义透传。



### ➤ 数据下发

本示例中使用腾讯云 **云 API** 套件实现透传指令下发，通过腾讯云搜索栏搜索云 API，进入云 API 产品，从 API Explorer>物联网开发平台>设备管理相关接口>设备透传指令控制，进入透传数据下发页面，Region 填写广州，参数输入方式表单，ProductId 为对应的产品 ID，Devicename 对应的设备名称，Topic 格式为 `$thing/down/raw/${productId}/${deviceName}`，其

中 $\{productId\}$ 需要替换为数据的产品 ID,  $\{deviceName\}$ 需要替换为实际的设备名称, 本例中, 产品 ID 为 C18KNDWKEA, 设备名称也为 C18KNDWKEA, Payload 为要下发的数据。填写完毕后, 点击下方发起调用。从串口调试助手页面, 可看到接收到数据 hello, it's tencent !!!



[10:36:50.870]收←◆hello, it's tencent !!!

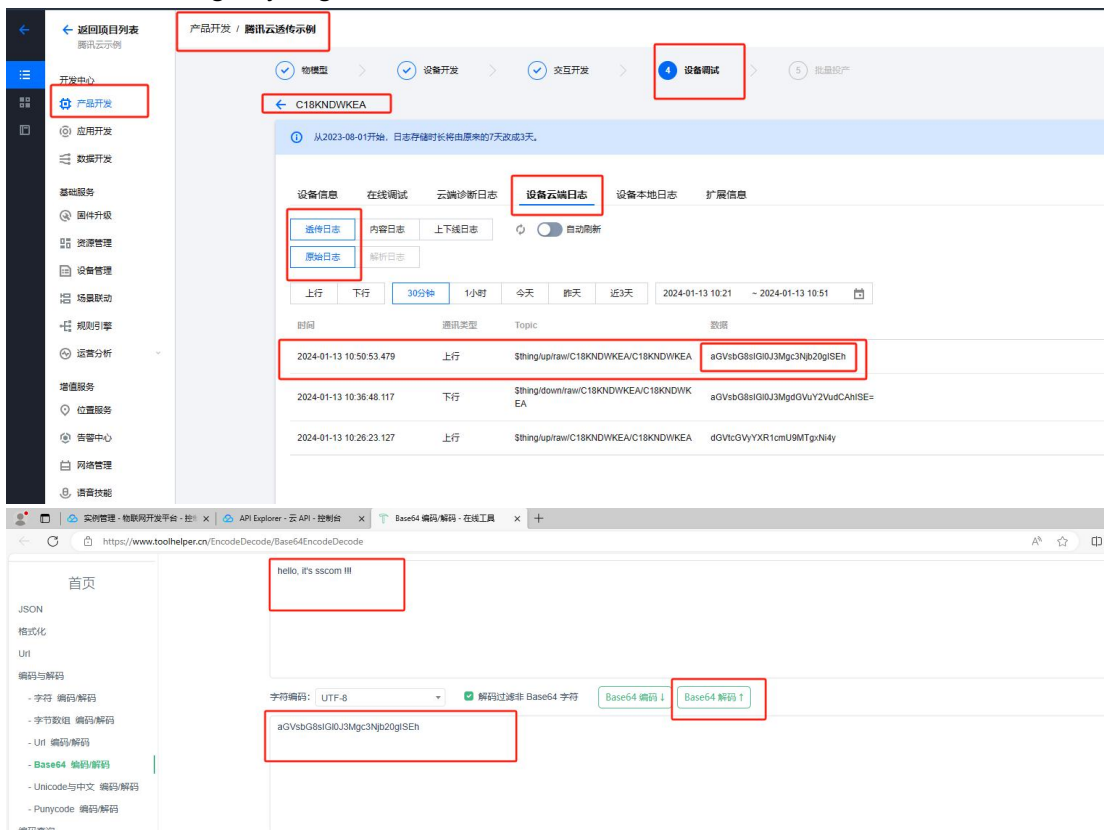


- 数据上传  
通过串口助手发送数据 hello, it's sscom !!!。

[11:26:03.154]发→hello, it's sscom !!!



通过腾讯云物联网平台对应产品对应设备下>设备云端日志，选择透传日志，原始日志通讯类型为上行表示上传的数据，此处显示数据为 base64 加密后的数据（此处为 aGVsbG8sIGl0J3Mgc3Njb20glSEh），经过 base64 解密后可得到原始数据 hello, it's sscom !!!。



### ➤ 编码函数

透传时编码函数固定为以下内容，若用户没有 LUA 语言编程基础不建议修改编码/解码函数。可在第 5 节查看。

基础配置 编码函数 解码函数 网络通道1 网络通道2 网络通道3 网络通道4

```
return function(ch, buf, len)
    return buf, len
end
```

➤ 解码函数

透传时解码函数固定为以下内容，若用户没有 LUA 语言编程基础不建议修改编码/解码函数。可在第 5 节查看。

参数配置(test3)

基础配置 编码函数 解码函数 网络通道1 网络通道2 网络通道3 网络通道4

```
return function(ch, buf, len, p1, p2, p3, p4, p5)
    return buf, len
end
```

3) 腾讯云物模型

使用物模型时首先需要创建产品，产品创建如上所述，注意在创建产品时数据协议栏选择物模型，物联网管理平台配置参数时也需要选择为物模型。

**新建产品**

产品名称 \* 腾讯云物模型示例  
支持中文、英文、数字、下划线、空格（非首尾字符）、中英文括号、-、@、\、/的组合，最多不超过40个字符

产品品类

设备类型

通信方式 \* 2G/3G/4G  
请根据业务场景正确选择产品的通信方式，否则会影响后续产品开发

认证方式

数据协议   ⓘ

描述 选填  
最多不超过80个字符

基础配置	编码函数	解码函数	网络通道1	网络通道2	网络通道3	网络通道4
禁用 <input type="checkbox"/> 启用 <input checked="" type="checkbox"/>						
通道类型	<input type="radio"/> MQTT <input type="radio"/> OneNet <input type="radio"/> 阿里云 <input type="radio"/> 百度云 <input checked="" type="radio"/> 腾讯云 <input type="radio"/> TCP <input type="radio"/> UDP <input type="radio"/> HTTP					
注册类型	<input checked="" type="radio"/> 动态注册 <input type="radio"/> 一机一密					
数据协议	<input type="radio"/> 自定义透传 <input checked="" type="radio"/> 物模型					
产品ID	OKM8GTCO34					
ProductSecret	gETdj6yYrNgZBrwUrcollqAO					
设备名称	\${productid}					
心跳包间隔	300 秒					
是否开启SSL	<input checked="" type="radio"/> 否 <input type="radio"/> 是					
发布QOS级别	<input checked="" type="radio"/> 0 <input type="radio"/> 1 <input type="radio"/> 2					
发布是否为保留消息	<input checked="" type="radio"/> 否 <input type="radio"/> 是					

使用物模型时需要在腾讯云对应产品下创建相关物模型，本例中创建两个物模型，温度（temperature）和温度系数（temperature\_ratio）。温度用于接收数据，温度系数用于下发数据。

#### 新增自定义功能

⚠ 注意：添加自定义功能将影响设备通过语音、中控面板控制，建议添加标准功能，若已有标准功能无法满足，您可提交 [意向单](#) 申请新增标准功能。

功能类型	<input checked="" type="button"/> 属性 <input type="button"/> 事件 <input type="button"/> 行为		
功能名称 *	温度		
	支持中文、英文、数字、下划线的组合，最多不超过20个字符		
标识符 *	temperature		
	第一个字符不能是数字，支持英文、数字、下划线的组合，最多不超过32个字符		
数据类型	<input type="button"/> 布尔型 <input type="button"/> 整型 <input type="button"/> 字符串 <input checked="" type="button"/> 浮点型 <input type="button"/> 枚举整型 <input type="button"/> 枚举字符串 <input type="button"/> 时间型 <input type="button"/> 结构体 <input type="button"/> 数组		
读写类型	<input type="button"/> 读写 <input checked="" type="button"/> 只读 ⓘ		
数值范围	- -100.000 + - - 100.000 +		
初始值	- 0.000 +		
步长	- 0.100 + ⓘ		
单位	摄氏度		
	0-12个字符		



功能类型	<input checked="" type="button" value="属性"/> <input type="button" value="事件"/> <input type="button" value="行为"/>
功能名称*	<input type="text" value="温度系数"/> <small>支持中文、英文、数字、下划线的组合，最多不超过20个字符</small>
标识符*	<input type="text" value="temperature_ratio"/> <small>第一个字符不能是数字，支持英文、数字、下划线的组合，最多不超过32个字符</small>
数据类型	<input type="button" value="布尔型"/> <input checked="" type="button" value="整型"/> <input type="button" value="字符串"/> <input type="button" value="浮点型"/> <input type="button" value="枚举整型"/> <input type="button" value="枚举字符串"/> <input type="button" value="时间型"/> <input type="button" value="结构体"/> <input type="button" value="数组"/>
读写类型	<input checked="" type="button" value="读写"/> <input type="button" value="只读"/> ⓘ
数值范围	- <input type="text" value="0"/> + - <input type="text" value="100"/> +
初始值	- <input type="text" value="0"/> +
步长	- <input type="text" value="1"/> + ⓘ

➤ 解码函数示例

此处 `ch == 3`，是因为本例中在物联网管理平台将腾讯云配置为网络通道 3

源码：

```
return function(ch, buf, len, p1, p2, p3, p4, p5)
    if ch == 3 then
        local json_obj, result, err = json.decode(buf)
        if result ~= 1 or json_obj == nil then
            return
        end
        if json_obj['method'] ~= "control" or json_obj['params'] == nil or
        json_obj['params']['temperature_ratio'] == nil then
            return
        end
        return json_obj['params']['temperature_ratio'],
        string.len(json_obj['params']['temperature_ratio'])
    end
end
```

➤ 编码函数示例

此处 `ch == 3`，是因为本例中在物联网管理平台将腾讯云配置为网络通道 3

源码：

```
return function(ch, buf, len)
    if ch == 3 then
        if string.find(buf, '=') ~= nil then
            local offset = string.find(buf, '=')
            local temperature_value = string.sub(buf, tonumber(offset) + 1)
            buf = string.format('{"method":"report","clientToken":
            "%d","params":{"temperature":%s}}', math.random(1, 10000),
            temperature_value)
        end
    end
end
```



```

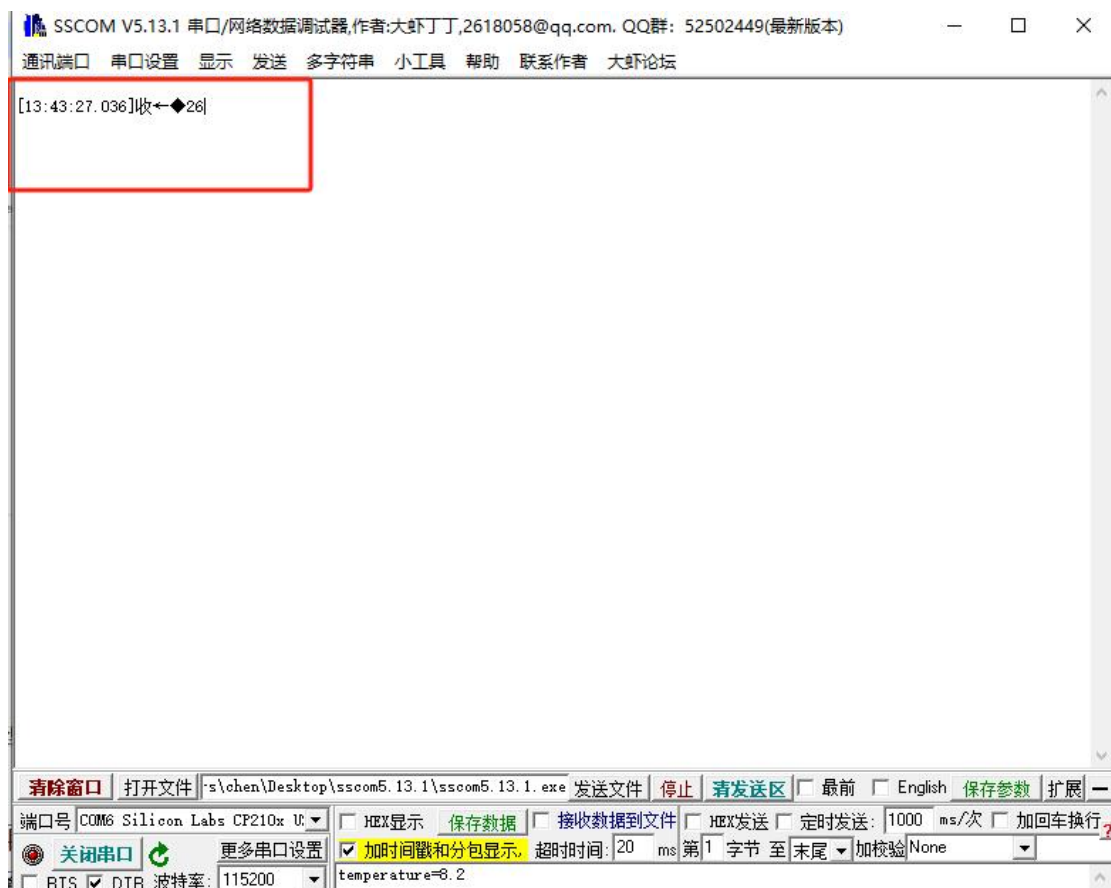
return buf, #buf
end
end
end
end
    
```

### ➤ 数据下发

在对应设备下>在线调试>属性调试中，勾选温度系数栏输入数据 26，点击发送，则数据下发完成。



通过串口调试助手查看数据，得到解析函数解析后的数据 26。



➤ 数据上传

通过串口助手发送数据 `temperature=8.2`，编码函数首先解析数据得到实际温度值 8.2，再将其打包为符合腾讯云物模型要求的数据包：

`{"method": "report", "clientToken": "6184", "params": {"temperature": 8.2}}`，最后 DTU 设备将该数据包上传到腾讯云，上报内容可通过对应设备下>在线调试>属性调试中，温度条目查看上报数据。



← 返回项目列表  
腾讯云示例

产品开发 / 腾讯云物模型示例

物模型 > 设备开发 > 交互开发 > 4 设备调试 > 5 批量投产

← OKM8GTC034

从2023-08-01开始, 日志存储时长将由原来的7天改成3天。

设备信息 **在线调试** 云端诊断日志 设备云端日志 设备本地日志 扩展信息

下发指令

属性调试 行为调用

<input type="checkbox"/> 功能名称/标识符	期望值	实时数据
<input type="checkbox"/> 地区区域码(lac)	(只读) 数值范围: 0-32,初始值: 0,步长: 1,单位:	-
<input type="checkbox"/> 基站码(cid)	(只读) 数值范围: 0-32,初始值: 0,步长: 1,单位:	-
<input type="checkbox"/> 移动网号(mnc)	(只读) 数值范围: 0-5,初始值: 0,步长: 1,单位:	-
<input type="checkbox"/> 移动国家号(mcc)	(只读) 数值范围: 0-5,初始值: 0,步长: 1,单位:	-
<input type="checkbox"/> 网络制式(networkType)	(只读) 1 - GSM,2 - CDMA,3 - WCDMA,4 - TD_WCDMA,5 - LTE	-
<input type="checkbox"/> 温度(temperature)	(只读) 数值范围: -100-100,初始值: 0,步长: 0.1,单位: 摄氏度	8.2

发送 重置

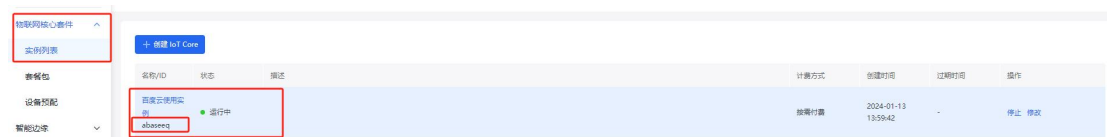
### 6.3.4 百度云

登录百度云，从搜索栏中输入物联网核心套件搜索，进入物联网核心套件。

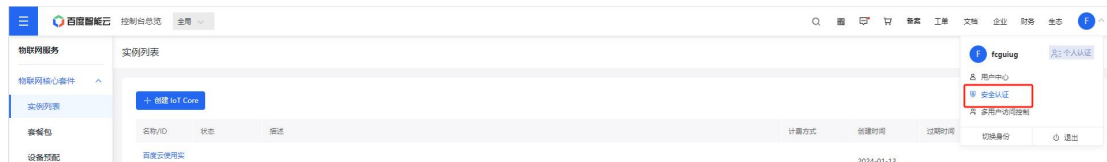
进入物联网核心套件后点击实例列表>创建 IoT Core，根据需要选择付费方式、配置信息。



创建完成后可在实例列表中查看实例 ID，图中 abaseeq 为 IoTCoreId。



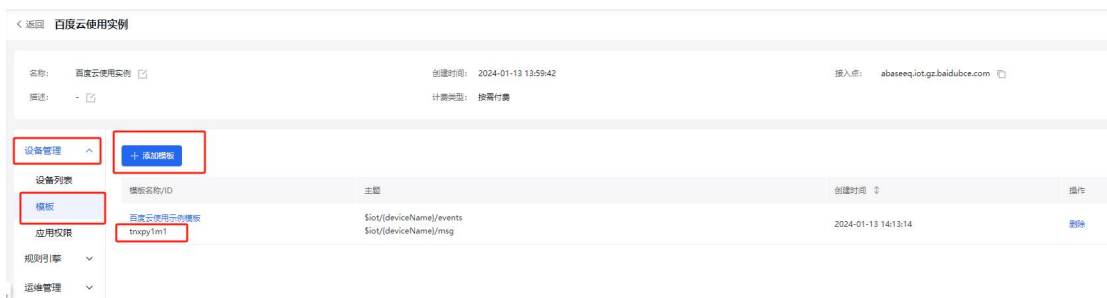
将鼠标置于右上角头像处，点击安全认证，进入安全认证界面。



创建 Access Key，获取自动注册时需要使用的 Access Key 和 Secret Key。



创建模板，进入创建的实例，从设备管理>模板>添加模板，创建模板，此处 tnxpy1m1 为该模板 ID。



#### 1) 配置平台参数配置

通道类型选择百度云。

#### ➤ 自动注册

此处设备名称使用通配符 \${imei}，实际设备名将被替换为设备的 imei 869020066533371。

基础配置	编码函数	解码函数	网络通道1	网络通道2	网络通道3	网络通道4
禁用 <input type="checkbox"/> 使能 <input checked="" type="checkbox"/>						
通道类型	<input type="radio"/> MQTT	<input type="radio"/> OneNet	<input type="radio"/> 阿里云	<input checked="" type="radio"/> 百度云	<input type="radio"/> 腾讯云	<input type="radio"/> TCP <input type="radio"/> UDP <input type="radio"/> HTTP
注册类型	<input checked="" type="radio"/> 自动注册	<input type="radio"/> 一机一密				
Accesskey	<input type="text" value="ALTAKomezwRDRxkXnz486XTsCa"/>					
Secret Key	<input type="text" value="6d88e6934cf14d2fa638d90cbd0ee4c8"/>					
地域(regionId)	<input type="text" value="广州(gz)"/>					
IoTCoreId	<input type="text" value="abaseeq"/>					
模板ID	<input type="text" value="tnxpy1m1"/>					
DeviceKey	<input type="text" value="{imei}"/>					
心跳包间隔	<input type="text" value="300"/>	<input type="text" value="秒"/>				
是否开启SSL	<input checked="" type="radio"/> 否	<input type="radio"/> 是				
发布QOS级别	<input checked="" type="radio"/> 0	<input type="radio"/> 1	<input type="radio"/> 2			
发布是否为保留消息	<input checked="" type="radio"/> 否	<input type="radio"/> 是				

### ➤ 一机一密

使用一机一密时，需要先在百度云对应实例下新增设备，配置参数中设备名称要与百度云平台创建的设备名称相同。设备创建完成后点击设备名称进入设备信息页，可获取**设备密钥**。将对应参数填入物联网管理平台的配置参数。

< 返回 百度云使用实例

名称: 百度云使用实例	创建时间: 2024-01-13 13:59:42	接入点: abaseeq.iotgz
描述: -	计费类型: 按量付费	

设备管理

设备列表

模板

应用权限

规则引擎

运维管理

+ 新增设备

设备名称	认证方式	描述	创建时间
869020066533371	密钥认证		2024-01-13 14:18:07

+ 加载更多

< 返回 **新增设备**

### 配置信息

\* 名称:

\* 认证方式:  证书认证  密钥认证

描述:

\* 模板:

标签: [+ 添加标签](#)

< 返回 **设备869020066533371详情**

[设备信息](#) [设备影子](#) [模拟设备](#)

#### 基础信息

名称: 869020066533371

创建时间: 2024-01-13 14:18:07

描述:

模板: 百度云使用示例模板

#### 连接信息

IoTCoreId:

DeviceKey:

DeviceSecret:  [重置](#)

服务鉴权证: [查看详情](#)

#### 基础配置

#### 编码函数

#### 解码函数

#### 网络通道1

#### 网络通道2

#### 网络通道3

#### 网络通道4

禁用  使能

通道类型:  MQTT  OneNet  阿里云  百度云  腾讯云  TCP  UDP  HTTP

注册类型:  自动注册  一机一密

地域(regionId):

IoTCoreId:

DeviceKey:

DeviceSecret:

心跳包间隔:  秒

是否开启SSL:  否  是

发布QOS级别:  0  1  2

发布是否为保留消息:  否  是

## 2) 百度云数据转发使用示例

本示例中通过百度云的数据转发功能，将设备上传数据通过百度云转发重新发送给备。

通过创建的模板可获取  $\$iot/\{deviceName\}/events$ (客户端向云端发送消息)和  $\$iot/\{deviceName\}/msg$ (云端向客户端发送消息)两个 Topic。

< 返回 百度云使用实例

名称: 百度云使用实例  创建时间: 2024-01-13 13:59:42 接入点: abaseeq.iot.gz.be

描述: -  计费类型: 按需付费

设备管理  + 添加模板

模板名称/ID	主题	创建时间
百度云使用示例模板 tnxpy1m1	$\$iot/\{deviceName\}/events$ $\$iot/\{deviceName\}/msg$	2024-01-13 14:13:14

设备列表

模板

应用权限

规则引擎

运维管理

在规则引擎>规则引擎列表中，创建规则。

< 返回 百度云使用实例

名称: 百度云使用实例  创建时间: 2024-01-13 13:59:42 接入点: abaseeq.iot.gz.be

描述: -  计费类型: 按需付费

设备管理  + 创建规则

设备列表

模板

应用权限

规则引擎

自定义目的地

名称/ID	状态	描述	创建时间	操作
百度云使用示例规则 3h5armedcu1m543jcm1208gm75sr	已禁用	-	2024-01-13 14:13:57	解除禁用 启用

创建完成后需要先编辑调试规则。

数据输入中，数据来源点击下方获取 topic 模板

< 返回 编辑规则

基本信息

规则名称: 百度云使用示例规则

规则描述:

IoT Core: abaseeq

数据输入

\* 数据来源:

选择上述中创建的模板 topic 选择  $\$iot/\{deviceName\}/events$ ，勾选全部设备，点击载入。

topic模板
✕

---

\* 设备模板: 百度云使用示例模板 ▾

\* 可选topic:  \$Iot/{deviceName}/events  
 \$Iot/{deviceName}/msg

\* 选定设备: 请输入设备名称 🔍

全部设备(+)  
 869020066533371  
 SYS\_LOGGER

+ 加载更多

---

取消
载入

添加数据目的地，选择 MQTT 主题，实例选择所创建的实例，Topic 格式为 \$Iot/\${devicename}/msg，其中 \${devicename} 表示当前设备名称，此处为 869020066533371。点击确定。

数据目的地
✕

---

\* 数据目的地: + 添加目的地

类型

MQTT主题

---

错误操作目的地

添加数据目的地
✕

---

\* 数据目的地: MQTT主题 ▾

\* IoT Core: 百度云使用实例(abaseeq) ▾

\* Topic ID: \$Iot869020066533371/msg

\* Qos: 1 ▾

取消
确定

保存规则后从规则列表中启用该规则。

### ➤ 编码函数

透传时编码函数固定为以下内容，若用户没有 LUA 语言编程基础不建议修改编码/解码函数。

基础配置
编码函数
解码函数
网络通道1
网络通道2
网络通道3
网络通道4

```
return function(ch, buf, len)
  return buf, len
end
```

### ➤ 解码函数



透传时解码函数固定为以下内容，若用户没有 LUA 语言编程基础不建议修改编码/解码函数。

```

参数配置(test3)
    基础配置    编码函数    解码函数    网络通道1    网络通道2    网络通道3    网络通道4
    return function(ch, buf, len, p1, p2, p3, p4, p5)
        return buf, len
    end
    
```

### ➤ 数据传输

通过串口调试助手发送 **hello,it's me !!!**，可看到串口助手发送的数据通过百度云又重新发回串口助手。本示例只作为简单演示功能，用户可通过百度云规则引擎功能实现更多功能。



## 6.3.5 MQTT 透传

### ➤ 平台参数配置

用户名、密码、客户端 ID 为 MQTT 连接三要素，对应填入即可，填写 MQTT 对应的服务器 ID 及端口号，发布主题、订阅主题用户可自定义，发布主题和订阅主题可使用通配符  $\${imei}$ 、 $\${clientId}$ ， $\${imei}$  将被替换为 DTU 设备的 IMEI， $\${clientId}$  将被替换客户端 ID。

下列发布主题为设备发布数据时使用的 Topic，订阅主题为设备接收数据 Topic。

禁用  使能

通道类型  MQTT  OneNet  阿里云  百度云  腾讯云  TCP  UDP  HTTP

用户名

密码

客户端ID

服务器IP或域名

服务器端口号

心跳包间隔  秒

发布主题

订阅主题

是否开启SSL  否  是

发布QOS级别  0  1  2

发布是否为保留消息  否  是

### ➤ 编码函数

透传时编码函数固定为以下内容，若用户没有 LUA 语言编程基础不建议修改编码/解码函数。

基础配置 编码函数 解码函数 网络通道1 网络通道2 网络通道3 网络通道4

```
return function(ch, buf, len)
    return buf, len
end
```

### ➤ 解码函数

透传时解码函数固定为以下内容，若用户没有 LUA 语言编程基础不建议修改编码/解码函数。

参数配置(test3)

基础配置 编码函数 解码函数 网络通道1 网络通道2 网络通道3 网络通道4

```
return function(ch, buf, len, p1, p2, p3, p4, p5)
    return buf, len
end
```

## 6.3.6 TCP 透传

### ➤ 平台参数配置

填写对应 IP 及端口号，若选择使能注册包，则在首次连接上 TCP 平台后将发送一个注册报文。若选择使能心跳包，则每隔一个心跳包间隔将发送一次心跳包

参数配置(TEST1)

基础配置 编码函数 解码函数 网络通道1 网络通道2 网络通道3 网络通道4

禁用  使能

通道类型  MQTT  OneNet  阿里云  百度云  腾讯云  TCP  UDP  HTTP

服务器IP或域名

服务器端口号

使能注册包  禁用  使能

使能心跳包  禁用  使能

是否开启SSL  否  是

### ➤ 编码函数

透传时编码函数固定为以下内容，若用户没有 LUA 语言编程基础不建议修改编码/解码函数。源码在第 5 节中查看。

基础配置 编码函数 解码函数 网络通道1 网络通道2 网络通道3 网络通道4

```
return function(ch, buf, len)
    return buf, len
end
```

### ➤ 解码函数

透传时解码函数固定为以下内容，若用户没有 LUA 语言编程基础不建议修改编码/解码函数。源码在第 5 节中查看。

参数配置(test3)

基础配置 编码函数 解码函数 网络通道1 网络通道2 网络通道3 网络通道4

```
return function(ch, buf, len, p1, p2, p3, p4, p5)
    return buf, len
end
```

## 6.3.7 UDP 透传

- 平台参数配置  
填写服务器 IP 端口号即可

### 参数配置(TEST1)

基础配置 编码函数 解码函数 网络通道1 网络通道2 网络通道3 网络通道4

禁用  使能

通道类型  MQTT  OneNet  阿里云  百度云  腾讯云  TCP  UDP  HTTP

服务器IP或域名

服务器端口号

- 编码函数  
透传时编码函数固定为以下内容，若用户没有 LUA 语言编程基础不建议修改编码/解码函数。源码在第 5 节中查看。

```
基础配置 编码函数 解码函数 网络通道1 网络通道2 网络通道3 网络通道4
```

```
return function(ch, buf, len)
    return buf, len
end
```

- 解码函数  
透传时解码函数固定为以下内容，若用户没有 LUA 语言编程基础不建议修改编码/解码函数。源码在第 5 节中查看。

```
参数配置(test3)
```

```
基础配置 编码函数 解码函数 网络通道1 网络通道2 网络通道3 网络通道4
```

```
return function(ch, buf, len, p1, p2, p3, p4, p5)
    return buf, len
end
```

## 6.3.8 HTTP 透传

### ➤ 平台参数配置

填写对应 Url，Http 操作模式固定为 POST，数据请求类型固定为 JSON，超时时间为设备 POST 数据后最长的等待回复时间，超时时间为 0 时表示永久等待，Basic 为请求 Header 中的 Authorization（非必填项）。

### 参数配置(TEST1)

基础配置
编码函数
解码函数
网络通道1
网络通道2
网络通道3
网络通道4

禁用  使能

通道类型  MQTT  OneNet  阿里云  百度云  腾讯云  TCP  UDP  HTTP

Url

Method POST

请求类型 JSON

超时时间  秒

Basic

### ➤ 编码函数

透传时编码函数固定为以下内容，若用户没有 LUA 语言编程基础不建议修改编码/解码函数。源码在第 5 节中查看。

基础配置
编码函数
解码函数
网络通道1
网络通道2
网络通道3
网络通道4

```
return function(ch, buf, len)
    return buf, len
end
```

### ➤ 解码函数

透传时解码函数固定为以下内容，若用户没有 LUA 语言编程基础不建议修改编码/解码函数。源码在第 5 节中查看。

### 参数配置(test3)

基础配置
编码函数
解码函数
网络通道1
网络通道2
网络通道3
网络通道4

```
return function(ch, buf, len, p1, p2, p3, p4, p5)
    return buf, len
end
```

## 7. 应用实例

本节介绍如何使用 DTU4G 模块获取 B-TH-RS-30 温湿度传感器数据，并且将采集的数据上传到腾讯云和阿里云，并且可通过腾讯云修改传感器设备地址。

主要包含如下功能点：

- 使用轮询指令，周期为 60 秒，定时采集传感器温湿度；
- 网络通道 1 配置为腾讯云并且数据协议使用物模型，DTU 模块将传感器数据解析并打包为符合腾讯云物模型的 JSON 格式数据；
- 网络通道 2 配置为阿里云数据协议使用自定义透传，DTU 模块直接将传感器实际回复数据上传至阿里云；
- 通过腾讯云修改传感器设备地址

B-TH-RS-30 温湿度传感器默认设备地址为 01，故轮询命令为 (HEX)：**01 03 00 00 00 02 c4 0b**，该命令为采集温湿度命令。通信成功后其返回值为 HEX 类型的 9Byte 数据：设备地址+功能码+字节数+温度数据+湿度数据+CRC(L)+CRC(H)。

### 2). 读多寄存器命令 读温湿度

总序号 (byte)	1	2	3	4	5	6	7	8
各字节说明	设备地址	功能码	寄存器起始地址		读取寄存器数量		CRC 校验 L H	
值 (hex)	01	03	00	00	00	02	C4	0B

#### 返回

总序号 (byte)	1	2	3	4	5	6	7	8	9
各字节说明	设备地址	功能码	字节数	温度数据		湿度数据		CRC 校验 L H	
值 (hex)	01	03	04	00	E9	02	6F	6A	8B

温度：00 E9 除 10 = 233/10 = 23.3℃

湿度：02 6F 除 10 = 623/10 = 62.3 %RH

修改设备地址命令码如下：

3).写单寄存器命令

修改设备地址

总序号 (byte)	1	2	3	4	5	6	7	8
各字节说明	设备地址	功能码	寄存器地址		写地址数据		CRC 校验	
							L	H
值 (hex)	01	06	00	64	00	02	49	D4

返回

总序号 (byte)	1	2	3	4	5	6	7	8
各字节说明	设备地址	功能码	寄存器地址		写地址数据		CRC 校验	
							L	H
值 (hex)	01	06	00	64	00	02	49	D4

修改设备地址为 02

## 7.1 创建腾讯云产品（物模型）

产品创建过程参考 5.3.3 章节，注意数据协议选择物模型。

创建好产品后，定义 3 个物模型功能点，分别为温度、湿度以及设备地址。

温度物模型功能点：功能类型选择属性，标识符为 temperature，数据类型为浮点数，读写类型选择只读，数据范围-100~+100。

功能类型  属性  事件  行为

功能名称 \* 温度  
支持中文、英文、数字、下划线的组合，最多不超过20个字符

标识符 \* temperature  
第一个字符不能是数字，支持英文、数字、下划线的组合，最多不超过32个字符

数据类型  布尔型  整型  字符串  浮点型  枚举整型  枚举字符串  时间型  结构体  数组

读写类型  读写  只读 ⓘ

数值范围

初始值

步长    ⓘ

单位 摄氏度  
0-12个字符

湿度物模型功能点：功能类型选择属性，标识符为 humidity，数据类型为浮点数，读写类型选择只读，数据范围 0~+100。



功能类型	<input checked="" type="radio"/> 属性	<input type="radio"/> 事件	<input type="radio"/> 行为
功能名称 *	湿度		
	支持中文、英文、数字、下划线的组合，最多不超过20个字符		
标识符 *	humidity		
	第一个字符不能是数字，支持英文、数字、下划线的组合，最多不超过32个字符		
数据类型	<input type="radio"/> 布尔型	<input type="radio"/> 整数型	<input checked="" type="radio"/> 浮点型
	<input type="radio"/> 字符串	<input type="radio"/> 枚举整型	<input type="radio"/> 枚举字符串
	<input type="radio"/> 时间型	<input type="radio"/> 结构体	<input type="radio"/> 数组
读写类型	<input type="radio"/> 读写	<input checked="" type="radio"/> 只读	<input type="radio"/> ①
数值范围	-	0.000	+
	-	100.000	+
初始值	-	0.000	+
步长	-	0.100	+
			①
单位	%RH		
	0-12个字符		

设备地址功能点：功能类型选择属性，标识符为 device\_addr，数据类型为整数型，读写类型选择读写，数据范围 0~+255。

功能类型	<input checked="" type="radio"/> 属性	<input type="radio"/> 事件	<input type="radio"/> 行为
功能名称 *	设备地址		
	支持中文、英文、数字、下划线的组合，最多不超过20个字符		
标识符 *	device_addr		
	第一个字符不能是数字，支持英文、数字、下划线的组合，最多不超过32个字符		
数据类型	<input type="radio"/> 布尔型	<input checked="" type="radio"/> 整数型	<input type="radio"/> 字符串
	<input type="radio"/> 浮点型	<input type="radio"/> 枚举整型	<input type="radio"/> 枚举字符串
	<input type="radio"/> 时间型	<input type="radio"/> 结构体	<input type="radio"/> 数组
读写类型	<input checked="" type="radio"/> 读写	<input type="radio"/> 只读	<input type="radio"/> ①
数值范围	-	0	+
	-	255	+
初始值	-	1	+
步长	-	1	+
			①
单位			
	0-12个字符		

## 7.2 创建阿里云产品（自定义透传）

产品创建过程参考 5.3.2 章节，注意数据格式为自定义透传。

## 7.3 重新编写编码与解码函数

### ➤ 编码函数

编码函数处理温湿度传感器通过 RS485 发送给 DTU 设备的数据，先判断其接收数据长



度是否为 9 个，然后判断其功能码与数据字节数是否正确，最后判断 CRC 校验码是否正确，若满足以上条件则认为数据合法，因为网络通道 1 对应腾讯云物模型所以将实际温度和实际湿度打包为满足腾讯云物模型的 JSON 格式。网络通道 2 对应阿里云，该通道使用自定义透传，直接将原始数据上传至阿里云平台。在校验 CRC 时使用到内部函数 `tools.crc16`，其用法在 5.3 章说明。

```
return function(ch, buf, len)
  if len ~= 9 then return end
  if string.byte(buf, 2) ~= 3 or string.byte(buf, 3) ~= 4 then return end
  local buf_data = string.sub(buf, 1, 7)
  local crc = tools.crc16(buf_data)
  if (crc & 0xFF) ~= string.byte(buf, 8) or ((crc >> 8) & 0xFF) ~=
string.byte(buf, 9) then return end
  if ch == 1 then
    buf =
string.format('{ "method": "report", "clientToken": "%d", "params": { "tempera
ture": %s, "humidity": %s } }', math.random(1, 10000), ((string.byte(buf, 4) <<
8) | string.byte(buf, 5)) / 10, ((string.byte(buf, 6) << 8) | string.byte(buf,
7)) / 10)
    return buf, #buf
  else return buf, len end
end
```

#### ➤ 解码函数

从腾讯云下发的数据中解析出设备地址字段，并将其打包为传感器修改地址命令。在校验 CRC 时使用到内部函数 `tools.crc16`，其用法在 5.3 章说明。

```
return function(ch, buf, len)
  if ch == 1 then
    local json_obj, result, err = json.decode(buf)
    if result ~= 1 or json_obj == nil then return end
    if json_obj['params'] == nil or json_obj['params']['device_addr'] ==
nil then return end
    local crc = tools.crc16(string.pack("BBBBBB", 0x01, 0x06, 0x00, 0x64,
0x00, tonumber(json_obj['params']['device_addr'])))
    return string.pack("BBBBBBBB", 0x01, 0x06, 0x00, 0x64, 0x00,
tonumber(json_obj['params']['device_addr']), (crc & 0xFF), ((crc >> 8) &
0xFF)), 8
  else return buf, len end
end
```

## 7.4 物联网管理平台配置设备参数

基础配置如下，编码解码函数如 7.3 中所写。

基础配置	编码函数	解码函数	网络通道1	网络通道2	网络通道3	网络通道4
<b>串口配置</b>						
波特率(bps)	9600					
数据位	<input checked="" type="radio"/> 8位 <input type="radio"/> 7位					
校验位	<input checked="" type="radio"/> 无校验 <input type="radio"/> 奇校验 <input type="radio"/> 偶校验					
停止位	<input checked="" type="radio"/> 1位 <input type="radio"/> 2位					
通讯方式	<input type="radio"/> TTL <input type="radio"/> 232 <input checked="" type="radio"/> 485					
<b>采集方式</b>						
采集方式	<input type="radio"/> 主动上报 <input checked="" type="radio"/> 轮询					
轮询周期	10 秒					
轮询指令	<input type="text" value="010300000002c40b"/> (+)					

网络通道 1 配置为腾讯云，使用数据协议为物模型。

基础配置	编码函数	解码函数	网络通道1	网络通道2	网络通道3	网络通道4
禁用	<input checked="" type="checkbox"/> 禁用 <input type="checkbox"/> 使能					
通道类型	<input type="radio"/> MQTT <input type="radio"/> OneNet <input type="radio"/> 阿里云 <input type="radio"/> 百度云 <input checked="" type="radio"/> 腾讯云 <input type="radio"/> TCP <input type="radio"/> UDP <input type="radio"/> HTTP					
注册类型	<input checked="" type="radio"/> 动态注册 <input type="radio"/> 一机一密					
数据协议	<input type="radio"/> 自定义透传 <input checked="" type="radio"/> 物模型					
产品ID	<input type="text" value="OKM8GTCO34"/>					
ProductSecret	<input type="text" value="gETdj6yyrNgZBrwUrcollqAO"/>					
设备名称	<input type="text" value="{imei}"/>					
心跳包间隔	300 秒					
是否开启SSL	<input checked="" type="radio"/> 否 <input type="radio"/> 是					
发布QOS级别	<input checked="" type="radio"/> 0 <input type="radio"/> 1 <input type="radio"/> 2					
发布是否为保留消息	<input checked="" type="radio"/> 否 <input type="radio"/> 是					

网络通道 2 配置为阿里云，使用的数据协议为自定义透传

基础配置	编码函数	解码函数	网络通道1	网络通道2	网络通道3	网络通道4
禁用 <input type="checkbox"/>	使能 <input checked="" type="checkbox"/>					
通道类型	<input type="radio"/> MQTT <input type="radio"/> OneNet <input checked="" type="radio"/> 阿里云 <input type="radio"/> 百度云 <input type="radio"/> 腾讯云 <input type="radio"/> TCP <input type="radio"/> UDP <input type="radio"/> HTTP					
注册类型	<input checked="" type="radio"/> 自动注册 <input type="radio"/> 一型一密 <input type="radio"/> 一机一密					
数据协议	<input checked="" type="radio"/> 自定义透传 <input type="radio"/> 物模型					
实例ID	<input type="text" value="iot-06z00ebjj28ajii"/>					
地域(regionId)	<input type="text" value="华东2(cn-shanghai)"/>					
ProductKey	<input type="text" value="k0lj53XSkob"/>					
AccessKey ID	<input type="text" value="LTAI5tCZzD2UANmKgTpPaSt"/>					
AccessKey Secret	<input type="text" value="4uwptUmzh56Ts4yClcibytsjCs2Hil"/>					
DeviceName	<input type="text" value="{imei}"/>					
心跳包间隔	<input type="text" value="300"/> 秒					
是否开启SSL	<input checked="" type="radio"/> 否 <input type="radio"/> 是					
发布QOS级别	<input checked="" type="radio"/> 0 <input type="radio"/> 1 <input type="radio"/> 2					
发布是否为保留消息	<input checked="" type="radio"/> 否 <input type="radio"/> 是					

## 7.5 数据上报

在腾讯云设备调试界面观察温度、湿度条目，可在实时数据栏中看到传感器采集温度与湿度。

产品开发 / 腾讯云物模型示例

物模型 > 设备开发 > 交互开发 > 4 设备调试 > 5 批量投产

869020066533371

从2023-08-01开始，日志存储时长将由原来的7天改成3天。

设备信息 | **在线调试** | 云端诊断日志 | 设备云端日志 | 设备本地日志 | 扩展信息

下发指令

属性调试 | 行为调用

功能名称/标识符	期望值	实时数据
<input type="checkbox"/> 移动国家号码(mcc)	(只读) 数值范围: 0-5,初始值: 0,步长: 1,单位:	-
<input type="checkbox"/> 移动国家号码(mcc)	(只读) 数值范围: 0-5,初始值: 0,步长: 1,单位:	-
<input type="checkbox"/> 网络制式(networkType)	(只读) 1 - GSM,2 - CDMA,3 - WCDMA,4 - TD_WCDMA,5 - LTE	-
<input type="checkbox"/> 温度(temperature)	(只读) 数值范围: -100-100,初始值: 0,步长: 0.1,单位: 摄氏度	22.1
<input type="checkbox"/> 湿度(humidity)	(只读) 数值范围: 0-100,初始值: 0,步长: 0.1,单位: %RH	41.5
<input type="checkbox"/> 设备地址(device_addr)	- 1 +	2

发送 | 重置

在阿里云监控运维>日志服务下，选择正确产品，输入对应的设备名称，点击查看业务类型为设备到云的消息，内容格式选择为Hex，在此可看见传感器回复的原始数据为0x01030400dd019eea31，0x01为设备地址，0x03为功能码，04为实际数据字节数，00dd为温度数据，019e为湿度数据，ea31为CRC校验码。



## 7.6 数据下发

通过腾讯云修改传感器设备地址，将其地址修改为 0x02。在腾讯云设备调试页面，勾选设备地址条目，在输入栏中输入 2，点击发送。



在修改设备地址之后，在腾讯云和阿里云可以看到设备不再上传数据，因为轮询指令中设备地址为 01，而此时设备地址以及被修改为 02。此时修改轮询指令将设备地址修改为 0x02、CRC 校验码也对应修改，最终为 02030000002C438。

基础配置	编码函数	解码函数	网络通道1	网络通道2	网络通道3	网络通道4
<b>串口配置</b>						
波特率(bps)	9600					
数据位	<input checked="" type="radio"/> 8位 <input type="radio"/> 7位					
校验位	<input checked="" type="radio"/> 无校验 <input type="radio"/> 奇校验 <input type="radio"/> 偶校验					
停止位	<input checked="" type="radio"/> 1位 <input type="radio"/> 2位					
通讯方式	<input type="radio"/> TTL <input type="radio"/> 232 <input checked="" type="radio"/> 485					
<b>采集方式</b>						
<input type="radio"/> 主动上报 <input checked="" type="radio"/> 轮询						
轮询周期	10 秒					
轮询指令	<input type="text" value="020300000002C438"/> (+)					

修改完成轮询指令后，等待设备更新配置，配置更新完成后可在腾讯云、阿里云看到数据上传。