

大家好，墨子号科技这里给大家介绍下使用 NB 模块来接入腾讯云使用的教程。采用 MQTT 协议接入方式。当然只要 NB 模块接入成功，4G 也是一样的。本质上 MQTT 没有区别。大家可以关注我们的技术群或者是公众号以及淘宝店铺索取最新资料。

## 目录

1. 腾讯云的注册 .....	1
2. 注册产品 .....	2
3. 物模型与小程序配置 .....	4
4. 添加设备 .....	10
5. 数据上报与下发 .....	17

## 1. 腾讯云的注册

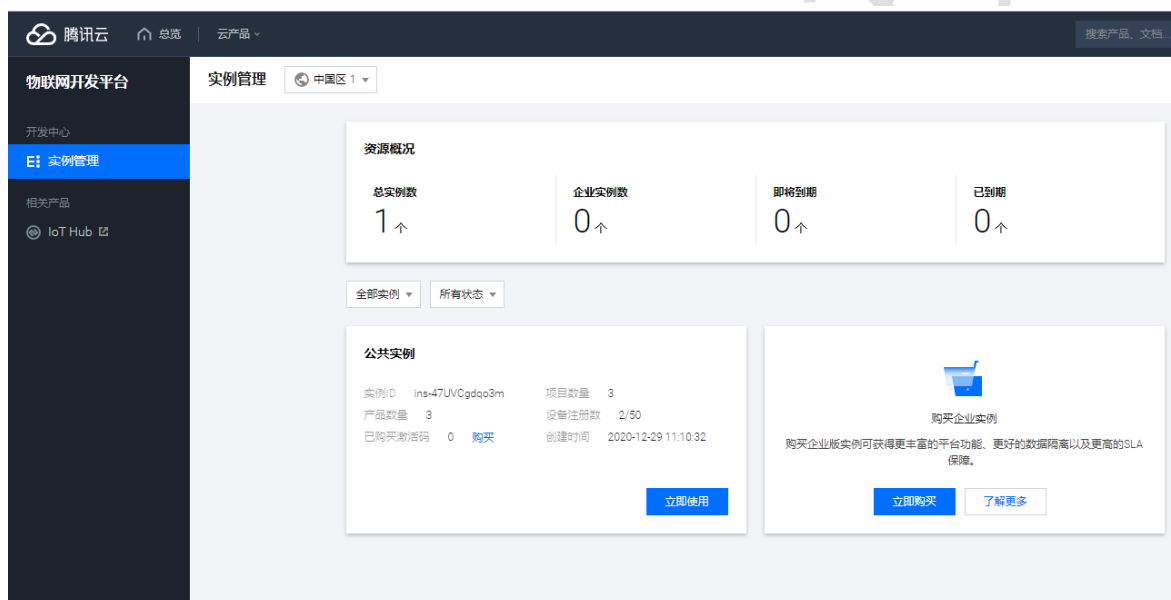
在百度当中输入腾讯云关键词，找到腾讯云注册平台，登录注册即可。



找到腾讯云官方平台点击进入注册即可。

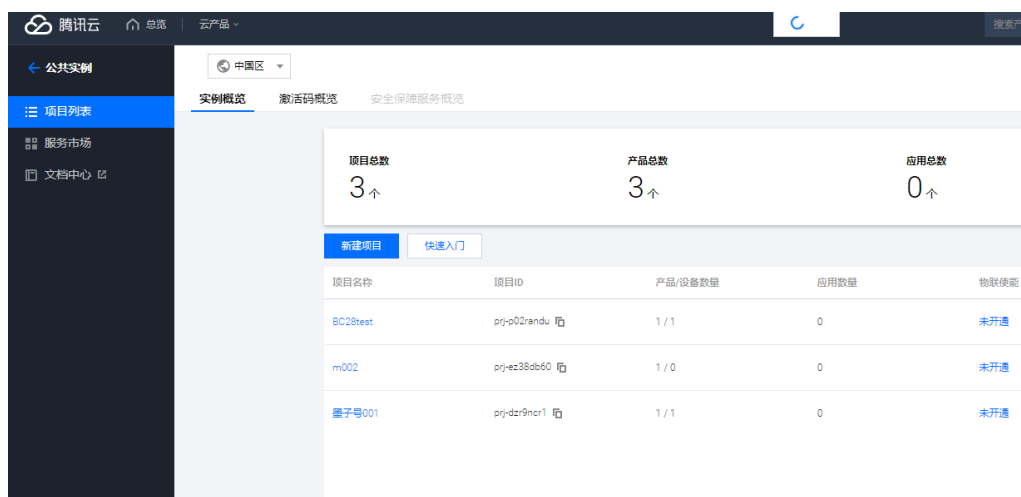


上面就是腾讯云的注册平台界面了，新用户根据提示注册下帐号即可。注册好之后，进入控制台找到物联网平台，进行产品和设备的注册。



## 2. 注册产品

进入产品服务里面，左边有实例管理，这里选用公共实例进入之后进行使用。这里我们主要给用户介绍下使用 MQTT 协议来接入腾讯云该如何注册以及模块端怎么使用。



找到项目列表，然后选择新建项目，因为新用户是没有项目的，所以需要先创建项目。



上面我们注册了一个 MOZIHAIQ\_TEST 的名称的项目，然后进入之后创建产品。

产品开发 / 新建产品

### 新建产品

产品名称 \*

BC28NBiot

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

智能城市 / 环境监测 / 温湿度检测

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

①

描述

选填

最多不超过80个字符

确定

取消

创建一个设备，设备类型一般正常都是选择设备默认即可，通信方式选择 2-4G 的方式。其他的认证方式和数据协议按照平台默认给出的即可。

产品名称	产品ID	产品品类	设备类型	状态	创建时间	操作
BC28NBiot	H264CYU8XX	智能城市-环境监测-温度检测	设备	开发中	2023-06-21 16:39:06	<a href="#">删除</a>

### 3. 物模型与小程序配置

产品开发 / BC28NBiot

1 物模型 > 2 设备开发 > 3 交互开发 > 4 设备调试 > 5 批量投产

[导入物模型](#) [查看物模型JSON](#)

**标准功能** [添加标准功能](#)

功能类型	功能名称	标识符	数据类型	读写类型	数据定义	操作
属性	地区区域码 <b>必选</b>	lac	整型	只读	数值范围: 0-32 初始值: 0 步长: 1 单位: -	<a href="#">编辑</a> <a href="#">删除</a>
属性	基站码 <b>必选</b>	cid	整型	只读	数值范围: 0-32 初始值: 0 步长: 1 单位: -	<a href="#">编辑</a> <a href="#">删除</a>
属性	移动网号 <b>可选</b>	mnc	整型	只读	数值范围: 0-5 初始值: 0 步长: 1 单位: -	<a href="#">编辑</a> <a href="#">删除</a>
属性	移动国家号码 <b>可选</b>	mcc	整型	只读	数值范围: 0-5 初始值: 0 步长: 1 单位: -	<a href="#">编辑</a> <a href="#">删除</a>
属性	网络制式 <b>可选</b>	networkType	枚举型	只读	1 - GSM 2 - CDMA 3 - WCDMA 4 - TD-LTE 5 - LTE	<a href="#">编辑</a> <a href="#">删除</a>

点击设备名称之后将进入物模型参数的设计，默认有一些标准功能，用户根据自己的需要，是否进行选择使用。这里我们会自己添加下物模型参数，比如温度和湿度数据。

**自定义功能** [新建自定义功能](#)

功能类型	功能名称	标识符	数据类型	读写类型	数据定义	操作
当前列表为空						

他在下面有一个自定义功能，我们将在这里新建自定义功能，来进行温湿度的参数的定义。当然了用户可以直接使用标准功能里面所拥有的参数，不是非必要一定进行自定义参数的。

新增自定义功能

×

① 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](#)或[申请新增标准功能](#)。

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 \*

temp

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

①

数值范围

—

0

+

—

100

+

初始值

—

0

+

步长

—

1

+

①

单位

0-12个字符

描述

远传

最多不超过80个字符

保存

取消

我们定义了一个自定义的温度功能。标识符很重要，因为后面上传数据就是由标识符来决定的。这里建议用户读写功能设置为只读，这样这个数据只能在小程序显示而不能在小程序的按钮上控制下发了。因为并不需要温度值的下发改动。

同理按照这个思路创建下湿度的功能。此外再添加一个下发按钮，实现下发控制。

自定义功能 ②							新建自定义功能
功能类型	功能名称	标识符	数据类型	读写类型	数据定义	操作	
属性	温度	temp	整数型	只读	数值范围：0-100 初始值：0 步长：1 单位：°C	编辑 删除	
属性	湿度	humi	整数型	只读	数值范围：0-100 初始值：0 步长：1 单位：%RH	编辑 删除	
属性	LED状态	ledstatus	布尔型	只读	0 - 关 1 - 开	编辑 删除	
属性	下发控制	ledctr	布尔型	读写	0 - 关 1 - 开	编辑 删除	

上面是设定的一些自定义功能，支持温湿度上报，LED 状态上报，LED 灯下发。

物模型 > 2 设备开发 > 3 交互开发 > 4 设备调试 > 5 批量投产

设备开发 Topic列表

请选择设备开发方式

基于模组开发

根据您的业务场景选择合适的通信模组，请查看文档详细了解如何[基于模组开发](#)

基于SDK开发

使用支持跨平台移植的C-SDK可基于不同硬件平台快速接入，请查看文档详细了解如何[基于C-SDK开发](#)

基于OS开发

根据您的业务场景选择合适的物联网操作系统，请查看文档详细了解如何[基于OS开发](#)

上一步

下一步

这一步是设备开发，这个地方直接选择下一步即可。有兴趣的用户可以了解下这些开发方式，提供了一些文档说明。

接入腾讯连连官方小程序

您选择使用平台的官方小程序控制产品

使用通用版APP控制产品

用户可以使用腾讯连连通用版APP控制产品，也可以使用开源版APP，目前APP集成APP SDK控制产品

配置小程序

配置APP

产品展示配置

您可以自定义产品在腾讯连连首页、设备列表中展示的产品图标和默认名称

配置

快捷入口配置

您可以自定义产品在设备列表、快捷功能区显示的快捷操作

配置

面板配置

您可以自定义产品控制面板的风格、布局、按钮样式等配置

配置

设备绑定引导

您可以自定义用户在绑定设备时的引导图和文案

配置

扫一扫产品介绍

您可以自定义用户在使用微信扫一扫添加设备时的产品介绍页

配置

智能联动配置

配置

这个界面比较重要，关系到小程序的配置。这个地方默认选择是接入腾讯连连的小程序，这样就可以在微信里面查看相关参数了，省去了自己设计 APP 的麻烦。然后下面做一下配置。

产品展示配置

产品展示图标

文件名：  
8ed8498aa3d7485d6ea942d7cfd279e3.png

重新上传 重置

建议上传.png格式的透明背景图，图片尺寸建议为64x64最大500kb

产品展示名称 \*

BC28NBiot

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

厂家名称 \*

墨子号

请输入厂家名称，100字符以内

产品型号 \*

MZH028

请输入产品型号，50字符以内

UI强制显示在线

开启后，不管设备是否在线，UI界面会一直显示设备在线

保存

预览

我的小家 腾讯连连

27° 多云 南山

实时温度 湿度 实时风向 北风

BC28NBiot

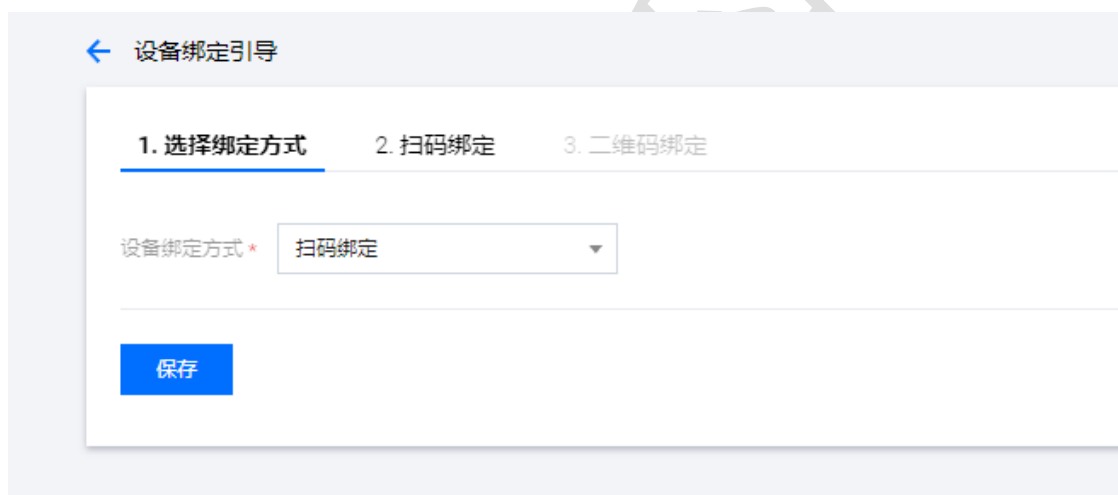
首页 智能 + 评测 我的

首先配置下产品展示配置，输入相关参数之后，进行保存。

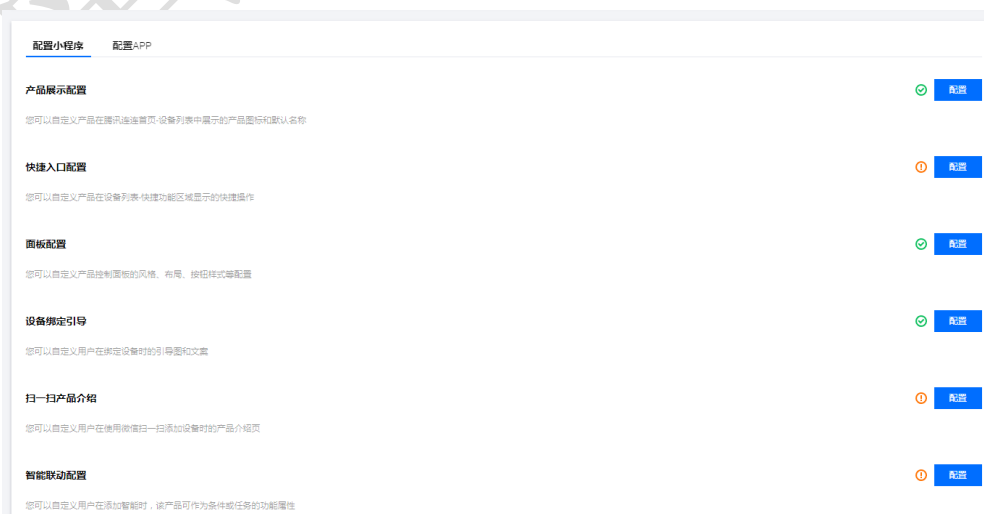
7



编辑面板, 这个地方一般标准功能定义的参数我们这里设置不显示, 只设置温湿度和按钮灯状态显示。设置好之后进行保存。



设备绑定方式选择扫码绑定, 实现扫描二维码进行设备的绑定。

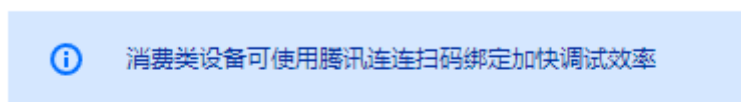




上面就是我们所设定的基本参数。用户可以根据自己的需要进行配置。然后下一步。



这样进入之后有个二维码，然后点击下二维码，用腾讯连连小程序扫码关注下进行设备绑定。



出来如上二维码，用户可以扫码进行关注绑定，使用腾讯连连。那么就会在手机小程序里面绑定此设备了。



上面是腾讯连连的界面，可以看到设备已经被绑定进来，这样只要模块端有数据上传后，这边也就会有数据同步进行显示了。

## 4.添加设备



产品开发创建好之后，下一步用将创建设备，在右边添加设备按钮创建下对应设备即可。

### 新建设备

所属产品

BC28NBlot

设备名称 \*

mzh028

支持英文、数字、下划线的组合，最多不超过48个字符

保存

取消

新建设备选择下所属产品，然后输入设备名称这样保存设备就被我们创建好了。

← mzh028

设备信息

设备属性

设备云端日志

在线调试

扩展信息

设备调试日志

设备信息

设备名称	mzh028	产品ID	H26ACYU8XX	所属产品	BC28NBlot
设备密钥	Ts99vX6uzZuuyXrkac9cwe	设备创建时间	2023-06-21 17:23:57	最后上线时间	-
激活时间	-	设备状态	未激活	固件版本	-

编辑

标签信息

设备标签	无标签信息
------	-------

编辑

设备本地日志

调试日志	关闭
日志等级	无

编辑

然后将创建好的设备打开其参数，这样就获取到了设备的相关信息参数，这些将对后面的设备登录非常重要，用户一定要保存使用。

下一步就要设备的连接了。首先我们先看下平台的相关文档说明，学习下先如何实现设备的连接。



用户可以搜索关键词 mqtt.fx 接入指引说明书, 学习下如何使用这个软件接入平台, 由于平台说明非常详细, 我们这里就不使用这个软件进行登录连接了。

然后我们看一下几个重要的参数来看下如何使用模块配置连接服务器。

参数	说明
Profile Name	用户自定义名称。
Broker Address	MQTT 服务器连接地址, 广州域设备填入: <code>PRODUCT_ID.iotcloud.tencentdevices.com</code> , 这里 <code>PRODUCT_ID</code> 为变量参数, 用户需填入创建产品时自动生成的产品 ID, 例如 <code>T****DS8G.iotcloud.tencentdevices.com</code> 。
Broker Port	MQTT 服务器连接端口, 填入: 1883。本文主要针对密钥认证类型的产品, 端口必须是1883, 如果您想通过8883接口接入, 建议使用证书认证型产品自行接入。
Client ID	MQTT 协议字段, 按照物联网通信约束填入: 产品 ID + 设备名, 如: <code>"TXXXXDS8Gdev001"</code> , <code>TXXXXDS8G</code> 是产品 ID, <code>dev001</code> 是设备名称。
Connection Timeout	连接超时时间 (秒)。
Keep Alive Interval	心跳间隔时间 (秒)。
Auto Reconnect	断网自动重连。

平台端有个非常重要的说明, 这个地方告知了接入地址端口和 ClientID 参数。详细介绍了参数怎么来的。那么下面先看下这 2 个参数。

MQTT 服务器地址是 `PRODUCT_ID.iotcloud.tencentdevices.com`



在产品信息里面看到了 `PRODUCT_ID`，所以这个地方的服务器地址是：  
`H26ACYU8XX.iotcloud.tencentdevices.com`。

`Client ID` 是产品 ID + 设备名，所以是：`H26ACYU8XXmzh028`。

5. 单击 **User Credentials**，填写 User Name 和 Password。

- **User Name**：MQTT 协议字段，按照物联网通信约束填入：产品 ID + 设备名 + SDKAppID + connid+expiry。创建完产品即可在产品列表页和产品详情页查看 ProductID，例如："TO\*\*\*\*DS8Gdev001;12010126;E4F3Q;1591948593"，仅替换示例中的产品 ID + 设备名即可，后面的三个参数本身由物联网通信接入 SDK 自动生成，也可由腾讯云物联网平台提供的 [生成小工具](#) 自动生成。
- **Password**：Password 必须填写，用户可以使用物联网平台提供的 [生成小工具](#) 自动生成 Password，也可以按照文档 [手动生成 Password](#)。

这个地方 user 和 password 是最为重要的参数，这将决定是否可以登录平台。  
这个是需要进行计算的，当然腾讯也提供了 C 语言的算法，这样在单片机里面可以自动计算，我们这里采用小工具的方式获取参数。

请输入设备信息：

ProductID:	<input type="text" value="H26ACYU8XX"/>
DeviceName:	<input type="text" value="mzh028"/>
DeviceSecret:	<input type="text" value="Ts99vX6uzZuuyvXrkac9cw=="/>
Hmac签名算法:	<input type="text" value="HMAC-SHA256"/>
<input type="button" value="Generate"/>	

结果：

UserName:	<input type="text" value="H26ACYU8XXmzh028;12010126;81R9F;1687947562"/>
Password:	<input type="text" value="2c9a914a51250748154ea6a13360cd8f505c809d431a5bad1da6973a6b498df0;hmacsha256"/>

上面就是利用小工具所计算来的用户名和密码，用户要保存，当然在单片机代码里面修改掉宏定义即可。

下面使用移远的 NB 模块 BC28CNV 来给大家演示下如何接入腾讯云平台。

```
+CSQ:25,99
OK
+CGATT:1
OK
460081109306208
OK
[10:19:24.702]发->◇ATI
□
[10:19:24.822]收<-◆
Quectel
BC28-CN
Revision:BC28CNVAA02A02
OK
[10:19:25.261]发->◇AT+CGATT?
□
[10:19:25.340]收<-◆
+CGATT:1
OK
```

在接入平台之前，一定要先确认卡号，注册网络，信号值等都正常后再进行下一步的连接操作。

## 6 举例

本章提供了一些示例以解释如何使用与 MQTT 相关的 AT 命令。

```
//为阿里云配置阿里设备信息。
AT+QMTCFG="aliauth",0,"oyjtmPI5a5j","MQTT_TEST","wN9Y6pZSily7Exa5qVzcmigEGO4kAazZ"
OK

AT+QMTOPEN=?
+QMTOPEN: (0),<host_name>,(0-65535)

OK

//开启 MQTT 客户端网络。
AT+QMTOPEN=0,"iot-as-mqtt.cn-shanghai.aliyuncs.com",1883
OK

+QMTOPEN: 0,0 //成功打开 MQTT 客户端网络。

AT+QMTOPEN?
+QMTOPEN: 0,"iot-as-mqtt.cn-shanghai.aliyuncs.com",1883

OK

AT+QMTCONN=?
+QMTCONN: (0),<clientID>,<username>,<password>

OK

//若连接阿里云，使用命令 AT+QMTCFG="ALIAUTH" 提前配置设备信息，之后便可省略参数
<username> 和 <password>
AT+QMTCONN=0,"clientExample" //将客户端连接到 MQTT 服务器。
OK

+QMTCONN: 0,0,0 //已成功将客户端连接到 MQTT 服务器。

AT+QMTSUB=?
+QMTSUB: (0),(1-65535),<topic>,(0-2)[,<topic>,(0-2)...]

OK
```

在 BC28 的 MQTT 手册里面有相关的指令介绍，用户可以根据指令来对接云平台，不管是阿里云还是其他云指令上都是区别不大的。

```
[10:24:14.792]发→◇AT+QIDNSCFG=8.8.8.8,8.8.4.4
□
[10:24:14.886]收←◆
OK
[10:24:18.142]发→◇AT+QMTOPEN=0,"H26ACYU8XX.iotcloud.tencentdevices.com",1883
□
[10:24:18.280]收←◆
OK
[10:24:19.095]收←◆
+QMTOPEN: 0,0
```

下一步是登录腾讯云平台，输入腾讯云平台的 IP 和端口，设置成功后，会

返回 0,0, 表明地址和端口没问题，下一步进行连接。由于 BC28 需要配置下 DNS 服务器才可以进行域名解析，所以设置了域名解析的指令然后就可以接入服务器了。

上面提及到获取 clientid,username,password 的参数了，下面就可以使用指令进行连接了。

```
AT+QMTCONN=?  
+QMTCONN: (0),<clientID>,<username>,<password>  
OK
```

用这个指令来进行配置，这个指令只要配置成功后，腾讯云的设备就会在线了。

```
[10:38:33.067]发->◇AT  
+QMTCONN=0,"H26ACYU8XXmzh028","H26ACYU8XXmzh028;12010126;G25YD;1688265396","3beadaaa932ca5735de3b2d0534c2d323e5a5cd4783019ba66d8348fde886c07;hmacsha256"  
[10:38:33.323]收<-◆  
OK  
[10:38:33.902]收<-◆  
+QMTCONN: 0,0,0
```

如上看到返回 0,0,0 代表连接成功，此时参数没有问题，就可以设备此时应该在线了。

← mzh028

设备信息

设备属性

设备云端日志

在线调试

扩展信息

设备调试日志

设备信息

设备名称	mzh028	产品ID	H26ACYU8XX
设备密钥	Ts99vX6uzZuuyvXrkac9cw==	设备创建时间	2023-06-21 17:23:57
激活时间	-	设备状态	在线

下一步就可以进行数据的上报与下发测试了。



## 5.数据上报与下发

设备在线之后，下一步进行数据的上报与下发测试。需要确认下上报数据的主题以及数据格式这样才好进行测试。

关于数据格式和主题下面这个链接有比较详细的介绍，用户可以学习了解。

<https://cloud.tencent.com/document/product/1081/34916>

数据点上报的主题，这里我们采用设备属性上行请求 TOPIC 主题来进行数据上报。

### 设备属性上报

1. 当设备需要向云端上报设备运行状态的变化时，以通知应用端小程序、App 实时展示或云端业务系统接收设备上报属性数据，物联网开发平台为设备设定了默认的 Topic：

- 设备属性上行请求 Topic： `$thing/up/property/{ProductID}/{DeviceName}`
- 设备属性下行响应 Topic： `$thing/down/property/{ProductID}/{DeviceName}`

对于我们关注的的数据上报问题，主要就是属性上报请求与上报响应。就是对应的发布与订阅，这一点与阿里云极其类似。

### 2. 请求

- 设备端请求报文示例：

```
{
  "method": "report",
  "clientToken": "123",
  "timestamp": 1628646783,
  "params": {
    "power_switch": 1,
    "color": 1,
    "brightness": 32
  }
}
```

对于数据格式也有要求，用户需要按照如上的数据格式来实现数据的上报，这样才会在腾讯云平台进行显示。

发布数据主题和数据内容都有相应的介绍了，那么下面根据这个说明来做一下数据上报功能测试下。

**AT+QMTPUB=0,0,0,0,"topic/pub"**

**>这是测试数据, hello MQTT.**

**OK**

**+QMTPUB: 0,0,0**

//发布消息。

//接收到 **>** 后，输入数据：“这是测试数据, hello MQTT”，数据的最大长度为 1024 字节，超过 1024 字节的数据将被忽略，输入完成后，点击“**Ctrl**”+“**Z**”发送。

上面是 AT 指令的上报说明，这个是每个模块对应的指令会有一些区别，用户需要根据自己所用的模块所对应的数据格式进行数据上报。

```
[10:48:07.191]发->◇AT+QMTFUB=0,0,0,0,"$thing/up/property/H26ACYU8XX/mzh028"  
[10:48:07.347]收<-◆  
>  
[10:48:09.848]发->◇{  
  "method": "report", "clientToken": "123", "params": {"temp": 12, "humi": 41}  
}  
[10:48:13.431]发->◇->□  
[10:48:13.482]收<-◆  
OK  
[10:48:13.619]收<-◆  
+QMTFUB: 0,0,0
```

这个是利用 AT 指令进行数据上报，如果主题没有错误的话，那么即便数据格式有问题，那么都会发布成功。在平台端有数据显示。注意上面没有上报时间戳，这样平台显示的时间就将是平台的本地时间了。用户可以省略时间戳的上报，此外标识符要与平台端的要对应上才可以，这样在平台才会有数据显示以及在腾讯连连上有显示。

```
[11:03:48.031]发->◇AT+QMTFUB=0,0,0,0,"$thing/up/property/H26ACYU8XX/mzh028"  
[11:03:48.175]收<-◆  
>  
[11:03:51.946]发->◇{"method": "report", "clientToken": "123", "params": {"temp": 12, "humi": 41}}  
[11:03:54.544]发->◇->□  
[11:03:54.594]收<-◆  
OK  
[11:03:54.732]收<-◆  
+QMTFUB: 0,0,0
```

这样上报，可以看到返回上报成功，那么看下平台端情况。



这样在腾讯云平台就显示出对应的数据了，这样数据上报就成功了。

← mzh028

设备信息 设备属性 设备云端日志 在线调试 扩展信息 设备调试日志

属性名称/属性标识符 Q

标识符	功能名称	历史数据	数据类型	最新值	更新时间
lac	地区区域码	<a href="#">查看</a>	整型	-	-
cid	基站码	<a href="#">查看</a>	整型	-	-
mnc	移动网号	<a href="#">查看</a>	整型	-	-
mcc	移动国家号码	<a href="#">查看</a>	整型	-	-
networkType	网络制式	<a href="#">查看</a>	枚举型	-	-
temp	温度	<a href="#">查看</a>	整型	12	2023-06-25 11:03:55.017
humid	湿度	<a href="#">查看</a>	整型	41	2023-06-25 11:03:55.017
ledstatus	LED状态	<a href="#">查看</a>	布尔型	-	-
ledctr	下发控制	<a href="#">查看</a>	布尔型	-	-

同样在腾讯连连也可以看到数据被更新了。



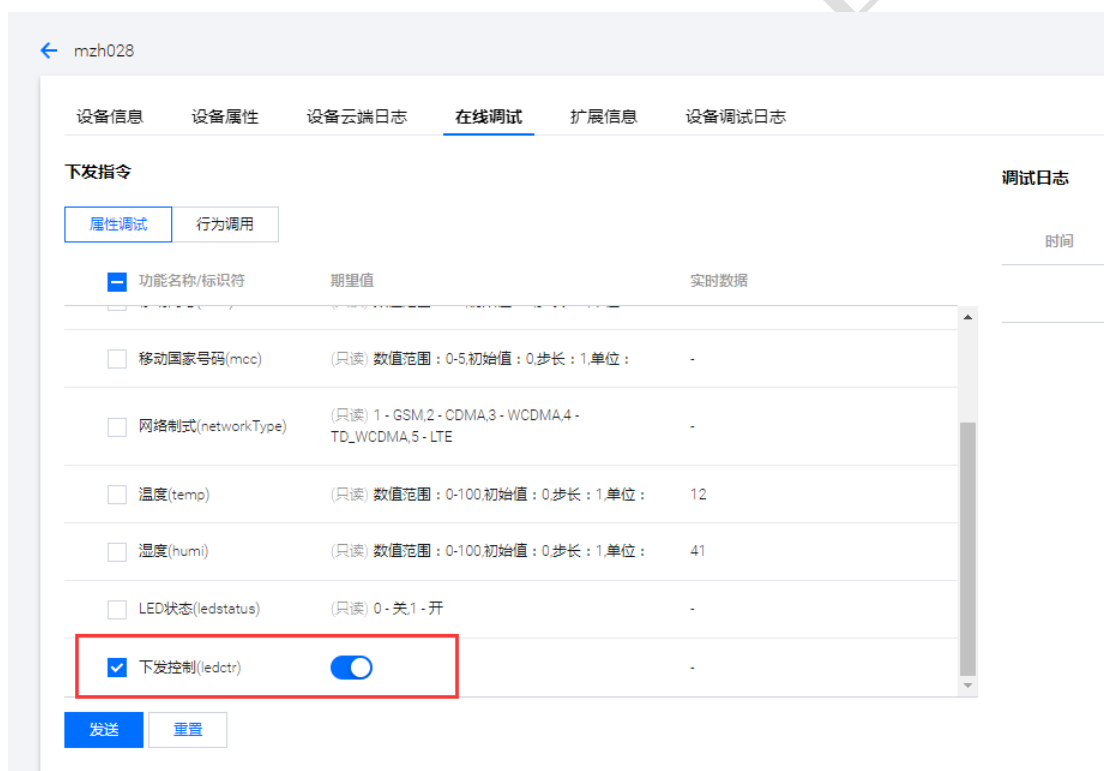
这样完成了腾讯云的数据的上报，下面测试下腾讯云的数据下发功能，对于模块而言，需要订阅下主题才可以进行收到腾讯云平台的数据下发，这点用户需要注意。

```
[10:56:02.488]发→◇AT+QMTSUB=0,1,"$thing/down/property/H26ACYU8XX/mzh028",0
[10:56:02.621]收←◆
OK

[10:56:03.471]收←◆
+QMTSUB: 0,1,0,0
```

首先发个指令订阅下主题，这个是用来接收平台下发数据的，不然不订阅收不到下发数据。

在平台端找到命令下发的地方或者用腾讯连连下发也是可以的。我们在平台端下发给客户看下。



打开在线调试界面，找到属性调试，下发指令，然后选择需要下发的指令参数并配置参数进行发送，这样在模块端就会被收到并打印出来。

```
+QMTRECV: 0,0,"$thing/down/property/H26ACYU8XX/mzh028",113,{"method":"control","clientToken":"v2149650409aRh1C::694fcf4c-c2db-420e-b4a2-10eefbcd6d5","params":{"ledctr":1}}
```

可以看到接收到了平台下发的数据了。这样对于模块的上传与下发功能都完成了。下发下来的参数数据格式在上面的协议介绍文档都有非常详细的介绍，用户可以根据自己的需要来将下发的功能代码进行使用即可，这里将下发的数据内容用户控制开发上的LED开关，并上报了属性。这样就可以实现远程采集温湿度并控制LED开关，这样可以很方便使用手机端来与板子之间进行交互了。

后续我们会针对主推的开发板，陆续更新到腾讯云物联网平台代码来，更加方便用户来使用腾讯云平台，谢谢！

墨子号科技