DTU 固件 Web 配置用户手册

版本:V2.2

版权声明
版权所有：深圳市银尔达电子有限公司,深圳市银尔达电子有限公司保留所有权利。
说明

本文档用于记录、指导研发流程和人员基本文档。
公司网站：http://www.yinerda.com
联系电话： 0755-23732189
联系地址： 深圳市龙华区大浪街道华宁路 117 号中安科技园 A 栋 2003-2005

修订记录

版本	修改内容	编写人	修订时间
V1.0	新建	杨洋	20210307
V1.1	添加支持 N58 系产品说明 添加常见问题描述	杨洋	20210420
V1.2	修改 N58 系列 DTU 服务器配置平台型号和设备对应关系	杨洋	20210505
V1.3	添加 4G 与 4G 点对点的传送数据的实现方式介绍	杨洋	20210514
V1.4	添加串口无数据重启时间和网络无数据重启时间介绍 删除自定义服务器描述	杨洋	20210605
V1.5	添加远程控制命令功能 添加 NTP 同步周期 添加 GPIO 功能 添加定位配置功能 新增功能 724 系列大于 1.1.0 固件版本支持，N58 系列不支持	杨洋	20210717
V1.6	添加远程控制实例 添加 GPIO 控制实例 添加定位信息上报实例 添加数据模板实例 添加固件升级实例	杨洋	20210805
V1.7	添加 HTTP 协议	杨洋	20210812
V1.8	添加阿里云企业版 MQTT 连接域名说明 更新远程升级说明	杨洋	20210927
V1.9	添加 GPS 说明 添加支持变量说明	杨洋	20220127
V2.0	添加视频教程链接	杨洋	20220727
V2.1	添加基本参数低功耗、LED、TTS 等参数说明 添加连接腾讯 IOT 参数 添加连接移动 Onenet 说明	杨洋	20221029
V2.2	添加系统 LED 状态描述	杨洋	20230103

目录

零、 视频教程 5
一、硬件资源说明 5
1.1、适合 Air724 模组 5
1.2、适合 N58 模组 6
1.3、适合条件 6
1.4、设备支持功能查询 6
1.5、设备网络状态 LED 描述 6
1.6、设备信号强度 LED 描述 6
二、 WEB 配置的优点 7
1)设备参数的集中管理 7
2)设备参数的可追溯管理 7
3)设备参数的批量下发 7
4)随时动态修改参数 7
三、 配置平台通用信息说明 8
3.1、通用自动替换-变量引用说明 8
3.2、网络通道-心跳包说明 9
3.3、网络通道-登录注册信息说明 9
3.4、网络通道-MQTT topic 订阅和发布说明 9
四、 WEB 配置使用流程 10
1)注册账号 10
2)添加设备 10
3)创建分组 10
4)分组添加设备 10
5)设置参数 10
五、 WEB 配置步骤 11
5.1、添加配设备 11
5.2、创建新分组 11
5.3、分组添加设备 12
5.4、配置参数 12
5.5、设备更新参数 13
5.6、设备的详情 13
六、 参数配置项含义 15
6.1、基本参数 15
6.2、串口参数 16
6.3、自动轮询 17
6.4、网络通道参数 17
6.4.1、TCP/UDP 参数 18
6.4.2、MQTT 参数 19
6.4.3、阿里云参数 20
6.4.4、HTTP 参数 21
6.4.5、腾讯 IOT 参数 23
6.4.6、移动 Onenet-TCP 协议参数 23

6.4.7、移动 Onenet-Modbus 协议参数 24
6.4.8、移动 Onenet-MQTT 物联网套件(新版)参数 25
6.5、GPIO 25
6.5.1、输出控制 26
6.5.2、输入电平周期上报 26
6.5.3、输入电平检查预警 26
6.5.4、输入电流周期上报 27
6.5.5、输入电流检查预警 27
6.5.6、输入电压周期上报 28
6.5.7、输入电压检查预警 28
6.6、定位 29
6.6.1、基站定位 30
6.6.2、GPS 定位 30
七、 TCP 测试实例 32
八、 DTU 自动轮询实现 modbus 实例 35
九、 阿里云一型一密实例 37
十、 4G 与 4G 点对点通讯实例 42
十一、 远程控制命令实例 48
十二、 GPIO 功能实例 50
十三、 定位实例 52
十四、 数据流模板实例 54
十五、 DTU 固件远程升级实例 56
十六、 腾讯 IOT 连接实例 58
十七、 移动 Onenet 多协议接入-TCP 连接实例 65
十八、 移动 Onenet 多协议接入-MODBUS 连接实例 70
十九、 移动 Onenet-MQTT 物联网套件新版连接实例 73
二十、 常见问题说明 77

零、视频教程链接

本文配套视频教程:https://www.bilibili.com/video/BV1364y117zc

一、硬件资源说明

1.1、适合 Air724 模组

产品型号	物理接口	特殊描述
YED-D724X	ttluart+rs485	5~36V 供电、ZH1.5 接线座
YED-D724X1		5~36V 供电、2EDG3.8 接线座
YED-D724W	rs232+rs485	5~36V 供电、ZH1.5 接线座
YED-D724W1		5~36V 供电、2EDG3.8 接线座
YED-G724W		5~36V 供电、标准 9 针 RS232 串口
YED-DG724W		5~36V 供电、导轨固定、2EDG5.0 接线座
YED-C724	ttluart	5~16V 供电、双排排针固定、IPEX 天线
YED-Core724		5~16V 供电、单排排针固定、SMA 天线
YED-M724		5~16V 供电、单排排针固定、IPEX 天线、超小体积
YED-Core724-IPEX		5~16V 供电、单排排针固定、IPEX 天线、翻盖卡、休眠管脚
YED-D724Y1	rs485	5~16V 供电、ZH1.5 接线座、IPEX 天线
YED-D724Y3		5~36V 供电、防水防尘防油外壳、航空接线头
YED-RY1880	rs232+rs485	5~36V 供电、8 路继电器输出，8 路光耦隔离电平输入
YED-RY18811	rs232+rs485	5~36V 供电、8 路继电器输出、8 路光耦电平输入、1 路 0~10V 电压采集、1 路 0~20ma 电流采集
YED-D820W1	rs232+rs485	5~36V 供电、4G+GPS 北斗透传、1 路 0~20ma 电流采集
YED-Core820	ttluart	5~36V 供电、4G+GPS 北斗透传、IPEX 天线
YED-RN102	无	5~36V 供电、2 路继电器、TTS 喇叭、导轨外壳
YED-RY1222	无	5~36V 供电、2 路 0~20ma 电流/电流/电压/NTC 采集，2 路继电器、2 路数字量输入，TTS 喇叭、导轨外壳
YED-AS8K8	ttluart	5~36V 供电,1 路 ttluart,8 路输出，8 路输入，TTS
YED-DG724W6	rs232+rs485	220V 供电，DTU，铁壳，导轨
YED-DG724W6-L	rs232+rs485	220V 供电，塑料外壳，导轨
YED-D724L1-Y	rs485	5~36V 供电，可以做低功耗，1 路电流检查，1 路数字量，1 路可控电源输出，SD 卡
YED-DG724W7	rs232+rs485	220V 供电，DTU，铁壳，导轨，SD 卡
YED-DG724W7-H	rs232+rs485	220V 供电，DTU，铁壳，导轨，RS232+RS485 隔离，SD 卡
YED-DG724T2	3 个 RS485	5~36V 供电，3 路独立 RS485,(2 路普通 485,1 路隔离 485)，SD 卡
YED-S724C	ttluart	5~36V 供电，1 路 TTL 串口，一路 5~30V ADC，1 路数字量
YED-S724Z	rs232	5~36V 供电，1 路 RS232 串口，一路 5~30V ADC，1 路数字量
YED-S724Y	rs485	5~36V 供电，1 路 RS485 串口，一路 5~30V ADC，1 路数字量
YED-RY1440	rs232+rs485	5~36V 供电，1 路 RS485 串口，4 路输入，4 路继电器，一路 5~30V 电源 ADC，1 路耳机口（可接音响功、放喇叭）
YED-J724U	ttluart	5~36V 供电，1 路 TTL 串口，封装兼容有人 7S1 Cat1 核心板
YED-C820	ttluart	5~12V 供电，2 路 TTL 串口，2 路 ADC，4 路 GPIO

1.2、适合 N58 模组

产品型号	物理接口	特殊描述
YED-D58X1	ttluart+rs485	5~36V 供电、2EDG3.8 接线座
YED-D58W1	rs232+rs485	5~36V 供电、2EDG3.8 接线座
YED-G58W		5~36V 供电、标准 9 针 RS232 串口、RS485 带隔离
YED-DG58W		5~36V 供电、导轨固定、2EDG5.0 接线座
YED-CoreN58	ttluart	5~16V 供电、单排排针固定、SMA/IPEX 天线

1.3、适合条件

1) SIM 卡为普通 SIM 卡，可以连接参数配置平台
2) 如果 SIM 是定向卡，SIM 需要提前添加平台域名
3) 如果 SIM 卡为专网卡，不适用 web 服务器配置

1.4、设备支持功能查询

Web 配置平台流程和功能是一样的。具体的设备型号支持某些功能，不支持某些功能。

具体支持的功能请参考对应系列模组手册：

Air724 系列参考《银尔达-Air724 系列 DTU 固件功能用户手册(必读》

N58 系列参考《银尔达-N58 系列 DTU 固件功能用户手册(必读》

1.5、设备网络状态 LED 描述

设备上面一般有 3 颗 LED 灯，其意义如下。如果只有 NET LED，就只需要看 NET LED 状态。

指示意义	现象	备注
供电正常	PWR LED 常亮，并且明亮	
供电异常	PWR LED 不亮，或者昏亮	
设备没出厂初始化	NET LED 和 RDY LED 2000ms 同时闪烁	联系销售处理,需要出厂初始化。
SIM 卡不识别	NET LED 和 RDY LED 5000ms 同时闪烁	
SIM 卡正常，但注册不了网络	NET LED 100ms 闪烁,RDY LED 熄灭	
注册网络成功，但没连上服务器	NET LED 500ms 慢闪,RDY LED 熄灭	没有任何通道链接服务器
成功连上服务器	NET LED 1000ms 慢闪,RDY LED 常亮	至少有一个通道链接服务器成功

1.6、设备信号强度 LED 描述

某些型号 DTU 有 5 颗信号 LED，可以用于指示信号强度

LED 亮个数	CSQ 范围	备注
5	26~31	极强
4	21~25	强
3	17~20	一般
2	12~16	差
1	6~11	很差（不能稳定通信）
0	<6	不能通信

二、WEB 配置的优点

1)设备参数的集中管理

全部的设备都在同一个平台上。

2)设备参数的可追溯管理

平台设置的设备参数都会保存，并且可以导出，需要查看的时候可以随时查看。

3)设备参数的批量下发

同一个分组的设备，其参数全部相同，可以同时对几百上千的设备批量设置，节省时间。

4)随时动态修改参数

当设备部署后，如果参数需要修改，可以在平台上直接配置参数，不需要到部署地方去本地配置。

三、配置平台通用信息说明

3.1、通用自动替换-变量引用说明

变量格式为${XXX},是 DTU 参数的一个占位符号，在 DTU 逻辑会自动把它匹配成设备运行的实际值，用来实现批量自动替换的功能。

变量分为通用变量和专用变量。

通用变量在基本全部的字符串模式都能匹配，比如${IMEI},${ICCID}等；专用变量只有特定功能应用，比如${DI1},${AII1}等。

比如在注册包里添加变量${IMEI},设备就会自动把${IMEI}替换成模组的实际值 865145689XXXX。参数支持的变量如下:

变量名	功能	固件支持版本	字符串	16进制	备注
${IMEI}	模块 IMEI	1.0.5	Y	Y	通用。 设备唯一编码，可以用于识别设备。 16 进制描述:比如 IMEI 为 867435053363255，固定 15 位，被替换的内容会在末尾添加 0 变成 0x86 0x74 0x35 0x05 0x33 0x63 0x25 0x50 最有一个 Byte 的低位填充 0
${ICCID}	SIM 卡 ICCID	1.0.5	Y	N	通用
${CSQ}	信号强度	1.0.5	Y	N	通用
${GPS}	基站定位固定格式	1.0.8	Y	N	通用
${LBSLON}	基站经度	1.1.7	Y	N	通用，开启基站定位
${LBSLAT}	基站纬度	1.1.7	Y	N	通用，开启基站定位
${GPSLONT}	GPS 经度东西	1.1.7	Y	N	通用，支持 GPS，并且开启 GPS
${GPSLON}	GPS 经度	1.1.7	Y	N	通用，支持 GPS，并且开启 GPS
${GPSLATT}	GPS 纬度南北	1.1.7	Y	N	通用，支持 GPS，并且开启 GPS
${GPSLAT}	GPS 纬度	1.1.7	Y	N	通用，支持 GPS，并且开启 GPS
${DIn}	数字量输入通道	1.1.7	Y	N	IO 功能处引用，支持输入，n 是数字量通道
${AIIn}	电流量输入通道	1.1.7	Y	N	IO 功能处引用，支持输入，n 是数字量通道
${AIVn}	电压量输入通道	1.1.7	Y	N	IO 功能处引用，支持输入，n 是数字量通道

3.2、网络通道-心跳包说明

心跳包包开关	打开，DTU 会按“心跳包发送间隔时间”定期发送“心跳包数据”给服务器 心跳包是 DTU 自己检查是否与服务器断开的方法。是否打开心跳包与自己的应用场景有关系，如果打开，服务器需要过滤心跳。
心跳包数据	**功能:**用于周期和服务器发送数据，保存网络连接 设置 Hex 16 进制比如要设置 0xaa 0x01 0xbb ，设置格式为 aa 01 bb 或者 aa01bb
	心跳包变量匹配说明: 在字符串模式下，支持${IMEI}、${ICCID}、${CSQ}等通用变量替换实际值自由组合上报数据。
心跳包发送间隔时间	心跳包频率 用于维持设备与服务器的连接，一般 60~180 秒内比较好 **说明:只有在不做任何数据交互的时候，才会发送心跳包数据**

3.3、网络通道-登录注册信息说明

登录注册信息	每次 DTU 新连接服务器后的第一个数据包
	固定格式说明:包括了 IMEI(设备编码),ICCID(SIM 卡标识),CSQ(信号强度)，和固件版本等数据的 json 字符串比如 {"fver":"YED_DTU_1.0.3_luat_v0030_rda8910_float","iccid":"89860411102070058856","imei":"862167052964826","csq":22} **变量自动匹配说明:** 支持引用变量${IMEI},${ICCID},${CSQ}等通用变量替换实际值自由组合上报数据。

3.4、网络通道-MQTT topic 订阅和发布说明

订阅 topic	单个 topic 如:/bbb1 多个 topic 以“;”分割，比如/bbb1;/bbb2 就同时订阅了/bbb1,bbb2。 **变量自动匹配说明:** topic 如果包含${IMEI}会自动替换成设备的实际 IMEI
发布 topic	单个 topic 如:/pus 多个 topic 如:headingcode:111/aaa1;headingcode:222/aaa2 **变量自动匹配说明:** topic 如果包含${IMEI}会自动替换成设备的实际 IMEI; **多 topic 发布说明:** 每个 topic 以“headingcode”开头,以“;”分割。 定义数据识别码，格式为:headingcode:识别码/。 数据识别码说明: 数据识别码为串口数据开头的某几个固定字符，用于确定这条串口数据使用哪一个 topic 发送给服务器。识别码可以是任意数字的字符串，当串口数据的头与识别码对应后，这一包的数据就会通过对应的 topic 发送给服务器。如果数据不匹配任何识别码，这包数据将丢失。 比如 headingcode:111/aaa1，定义了数据识别码为 111，发布的 topic 为 aaa1;如果串口的前面 3 个字节等于“111”,这一包数据就会通过 topic aaa1 传送给服务器。

四、WEB 配置使用流程

WEB 配置平台地址:https://dtu.yinerda.com/

1)注册账号

可以自己使用手机号注册平台账号。手机号为账号唯一识别码，请妥善保管。

2)添加设备

购买产品后，登录平台，在设备列表里面使用模组 IMEI 添加设备。

3)创建分组

创建新的分组，可以配置串口波特率 ，通讯协议等内容，保存参数

4)分组添加设备

一个分组下的设备，参数全部相同，包括产品型号。

5)设置参数

编辑分组，保存参数即可，注意每次保存都会导致参数变化，设备就会自动更新。如果你只是浏览参数而没有修改，建议不要点击保存，直接返回。

6)设备更新参数

重启设备(断电重启或者按 Reload 5 秒恢复出厂设置)或者 24 小时后，如果服务器有参数更新，设备会自动更新设置的参数，如果无更新，不做任何处理。

7)确认参数更新

通过对比服务器参数版本和设备参数版本，当 2 个参数版本一样表示设备已经更新参数。当设备参数小于服务器参数版本，表示设备还没有更新。

五、WEB 配置步骤

本节视频教程:https://www.bilibili.com/video/BV1364y117zc?p=17

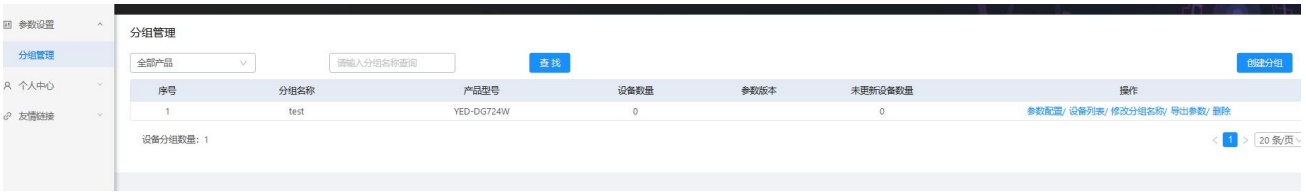
5.1、添加配设备

当您注册新账号后，如果设备列表没有购买的设备，需要添加设备。可以用微信扫码设备 IMEI 后添加（**不用填写 SN**），如果添加成功，能看到**产品型号**和设备 IMEI。**产品型号**后面创建分组需要使用。

如果提示:“找不到此设备”表示设备没授权，需要联系淘宝销售授权后再添加。

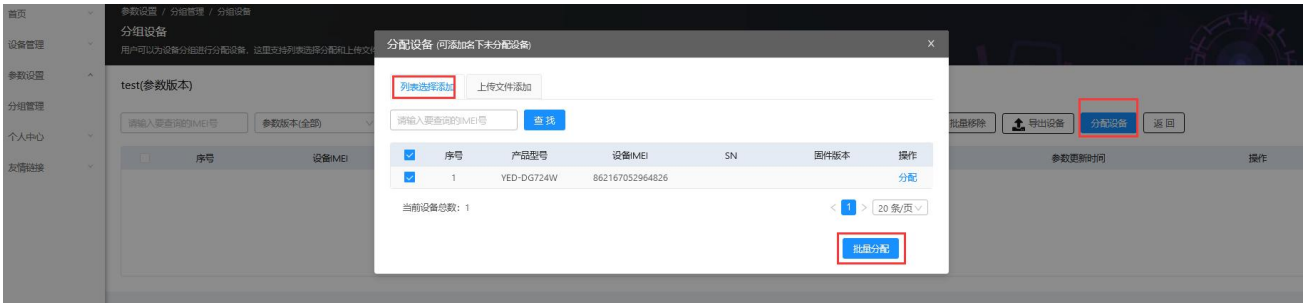
5.2、创建新分组

分组名称自定义，分组产品型号必须和**设备列表显示的“产品型号”**一样。分组是设备参数的载体，同一个分组的所有设备参数都一样(包括产品型号)，**如果你的设备每个参数都不相同(不能通过变量自动引用)，就需要每个设备创建一个分组**。创建分组的名字可以任意定义，选择的产品型号后就不能在修改。如果需要删除分组，需要把分组的设备全部移除才能删除。



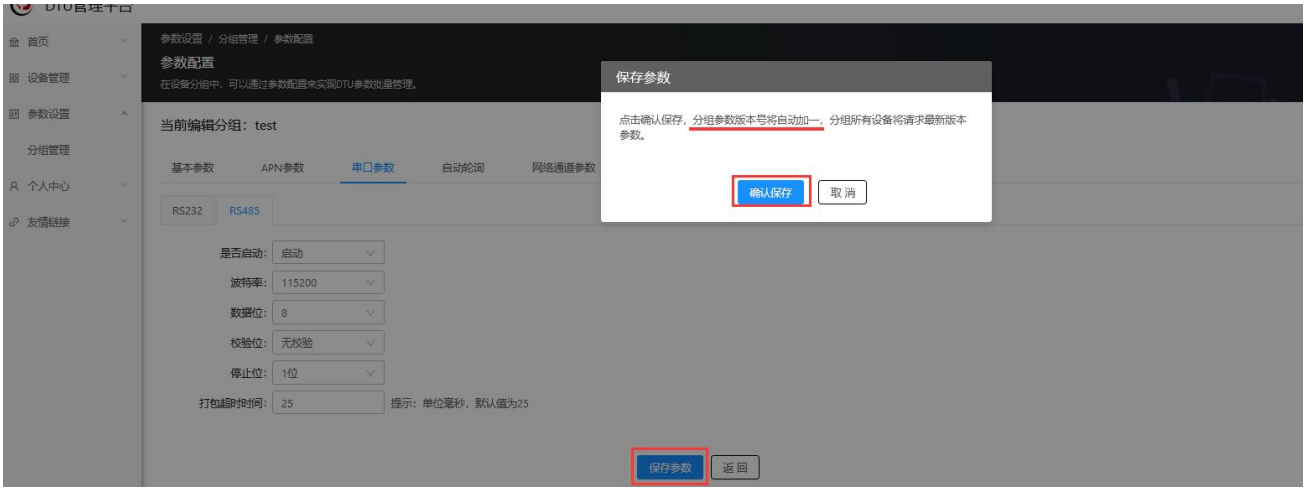
5.3、分组添加设备

在分组里面添加设备，这些设备的参数就会和分组一样。可以点击查找或者 F5 刷新页面数据。



5.4、配置参数

进入参数配置，当配置好参数后，点击保存，确认保存后，就修改了参数。注意任何确认保存都会修改服务器参数，导致 DTU 重新更新参数。如果您只是浏览参数，就不要点击保存直接返回即可。当保存参数退出后，就可以看到分组的参数版本变化了，并且提示没有更新的设备数量。



5.5、设备更新参数

确认设备能正常联网后(NET LED 500ms 闪烁)，**重启设备**，设备就会自动更新参数。然后再设备管理->设备列表里面查看更新结果。当分组参数版本等于设备参数版本时表示设备参数更新成功。设备信息不会自动刷新，可以点击查找或者 F5 更新界面。

如果设备已经部署出去，就需要想办法让设备重启或者再分组管理->基本参数->设置分组参数更新为自动更新，**就会 24 小时自动请求一次服务器**。请求一次，如果服务器没更新参数会使用少量的请求流量，大概 500 多 Byte。

5.6、设备的详情

设备列表->详情，可以看到设备的更多的东西比如 ICCID 和固件版本号。ICCID 可以用来查询 SIM 卡是否停机，固件版本号可以用来确认产品是否有更新最新功能。

设备详情
IMEI：862167052964826
SN：
分组名称：test
ICCID：89860411102070058856
归属用户：13560789567
设备添加日期：2021-03-07 15:09:35
产品型号：YED-DG724W
固件版本：YED_DTU_1.0.3_luat_v0030_rda8910_float
服务器参数版本：1
设备参数版本：1
参数更新时间：2021-03-07 15:20:36

六、参数配置项含义

不同的设备，根据资源，开放的配置 UI 不一样。

6.1、基本参数

项目	含义
网络分帧时间	网络数据分包超时时间，当 2 个字节的数据间隔大于这个时间后，将发送给串口
设置参数密码	如果设置了密码，当下一次设置的时候必须输入同样的密码才能设置，否者只能本地复位模组参数，才能更新新参数(注意当密码错误后，服务器显示参数更新，但是设备实际更新不成功)。 同时本地串口必须验证密码后才能读取参数，可以防止非法访问设备的参数。
DTU 日志输出	控制 DTU 调试日志。日志会有设备信息，可以用于调试设备
DTU 固件更新	控制 DTU 固件升级功能。可以升级成最新的 DTU 固件。当然固件是稳定的，就不建议随便升级。
分组参数更新	自动更新表示设备在不重启的情况下，24 小时后会自动请求一次。 手动更新表示设备只会更新第一次参数，后面不会主动更新参数，除非设备恢复出厂设置。
DTU 自动重启时间	可以设置 DTU 在运行多久时间后，自动重启
串口无数据重启时间	设置后，启用的串口，如果超过这个时间没有接收到任何数据，DTU 自动重启
网络无数据重启时间	设置后，启用的网络通道，如果超过这个时间没有接收到任何数据，DTU 自动重启
远程控制命令	目标服务器是否能下发配置命令配置设备。(和串口执行命令相同) 设备默认关闭
NTP 同步周期	NTP 网络时间同步功能，设备默认关闭，可以解决基站时间不对的问题
新连接清除缓存	网络通道断开重连的时候，删除本通道的发送缓存数据

SIM 卡切换	有的设备支持双卡，单待。目前只支持默认。插了外置卡，默认用外置卡
启用低功耗	支持低功耗的设备，再保持服务器连接(及时响应交互数据)，能名下降低功耗。已 S724C 为例，不启用低功耗，大约 5V 30ma.启用低功耗大约 5V 4.5ma
TTS 播报通道	设备支持耳机孔(可外接功放喇叭)和 SPK 口（直接驱动 8 欧 2W 喇叭），选择不同的接口播放
关闭 LED	关闭 LED 能减少更多的能耗

6.2、串口参数

项目	含义
是否启动	启动表示打开串口，如果关闭，串口不能通信
波特率	串口波特率 1200~921600
数据位	Air724 系列只支持 8 位数据，不支持 7 位数据
校验位	无校验，奇、偶校验
停止位	1、2 位
打包超时时间	串口数据分包超时时间，当 2 个字节的数据间隔大于这个时间后，将发送给服务器
EN 翻转时间	默认为 0 的时候，延迟时间为 5 个波特率时钟，比如 9600 波特率 EN 时间是 1000/9600=104us。 RS485 匹配 EN 翻转需要，如果设置需要根据波特率计算延迟时间 如果发现 DTU 接收 485 数据 固定前面丢失，说明 EN 太长了，需要减小时间 如果发现您的设备接收不全，固定最后数据丢失，说明 EN 太短，需要增加时间

6.3、自动轮询

自动轮询，是提前在 DTU 内保存对外轮询的命令，当设置命令后，DTU 就会按照间隔自动发送数据给串口。当收到串口数据后，在自动发送到服务器。这样可以减少服务器的轮询流量。

项目	含义
是否启动	启动后，就会发送对应的数据
轮询应答超时时间	当发送一条命令后，等待这个时间发送下一跳轮询命令。两条命令之间间隔
轮询周期时间	当设置的全部命令轮询完成后，等待这个时间后，继续从第一条命令开始轮询
轮询命令格式	轮询的数据格式，可以是字符串或者 Hex16 进制。DTU 会根据类型转数据
轮询命令	需要发送的命令 当设置字符串的时候，轮询原封不动的发送出去 设置 Hex 16 进制格式的时候，比如要设置 0xaa 0x01 0xbb ，设置格式为 aa 01 bb 或者 aa01bb Air724 系列支持 50 条；N58 系列支持 20 条

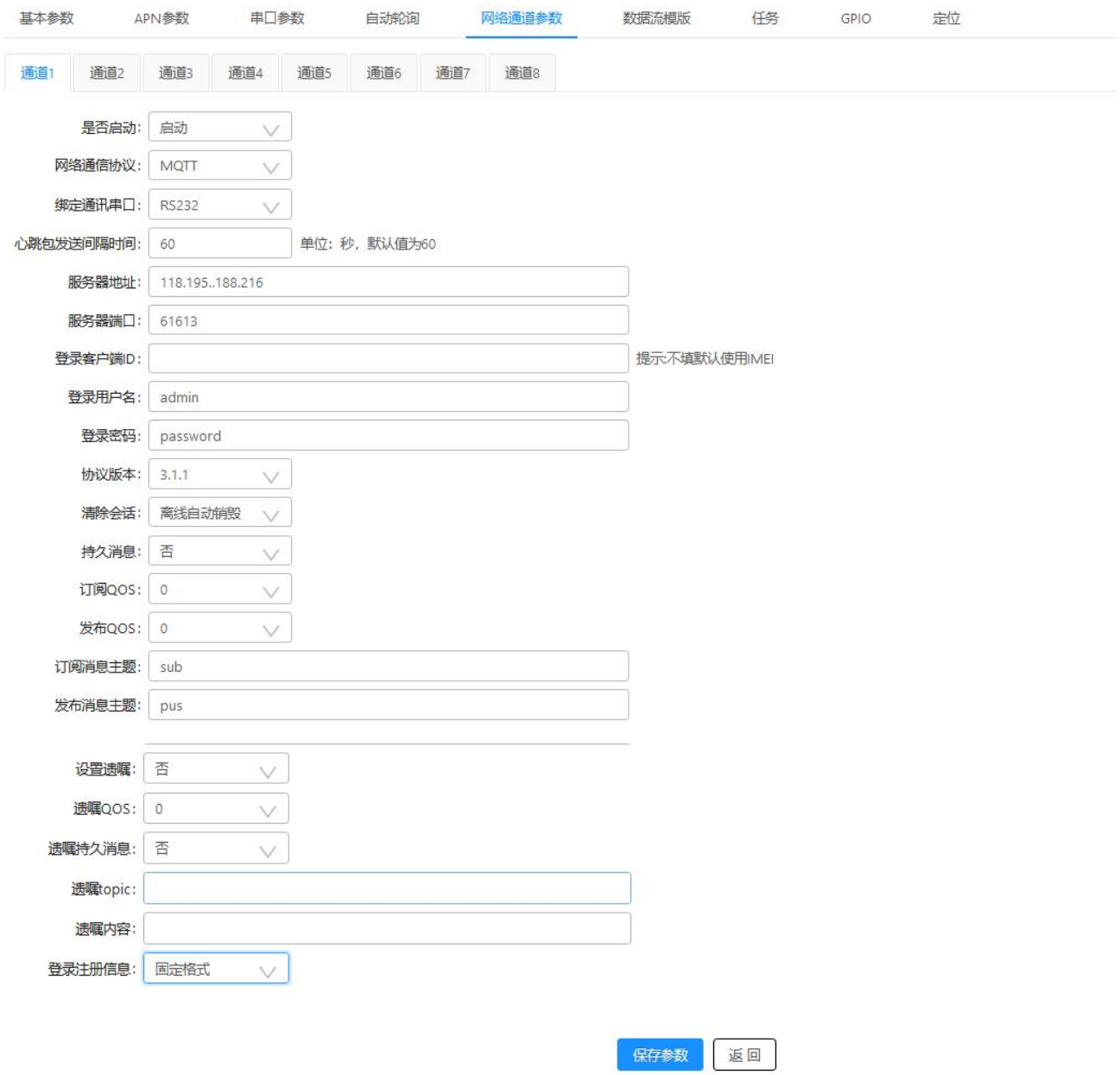
6.4、网络通道参数

网络通道参数，每个通道表示一个连接，有多少个通道就最多可以连接多少个服务器。设置 DTU 连接服务器的通信协议和对应的参数，分别支持 TCP、UDP、MQTT、阿里云的一型一密和一机一密，目前都不支持 SSL 加密连接方式。

6.4.1、TCP/UDP 参数

项目	含义
是否启动	是否启动本通道
网络通信协议	TCP/UDP
绑定通信串口	本网络通道绑定串口硬件
心跳包包开关	支持变量，参考第三章，网络通道心跳包说明
心跳包数据	
心跳包发送间隔时间	
服务器地址	DTU 需要连接的地址，支持域名和 IP
服务器端口	服务器端口号
数据前置字段 数据后置字段	**功能:**在有效数据的前面或者后面加上固定的标记 对心跳、注册包以外的所有数据有效
	支持数据变量匹配说明: 在字符串模式下，支持${IMEI}、${ICCID}、${CSQ}等通用变量替换实际值自由组合上报数据。
登录注册信息	**功能:**每次 DTU 新连接服务器后的第一个数据包
	支持变量，参考第三章，网络通道登录注册信息说明

6.4.2、MQTT 参数



项目	含义
是否启动	本通道是否启动
网络通信协议	MQTT
绑定通信串口	本网络通道绑定串口硬件
心跳包发送间隔时间	MQTT 协议要求的心跳时间，一般建议 60~120 秒
服务器地址	支持域名或者 IP
服务器端口	
登录客户端 ID	如果不填写或者${IMEI}默认使用模组的 IMEI
登录用户名	MQTT 服务器的用户名
登录密码	MQTT 服务器的密码

协议版本	
清除会话	
持久消息	
订阅 QOS	订阅 topic 的 QOS 等级
发部 QOS	发部 topic 的 QOS 等级
订阅 topic	支持变量,支持多 topic，参考第三章，网络通道 MQTT topic 说明
发布 topic	
设置遗嘱	一般不用设置
遗嘱 QOS	
遗嘱持久消息	
遗嘱 topic	
遗嘱内容	
登录注册信息	每次 DTU 新连接服务器后的第一个数据包
	支持变量，参考第三章，网络通道登录注册信息说明

6.4.3、阿里云参数

阿里云通道是连接阿里云物联网 IOT 平台的，分为一型一密和一机一密。一型一密，可以保证一个分组，添加全部的设备；一机一密需要每个设备分一个组(devicename 不同)。

注意:DTU 只能配置一个通道连接阿里云 IOT。

一型一密的设备，如果设备换了参数，需要本地恢复出厂设置后，才能连接新参数。

项目	含义
是否启动	本通道是否启动
通信协议	阿里云
绑定通信串口	本通道的串口数据是通过那个硬件发送的
连接类型	一型一密，一机一密
心跳包间隔	协议自带的，推荐设置 60 秒
地域代码	阿里参数和你的账号注册有关系，要更加实际填写，不能填写错，华东 2 默认是上海
ProductKey	阿里产品参数
ProductSecret	阿里产品参数（一型一密）
DeviceSecret	阿里产品参数（一机一密）
Devicename	阿里产品参数（一机一密）
清除会话	
QOS 级别	阿里订阅与发布 QOS 是一样的
订阅主题	支持变量，参考第三章，网络通道 MQTT Topic 信息说明
发布主题	**阿里 topic devicename 说明:** 阿里 topic 格式如下 /a1Ul8cgfQCv/${deviceName}/user/get1 ${deviceName}需要替换成真实的参数。如果是一型一密，用${IMEI}替换
登录注册信息	每次 DTU 新连接服务器后的第一个数据包
	支持变量，参考第三章，网络通道登录注册信息说明
阿里 MQTT 域名	阿里企业版和 2021 年 7 月 30 日后开通的公共版需要填写 查询连接:连接实例->查看开发配置->MQTT 设备接入域名

6.4.4、HTTP 参数

Air724 系列 DTU 固件版本大于 1.1.2 版本支持。最多同时支持 2 路 HTTP 协议通道。 适合小数据的传送，比如传感器数据上报。不适大数据传送，比如下载文件，固件升级。

项目	含义
是否启动	是否启动本通道
网络通信协议	HTTP
绑定通信串口	本网络通道绑定串口硬件
请求方法	GET/POST
服务器地址	DTU 需要连接的地址，支持域名和 IP 1、www.yinerda.com 2、http://www.yinerda.com 3、http://192.168.1.1 192.168.1.1
服务器端口	服务器端口号:80
请求的 Url	请求的地址：/XXX **URL 说明:** get 会把串口的数据拼接到 url 上面，编码规则需要提前转换好 设置服务器地址为 www.yinerda.com,服务器端口为 80，url 为/test?aaa=。 DTU 最终请求的 url 为 http://www.yinerda.com:80/test？aaa=串口数据
等待超时时间	等待 HTTP 返回超时时间 设置的是 DTU 内部，HTTP 超时时间。实际上 DTU 自己实现 7 次重传。7 次重传都失败后会自动重启服务器，最大响应时间是（设置时间+5)*7 秒
是否自定义头部	**自定义 head 说明:** head 参数传入自定义请求头，如果自定义请求头中存在 Host、Connection、Content-Length 三个请求头，其他的需要自己添加。

	head 可以自定义添加，每个变量已/0d/0a 结尾 比如:Content-Type=application/octet-stream/0d/0a
返回数据过滤	是否过滤 code ，head 数据
登录注册信息	每次 DTU 新连接服务器后的第一个数据包

6.4.5、腾讯 IOT 参数

项目	含义
是否启动	是否启动本通道
网络通信协议	腾讯 IOT-IoT Explorer
绑定通信串口	本网络通道绑定串口硬件
心跳包间隔	MQTT 协议的心跳包时间
Region	区域 ID，现在默认支持“ap-guangzhou”
ProductID	产品 ID
ProductSecret	产品密码
DeviceName	设备名字，建议用${IMEI}自动引用设备 IMEI 作为名字，方便管理
清除会话	MQTT 协议的参数
消息 QOS	MQTT 协议的参数
订阅消息主题	支持变量，参考第三章，网络通道 MQTT Topic 信息说明
发布消息主题	**topic devicename 说明:**topic 格式如下 /LP74R9AQ3M/${deviceName}/data ${deviceName}需要替换成真实的参数。如果是一型一密，用${IMEI}替换
登录注册信息	每次 DTU 新连接服务器后的第一个数据包
	支持变量，参考第三章，网络通道登录注册信息说明

6.4.6、移动 Onenet-TCP 协议参数

项目	含义
是否启动	是否启动本通道
网络通信协议	移动 Onenet
绑定通信串口	本网络通道绑定串口硬件
通讯协议	移动旗下，多协议接入 TCP 协议
心跳包包开关	支持变量，参考第三章，网络通道心跳包说明
心跳包数据	
心跳包发送间隔时间	
服务器地址	目前固定
服务器端口	目前固定
产品 ID	产品参数
设备注册码	产品参数
脚本名称	产品参数
设备名称	建议使用${IMEI},方便管理
鉴权信息	建议使用${IMEI},方便管理

6.4.7、移动 Onenet-Modbus 协议参数

项目	含义
是否启动	是否启动本通道
网络通信协议	移动 Onenet
绑定通信串口	本网络通道绑定串口硬件
通讯协议	移动旗下，多协议接入 Modbus 协议
心跳包间隔	
产品 ID	产品参数
Master-APIKey	产品参数
设备名称	如果为${IMEI}自动替换设备的 IMEI
设备序列号	如果为${IMEI}自动替换设备的 IMEI 的后 11 位
设备密码	如果为${IMEI}自动替换设备的 IMEI 的后 8 位

6.4.8、移动 Onenet-MQTT 物联网套件(新版)参数

项目	含义
是否启动	是否启动本通道
网络通信协议	移动 Onenet
绑定通信串口	本网络通道绑定串口硬件
通讯协议	移动旗下，多协议接入 Modbus 协议
心跳包间隔	
产品 ID	产品参数
AccessKey	产品参数
设备名称	建议使用${IMEI}自动替换设备的 IMEI，方便管理
消息 QOS	
订阅应答	是否接收上报数据后服务器的确认应答

6.5、GPIO

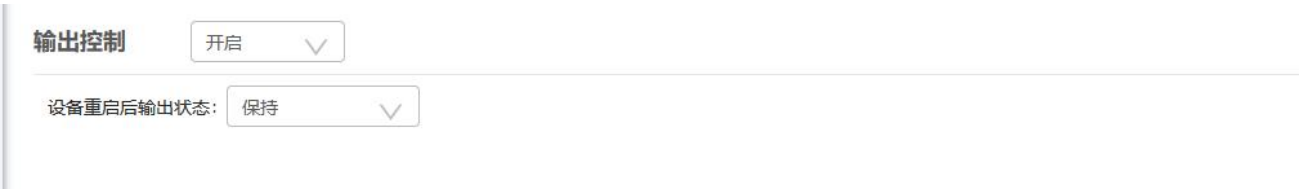
GPIO 功能只有具有 DI 输入、DO 输出、AI 电流、AV 电压输入功能的设备才有功能。

继电器的重启后的状态:比如充电桩开关，当设备网络重启设备后，还以维持之前的控制状态;

输入电平周期上报、输入电流周期上报、输入电压周期上报，可以上报设备本地输入到服务器。

输入电平检查预警、输入电流检测预警、输入电压检测:可以当某个输入状态满足触发条件后，上报对应数据到服务器并且可以本地联动输出。

6.5.1、输出控制

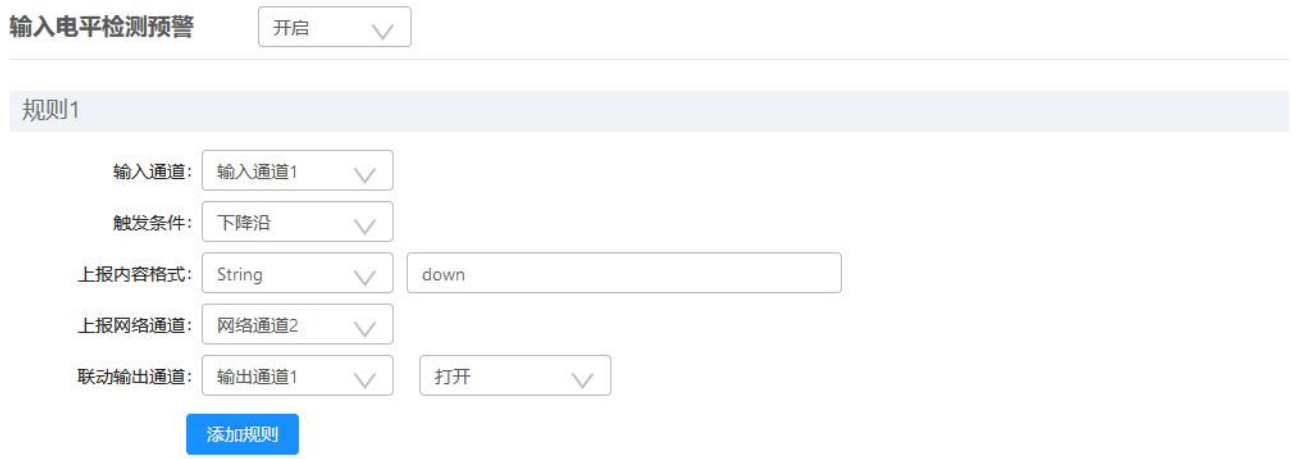


项目	含义
设备重启后输出状态	默认:默认全部关闭 保持:保持重启之前的输出状态

6.5.2、输入电平周期上报

项目	含义
上报周期	上报状态服务器的周期
上报内容	支持${DIn}和通用变量替换实际值自由组合上报数据。 ${DI1}表示通道 1，${DI8} 表示通道 8，和硬件通道一一对应
上报网络通道	状态数据上报的网络通道，需要提前配置网络通道

6.5.3、输入电平检查预警

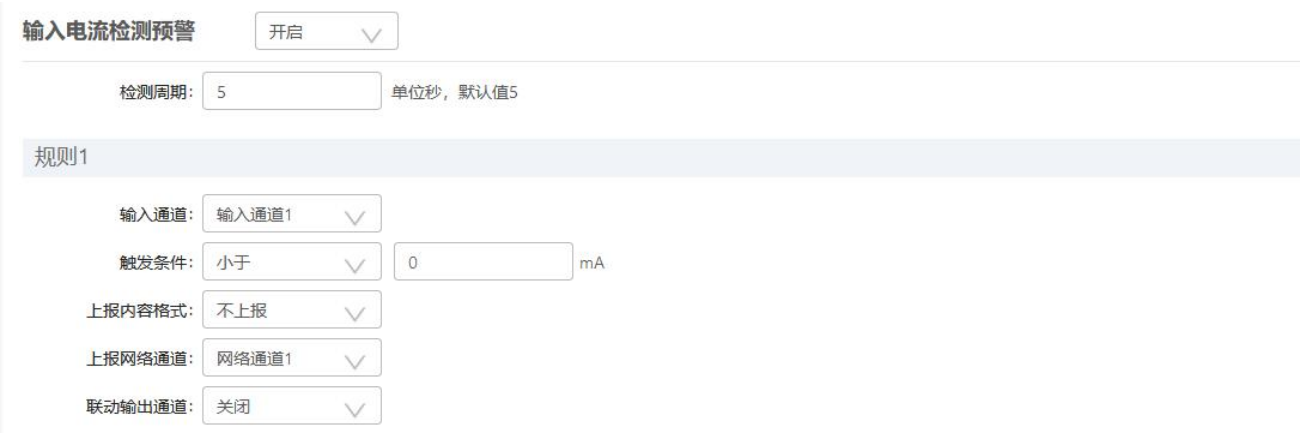


项目	含义
输入通道	对应硬件的输入通道
触发条件	上升沿:设备内部从低电平到高电平 下降沿:设备内部从高电平到低电平 基本上设备默认是高电平，当触发后变成低电平，比如 YED-RY880
上报内容格式	字符串格式:支持${DIn}和通用变量替换实际值自由组合上报数据。
上报网络通道	上报数据的网络通道，需要提前配置网络通道 如果格式选择不上报，就不会上报数据
联动输出通道	对应硬件的输出通道 打开:继电器是常合，高低电平是 高电平 关闭:继电器是常开，高低电平是 低电平

6.5.4、输入电流周期上报

项目	含义
上报周期	上报状态服务器的周期
上报内容	支持${AIIn}和通用变量替换实际值自由组合上报数据。 ${AII1}表示通道 1，${AII2} 表示通道 2，和硬件通道一一对应
上报网络通道	状态数据上报的网络通道，需要提前配置网络通道

6.5.5、输入电流检查预警



项目	含义
检测周期	电流检测的间隔
输入通道	对应硬件的输入通道
触发条件	区间:满足这个区间范围会上报
上报内容格式	字符串格式:支持${AIIn}和通用变量替换实际值自由组合上报数据。
上报网络通道	上报数据的网络通道，需要提前配置网络通道 如果格式选择不上报，就不会上报数据
联动输出通道	对应硬件的输出通道 打开:继电器是常合，高低电平是 高电平 关闭:继电器是常开，高低电平是 低电平

6.5.6、输入电压周期上报

项目	含义
上报周期	上报状态服务器的周期
上报内容	支持${AIVn}和通用变量替换实际值自由组合上报数据。 ${AIV1}表示通道 1，${AIV2} 表示通道 2，和硬件通道一一对应
上报网络通道	状态数据上报的网络通道，需要提前配置网络通道

6.5.7、输入电压检查预警

项目	含义
检测周期	电流检测的间隔
输入通道	对应硬件的输入通道
触发条件	区间:满足这个区间范围会上报
上报内容格式	字符串格式:支持${AIVn}和通用变量替换实际值自由组合上报数据。
上报网络通道	上报数据的网络通道，需要提前配置网络通道 如果格式选择不上报，就不会上报数据
联动输出通道	对应硬件的输出通道 打开:继电器是常合，高低电平是 高电平 关闭:继电器是常开，高低电平是 低电平

6.6、定位

定位功能包含了基站定位和 GPS 定位。

Air724 支持基站定位

Air820 支持基站定位和 GPS 定位

N58 不支持基站定位上报，支持串口查询基站定位。

基站定位信息会产生延迟和额外流量消耗。也不应当一定能定位成功，至少需要 3 颗基站，并且基站信息在服务器数据库才行，有定位失败的可能，误差 500M 到几公里，用于区域定位。

GPS 定位需要再室外，不能用于室内

6.6.1、基站定位

项目	含义
更新时间	定位信息更新时间，最少 60 秒， 建议时间设置长一些减少流量消耗
主动上报内容格式	不上报:设备只更新，不上报，可以再其他地方引用变量。 固定格式:字符串，格式为纬度_经度，比如 113.9736648_022.6940965。 自定义格式:可以引用${GPS} ${LBSLON},${LBSLAT}等通用变量替换实际值自由组合上报数据。
主动上报网络通道	数据上报网络通道，需要提前配置网络通道

6.6.2、GPS 定位

项目	含义
更新时间	定位信息更新时间，最少 5 秒，

主动上报内容格式	不上报:设备只更新，不上报，可以再其他地方引用变量。 固定格式:字符串，格式为纬度_经度，比如 113.9736648_022.6940965。 自定义格式:可以引用${GPSLONT},${GPSLON},${GPSLATT},${GPSLAT}等通用变量替换实际值自由组合上报数据。
主动上报网络通道	数据上报网络通道，需要提前配置网络通道

七、TCP 测试实例

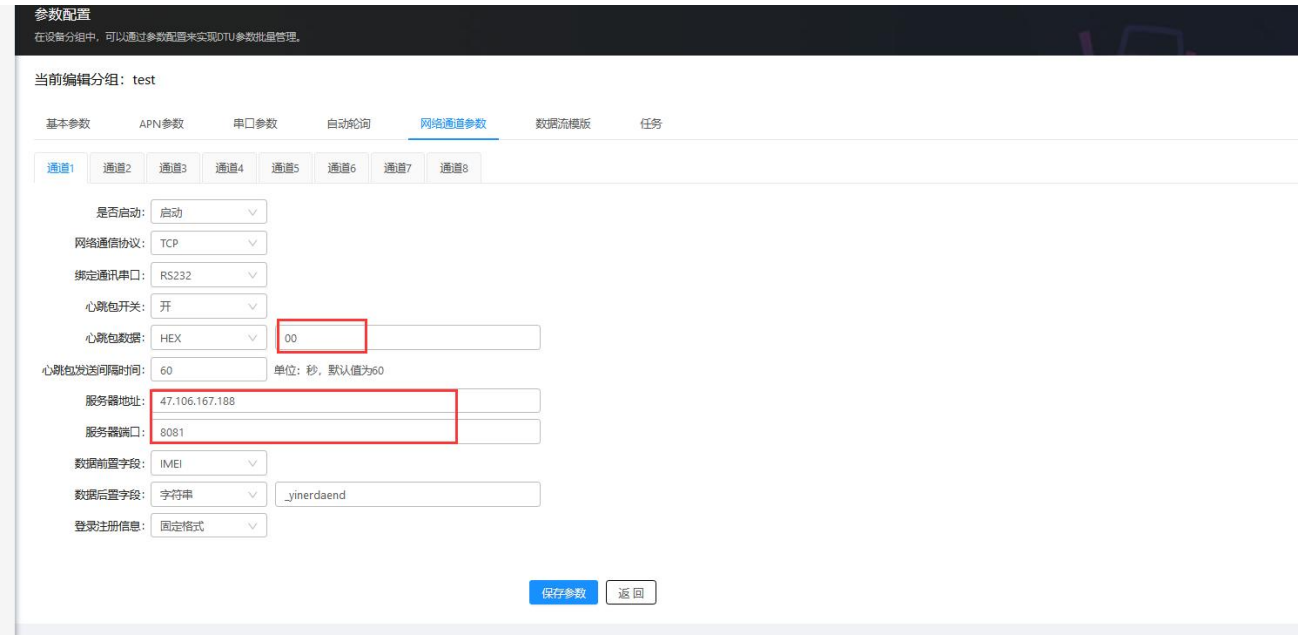
以 RS232 通讯为例，RS485，TTLUART 方法都一样；UDP 和 TCP 方法一样；

$启动 RS232 串口

$启动通道 1，

设置的必要信息心跳包、服务器 IP 和端口，心跳包设置的是 0x00,60 秒发送一次

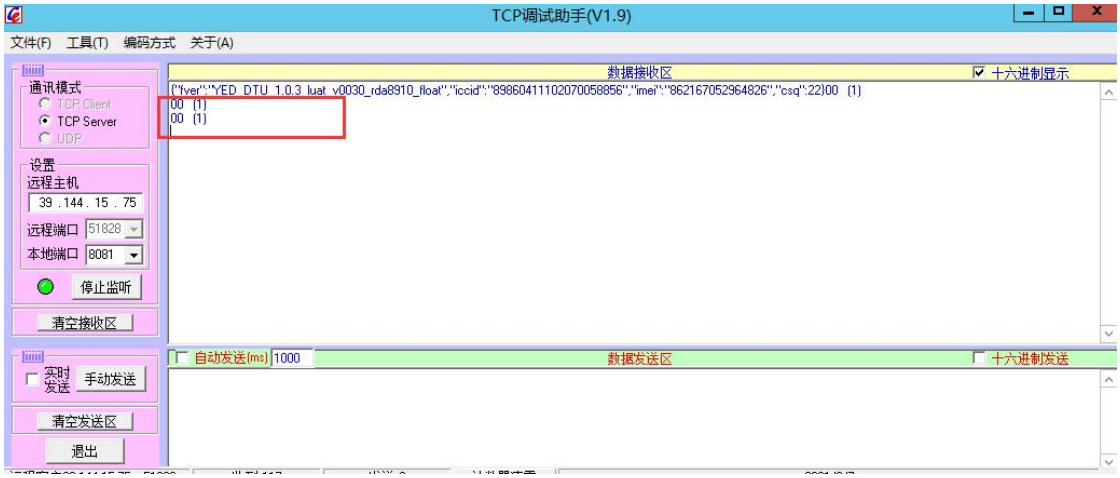
根据实际需求设置其他信息，比如数据前置段，数据后置段，或者登录注册信息



$数据内容分析

下图为设备登录注册信息，注册信息为固定格式。

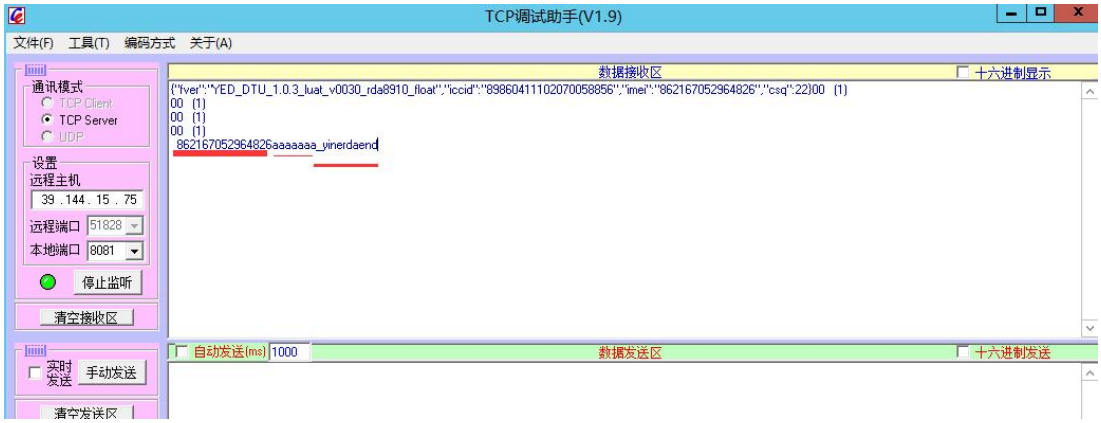
下图为心跳包数据,再 hex 16 进制模式下看到是 0x00



下图为 RS232 上报的数据，服务器收到的数据。

RS232 实际发送数据 aaaaaaa 给服务器

在服务器上可以看到 aaaaaaa 是串口发送的实际数据；862167052964826 是前置数据 IMEI；_yinerdaend 是后置数据。


TCP调试助手(V1.9)
文件(F) 工具(T) 编码方式 关于(A)
通讯模式
TCP Client
TCP Server
UDP
设置
远程主机
39 . 144 . 15 . 75
远程端口 51828
本地端口 8081
停止监听
清空接收区
实时发送
手动发送
清空发送区
数据接收区
十六进制显示
{"fver":"YED_DTU_1.0.3_luat_v0030_rda8910_float","iccid":"89860411102070058856","imei":"862167052964826","csq":22}00 (1)
00 (1)
00 (1)
00 (1)
862167052964826aaaaaaa_yinerdaend
自动发送(ms) 1000
数据发送区
十六进制发送

八、DTU 自动轮询实现 modbus 实例

以 RS485+TCP 自动轮询为例，TTL UART、RS232 是一样的。MQTT 或者阿里云也是一样的。

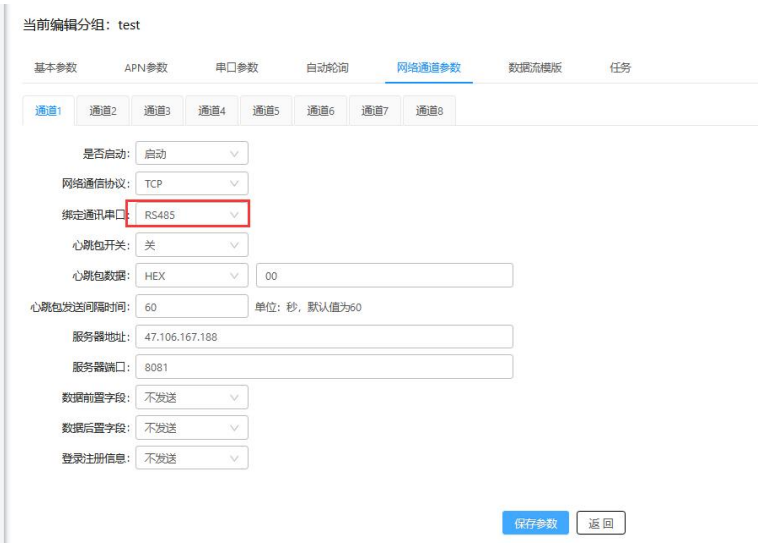
$启动 RS485

$启动 RS485 自动轮询

设置了应答超时为 500ms,周期 5000ms 和 3 条 16 进制的轮询命令

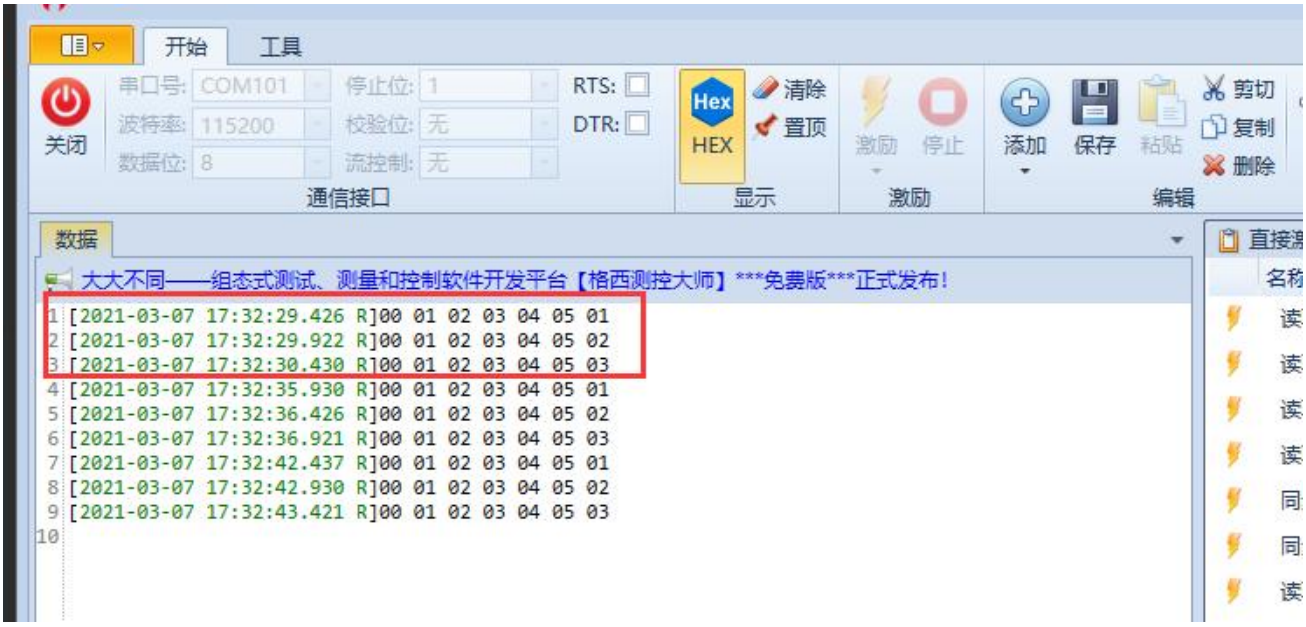
$设置 TCP 网络

绑定通信串口为 RS485，关闭心跳包。是否关闭心跳包，这个和服务器逻辑有关系。如果开了心跳包，服务器需要过滤心跳包。

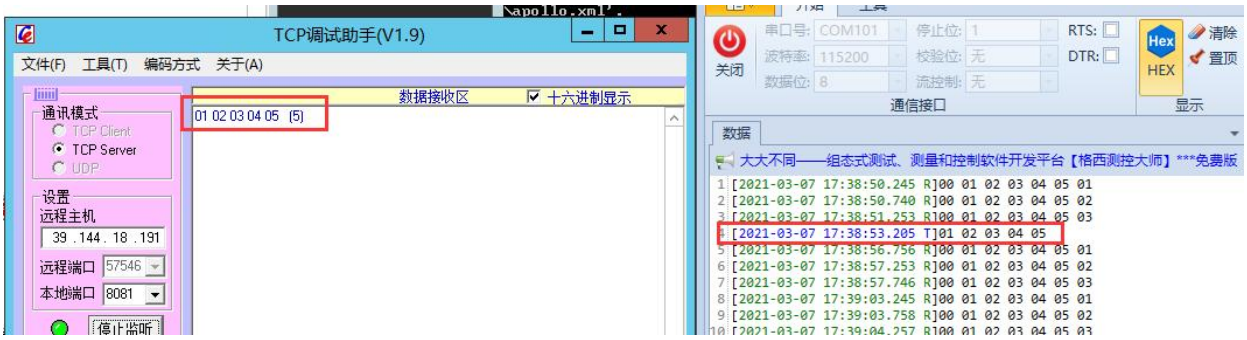


$观察串口轮询命令

可以观察到串口每隔 500ms 后发送 1 条命令，当连续发送 3 条数据后，停顿了 5 秒，继续发送第 1 条数据



$当 RS485 发送一条命令后，服务器会收到相应的数据

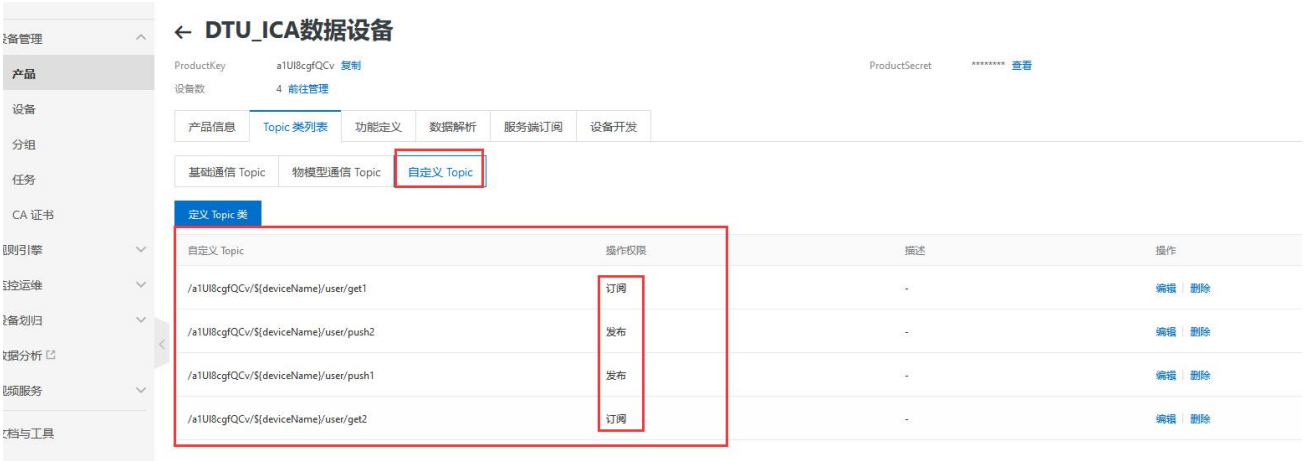


九、阿里云一型一密实例

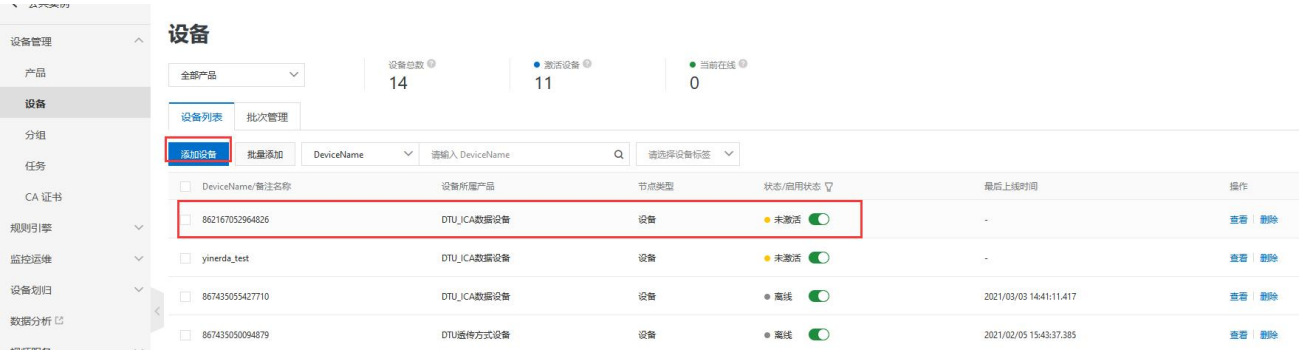
本例演示阿里云订阅和发布多个 topic 的方法。单个 topic 的方法更简单

MQTT 的连接和使用方法和这个类似。只是服务器变成自己的了。

$首先再阿里云 IOT 创建一个产品，并且开通动态注，然后添加一个自定义的 topic,并且添加 DTU IMEI 号相同的设备。



添加的 862167052964826 设备就是 DTU 的 IMIE，现在显示未激活



$设置 DTU

启动 RS232

设置 ProductKey 和 ProductSecret

地域代码和账号有关系，不要乱填，华东 2 是上海

填写订阅 topic 和发布 topic。需要把 topic 的${devicename} 替换成真实的设备名称，比如一型一密需要用${IMEI}替换

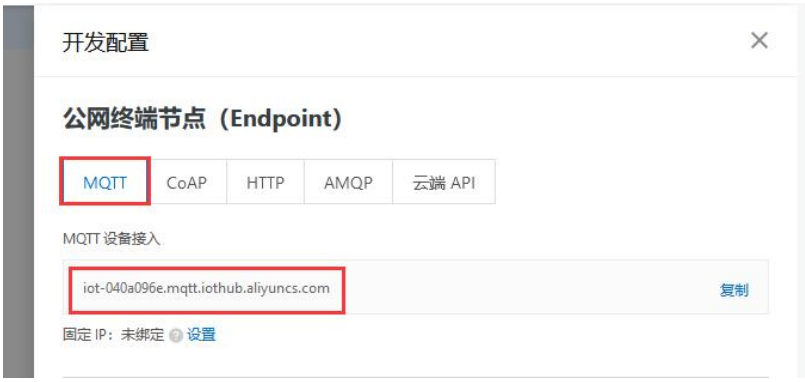
订阅的内容为:

/阿里 PK/${IMEI}/user/get1

/阿里 PK/${IMEI}/user/push1

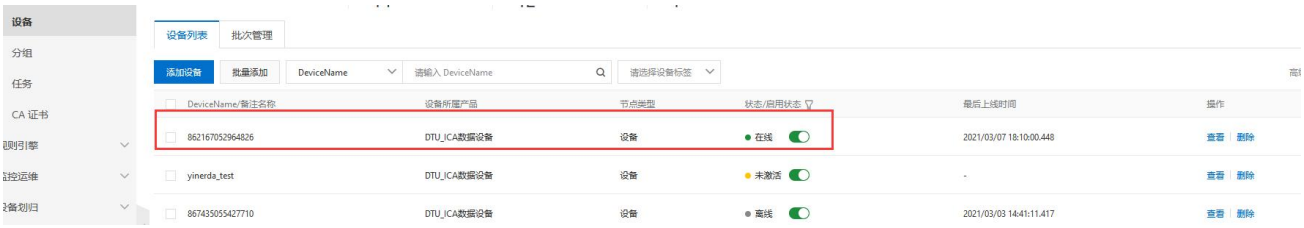
如果是企业版本和 2021 年 7 月 30 日后开通的公版实例，需要填写连接域名。

域名查找路径：连接实例->查看开发配置->MQTT 设备接入域名



$更新设备参数

当设备参数如果正确，阿里会显示设备激活

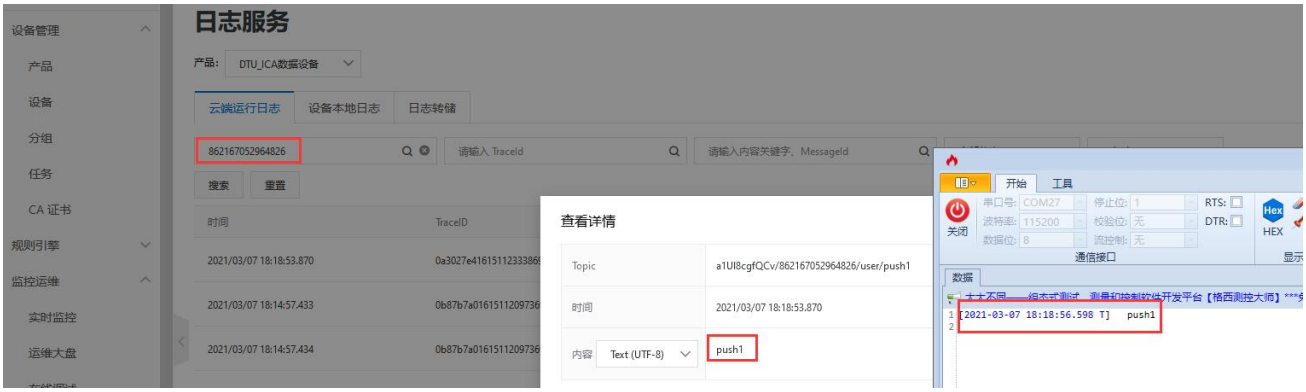


当 topic 订阅成功后，再设备的 topic 列表利可以看到订阅的 topic

$测试下发数据

阿里通过 get1 topic 下发数据"get1"给 DTU 的 RS232；

$测试 DTU 给阿里云发数据



到此，设备与阿里 IOT 就连接了。你要做的就是自己的服务器与阿里 IOT 对接。这个需要找阿里的技术支持或者看文档。

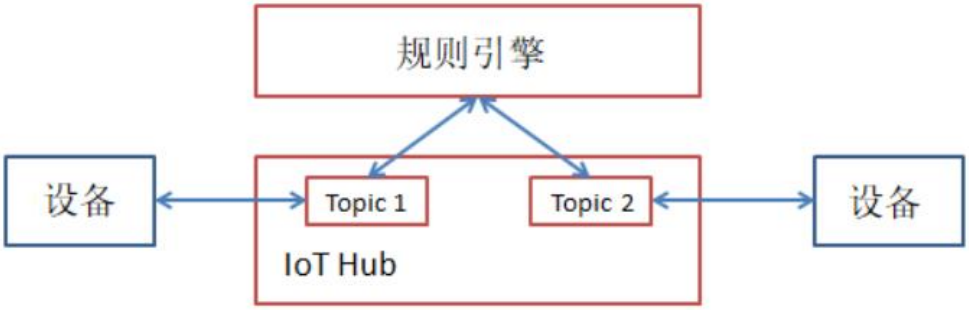
注意的关键点是：阿里添加设备和 DTU IMEI 号相同的设备，并且产品开通动态注册。

如果使用物模型 topic,串口数据必须要符合阿里物模型数据规则，服务器才会识别，模型数据参考阿里云的官方文档。

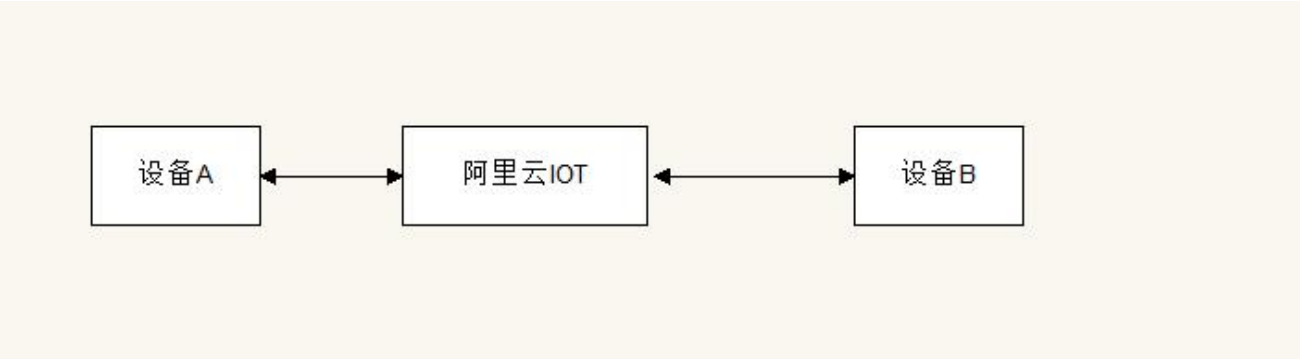
十、4G 与 4G 点对点通讯实例

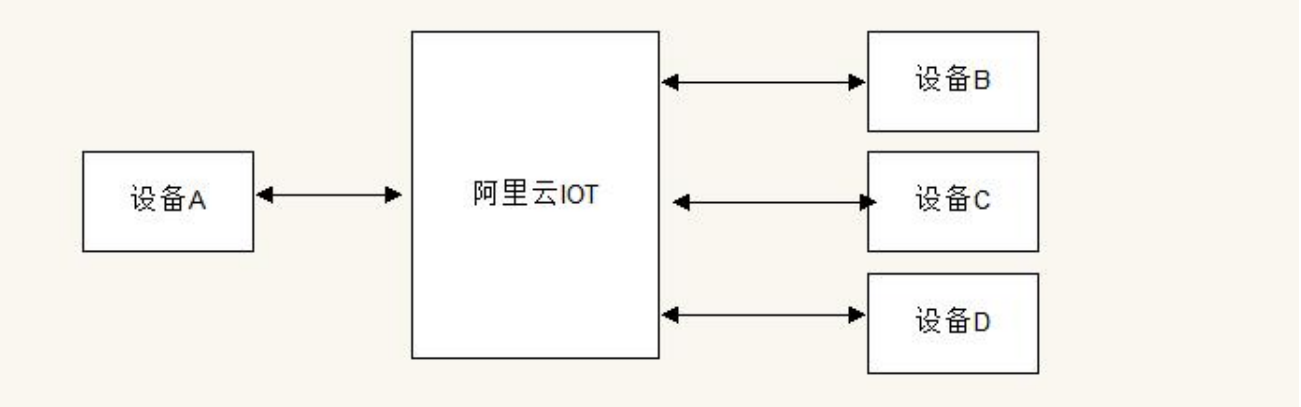
要实现设备与设备之间通讯，只需要在阿里 IOT 的基础上，添加一个云产品流转功能即可，阿里 IOT 的方法，参考本文的阿里 IOT 实例，下文主要讲解使用 IOT 的规则引擎实现数据转发功能。

实例框架:



实现功能:一对一和一对多





1) 、创建产品

创建产品按“阿里云 IOT 一型一密配置”讲解的创建产品和定义 topic,并且配置设备的参数，订阅 get1 topic 发布 pus1 topic

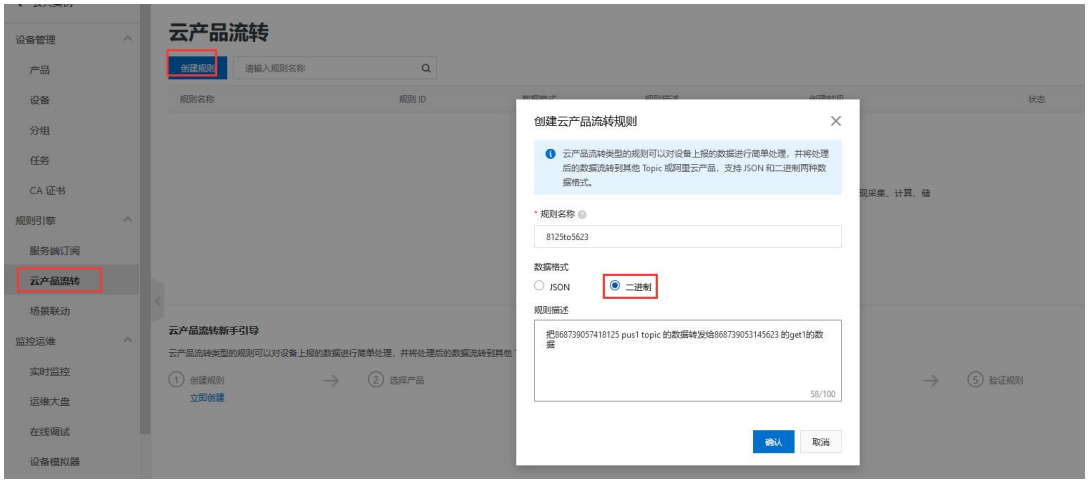
实现的功能是

把 868739057418125 pus1 topic 的数据转发给 868739053145623 的 get1 的数据；

把 868739053145623 pus1 topic 的数据转发给 868739057418125 的 get1 的数据

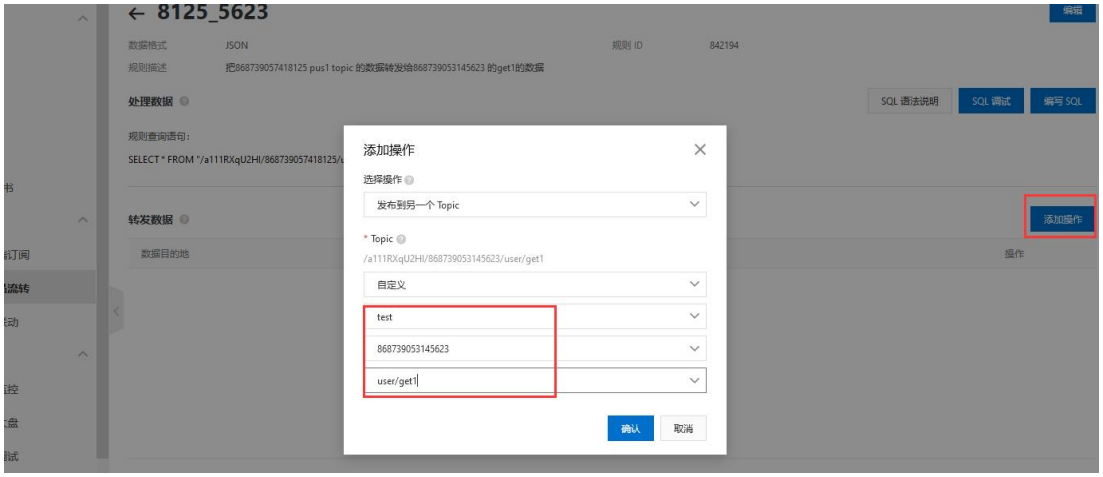
2)、创建云产品流转

把 868739057418125 pus1 topic 的数据转发给 868739053145623 的 get1 的数据

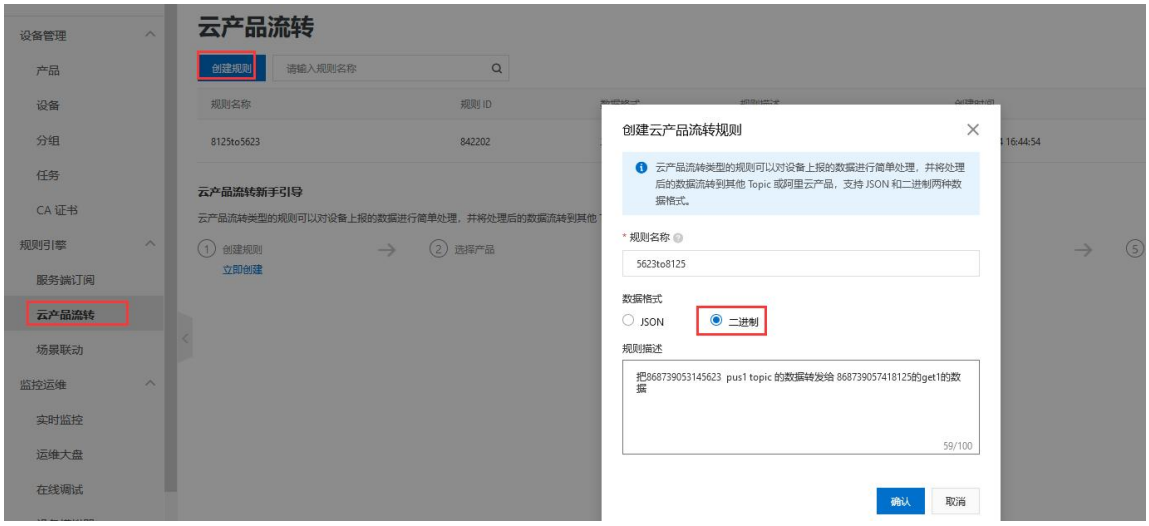


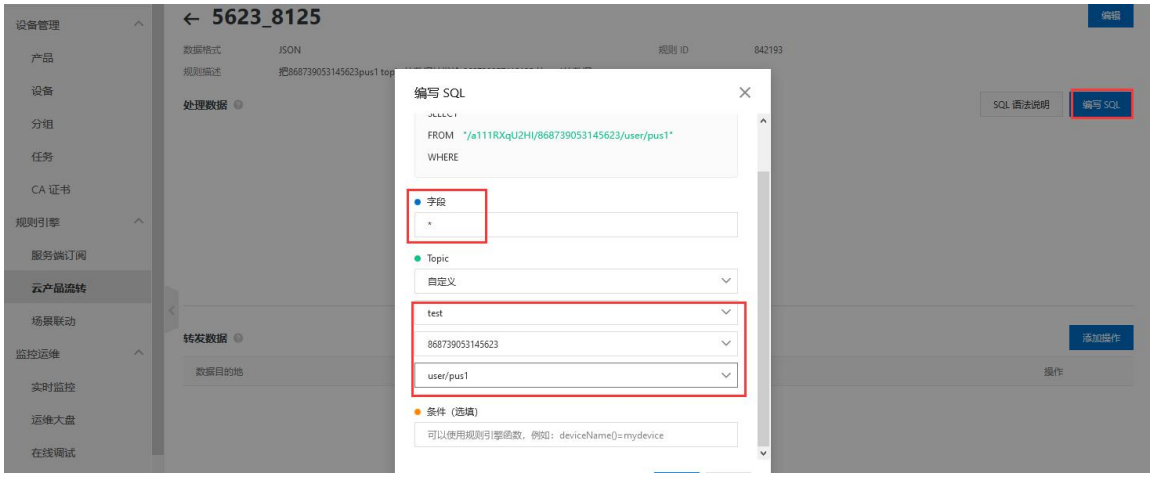
编写 SQL 如下

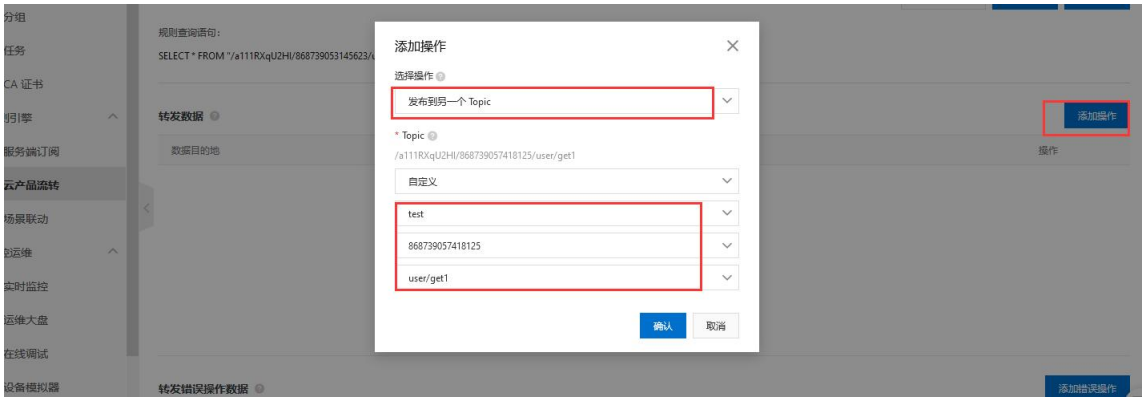
编写转发数据如下



把 868739053145623　pus1 topic 的数据转发给 868739057418125 的 get1 的数据

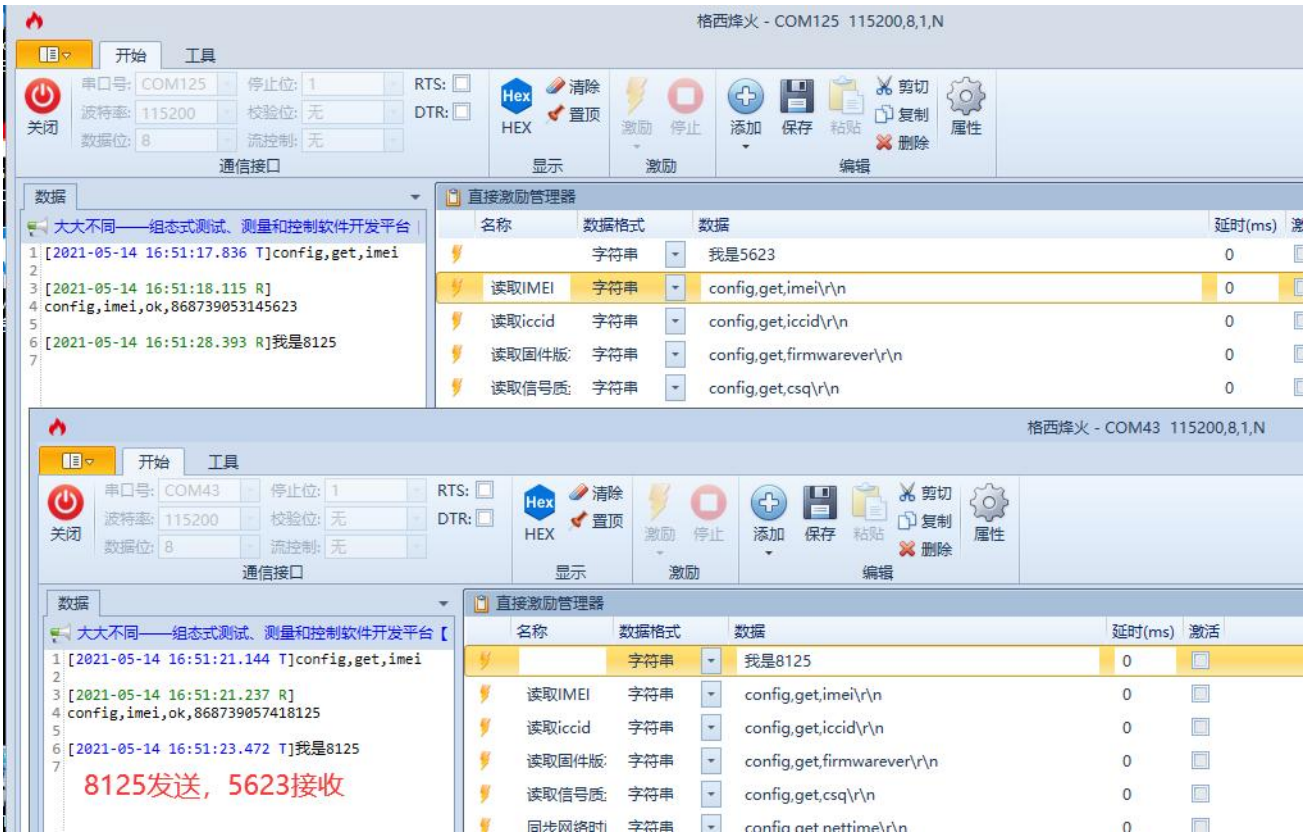


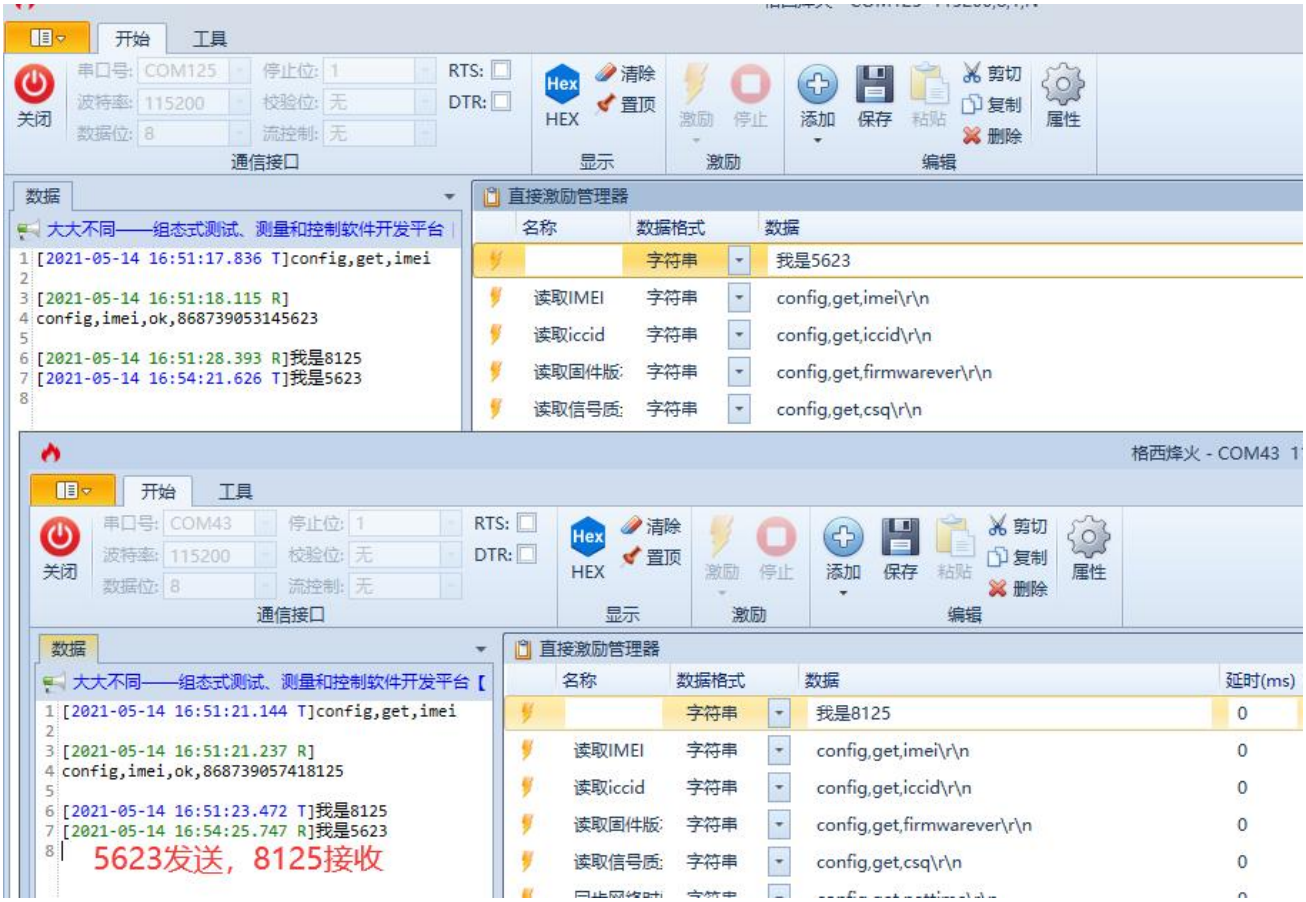




3) 、启动流转规则

4) 、确保 DTU 更新参数后，测试数据





5)、一对多通讯设置方法

如果使用 1 对多功能，就是主机的在设置转发数据的时候，多添加几个目标设备；从机在设置转发的的时候全部都指向主机即可。

十一、远程控制命令实例

远程控制命令是通过 DTU 连接的目标服务器，下发 config 控制命令控制设备，下发的命令为《银尔达-DTU 固件串口配置命令手册.pdf》描述的命名。可以实现：

1) DTU 设备的参数配置和修改、查询。比如读取 IMEI，ICCID，信号等
2) RTU 设备的输入查询，输出的控制和查询。比如 YED-RY1880 继电器输出和光耦输入检查
3) 进行远程参数修改

1、开启远程控制命令

2、配置网络通道

控制命令可以通过任何网络协议下发。收到控制命令后 DTU 解析控制命令，非控制命令，透传到对应串口。下图配置 TCP 协议连接。服务器参数换成自己的。

3、设备更新参数

保存参数后，重启设备，就会自动更新参数。重启后，等 30 秒后刷新设备列表，查看分组参数版本和设备参数版本是否相等，如果相等，表示参数更新成功，如果不相等说明参数更新失败。如果失败需要通过 LED 状态，检查网络状态。

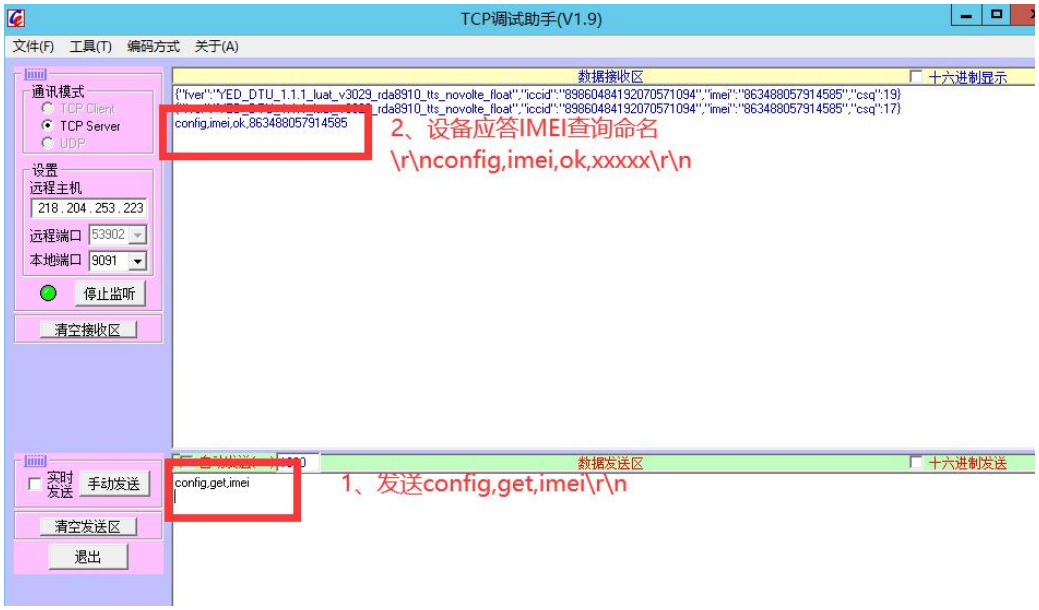
打开服务器，设备已经成功连上服务器，并且发送了注册包

4、服务器下发控制命令

服务器下发 config,get,imei\r\n（\r\n 用回车建代替），DTU 应答了服务器，表示远程命令成功，其他命令同理，比如查询 iccid,csq,控制继电器的命令 doout 和查询光耦命令 diin 等。

下发命令的时候注意事项:

- 服务器下发的支持的命令和串口配置命令相同
- 确认 DTU 开启了远程控制命令
- 有的工具无法发送\r\n 回车换行，工具把\r\n 当成了 \ r \ n 4 个字符，而不是\r\n 2 个字符串，可能需要编写代码发送才行（阿里 IOT 测试工具和 MQTT.fx 测试工具都是这样）
- 不响应的命令都会透传到串口，可以用电脑 16 进制模式和串口命令对比一下判断服务器发的原始数据



十二、GPIO 功能实例

GPIO 功能只有具有输入输出的设备才有，比如 YED-RY1880，YED-RY1210。可以实现输出状态的保持，输入状态的周期上报和状态变化预警联动功能。

GPIO 功能只有在 web 配置，串口没有开放对应命令。

输出控制:可以设置设备重启后，继电器输出的状态是默认全部关闭，还是保持重启之前的继电器状态。

输入电平周期上报:可以设置设备按设置时间，定时上报光耦的输入状态，注意光耦默状态默认为 1，外部触发光耦导通后，状态为 0。

输入电平检查预警：检查光耦输入电平重一个状态变化到另一个状态后，主动给服务器上报设置的内容，服务器可以通过内容及时处理数据。还可以设置本地联动，控制本地继电器输出。

1、配置网络通道

控制命令可以通过任何网络协议下发。收到控制命令后 DTU 解析控制命令，非控制命令，透传到对应串口。下图配置 TCP 协议连接。服务器参数换成自己的。

2、配置 GPIO

3、设备更新参数

保存参数后，重启设备，就会自动更新参数。重启后，等 30 秒后刷新设备列表，查看分组参数版本和设备参数版本是否相等，如果相等，表示参数更新成功，如果不相等说明参数更新失败。如果失败需要通过 LED 状态，检查网络状态。

7	YED-D724W_W1_G724W	863488057891155	D724W测试	9	9	2021-07-23 14:34:56	详情/删除
8	YED-DG724W	863488057914502	YED-DG724W测试	6	6	2021-08-03 14:22:24	详情/删除
9	YED-DG724W	863488057914452	YED-DG724W测试	6	6	2021-08-03 14:38:22	详情/删除
10	YED-RY1880	863488057914585	RY1880Test	14	14	2021-08-05 10:16:39	详情/删除

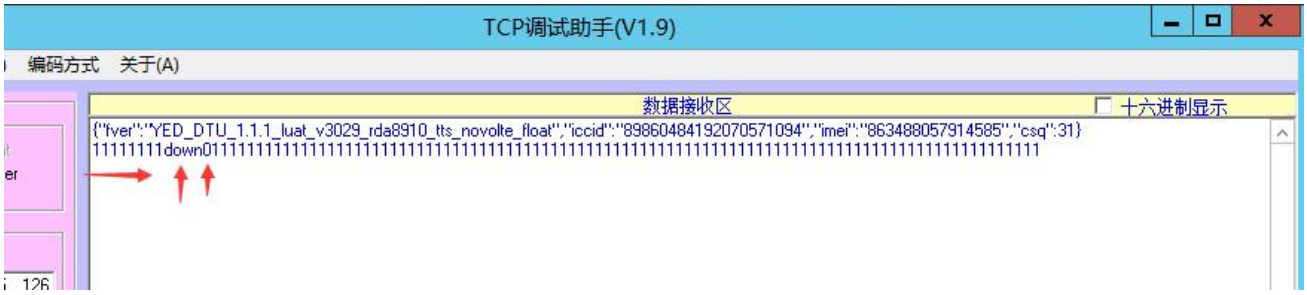
当前设备总数：10

当更新参数后，DTU 每 5 秒给服务器发送输入状态

1,1 分别表示输入通道 1 和 2 都没触发状态。

down 表示输入 1 被触发后，上传给服务器的数据,这时候也控制了继电器。

0,1 表示输入通道 1 触发，输入通道 2 没有触发。


TCP调试助手(V1.9)
编码方式
关于(A)
数据接收区
十六进制显示
{"fver":"YED_DTU_1.1.1_luat_v3029_rda8910_tts_novolte_float","iccid":"89860484192070571094","imei":"863488057914585","csq":31}
11111111down01111111111

十三、定位实例

目前 Air724 系列支持基站定位，Air820 系列支持基站和 GPS 定位，N58 系列不支持，定位功能主要周期更新位置信息，可以把更新的位置信息直接传送 给服务器，也可以通过变量的方式，在 TCP、UDP 心跳包里面引用，然后上报给服务器。支持的变量参考“**变量引用说明**”。

1、配置网络通道

2、设置定位信息

下图设置是已固定格式把基站定位信息，通过网络通道 1，上次给服务器。

3、设备更新参数

保存参数后，重启设备，就会自动更新参数。重启后，等 30 秒后刷新设备列表，查看分组参数版本和设备参数版本是否相等，如果相等，表示参数更新成功，如果不相等说明参数更新失败。如果失败需要通过 LED 状态，检查网络状态。

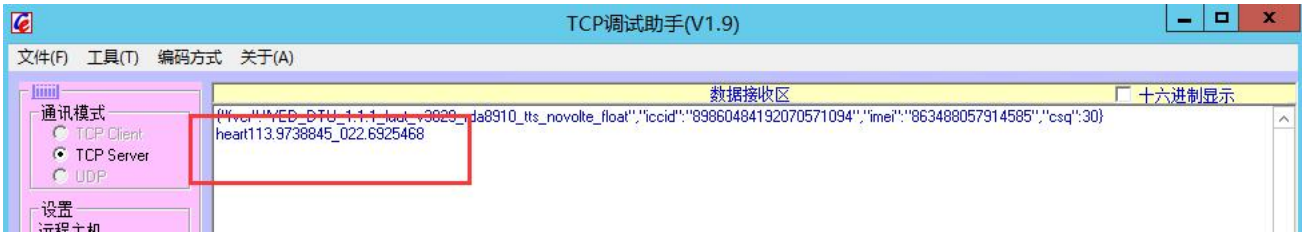
下图的固定格式信息为“113.9739972_022.6938968”其中 113.9739972 是经度 022.6938968 是纬度

4、把定位设置成不上报

5、在 TCP 心跳包里面引用定位信息

引用 GPS 定位信息，只能在字符串模式下引用

设备更新参数后，会以心跳包的方式，把定位信息上传给服务器



十四、数据流模板实例

本节对应视频教程:https://www.bilibili.com/video/BV1364y117zc?p=19

更多的 API 调用，查看《银尔达-Air724 数据模板和任务支持的 API 文档》和《银尔达-DTU 数据模板格式说明及实例.txt》

数据模板和任务配合的高级教程对应 P19~P25 可以根据视频教程观察是满足自己的需求；

视频对应的示例源码在 WIKI 和 QQ 群都有下载，直接拷贝测试即可。如果有需要可以联系商务代写。

数据流模板，是一段动态运行的代码，可以对需要发送的数据最添加、转换、过滤处理。

比如串口接受到的是 hex 数据，服务器需要的是 json 数据。就可以在**发送数据模板**里面，把串口数据转换成服务器的数据格式，然后再上传给服务器，同理可以在**接收数据模板**里面把服务器的数据转换成设备需要的数据，然后通过串口传给设备。

比如需要在串口原始数据前面添加自己的设备 ID，可以在数据模板完成。

比如平常设备发的是一些无效数据，不需要上传给服务器，需要过滤。

1、使用前提

Air724 系列 DTU 使用数据模板，是一段动态运行的 lua 代码，所以需要会 lua 语法。

如果数据模板里面出错了，会导致程序崩溃，所以需要确认自己知道在做什么。

2、数据模板格式

以 function 开头　end 结尾　中间是一串可执行的 lua 代码。

function　end 是固定格式不可修改。

...是原始数据。

return 的数据是处理完，需要发送的数据，如果 return “” 或者 nil 表示不发送数据。

如下模板:在原始数据前面添加一个字符串“send prot”,最终发出去的数据就是 send prot+原始数据。

```
function
    local str=...
    log.info("send prot")
    return "send prot"..str
end
```

3、发送数据流模板

发送模板的数据流程是　串口数据--->发送数据流模板-->服务器。原始数据是串口接收的数据

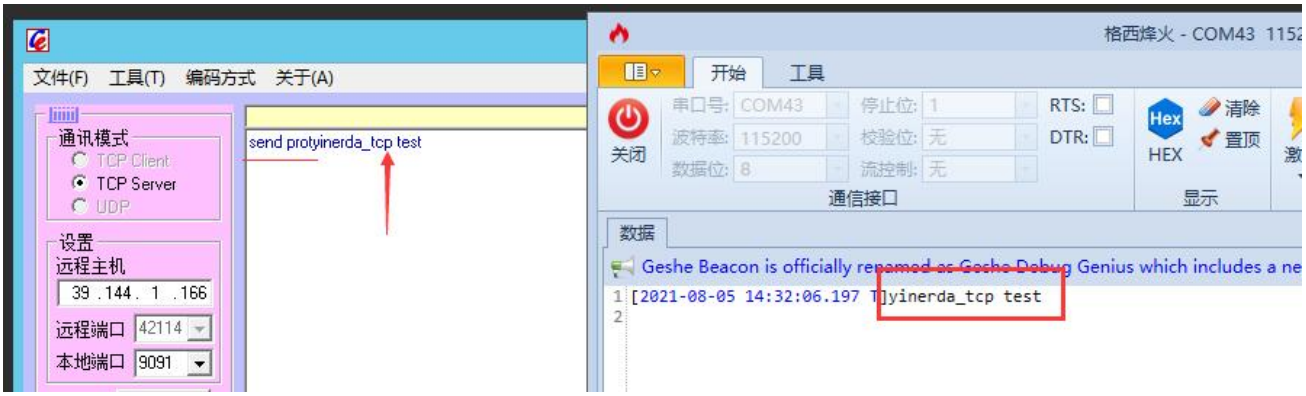
4、接收数据流模板

接收模板的数据流程是　服务器--->发送数据流模板-->串口数据。原始数据是网络接收的数据

5、启动网络通道 1 的数据流模板

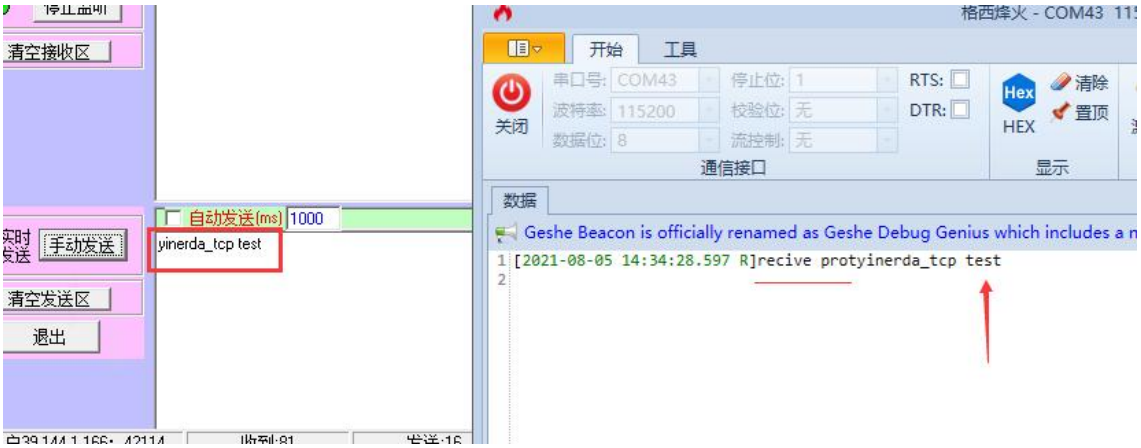
6、串口发数据服务器收到的情况

串口发送“yinerda_tcp test” 通过发送数据流模板 添加了 send prot,所以服务器收到了 “send protyinerda_tcp test”



7、服务器发送数据串口收到的情况

服务器发送“yinerda_tcp test” 通过接收数据流模板 添加了 recive prot,所以串口收到了 “recive protyinerda_tcp test”



十五、DTU 固件远程升级实例

提示:批量生产测试稳定，满足需求，请谨慎升级。

新购买设备、体验新功能，解决 BUG，建议升级。

如果网络异常导致固件失败，不会影响之前功能，模块会回滚到之前版本。

目前 Air724 系列模块支持，N58 系列不支持。

1、首先确认 DTU 固件版本

如果之前有通过 WEB 配置参数，设备获取过参数后，在**设备管理-设备列表-详情**可以查看到程序版本。

2、开启 DTU 固件更新

开启 DTU 固件更新后，保存参数。分组参数版本从 24 变成 25

3、重启设备，自动更新参数

固件更新根据网络情况，有延迟，升级后会自动重启这个过程可能会有 1~5 分钟，也可能升级失败。

4、再次更新参数

再次点击保存参数，让参数版本更新，重启设备，设备再次更新参数。在设备列表详情查询，固件从 1.0.6 变成了 1.1.0,表示固件更新成功

5、当需要的设备升级成功后，关闭固件升级功能，防止意外升级，影响设备。

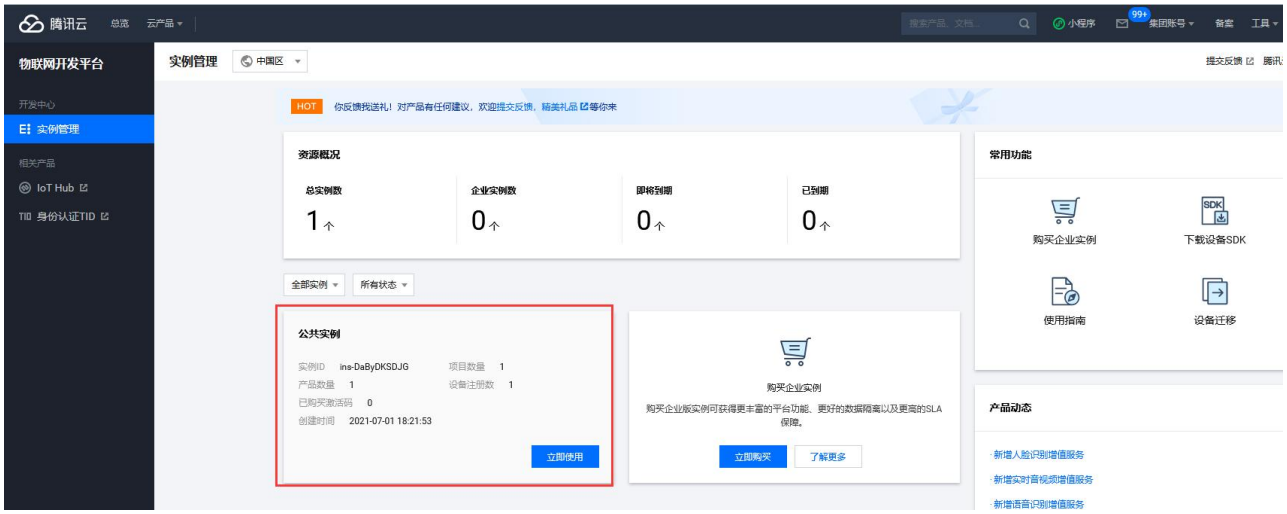
十六、腾讯 IOT 连接实例

本文档记录银尔达 DTU 固件连接腾讯物联网开发平台(IoT Explorer)。支持使用 DTU IMEI 作为设备名称，密钥认证的方式，自动注册腾讯 IOT，实现云端和 DTU 之间数据透传功能。

DTU 适合使用透传功能的场景。支持系统透传 topic 和自定义透传 topic。

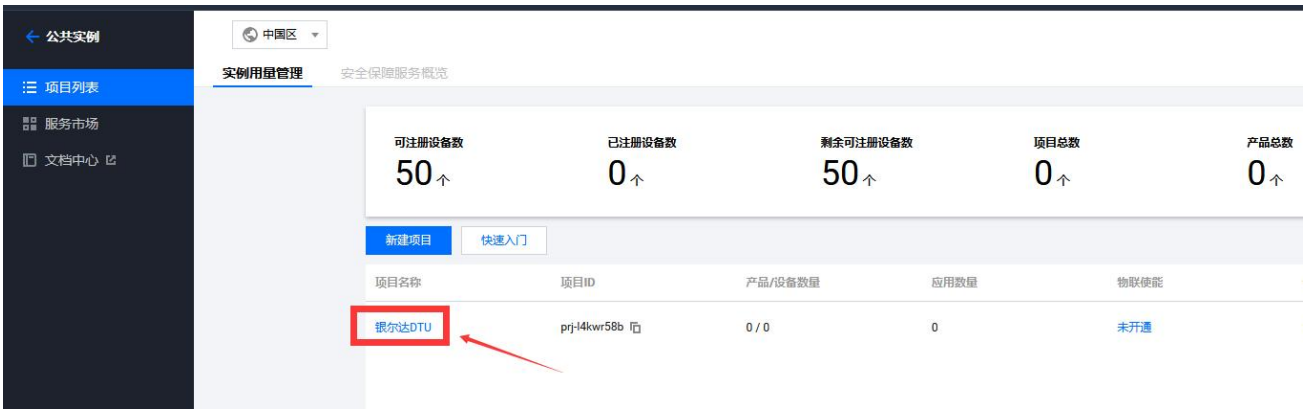
本文档时候 DTU 透传固件 V1.2.5 以上版本的固件支持。

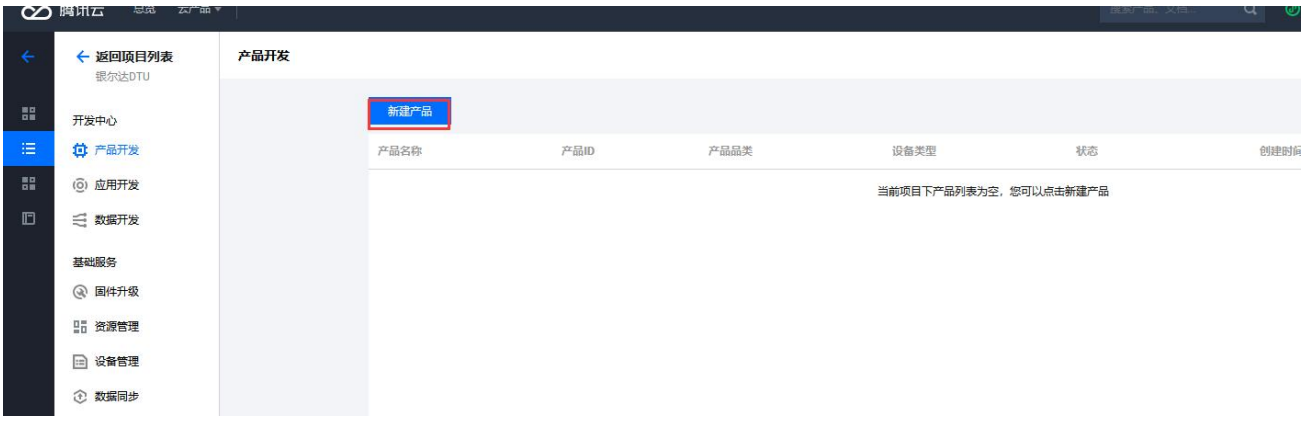
1) 、公共实例创建项目



2)、创建项目和产品

进入项目，创建产品，创建一个密匙认证，2G/3G/4G 通信，自定义透传的产品





3)、获取 ProductID 和 ProductSecret

设置产品，打开动态注册配置，获取 ProductID 和 ProductSecret，开启自动创建设备。

4) 、查询可以订阅和发布的 topic

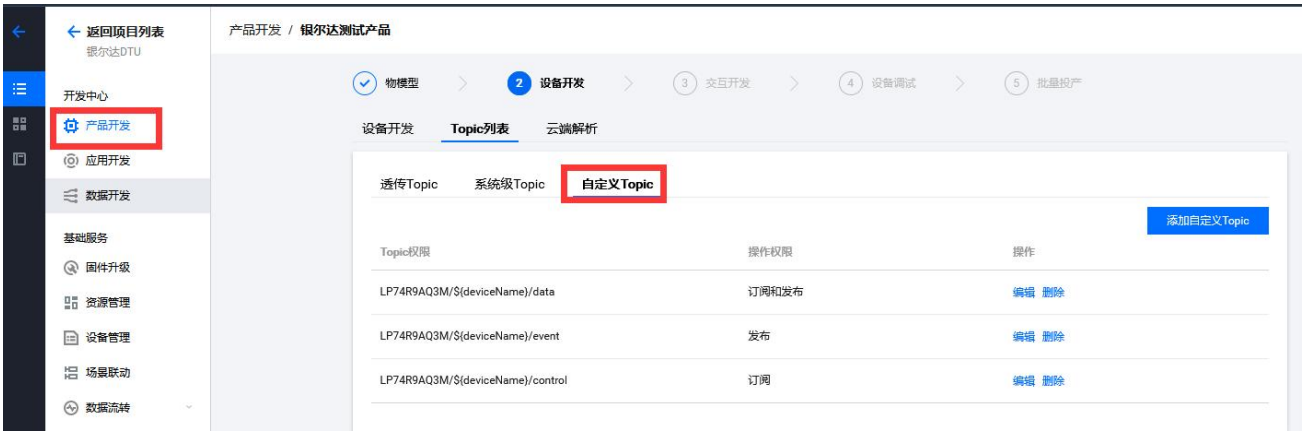
透传 topic 与自定义 topic 的区别在于，透传 topic 可以支持腾讯的物模型解析。

自定义 topic，不能使用物模型，相当于利用腾讯的透传属性，只做一个中转到自己的业务服务器。

DTU 测试使用自定义透传 topic 测试。

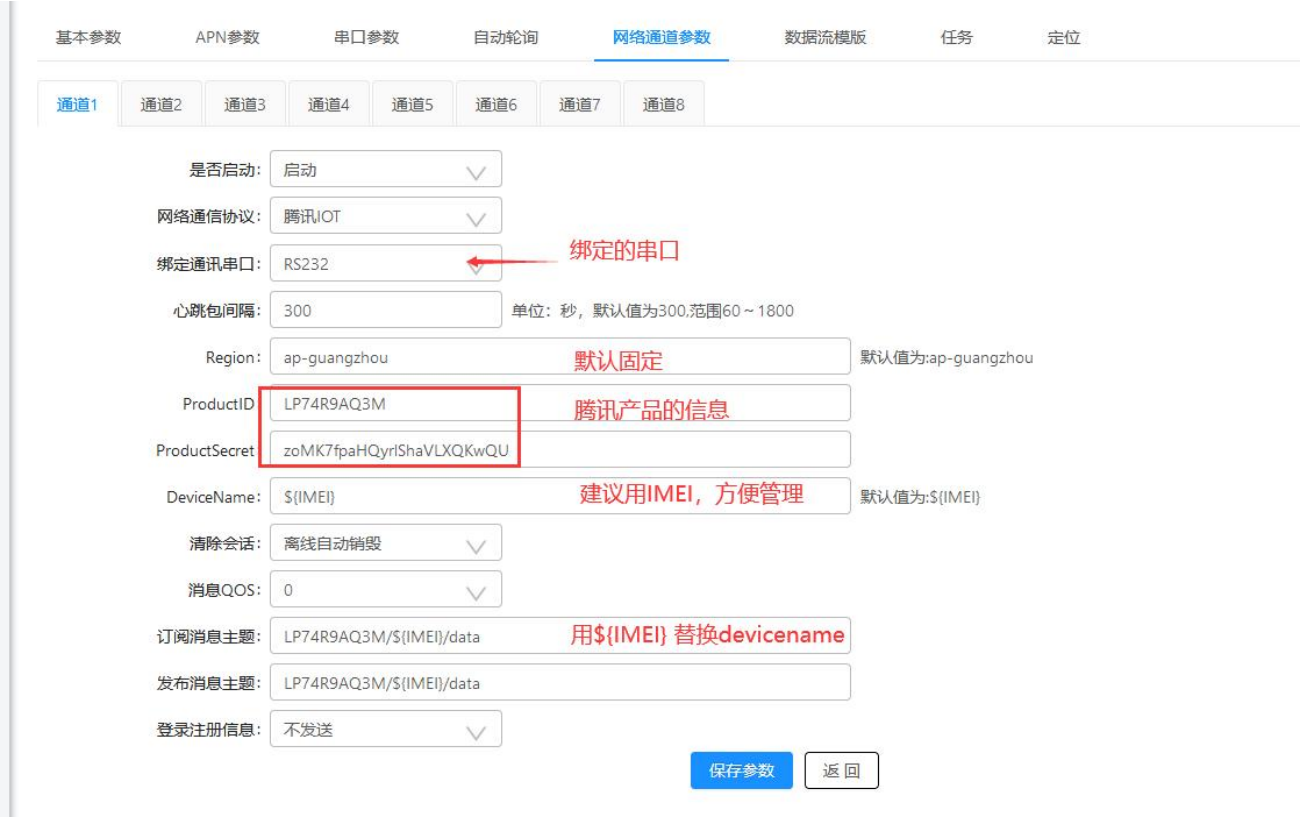
Topic 中的${deviceName} 一般替换成真实的名字。在 DTU 里面可以使用${IMEI}自动替代模块的 IMEI。

比如 LP74R9AQ3M/${deviceName}/data，修改成 LP74R9AQ3M/${IMEI}/data。



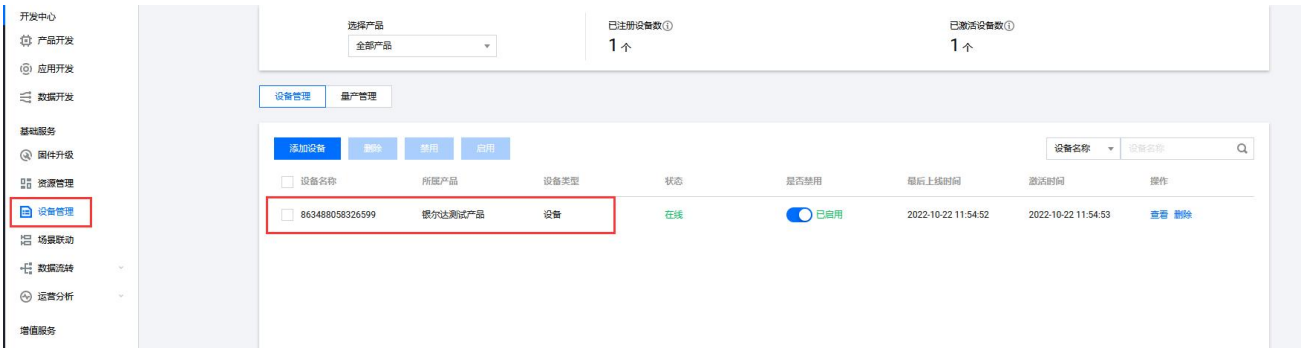
5) 、设置 DTU 参数

使用 WEB 和串口工具都可以设置参数。这里只讲解网络通道参数配置，其他参数参考“银尔达-DTU 固件 Web 配置用户手册”第 5 章去配置



保存参数后重启设备，更新参数。

如果设备参数正常，网络正常，腾讯云就会自动创建和注册设备，并且提示设备在线。

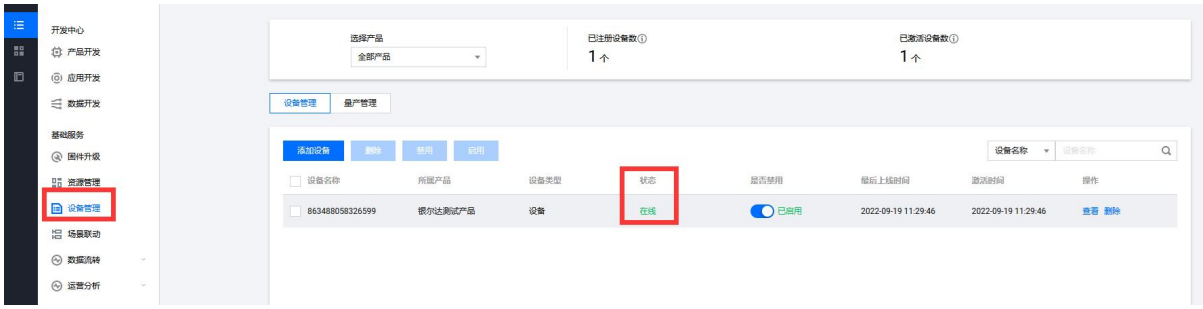


注意:腾讯 IOT 设备只能激活一次，激活的设备不能按 reload 按键恢复出厂设置，会导致内部保存的数据失效。如果回复出厂设置了要删除腾讯的设备，重新添加。

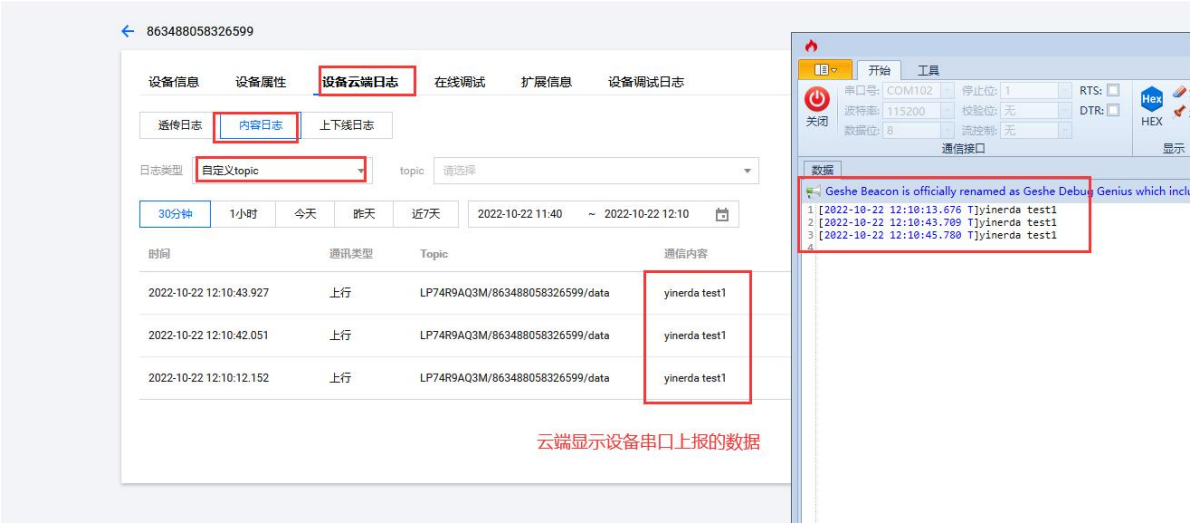
6)、调试设备

腾讯 IOT 需要操作 API，才能发送数据，没用工具单独对设备订阅的 topic 下发数据。这里只是利用腾讯的规则引擎简单的测试腾讯的数据流，详细的使用咨询腾讯 IOT 的技术人员。

DTU 重启更新参数后，连接成功，会显示设备在线

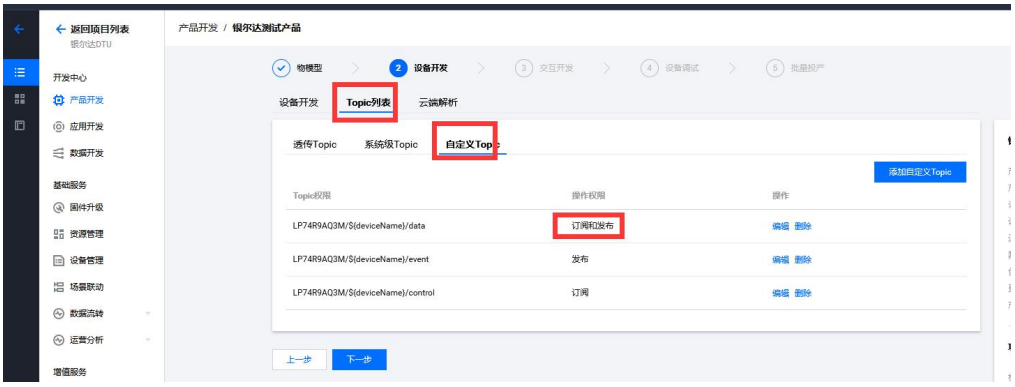


设备上报数据到腾讯云



腾讯云下发数据到 DTU

腾讯 IOT 自定义 topic 没有下发的测试界面，必须使用 API 发送。如果要测试，可以让设备订阅和发布同时具有发布和订阅的 topic，,然后通过规则引擎转发，实现自发自收。

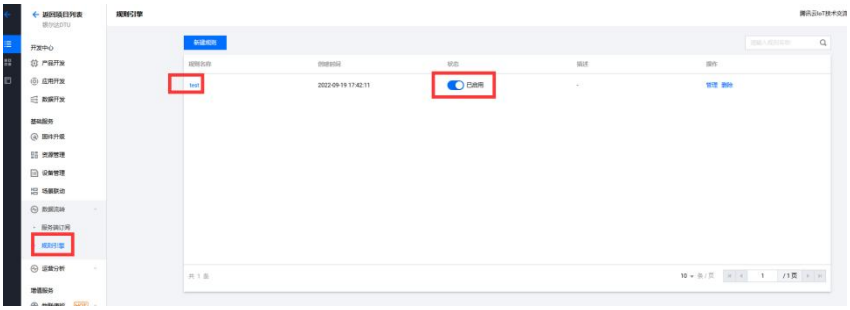


在数据流转->规则引擎 里面创建规则引擎

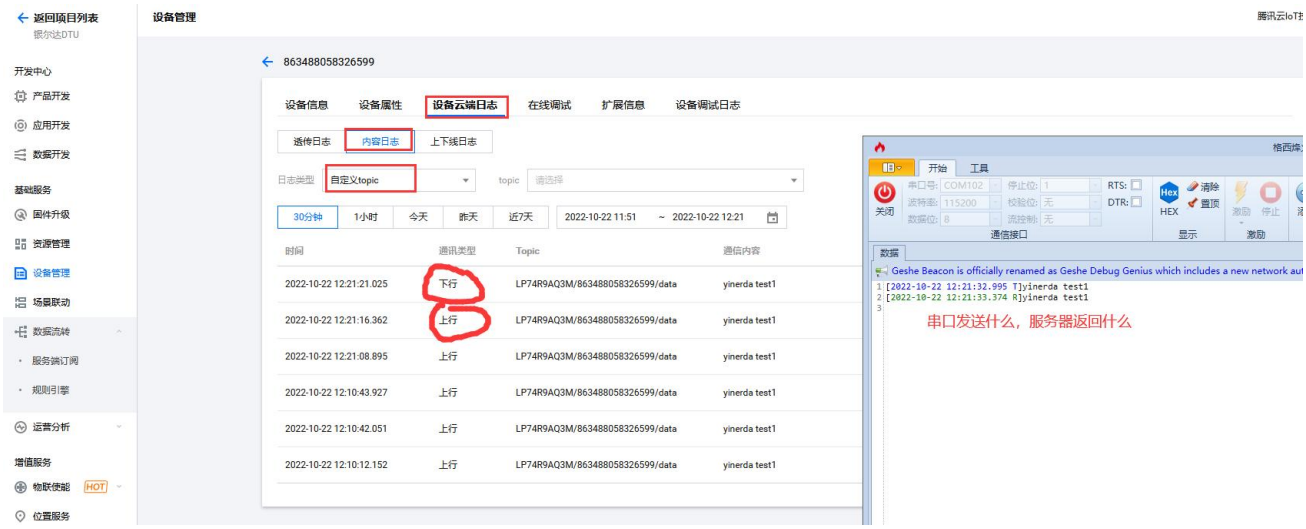
编写数据筛选条件

编写转发的设备

启动规则引擎



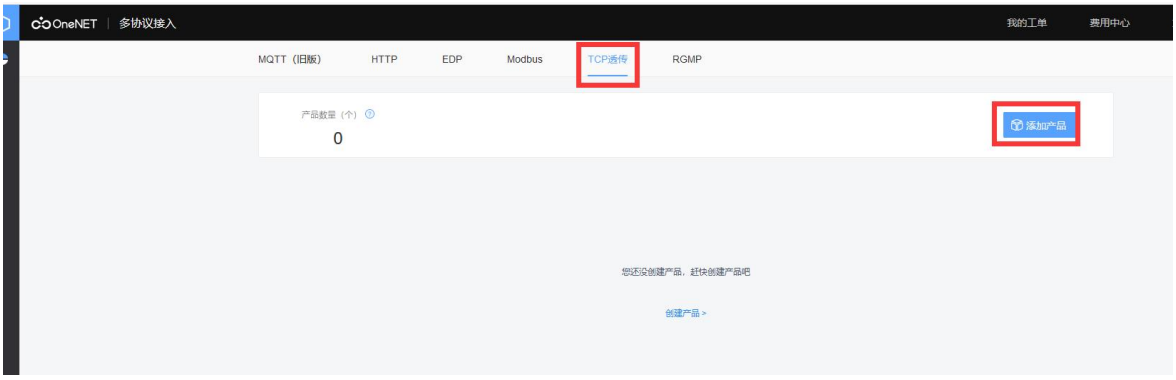
串口发送数据，服务器返回数据



DTU 只是透传设备，具体腾讯业务使用，请咨询腾讯的相关人员。

十七、移动 Onenet 多协议接入-TCP 连接实例

1) 、创建产品



2) 、记录设备注册码

DTU 固件通过产品 ID 和设备注册码，可以自动通过设备的 IMEI 为编码注册设备。这样就不需要提前在平台注册设备。

3) 、编写解析脚本

脚本是服务器解析透传数据的代码，必须是 lua 代码，限制规则查看平台说明。

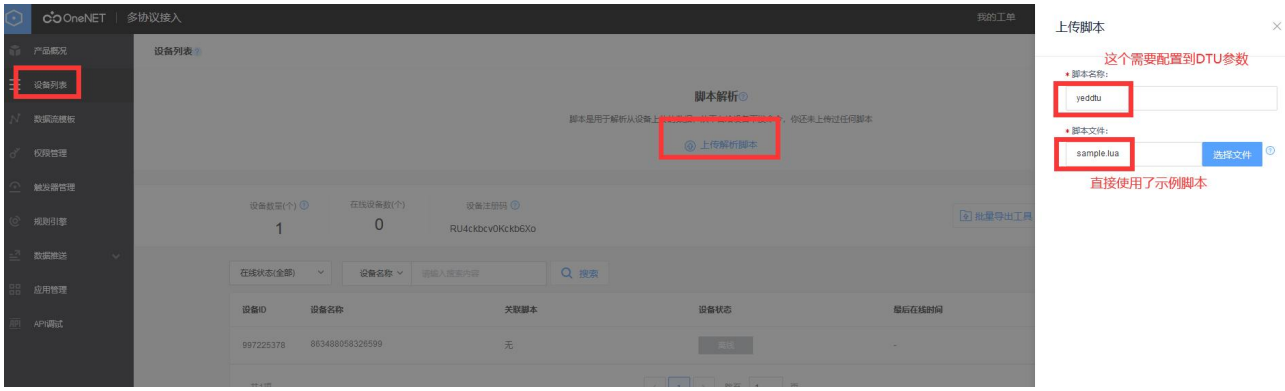
比如设备上报的是 hex 数据，业务服务器要的是 json 数据。那么脚本就需要 hex 与 json 之间转换。

在脚本示例里面以 function device_data_analyze(dev) 函数入口，解析设备的数据和发送数据

函数进入的是设备的数据，中间可以应答设备，函数返回的是发给业务服务器的数据。

文档地址:https://open.iot.10086.cn/doc/v5/develop/detail/495

这里直接加载了官方的 demo “sample.lua”，名字自己命名这里命名的是”yeddtu” 需要填写在配置参数里面。



4) 、添加数据流模板

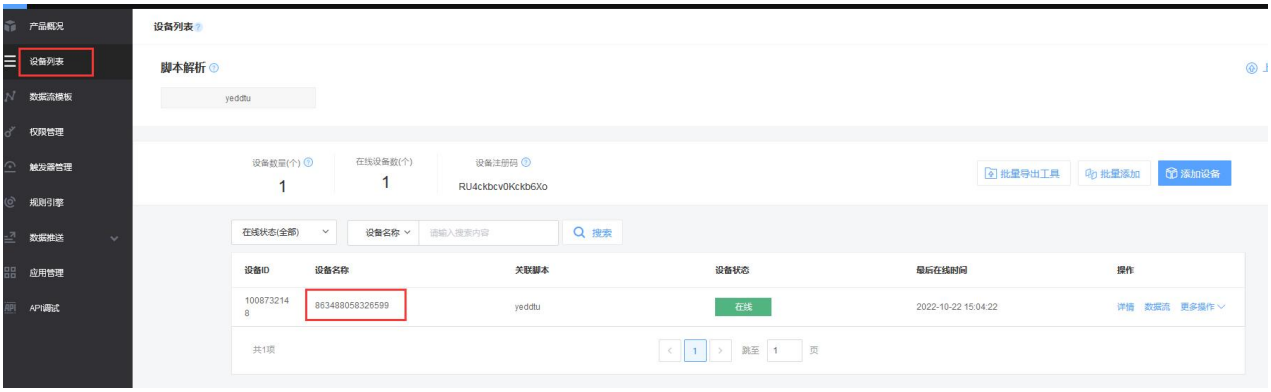
数据流名称，应该就是数据点，这个是与模板定义的名字一样，用来显示你解析的数据。

```
73  ----------------------------------------------------------------------
74  function device_data_analyze(dev)
75      local t={}
76      local a=0
77
78      -- 添加用户自定义代码 --
79      -- 例如:  --
80      local s = dev:size()
81      add_val(t,"ds_test",a,dev:bytes(1,s))
82      add_val(t,"ds_test",a,dev:bytes(1,s),"dev1")
83      add_val(t,"ds_test",a,dev:bytes(1,s),"dev2")
84      dev:response()
85      dev:send("received")
86
87      -- return $1,$2 --
88      -- 例如:  --
89      return s,to_json(t)
90  end
```

5)、配置参数

这里只讲解网络通道的参数配置，其他参数，参看《银尔达-DTU 固件 Web 配置用户手册》第 5 章

保持参数后，重启设备，参数更新，如果参数和网络正常平台会自动创建一个设备，并且在线。

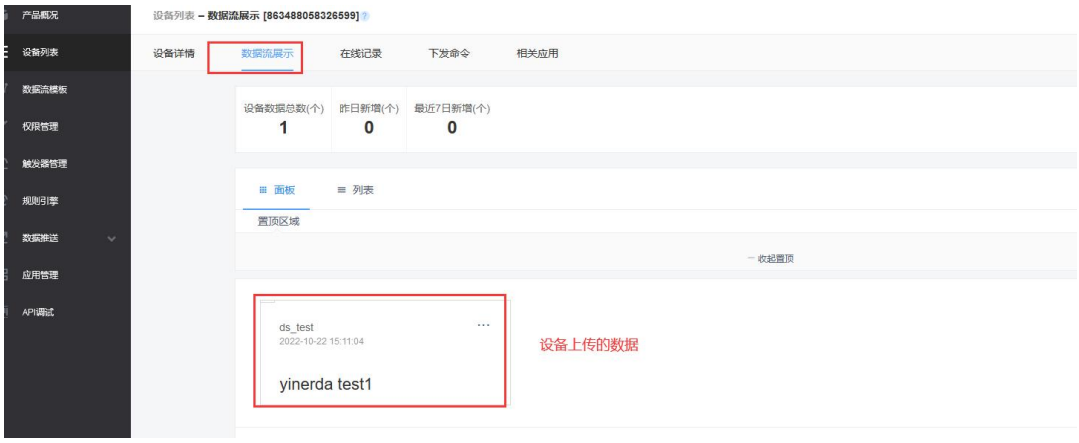


6) 、调试设备上报和下发数据

官方的脚本会自动定时下发”test” 命令，DTU 串口就会打印对应的命令。

当 DTU 串口发送数据给服务器后，服务器会返回“received”

在数据流展示中，就能看到 DTU 上传的数据

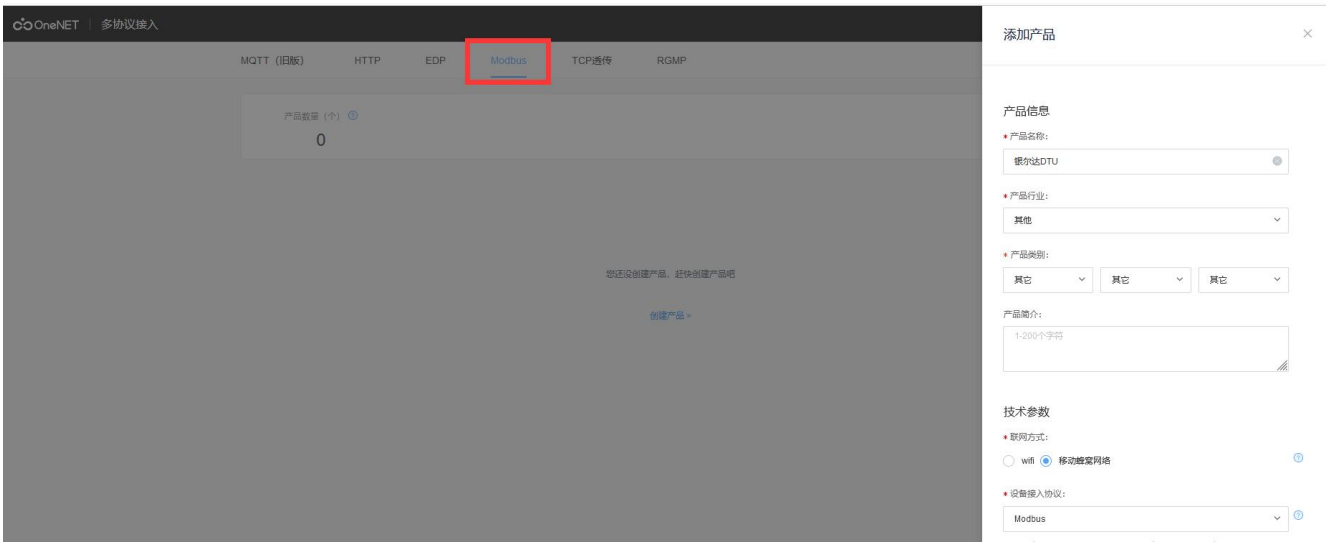


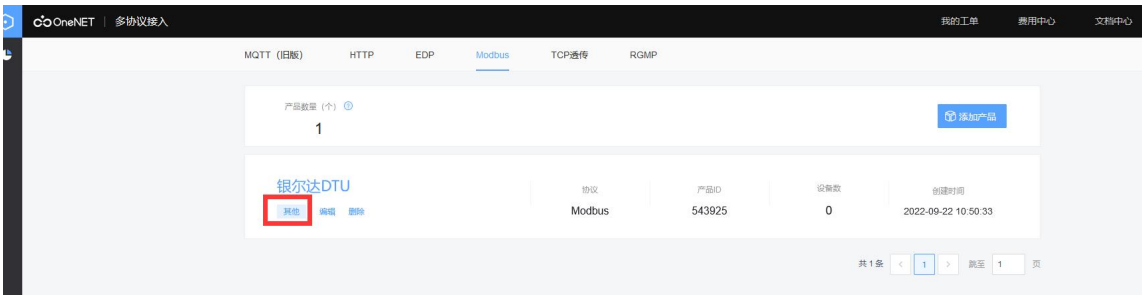
DTU 连接过程就完成了，关于移动云的其他业务开发和 API 开发，请咨询移动技术人员。

十八、移动 Onenet 多协议接入-MODBUS 连接实例

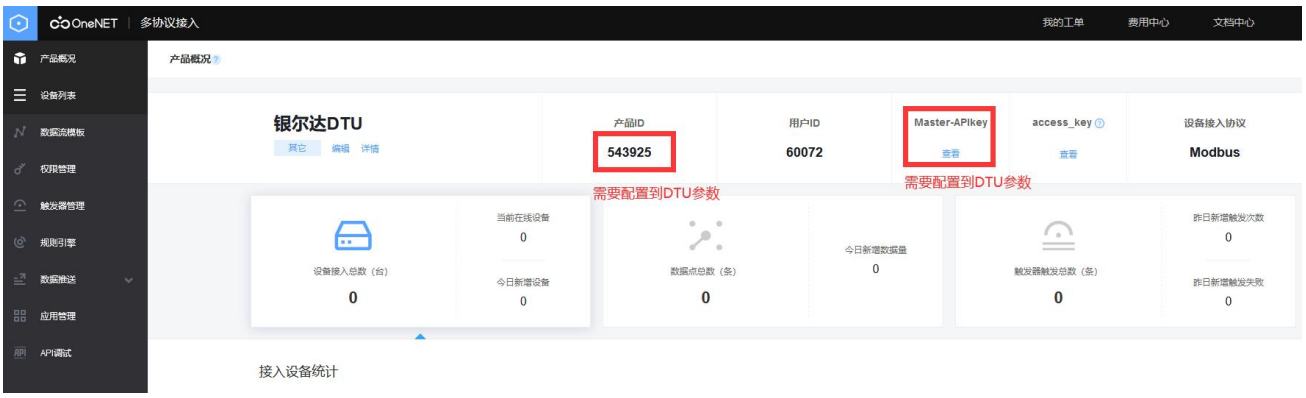
OneNET 支持的 Modbus 协议是基于 TCP 连接，即 Modbus over TCP,OneNET 作为主机，将数据封装在 TCP 的数据中进行收发数据，利用 DTU 实现的简单的透传能力，可以实现总线设备与平台的 Modbus 协议通信，可以广泛应用到使用 Modbus 协议的多种行业中。

1) 、创建产品





2) 、记录产品 ID 和 Master-APIkey



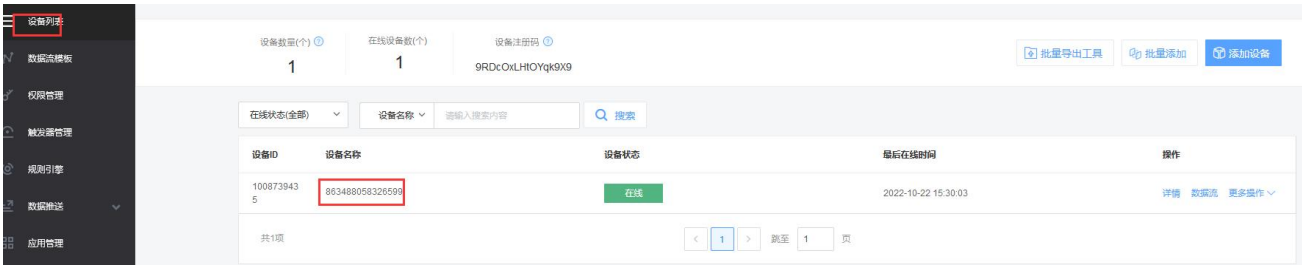
3)、添加设备

DTU 可以填写参数后自动注册设备，建议用 DTU 的 IMEI 去做，IMEI 是固定 16 位，前面 4 位基本是固定的，IMEI 的后 11 位作为序列号，能保证唯一，后 8 位作为密码。基本是够用了。

4)、配置参数

这里只讲解网络通道的参数配置，其他参数，参看《银尔达-DTU 固件 Web 配置用户手册》第 5 章。

保持参数后，重启设备，参数更新，如果参数和网络正常平台会自动创建一个设备，并且在线。



5)、调试设备上报和下发数据

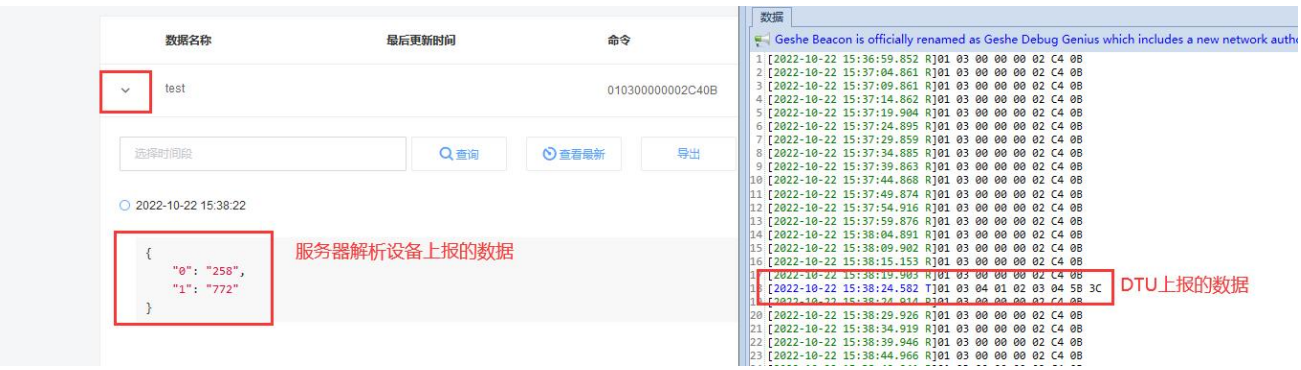
创建 modbus 轮询命令，onenet 作为主机就会自动发送轮询数据。当设备返回对应的结果后，服务器会自动对应并且解析数据。

创建数据流采集，填写 modbus 采集命令

根据自动实际设备去填写

设备的串口就能周期 5 秒收到服务器轮询数据

当设备在轮询命令的时候应答正确的命令，服务器会显示解析的数据。比如”01 03 04 01 02 03 04 5B 3C”

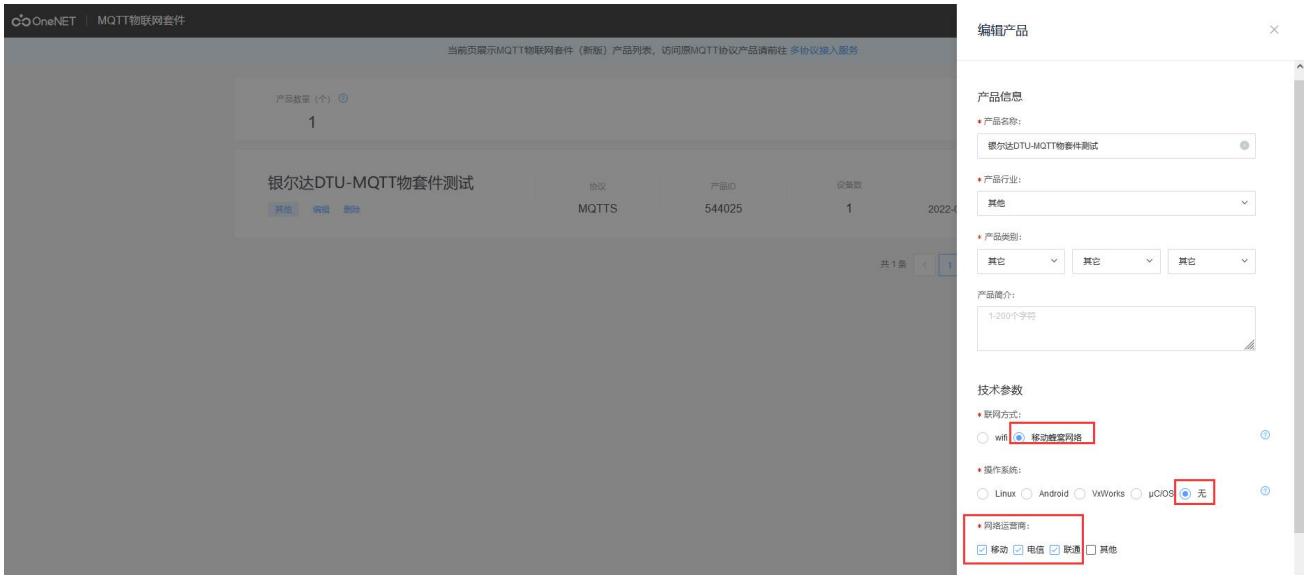


DTU 连接过程就完成了，关于移动云的其他业务开发和 API 开发，请咨询移动技术人员。

十九、移动 Onenet-MQTT 物联网套件新版连接实例

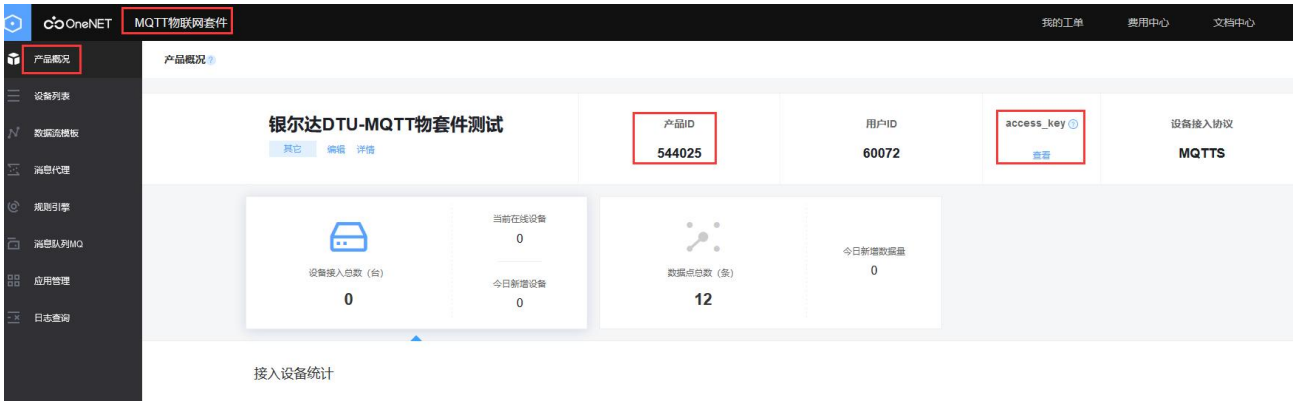
移动 Onenet MQTT 物联网套件，服务器发给 DTU 是纯数据透传，DTU 上报给服务器的数据是 json 格式的数据流格式。

1) 、创建产品



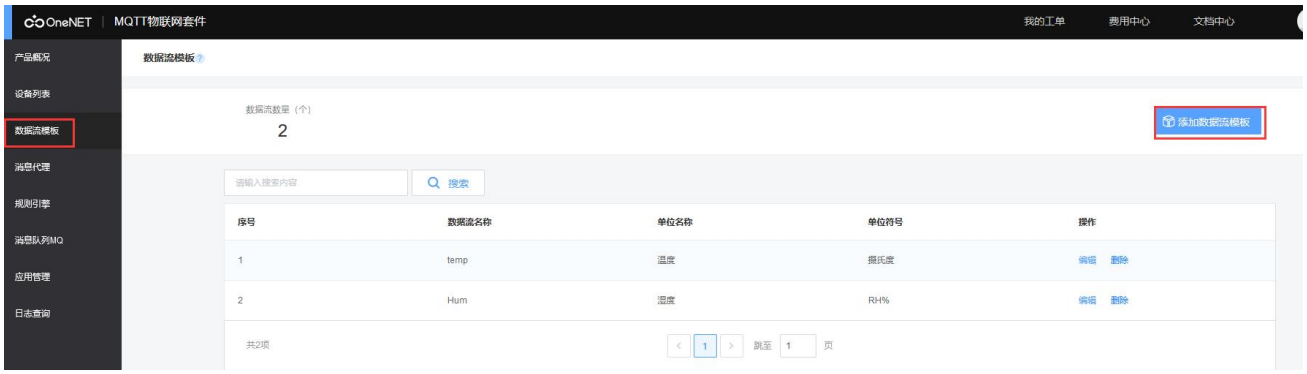
2) 、记录产品的产品 ID 和 Access_key

产品的 ID 和 access_key 需要填写到设备参数里面。



3) 、创建数据流模板

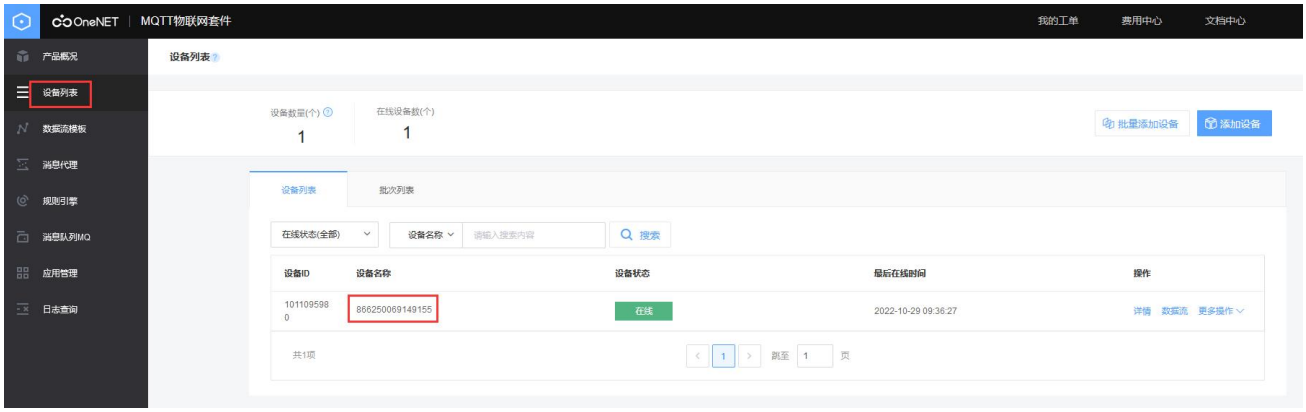
数据流模板，就是数据点，DTU 上报对应的 json 数据点，服务器记录相应的数据。服务器不关心数据点数据类型，只关心数据名称，类型由你上传的数据格式为准。这里创建一个温度(Temp)和湿度(Hum)用于测试。



4)、配置参数

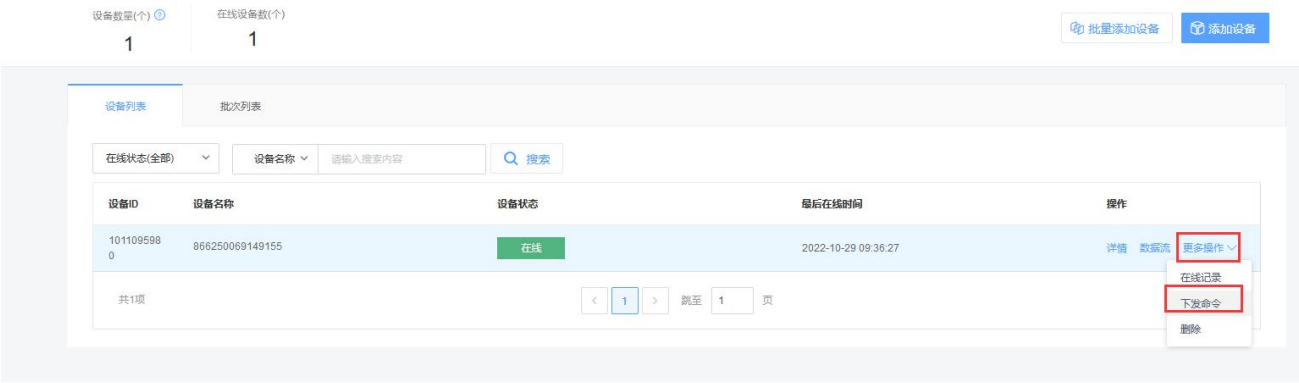
这里只讲解网络通道的参数配置，其他参数，参看《银尔达-DTU 固件 Web 配置用户手册》第 5 章

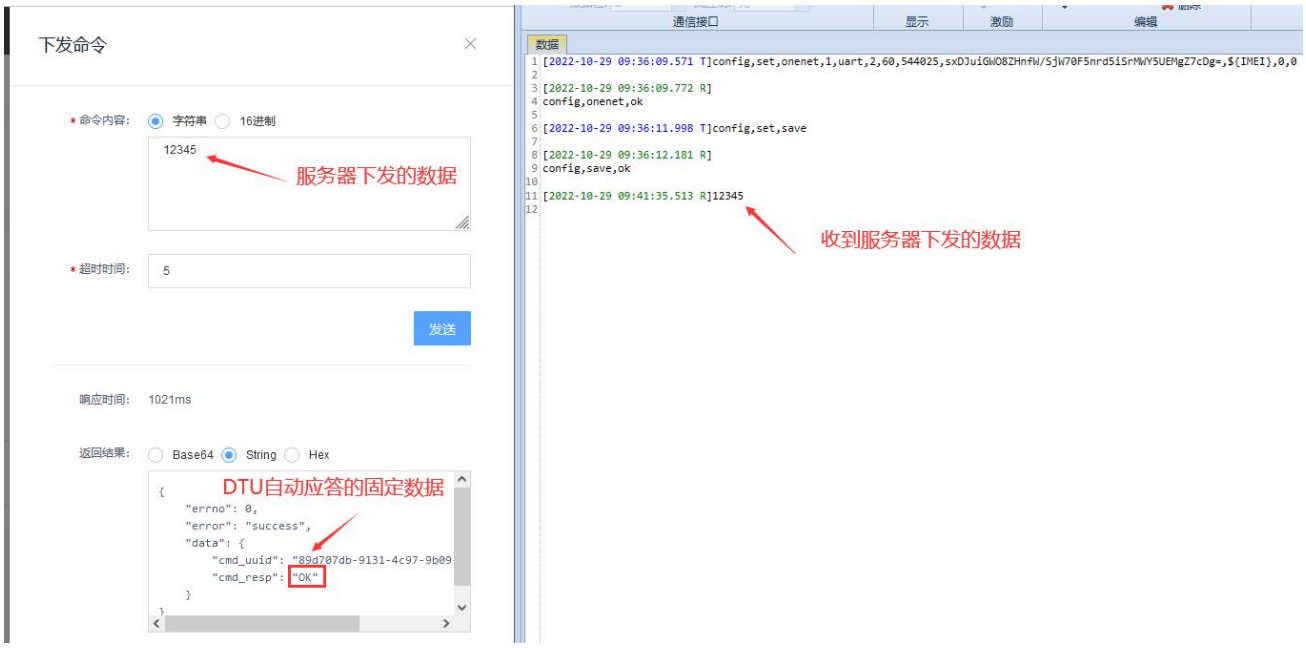
保持参数后，重启设备，参数更新，如果参数和网络正常平台会自动创建一个设备，并且在线。



5)、调试接收服务器命令

服务器下发命令调用测试接口下发，如果设备收到命令后，固定返回一个“OK”。





6) 、调试 DTU 上报数据

数据形式是 json 格式，数据内容要遵循移动的格式。

资料连接:https://open.iot.10086.cn/doc/v5/develop/detail/251　格式如下:

设备可以通过向系统固定 topic：$sys/{pid}/{device-name}/dp/post/json 发送数据点存储消息，消息中payload字段数据内容仅支持json格式，示例如下：

```
{
    "id": 123,
    "dp": {
        "temperatrue": [{
            "v": 30,
            "t": 1552289676
        }],
        "power": [{
            "v": 4.5,
            "t": 1552289676
        }],
        "status": [{
                "v": {
                    "color": "blue"
                },
                "t": 1552289677
            },
            {
                "v": {
                    "color": "red"
                },
                "t": 1552289678
            }
        ]
    }
}
```

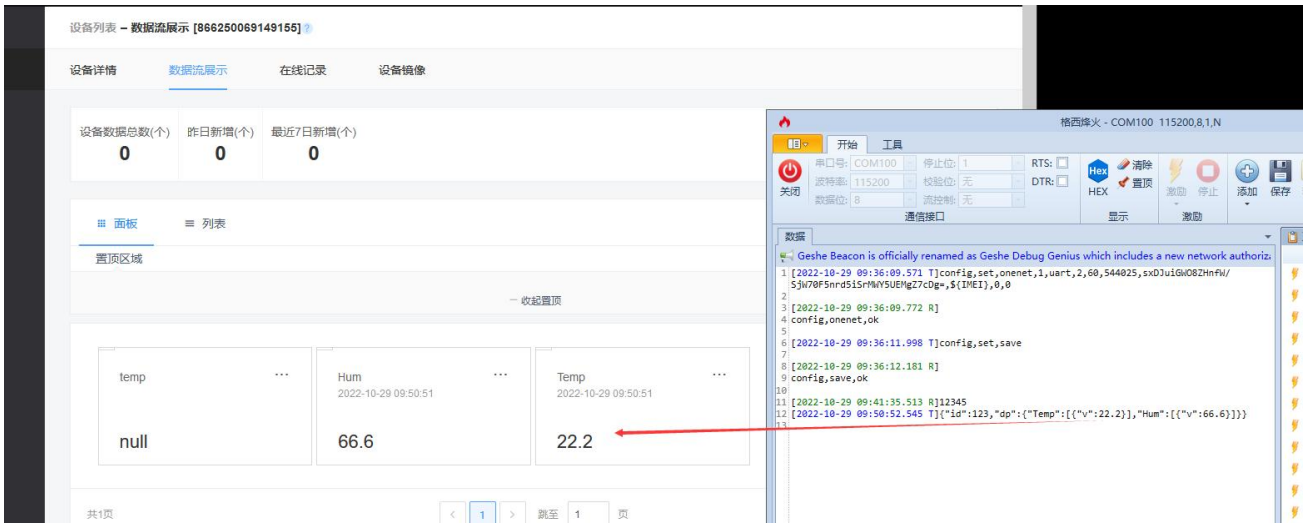
固定的

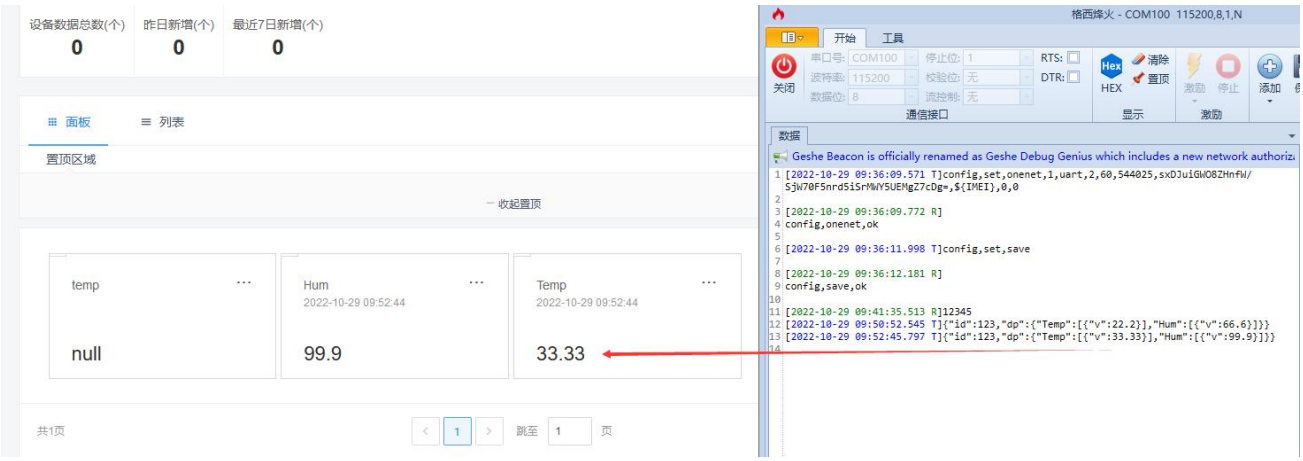
数据流模板名称

数据的内容，可以是任何类型

时间，可以不填写

第二个数据流模板名称





DTU 连接过程就完成了，关于移动云的其他业务开发和 API 开发，请咨询移动技术人员。

二十、常见问题说明

1、添加设备提示“找不到此设备”

是由于设备没授权导致，默认应该都要授权，可能是生产遗漏导致，发送设备型号(YED-XXXX)和设备 IMEI 给销售授权后再添加。

2、设备列表添加设备后，在分组的设备列表里面无法分配设备

确认创建分组的时候选择的产品型号是否和添加设备的型号一致

3、远程控制命令不响应

●服务器下发的支持的命令和串口配置命令相同

●确认 DTU 开启了远程控制命令

●有的工具无法发送\r\n 回车换行，工具把\r\n 当成了 \ r \ n 4 个字符，而不是\r\n 2 个字符串，可能需要编写代码发送才行（阿里 IOT 测试工具和 MQTT.fx 测试工具都是这样）

●不响应的命令都会透传到串口，可以用电脑 16 进制模式和串口命令对比一下判断服务器发的原始数据