

WG40配置软件使用手册

1.1 安装要求

- CPU：P4 1G 以上或相当型号
- 内存：最少 512MB，推荐 1GB
- 显示器：VGA、SVGA 或支持桌面操作系统的任何图形适配器。
- 鼠标：任何 PC 兼容鼠标
- 操作系统：Win XP（sp2）及以上。

1.2 使用入门

建立新工程

要建立新的工程，请首先为工程指定工作目录（或称“工程路径”）。用工作目录标识工程， 不同的工程应置于不同的目录。工作目录下的文件由网关配置软件自动管理。

项目设置向导

网关型号

网关选择RTU2MQTT01

硬件信息

更新模式以太网口

Io数量2

串口数量2

设备数量2

变量数量10

订阅数量5

发布数量5

MQTT数量5

项目名称

项目存储路径

获取路径

网关属性

项目类型

项目属性

确定

取消

构造数据库

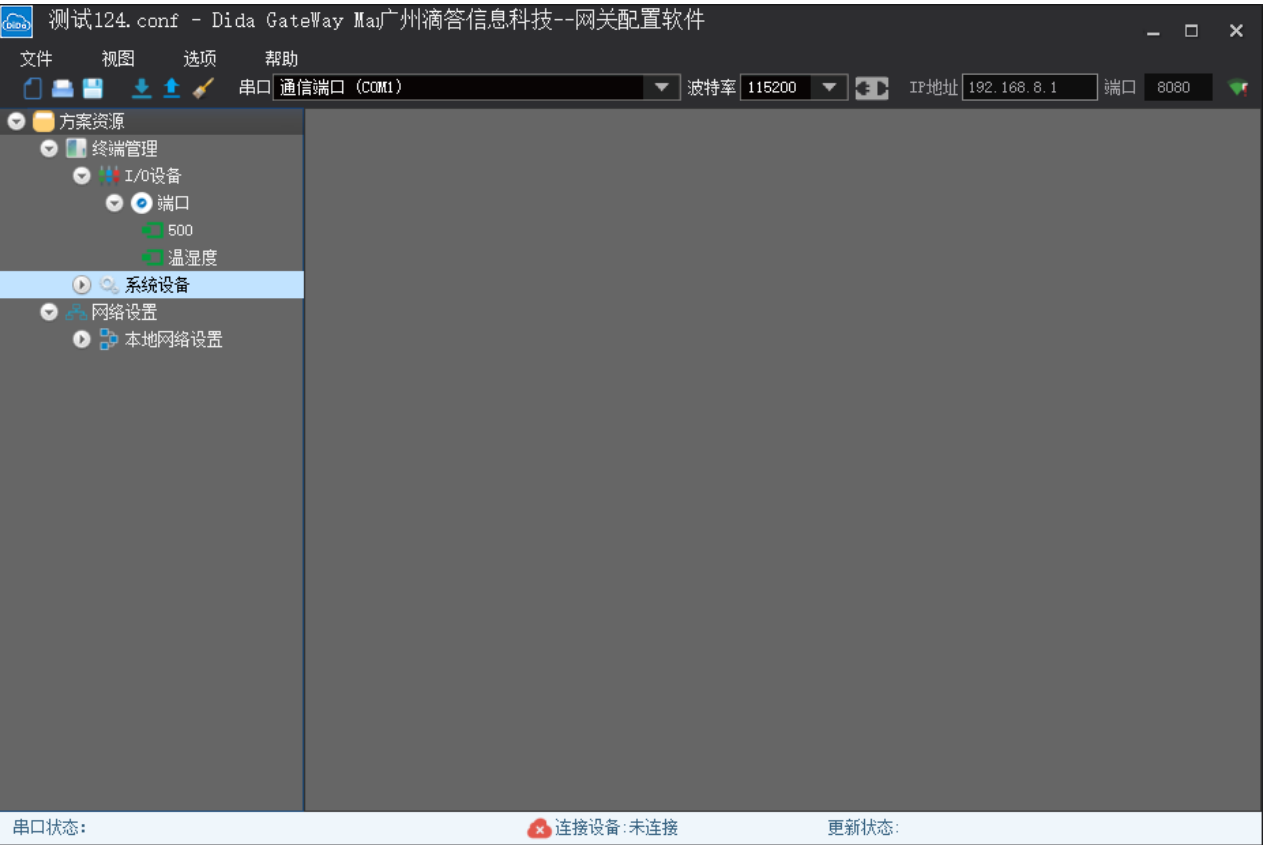
一个工程最基础的工作就是构造数据库，将实际需要的 IO 变量定义在工程中，网关配置软件 采用面向对象技术，通过端口对象、设备对象、寄存器对象的结构来定义 IO 变量表。实际工作中按照端口——>设备——>寄存器的步骤来构造数据库，不同于传统的组态软件一张大的 IO 变量表的方式来定义 IO 变量，通过不同设备来定义变量，便于技术人员在数据库中快速操作（查找、修改、删除）IO 变量。

1.3 开发环境——工程管理器

工程管理器是 网关配置软件 的一个重要组成部分，它将数据库、脚本、配置等工程元素集中管理，工程人员可以一目了然地查看工程的各个组成部分。工程浏览器简便易学，操作界面和 Windows 中的资源管理器非常类似，为工程的管理提供了方便高效的手段。

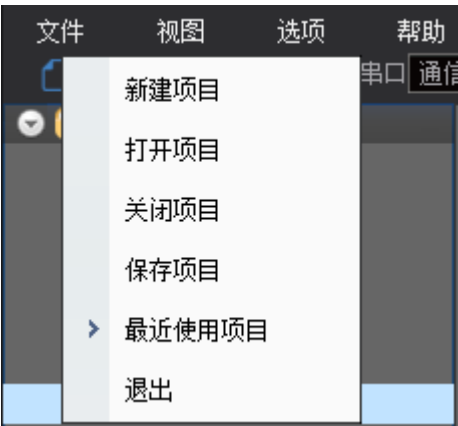
工程管理器概述

工程管理器的结构如图 3.1 所示。工程浏览器左侧为类似于资源管理器的树状结构图，主要展示工程的各个组成部分。主要包括“数据库”，“配方管理”，“全局脚本”，“工程设置”等部分。



工程菜单

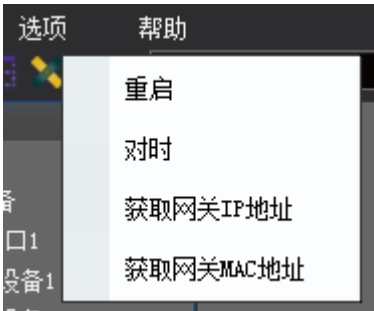
主要有“新建项目”、“打开项目”、“保存项目”、“关闭项目”等菜单项，实现项目的新建、打开、保存和关闭。



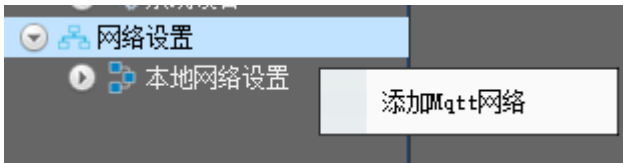
工具菜单

>2

“下载配置”用于下载工程到嵌入式设备中。 “提取配置”用于从嵌入式设备中上载工程到PC。 “校对”用于将PC时间发布到嵌入式设备中。 “重启”用于在PC上重新启动嵌入式设备。



“工程设置”用来设置从站属性、嵌入式设备的 IP 地址等。



构造数据库

方案资源管理器是“网关设置”最核心的部分。以实时数据库为中介环节，是联系上位机和下位机的桥梁。方案资源管理器从设备入手，在每个设备下定义相关的变量，与传统组态软件建立设备后，变量在一张数据表中相比，更加直观，查找变量更加方便。

方案资源管理器拥有“I/O 设备”，“I/O 设备”拥有“端口”，“端口”拥有“设备”，“设备”拥有“变量”。

IO 设备管理

端口

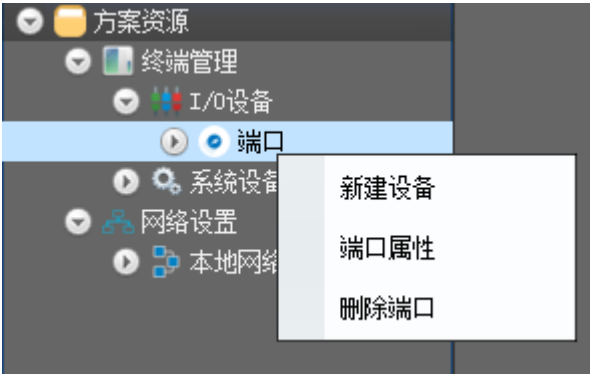
选择方案资源管理器，“终端管理”下的“I/O 变量”，鼠标点击右键，出现“新建端口”菜单，点击“新建端口”，出现端口属性对话框。端口名称具有唯一性，不可以重复。端口就是网关的串口或网口（高级型号），端口下面的设备就是连在这个串口（或网口）上的设备。



端口属性对话框

根据实际工程设备需求，可以选择设备类型，如“ModbusRTU”，选择端口类型为“串口”，设置好串口参数，点击“确定”即完成端口的定义。定义完成后会在“I/O 变量”下面生成“端口 1”，右键点击“端口 1”，出现如下子

菜单项。



新建设备

点击新建设备，出现如下对话框。



设备属性对话框

根据实际要求，修改设备名称和地址。点击确定后在“端口 1”下出现“设备 1”。



设备

假如工程有 2 个串口，第一个串口有 2 个“ModbusRTU”设备，第二个串口下有 3 个“ModbusRTU”设备，则可以建立“端口 1”，“端口 2”，“端口 1”下有 3 个设备，“端口 2”下有 2 个设备。如下图所示。



变量类型

变量的基本类型共有三类： I/O 变量、内存变量、系统变量。IO 变量是指可与外部数据采集程序直接进行数据交换的变量。

变量的数据类型

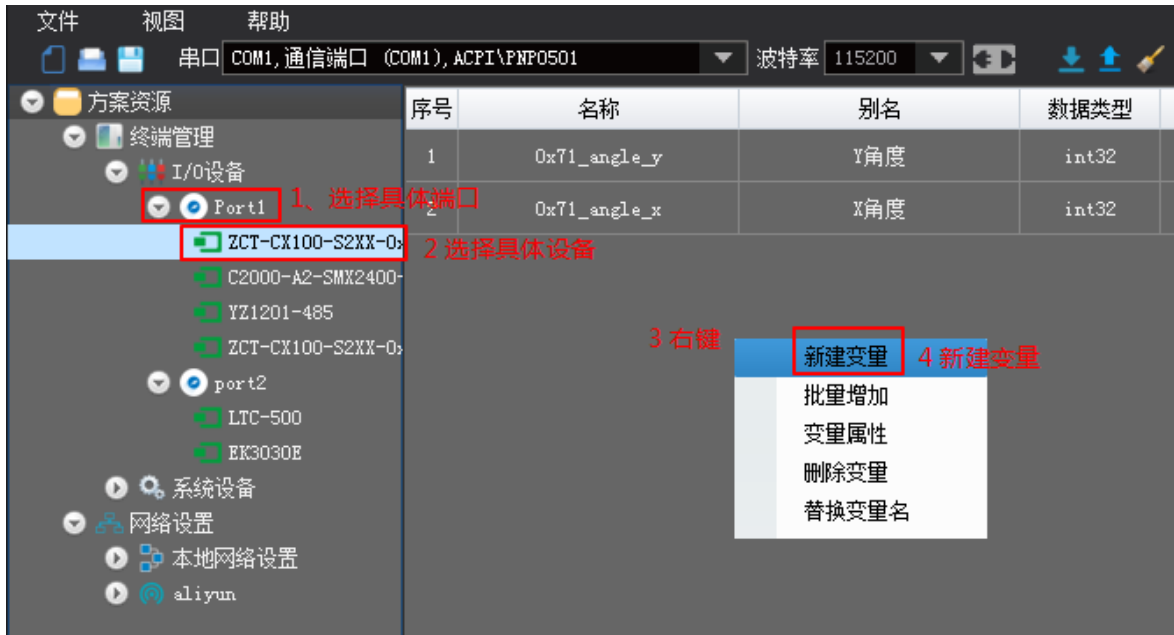
变量定义

选择相应的设备，然后右键点击右侧的变量显示区，出现如下子菜单。



变量操作

有两种定义变量的方式，如果变量寄存器地址是分散的，不连续的，可以点击“新建变量”来创建单个变量，如果变量寄存器地址是连续的，可以点击“批量增加”来创建多个变量。



变量作为一个对象，其属性分为基本属性、存盘属性、报警属性、量程变换，如下所示。



变量属性

基本属性有名称、描述、寄存器类型、寄存器地址、数据类型、最小值、最大值、小数位数、读写属性等。

名称：变量在整个数据库中的唯一标志，不可重复。只能以字母开头。也是alink json协议的关键词，在阿里云设置产品属性时必须以此设置，alink协议才接收并解析

别名：对变量名称的中文注释。

数据类型：变量对应的数据格式。

小数位数：变量值的小数有效位数。

读写属性：变量是否只读。

寄存器类型：不同的通讯规约，寄存器类型不一样，代表读取 的数据类型。

寄存器地址：变量对应的地址。

序号	名称	别名	数据类型	小数位	单位	寄存器地址	寄存器类型	读写属性	值
1	test	ReferTest	bool	1	int	1234	BO		

名称	描述
操作类型	指操作Modbus的功能码，具体如下： 输入状态，对应的功能码：0x02（读离散输入） 线圈状态，对应的功能码：0x01、0x0F（读、写线圈） 保持寄存器，对应的功能码：0x03、0x10（读、写保持寄存器） 输入寄存器，对应的功能码：0x04（读输入寄存器）
寄存器地址	指Modbus的寄存器的操作地址
数据类型	如采集的温度值的数据类型为浮点型，一个寄存器保存的是16位，需要两个寄存器保存的是32位，具体数据类型要看说明书
大小端	如采集的温度值占用一个寄存器（16位），但是对采集后的原始数据要进行高低字节的交换才能生成真实的值（modbus默认是大端，即高位在前，具体看说明书）
寄存器高低位	如采集的振动值占用两个寄存器（32位），但是对采集后的原始数据要进行前后寄存器的交换才能生成真实值
变量量程变换	指缩放系数，如采集的值为100，但是真实的值为10，因此需要缩放10倍，故缩放因子填

	写0.1即可。如放大10倍（即真实的值为1000），则放大因子为10
--	------------------------------------

一个寄存器保存16位二进制位，即两个字节，这两个字节一个地址高，一个地址低，他们的顺序关系就是寄存器大小端模式。高位数据在前面的地址为大端模式，低位数据在前面的地址为小端模式。

寄存器数据 0xAB （数据的左边是数据高位）

在内存中保存的方式（左边是前面地址低位，右边是后面地址）

大端 AB <= 一般Modbus设备为大端模式
小端 BA

如果一个变量是有两个寄存器保存，那这两个寄存器保存的值哪个在前，就需要设置寄存器（高低）位在前。只由一个寄存器保存的变量是不需要设置寄存器（高低）位在前的。

连续的寄存器的排序

数据 0x1122 （左边是数据的高位），在内存中的存储方式（左边的地址是前面）
高位在前时 11 22 （高位的数据在前面的地址）
低位在前时 22 11 （低位的数据在前面的地址）

举例说明：例如某个变量用两个寄存器保存，是无符号的值（就是没有负数的），那它的数据类型是int32，这个值是 0x1234
我们知道在mcu里实际的存储单位是字节，那么这个32位的无符号整数就有四个字节。那四个字节在存储区中是有以下几种情况：

低地址 ---> 高地址

- 1 2 3 4 就是 大端模式、寄存器（低）位在前；
- 3 4 1 2 就是 大端模式 寄存器（高）位在前；
- 2 1 4 3 就是 小端模式、寄存器（低）位在前；
- 4 3 2 1 小端模式、寄存器（高）位在前

下面以一个温湿度传感器modbus设备为例来说明如何填写

寄存器地址	内容	操作
40001H	湿度(单位 0.1%RH)	只读
40002H	温度(单位 0.1℃)	只读

温度：
当温度低于 0℃时以补码形式上传
FF9B H(十六进制)=-101=>温度=-10.1℃
湿度：
292 H(十六进制)=658=>湿度=65.8%RH

变量属性

基本属性

存盘属性

报警属性

量程变换

名称

Temperature

别名

Temperature

数据类型

int16

小数位

1

单位

int

量程转换系数

0.1

寄存器

地址

0000

☒十六进制

类型

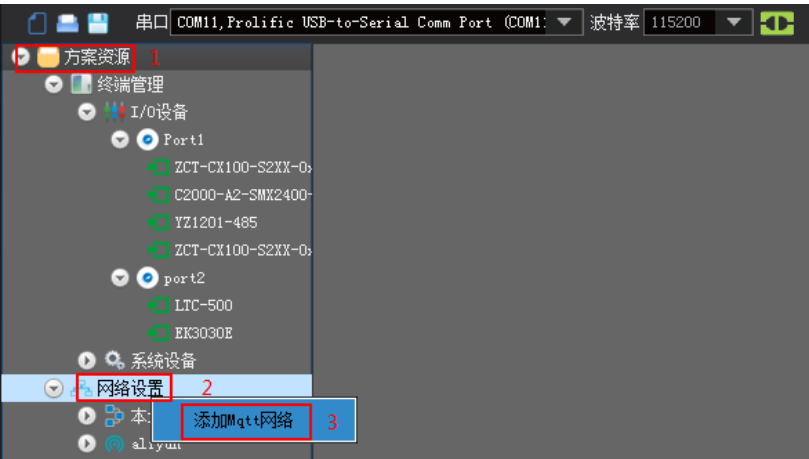
W3(保持寄存器)

因为是一个寄存器地址保存的温度所以是 16位的，我们选择int16；因为保存的温度单位0.1，所以我们设置量程转换系数为0.1；小数位设为1。



创建MQTT连接实体

在网关配置软件的工具栏，选择与网关连接的串口，成功连接之后，在配置软件左侧【方案资源】→【网络设置】→【添加Mqtt网络】，在对话框输入MQTT实体名称，添加MQTT连接实体。



- 【MQTT名称】：MQTT连接配置实体名称，**唯一性，不可重复**；支持英文字母、数字和特殊字符-_.:，最长长度为128字符；
- 【服务器地址】：MQTT远端服务器地址，可以是IP地址/域名，最长长度为128个字符；
- 【服务器端口】：MQTT远端服务器端口号，默认值：1883；
- 【服务器类型】：MQTT服务器端类型，可选：基本类型、阿里云；
- 【客户端ID】：MQTT客户端ID，最长长度为128字符，**支持英文字母、数字和特殊字符-_.:，最长长度为128字符**；
- 【用户名】：MQTT用户名，**支持英文字母、数字和特殊字符-_.:，最长长度为128字符**；
- 【连接密钥】：MQTT密钥，**支持英文字母、数字和特殊字符-_.:，最长长度为128字符**；
- 【连接心跳】：MQTT心跳周期，单位为秒；
- 【发布主题】：网关发布主题

MQTT

MQTT设置

MQTT名称

name

服务器地址

aIdmYxUsCJG.iot-as-mqtt.cn-shanghai.aliyuncs.com

服务器端口

1883

客户端ID

9Uart_TestaIdmYxUsCJG|securemode=3,signmethod=hmacsha1,timestamp=132323232|

用户名

9Uart_TestaIdmYxUsCJG

连接密钥

1F49338DAEA73239988DF44314

连接心跳

60

默认报警主题

PubAlarm

Qos0

enable

☐清除回话

☒是否开启连接

选择层安全

null

恢复出厂设置

重启设备

获取配置

写入配置

阿里云密钥计算

保存配置

图17 MQTT参数配置

为了方便转换阿里云物联网平台的配置参数到网关的MQTT实体配置参数，提供了【阿里云密钥计算】功能。该功能是将阿里云物联网平台提供的设备三元组信息，转换成网关配置的实际参数。

MQTT

MQTT阿里云密钥计算

地域(Region)

华东 上海[cn-shanghai]

连接域名

a16RJ2vzyR5.iot-as-mqtt.cn-shanghai.aliyu

DeviceName

Alink_Demo_09

ProductKey

a16RJ2vzyR5

DeviceSecret

UbhncDztqwqLr

TimeStamp

132323232

连接密钥

密钥生成

图18 阿里云密钥计算

- MQTT阿里云密钥计算参数说明：
- 【地域】：阿里云物联网平台所属地域，该参数需要根据实际的阿里云账号所属地域来选择；
 - 【连接域名】：选择好地域之后，软件自动生成对应的连接域名；
 - 【DeviceName】：阿里云物联网平台提供的设备名称；
 - 【ProductKey】：阿里云物联网平台提供的产品关键字；
 - 【DeviceSecret】：阿里云物联网平提供的设备密钥；
 - 【TimeStamp】：时间戳，可默认；

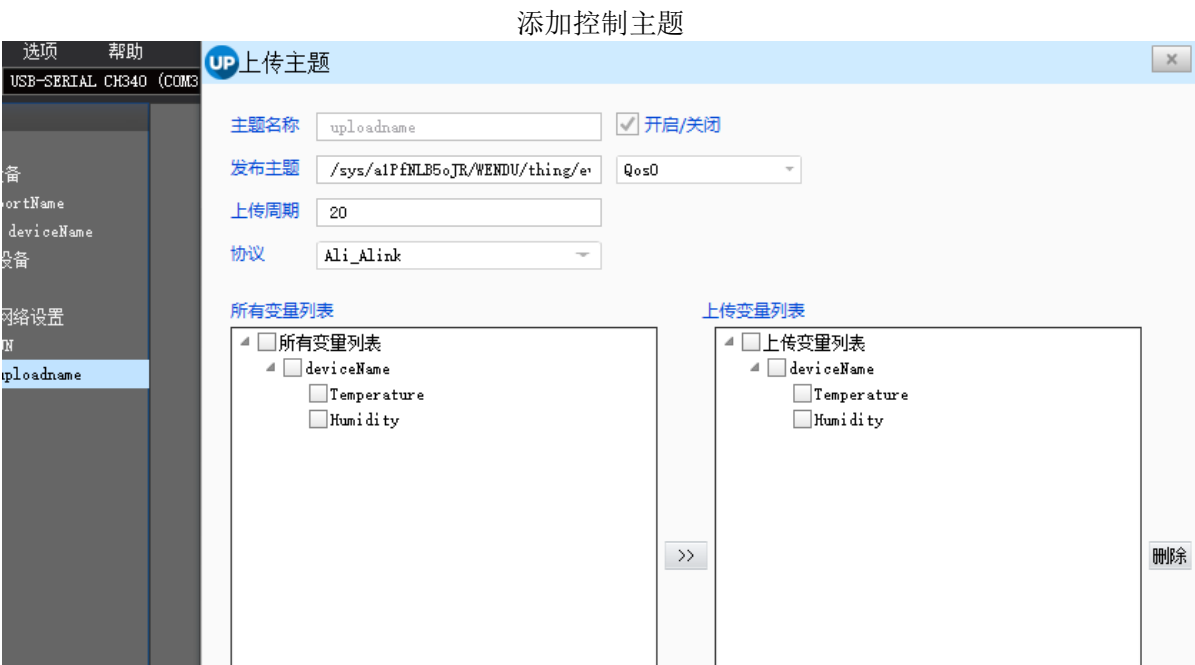
【连接密钥】：自动生成的连接密钥。

- MQTT连接参数说明：
- 【主题内容】：发布具体主题；
 - 【QOS】：发布主题对应的QOS值；
 - 【启用主题】：是否启用主题
 - 【控制协议】：主题的通信协议 Alink还是透传；
 - 【主动上报时间间隔(s)】：发布类主题，如是定时上报类，则需设置该值；
 - 【主题内容】：订阅具体主题；
 - 【QOS】：订阅主题对应的QOS值，该值为订阅主题的最大QOS值，网关可以接收QOS值小于等于该值的消息；
 - 【启用主题】：是否启用主题
 - 【有效负载格式】：主题的通信有效负载格式；



添加mqtt服务器配置

添加控制主题



添加上传主题

在阿里云上的设置，标识符必须和上位机设置的变量名称一样，设置为单精度浮点型，

功能类型	功能名称	标识符	数据类型	数据定义
属性	温度	Temperature	float (单精度浮点型)	取值范围: -40 ~ 80
属性	湿度	Humidity	float (单精度浮点型)	取值范围: 0 ~ 100

运行状态

湿度 45.9 %RH 2019/03/30 13:40:19	查看数据	温度 29.3 °C 2019/03/30 13:40:19	查看数据
---------------------------------------	----------------------	--------------------------------------	----------------------

2. 操作实例

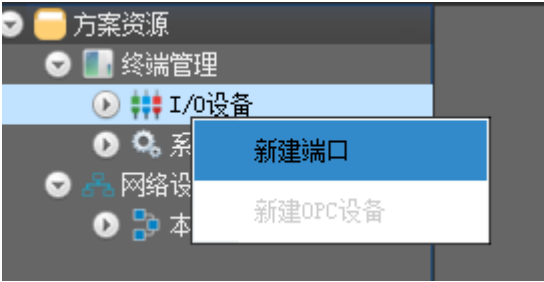
下面以WG20 以太网mqtt网关为例说明，通过上位机设置网关连接到阿里云mqtt服务器，把modbus寄存器的值在阿里云显示。

步骤1：新建项目，选择正确的设备型号 WG20，在下面的项目名称录入，以后就保存所有相关配置在此项目中，选择或创建合适的文件夹来保存项目。





步骤2：新建端口，选择左边窗口中的I/O设备，右键选择新建端口，此端口对应的就是网关上的串口。



在端口设置的窗口内，录入端口名称，端口类型, 设备类型。在串口参数栏内设置 串口的波特率和校验位、数据位、停止位。



用此方式我们添加了两个端口，COM1和COM2，下面我们要把三相电力检测设备DS9L连接到COM2上，那我们就要在COM2新建设备，选中COM2右键选择新建设备，出现设备属性对话框。录入设备名称、设备地址，选择此设备的寄存器大小模式和高低位。这些参数都会在设备的说明书内找到。



步骤3：增加变量

在左侧工程栏，选择我们刚刚增加的设备DS9L，，然后右键点击右侧的变量显示区，出现如下子菜单，选择批量增加。批量增加变量用来增加地址连续、数据类型一致的数个变量，避免了一个个增加变量的繁琐。



在批量增加变量对话框，我们录入要批量增加的变量个数和变量前缀名，数据类型，小数的位数和量程转换系数，和此批变量的开始地址。地址通常是十六进制的，可勾选设置。



批量增加变量后，右边变量窗口即可显示增加的变量表，可以看到增加的变量均以字母R为开头，寄存器地址是从4000连续增加，这个变量名称就是和服务端通信是JSON接口中的变量名称。可以点击来改变每个变量的名称和编辑属性

DS9L.conf - Dida GateWay Manager

广州滴答信息科技--网关配置软件

文件 视图 选项 帮助

串口 ELTIMA Virtual Serial Port (COM5->COM4) 波特率 115200 IP地址 192.168.8.1 端口 8080

方案资源

终端管理

I/O设备

COM1

COM2

DS9L

系统设备

网络设置

本地网络设置

序号	名称	别名	数据类型	小数位	单位	寄存器地址	寄存器类型	读写属性	值
1	R0	ReferTest	int32	1		4000	W4		
2	R1	ReferTest	int32	1		4002	W4		
3	R2	ReferTest	int32	1		4004	W4		
4	R3	ReferTest	int32	1		4006	W4		
5	R4	ReferTest	int32	1		4008	W4		
6	R5	ReferTest	int32	1		400a	W4		
7	R6	ReferTest	int32	1		400c	W4		
8	R7	ReferTest	int32	1		400e	W4		
9	R8	ReferTest	int32	1		4010	W4		
10	R9	ReferTest	int32	1		4012	W4		

通过单独添加变量或批量添加变量然后更改变量名称，我们得到如下的变量表

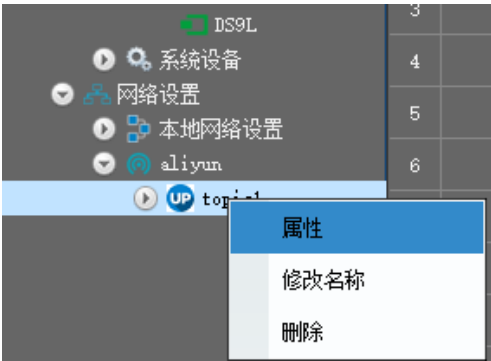
序号	名称	别名	数据类型	小数位	单位	寄存器地址	寄存器类型	读写属性	值
1	Ua	DATA	int32	1	Unit	4000	W3		
2	Ub	DATA	int32	1	Unit	4002	W3		
3	Uc	DATA	int32	1	Unit	4004	W3		
4	Uab	DATA	int32	1	Unit	4006	W3		
5	Ubc	DATA	int32	1	Unit	4008	W3		
6	Uca	DATA	int32	1	Unit	400a	W3		
7	Ia	DATA	int32	1	Unit	400c	W3		
8	Ib	DATA	int32	1	Unit	400e	W3		
9	Ic	DATA	int32	1	Unit	4010	W3		
10	Pa	DATA	int32	1	Unit	4012	W3		

步骤4：添加MQtt网络

设备最多可以支持连接四个MQTT服务器，连接一个MQTT服务器就要添加一个MQTT网络，连接服务器的相关账户、密码、上传的主题和控制的主题都保存在MQtt网络的设置中。选中左侧工作栏的网络设置，右键点击添加MQtt网络。



选中刚刚添加的MQTT网络aliyun，右键选择属性，编辑MQtt网络的属性，即设置服务器地址、服务器端口、客户端ID、用户名、连接密钥、心跳时间（单位为S）。**连接阿里云，阿里云平台规定心跳时间至少是30s。**



连接的是客户自己的mqtt服务器，直接填写相关信息即可，如果连接的是阿里云mqtt服务器，可以用在Mqtt设置对话框右下角的阿里云密钥计算工具，把阿里云mqtt服务器的 三元组 信息转为对应的 客户端id和秘钥。

MQTT

Mqtt设置

MQTT名称

aliyun

服务器地址

500.iot-as-mqtt.cn-shanghai.aliyuncs.com

服务器端口

1883

客户端ID

clientId

用户名

userName

连接密钥

password

连接心跳

60

默认报警主题

PubAlarm

Qos0

enable

☐清除回话

☒是否开启连接

选择层安全

null

恢复出厂设置

重启设备

获取配置

写入配置

阿里云密钥计算

保存配置

步骤五创建产品

登录阿里云，开通物联网功能，添加产品，在产品属性那里我们可以看到三元组信息，即ProductKey、DeviceName、DeviceSecret，在相应的参数旁边都有复制按键，点击可以复制。

如果是新产品，创建产品看阿里云文档

https://help.aliyun.com/document_detail/73728.html?spm=a2c4g.11186623.6.561.112d11bdduiGLw

物联网平台有两个规格，基础版和高级版。本文档介绍如何在控制台上创建高级版产品。

1. 登录[物联网平台控制台](#)。
2. 左侧导航栏选择设备管理 > 产品，单击创建产品。
3. 按照页面提示填写信息，然后单击完成。

新建产品 / 第二步：填写产品信息 (共二步) ×

产品信息

* 产品名称

Product

* 所属分类 ?

自定义品类 ▼ 功能定义

节点类型

* 节点类型

☒ 设备

☐ 网关 ?

* 是否接入网关

☐ 是

☒ 否

连网与数据

* 连网方式

WiFi ▼

* 数据格式

ICA 标准数据格式 (Alink JSON) ▼ ?

* 使用 ID* 认证：?

☐ 是

☒ 否

我们要设置节点类型为设备；设置数据格式为 Alink JSON，所属分类为自定义品类，

页面参数设置如下：

新建产品参数说明：

- 【产品名称】：产品名称；
- 【所属分类】：选择自动与
- 【节点类型】：选择设备，固定为设备；
- 【是否接入网关】：否；
- 【联网方式】：蜂窝 (3G/3G/4G)；
- 【数据类型】：ICA标准数据格式 (Alink JSON)；
- 【是否使用ID2认证】：否。

创建好产品之后，需要在产品下创建设备。

添加设备

特别说明: deviceName可以为空, 当为空时, 阿里云会颁发全局唯一标识符作为deviceName。

产品 :

Test20190716

DeviceName :

test001

备注名称 :

测试备注内容

确认

取消

图20 添加设备

添加设备参数说明：
【DeviceName】：设备名称；
【备注名称】：备注内容。

创建好设备之后，获取到设备的三元组信息。如图21所示。

查看设备证书

设备证书用于云端对接入的设备做鉴权认证，请妥善保管！

ProductKeya1j9fVB6aDe复制

DeviceNameTest001复制

DeviceSecret*****显示

一键复制

关闭

图21 设备三元组

在产品的设备列表里面中，可以方便查阅设备相关的详细参数。如图22所示。

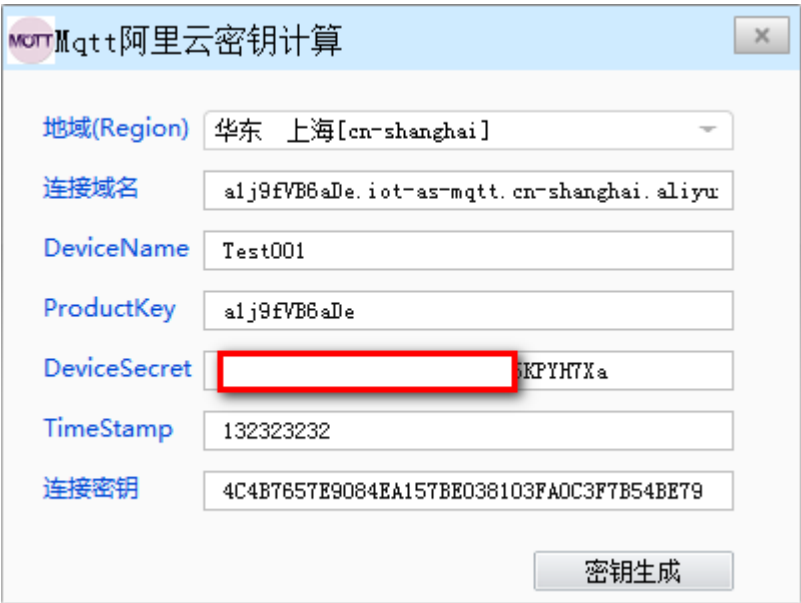
设备信息

产品名称	Test20190716	ProductKey	a1j9fVB6aDe 复制	区域	华东2（上海）
节点类型	设备	DeviceName	Test001 复制	DeviceSecret	***** 显示
备注名称	测试备注内容 编辑	IP地址	-	固件版本	-
添加时间	2019/07/16 18:10:18	激活时间		最后上线时间	
当前状态	未激活	实时延迟	测试		

图22 设备详情

从阿里云平台获取到设备对应的三元组信息，结合网关配置软件，点击阿里云秘钥计算按钮，弹出阿里云秘钥计算对话框，我们看到选择地域，通常是华东 上海，连接域名，然后就是三元组信息，通过拷贝阿里云上的

参数，黏贴过来，TimeStamp输入一个数字，点击密钥生成即可生成连接密钥。配置网关连接到阿里云物联网平台。如图23所示。点击密钥生成，相关信息自动生成并填入到 Mqtt设置的对话框的相关信息中，点击保存设置即可。



该对话框用于生成MQTT连接所需的密钥。它包含以下输入项：

- 地域(Region): 华东 上海[cn-shanghai]
- 连接域名: alj9fVB6aDe.iot-as-mqtt.cn-shanghai.aliyun
- DeviceName: Test001
- ProductKey: alj9fVB6aDe
- DeviceSecret: [输入框] KPYHTXa
- TimeStamp: 132323232
- 连接密钥: 4C4B7657E9084EA157BE038103FA0C3F7B54BE79

底部有一个“密钥生成”按钮。

图23 填入三元组信息

步骤6: 添加主题

右键选择添加上传主题，出现上传主题对话框，为上传主题录入名称。在左侧工作栏，选中刚刚添加的上传主题，右键选择属性，编辑上传主题。主题名称是为了大家方便管理而给主题起的名称。



该对话框用于添加新的上传主题。它包含以下输入项：

- 上传主题名称: topic1

底部有“确定”和“取消”按钮。

物联网平台的Topic分为系统定义的Topic和自定义的Topic，您可以直接使用系统定义的Topic或参见[自定义Topic](#)添加更多自定义Topic类。本文主要列出了系统定义的Topic。

/sys/\${YourProductKey}/\${YourDeviceName}/thing/event/property/post：用于设备上报属性。设备操作权限：发布。

/sys/\${YourProductKey}/\${YourDeviceName}/thing/service/property/set：用于设置设备属性。设备操作权限：订阅。

/sys/a1dmYxUsCJG/9Uart_Test/thing/event/property/post

/sys/a1dmYxUsCJG/9Uart_Test/thing/service/property/set

/sys/a1dmYxUsCJG/9Uart_Test/thing/event/\${tsl.event.identifer}/post

/sys/a1dmYxUsCJG/9Uart_Test/thing/service/\${tsl.event.identifer}

/sys/a1dmYxUsCJG/9Uart_Test/thing/deviceinfo/update

/a1dmYxUsCJG/9Uart_Test/user/update

/a1dmYxUsCJG/9Uart_Test/user/update/error

/a1dmYxUsCJG/9Uart_Test/user/get

设备管理 > 设备详情

9Uart_Test 在线

产品：MQTT网关9UART 查看

设备信息 Topic列表

将Topic列表里的第一个主题，填写到上传主题里面，根据阿里云物联网平台使用限制规定，设备的上报周期为30-1200秒，建议取值300秒以上。

选择右侧栏里的变量，然后点>>即可，选择要上传的变量列表，设置好上传周期和发布主题。

UP 上传主题

主题名称: ☒ 开启/关闭

发布主题: Qos1

上传周期:

协议:

所有变量列表

- ☐ 所有变量列表
- ☐ DS9L
 - ☐ Ua
 - ☐ Ub
 - ☐ Uc
 - ☐ Uab
 - ☐ Ubc
 - ☐ Uca
 - ☐ Ia
 - ☐ Ib
 - ☐ Ic
 - ☐ Pa
 - ☐ Pb
 - ☐ Pc
 - ☐ P
 - ☐ Qa
 - ☐ Qb
 - ☐ Qc
 - ☐ Q
 - ☐ Sa
 - ☐ Sb
 - ☐ Sc
 - ☐ S
 - ☐ PPa
 - ☐ PPb

>>

上传变量列表

- ☐ 上传变量列表
- ☐ DS9L
 - ☐ Ua
 - ☐ Ub
 - ☐ Uc
 - ☐ Uab
 - ☐ Ubc
 - ☐ Uca
 - ☐ Ia
 - ☐ Ib
 - ☐ Ic
 - ☐ Pa
 - ☐ Pb
 - ☐ Pc
 - ☐ P
 - ☐ Qa
 - ☐ Qb
 - ☐ Qc
 - ☐ Q
 - ☐ Sa
 - ☐ Sb
 - ☐ Sc
 - ☐ S
 - ☐ PPa
 - ☐ PPb

删除

确定 取消

步骤7：电脑连接网关，下载配置。

如下图连接好硬件设备，接上电源，用usb转232的db9接口接到设备的端口1，把modbus设备的485线A和B接到以太网mqtt网关的串口2。

IP地址 192.168.2.120

小数位 单位

设备连接测试--成功

项目完成下载

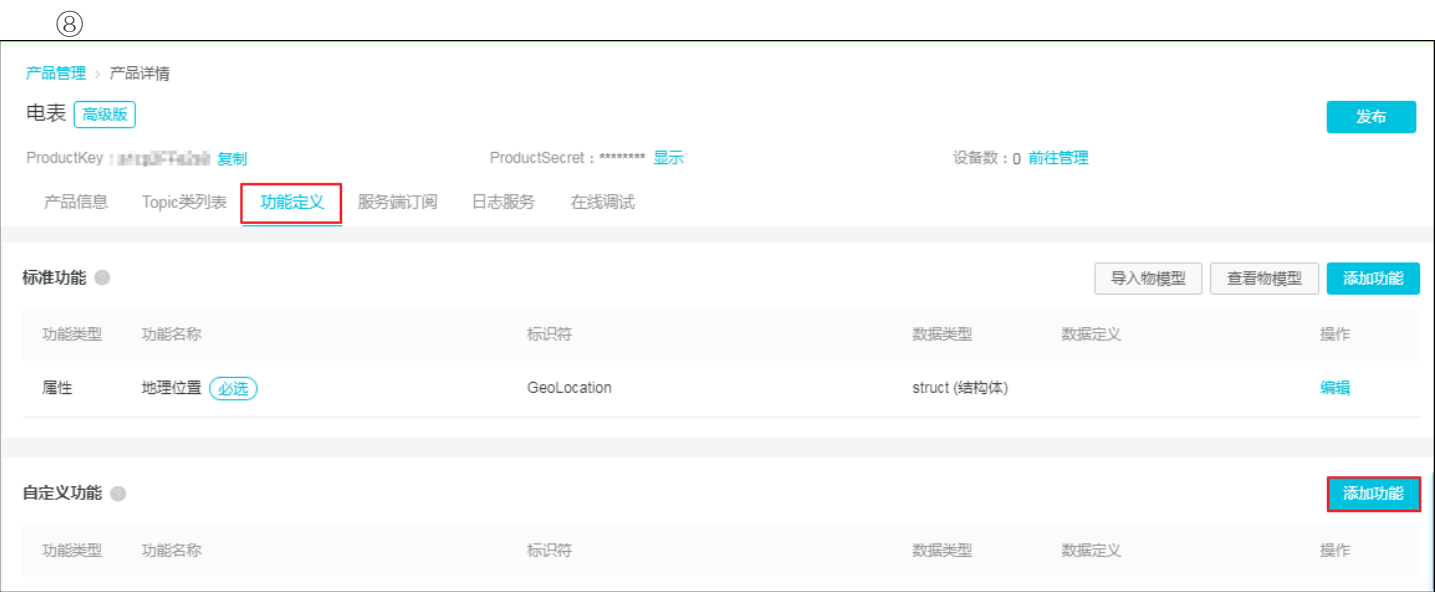
步骤8 创建物模型

阿里云mqtt高级版，进入产品详情，新增物模型，即定义产品功能（包括属性、事件和服务）。参见阿里云文档：
https://help.aliyun.com/document_detail/88241.html?spm=a2c4g.11186623.2.8.3474f9bcZPAaRi#task-qhm-d3j-w2b

- ① 登录[物联网平台控制台](#)。
- ② 左侧导航栏选择**设备管理 > 产品**。
- ③ 在**产品管理**页面产品列表中，单击产品所对应的**查看**操作按钮。
- ④ 单击**功能定义**。
- ⑤ 添加标准功能。单击**标准功能**栏对应的**添加功能**按钮，然后在弹出对话框中，选择适用于该产品的物联网平台预定义的标准功能。



- ⑦ 添加自定义功能。单击**自定义功能**栏对应的**添加功能**为产品新增自定义功能。您可为产品自定义属性、服务和事件。



自定义属性。在**添加自定义功能**页面，选择功能类型为**属性**。设置参数完成后，单击**确认**。

添加自定义功能

* 功能类型：

属性

服务

事件

* 功能名称：

工作电压

* 标识符：

LightVolt

* 数据类型：

float (单精度浮点型)

* 取值范围：

0

~

4

* 步长：

0.1

单位：

伏特 / V

读写类型：

☐ 读写

☒ 只读

属性唯一标识符，在产品中具有唯一性。即 Alink JSON 格式中的*identifier*的值，作为设备上报该属性数据的 Key，云端根据该标识符校验是否接收数据。要和上位机设置的变量名称（不是别名）一致。

序号	名称	别名	数据类型	小数位	单位	寄存器地址	寄存器类型	读写属性	值
1	Ua	DATA	int32	1	Unit	4000	W3		
2	Ud	DATA	int32	1	Unit	4002	W3		

属性参数设置如下：

参数	描述
功能名称	属性的名称，如用电量。同一产品下功能名称不能重复。 支持中文、大小写字母、数字、短划线和下划线，且必须以中文、英文或数字开头，不超过30个字符。 如果您创建产品时选择了功能模板，输入功能名称时，将从标准功能库中筛选匹配的标准属性，供您选择。 说明 当接入网关协议为Modbus时，不支持标准属性，仅支持自定义属性。
标识符	属性唯一标识符，在产品中具有唯一性。即 Alink JSON 格式中的<i>identifier</i>的值，作为设备上报该属性数据的 Key，云端根据该标识符校验是否接收数据。可包含英文、数字、下划线，长度不超过50个字符，如 PowerConsumption。 说明 不能用以下系统保留参数作为标识符：set、get、post、time、value。
数据类型	int32：32位整型。需定义取值范围、步长和单位符号。 float：单精度浮点型。需定义取值范围、步长和单位符号。 double：双精度浮点型。需定义取值范围、步长和单位符号。 enum：枚举型。定义枚举项的参数值和参数描述，如 1-加热模式、2-制冷模式。 bool：布尔型。采用 0 或 1 来定义布尔值，如 0-关、1-开。

步长	属性值和事件以及服务中输入输出参数值变化的最小粒度。数据类型为int32、float、double时，需要设置步长。
单位	单位可选择为无或根据实际情况选择。
读写类型	读写：请求读写的方法支持 GET（获取）和 SET（设置）。 只读：请求只读的方法仅支持 GET（获取）。 说明 当接入网关协议为Modbus时，无需设置该参数。
描述	输入文字，对该功能进行说明或备注。长度限制为100字。

设备管理 > 设备详情

DS9L_1 在线

产品：DS9L 查看

ProductKey：a1nXM1LhF1R 复制

DeviceSecret：***** 显示

设备信息 Topic列表 运行状态 事件管理 服务调用 日志服务

运行状态 设备数据上报的最新属性值，点击“查看数据”可以查看指定属性的历史数据

实时刷新 ☐ 表格 图表

总有功电度 查看数据 17202 Wh 2019/02/22 12:27:44	总无功电度 查看数据 0 kvarh 2019/02/18 16:16:48	频率 查看数据 5002 Hz 2019/02/22 12:27:44	相电流A 查看数据 0 mA 2019/02/22 12:27:44
相电流B 查看数据 0 mA 2019/02/22 12:27:44	相电流C 查看数据 0 mA 2019/02/22 12:27:44	总有功功率 查看数据 0 w 2019/02/22 12:27:29	总功率因素 查看数据 0 2019/02/22 12:27:44

添加控制主题

/sys/a1dmYxUsCJG/9Uart_Test/thing/event/property/post

/sys/a1dmYxUsCJG/9Uart_Test/thing/service/property/set

/sys/a1dmYxUsCJG/9Uart_Test/thing/event/\${tsl.event.identifer}/post

/sys/a1dmYxUsCJG/9Uart_Test/thing/service/\${tsl.event.identifer}

/sys/a1dmYxUsCJG/9Uart_Test/thing/deviceinfo/update

/a1dmYxUsCJG/9Uart_Test/user/update

/a1dmYxUsCJG/9Uart_Test/user/update/error

/a1dmYxUsCJG/9Uart_Test/user/get

这是一个主题的例子，我们可以看到 把上图Topic列表里的第二个topic（设备属性设置set）填写到下图的订阅主题里，把上图Topic列表里的第一个topic（post）填写到下图的

发布主题。

控制主题

主题名称

ThemeName

☐ 开启/关闭

发布主题

pub

Qos0

订阅主题

sub

Qos0

控制协议

null

确定

取消

向控制主题发送以下格式的协议

```
{ "id": "20", "version": "1.0", "method": "thing.service.property.set", "params": { "varName": 200 } }
```

其中id是代表下发指令的id，用于区分下发。“params”:后的 {} 内是 要操作的寄存器名称和操作值

网关返回

```
{ "id": "20", "code": 200, "data": {} }
```

Alink请求数据格式:

```
{
  "id": "123",
  "version": "1.0",
  "params": {
    "Power": {
      "value": "on",
      "time": 1524448722000
    },
    "WF": {
      "value": 23.6,
      "time": 1524448722000
    }
  },
  "method": "thing.event.property.post"
}
```

表 1. 请求参数说明

参数	类型	说明
id	String	消息ID号，String类型的数字。
version	String	协议版本号，目前协议版本1.0。
params	Object	请求参数。如以上示例中，设备上报了的两个属性Power和WF。具体属性信息，包含属性上报时间（time）和上报的属性值（value）。
time	Long	属性上报时间。
value	object	上报的属性值。
method	String	请求方法。

Alink响应数据格式:

```
{
  "id": "123",
  "code": 200,
  "data": {}
}
```

表 2. 响应参数说明

参数	类型	说明
----	----	----

id	String	消息ID号，String类型的数字。
code	Integer	结果状态码。具体参考 设备端通用code 。
data	String	请求成功时，返回的数据。

与服务器通信的Json协议例子，发送消息示例：

```
{
  "id": "51",
  "version": "1.0",
  "method": "thing.event.property.post",
  "params": {
    "Ua": 2226,
    "Ub": 0,
    "Uc": 0,
    "Uab": 2226,
    "Ubc": 2226,
    "Uca": 0,
    "Ia": 0,
    "Ib": 0,
    "Ic": 0,
    "Pa": 0,
    "Pb": 0,
    "Pc": 0,
    "P": 0,
    "Qa": 0,
    "Qb": 0,
    "Qc": 0,
    "Q": 0,
    "Sa": 0,
    "Sb": 0,
    "Sc": 0,
    "S": 0,
    "PFa": 0,
    "PFb": 0,
    "PFc": 0,
    "PF": 0,
    "F": 4996,
    "EP": 17202,
    "EQ": -8982
  }
}
```

设备端开发完成后，您可以使用物联网平台的在线调试功能，从控制台下发指令给设备端进行功能测试。目前仅高级版产品下的设备支持此功能。

操作步骤

在物联网控制台，左侧导航栏选择**监控运维 > 在线调试**。

在**在线调试**页，选择本次调试的设备。

选择设备后，页面会自动跳转至调试设置页。

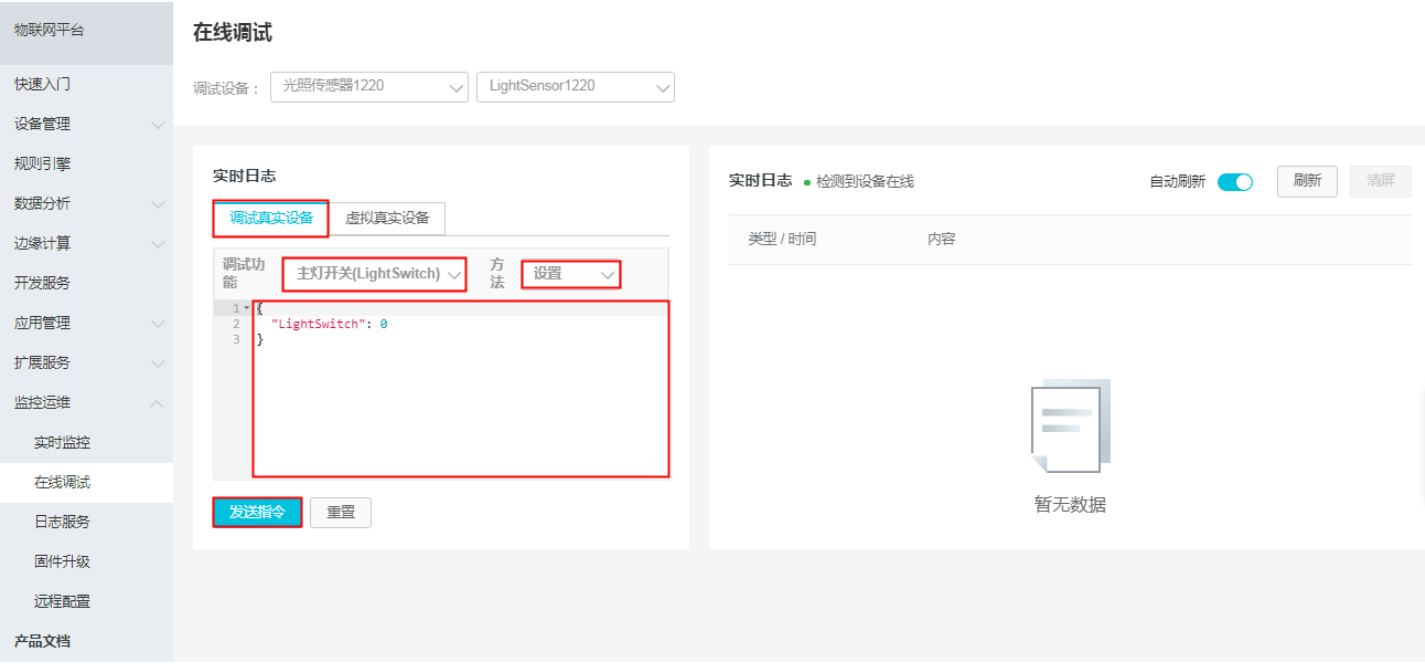


选择**调试真实设备**。

选择要调试的功能。

选择调试属性时，需选择方法为**设置**或**获取**。

选择调试事件时，需选择方法为**获取**。



发送调试指令。

设置属性：在空白栏，输入属性参数，格式为{"Yourpropertyidentifier": xxxx}，然后单击**发送指令**。属性设置结果，可查
看设备实时日志。

获取属性：单击**发送指令**，空白栏处显示设备最新上报的属性内容。

调用服务：在空白栏，输入服务入参，格式为{"Yourserviceinputparam": xxxx}，然后单击**发送指令**。服务调用结果，可
查看设备实时日志。

获取事件：单击**发送指令**，空白栏处显示设备设备最新上报的事件内容。