

开发手机小程序

1.产品设备初始化

- 创建产品Wechatdemo，下设两个设备WeChatcontrol（与微信小程序端相连，代表虚拟设备），ArduinoDev（与arduino端相连，代表真实设备）

DeviceName/备注名称	设备所属产品	节点类型	设备状态	最后上线时间	启用/禁用	操作
Wechatcontrol 虚拟设备	Wechatdemo	设备	● 在线	2023/10/30 16:46:31.798	<input checked="" type="checkbox"/>	查看 删除
ArduinoDev 真实设备	Wechatdemo	设备	● 在线	2023/10/30 16:18:12.910	<input checked="" type="checkbox"/>	查看 删除

- 导入物模型，为产品创建6个属性

注意产品名称与标识符的对应性

当前展示的是已发布到线上的功能定义，如需编辑请先撤销发布

前往编辑草稿 物模型 TSL 配置时序存储

请输入模块名称

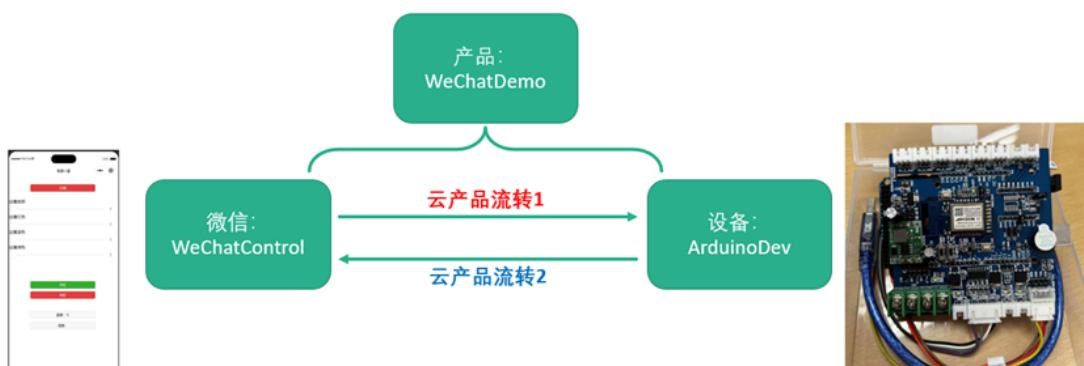
默认模块

功能类型	功能名称(全部)	标识符	数据类型	数据定义	操作
属性	蓝灯亮度 自定义	ColorBlue	int32(整数型)	取值范围: 0 ~ 255	查看
属性	绿灯亮度 自定义	ColorGreen	int32(整数型)	取值范围: 0 ~ 255	查看
属性	红灯亮度 自定义	ColorRed	int32(整数型)	取值范围: 0 ~ 255	查看
属性	光照 自定义	photores	int32(整数型)	取值范围: 0 ~ 1023	查看
属性	温度 自定义	temperature	float(单精度浮点型)	取值范围: 0 ~ 100	查看
属性	灯开关 自定义	light	bool(布尔型)	布尔值: 0 - 关 1 - 开	查看

- 为两个设备分别添加自定义topic,用于云端数据流转和设备及小程序端的通信

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
/k09l88hcMxJ/\${deviceName}/user/Wechatcontrol	发布和订阅	<input checked="" type="checkbox"/>	未开启	-	
/k09l88hcMxJ/\${deviceName}/user/ArduinoDev	发布和订阅	<input checked="" type="checkbox"/>	未开启	-	

- 创建云产品流转



云产品流转1：将微信的控制数据转发给设备

- 数据来源：微信上报的参数，操作权限是“发布”
`/k08lqrptqsF/WeChatControl/user/WeChatControl`
- 数据去向：设备属性设置Topic，操作权限是“订阅”
`/sys/k08lqrptqsF/ArduinoDev/thing/service/property/set`

云产品流转2：将设备的数据转发给微信

- 数据来源：设备上报的参数，操作权限是“发布”
`/sys/k08lqrptqsF/ArduinoDev/thing/event/property/post`
- 数据去向：微信自定义Topic，操作权限是“订阅”
`/k08lqrptqsF/WeChatControl/user/get`

- 在编写SQL选项中输入数据来源SELECT * FROM "/k09l88hcMxJ/Wechatcontrol/user/Wechatcontrol", 在转发操作的添加操作中输入数据去向/sys/k09l88hcMxJ/ArduinoDev/thing/service/property/set

← Arduino订阅的topic

数据格式 JSON 规则 ID 1722583
 规则描述 微信数据发送到Arduino上
 处理数据 编辑

规则查询语句:
 SELECT * FROM "/k09l88hcMxJ/Wechatcontrol/user/Wechatcontrol"

转发数据 添加操作

数据目的地	状态	操作
操作 ID: 1563582, 该操作将数据发布到另一个Topic中: /sys/k09l88hcMxJ/ArduinoDev/thing/service/property/set	● 正常执行	编辑

转发错误操作数据 添加错误操作

数据目的地	状态	操作
-------	----	----

- 从Arduinodev到Wechatcontrol的规则创建

在编写SQL选项中输入数据来源SELECT * FROM "/k09l88hcMxJ/ArduinoDev/thing/event/property/post"数据去向为 /k09l88hcMxJ/Wechatcontrol/user/get

物联网平台 / iot-06200felpb0ssm7 / 消息转发 / 云产品流转 / 数据转发的topic

← 数据转发的topic

数据格式 JSON 规则 ID 1727461
 规则描述 将数据从arduino转发到微信数据端
 处理数据 编辑

规则查询语句:
 SELECT * FROM "/k09l88hcMxJ/ArduinoDev/thing/event/property/post"

转发数据 添加操作

数据目的地	状态	操作
操作 ID: 1568069, 该操作将数据发布到另一个Topic中: /k09l88hcMxJ/Wechatcontrol/user/get	● 正常执行	编辑

转发错误操作数据 添加错误操作

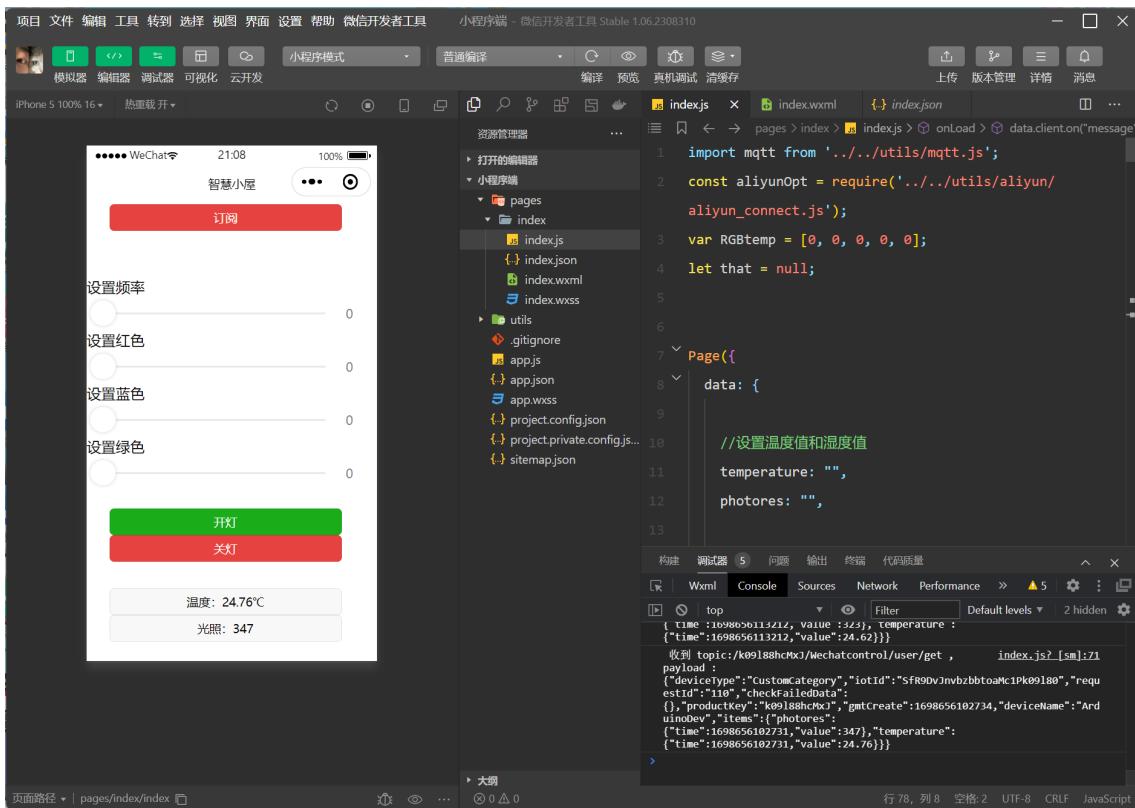
数据目的地	状态	操作
-------	----	----

注意必须将创建的规则点击运行才能生效

规则名称	规则 ID	数据格式	规则描述	创建时间	状态	操作
数据转发的topic	1727461	JSON	将数据从arduino转发到...	2023/10/30 13:55:14	● 运行中	查看 停止
Arduino订阅的topic	1722583	JSON	微信数据发送到Arduino上	2023/10/23 14:40:19	● 运行中	查看 停止

2.微信小程序端初始化

- 注册微信公众平台并下载微信小程序开发工具，导入工程项目



其中index.js用来存放主程序，index.wxml用来存放小程序图形生成页面代码

- 将设备Wechatcontrol的三元组连接参数填写到程序中data对应位置，如aliyun平台连接失败可通过password计算小程序重新计算。

```

    resubscribe: true, // 如果未按顺序重新连接，则云开发自动尝试重连（默认为
                      // true）

    clientId: 'Wechatcontrol',
    password: 'E21A29DE7621D9A81594B8022BB9C1B5F304E9C7',
    username: 'Wechatcontrol&k09l88hcMxJ',
}

aliyunInfo: {
    productKey: 'k09l88hcMxJ', // 阿里云连接的三元组，请自己替代为自己的产品信息！
    deviceName: 'Wechatcontrol', // 阿里云连接的三元组，请自己替代为自己的产品信息！
    deviceSecret: '6026f05ad3f4b999331c221a005084ba', // 阿里云连接的三元组，请自己
                                                       // 替代为自己的产品信息！
}
  
```

点击编译按钮，小程序端提示连接成功

智慧小屋



订阅

设置频率



设置红色



设置蓝色



设置绿色



0

0

0

0

开灯

关灯

温度: °C

光照:

阿里云平台中Wechatcontrol设备上线

3.设备端初始化

- 复制Arduinodev设备的三元组连接参数输入到json.v4程序中

```
//User Modified Part
#define wifi_ssid      "wifi"
#define wifi_psw       "jyrr0915"
#define clientIDstr   "ArduinoDev"
#define timestamp      "999"
#define ProductKey    "k09188hcMxJ"
#define DeviceName     "ArduinoDev"
#define DeviceSecret   "5e572ea49a6636eb6e798b6c40f5fa44"
#define password       "0ED6BC296FB6AB31432D22260DDF73BC320C4DAE"
```

此处注意一定要将手机热点调至2.4G频段，否则无法连接，并且注意securemode与signmethod的统一！！

```
#define AT_MQTT_CID          "AT+MQTTCID=%s|securemode=3\\\",signmethod=hmacsha1\\\",timestamp=%s\\r\\n"
```

- 对程序中loop函数进行修改，其中加入了对温度和光照强度的读取以及延时的设置

```

// TT (frequency==0) || frequency==1; }

//Serial.println(frequency);
if(Serial3.available()==0) {

    //端口7,8,9混色
    analogWrite(9, ColorBlue);
    analogWrite(8, ColorGreen);
    analogWrite(7, ColorRed);
}

if((millis()-timeStart)>10000)
{
    Upload();
    timeStart=millis();
    Serial.print(F("Temperature = "));
    Serial.print(bmp.readTemperature());
    Serial.println(" *C");
    Serial.print(F("PhotoRes = "));
    Serial.print(analogRead(A2));
}

```

- 点击上传到开发板听到三声滴滴后，说明连接成功，阿里云平台Arduinodev上线

<input type="checkbox"/> DeviceName/备注名称	设备所属产品	节点类型	设备状态	最后上线时间	启用/禁用	操作
<input type="checkbox"/> Wechatcontrol 虚拟设备	Wechatdemo	设备	● 在线	2023/10/31 21:29:03.462	<input checked="" type="checkbox"/>	查看 删除
<input type="checkbox"/> ArduinoDev 真实设备	Wechatdemo	设备	● 在线	2023/10/31 21:29:50.887	<input checked="" type="checkbox"/>	查看 删除

至此初始化全部完成，接下来进行微信小程序开发以及数据流转

4. 小程序端设置参数，设备端接收

- 将开发程序中sendData部分按json格式修改，参数与红蓝绿光标识符对应（此处注意标识符的规范性）

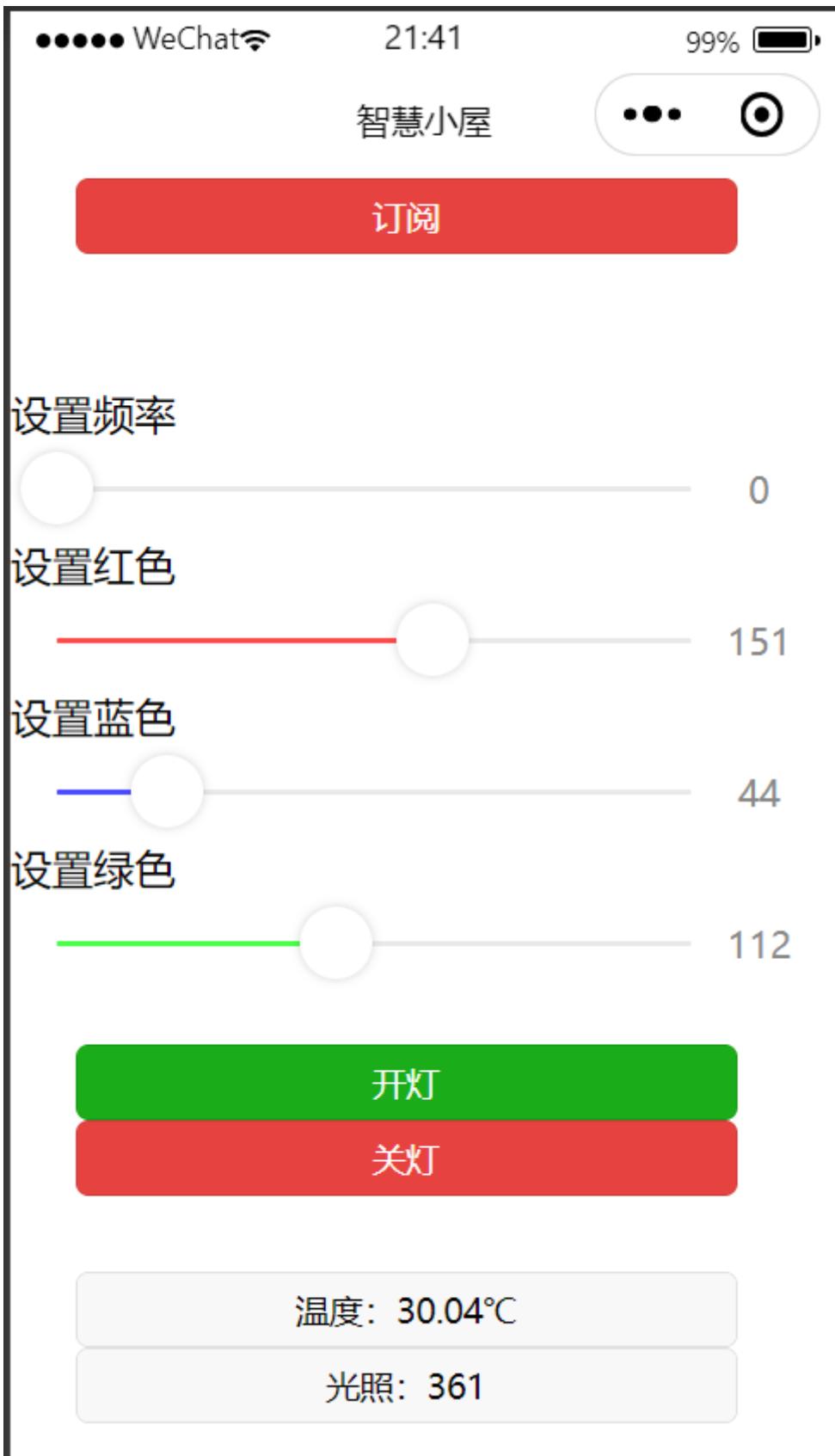
```
sendCommand(cmd, data) {  
    let sendData = {  
        id: '12233443',  
        version: '1.0',  
        params:{  
            ColorRed:RGBtemp[1],  
            ColorGreen:RGBtemp[3],  
            ColorBlue:RGBtemp[2]  
        }  
    };  
}
```

- 将自定义关于Wechatcontrol的topic输入程序中对应部分，可顺便将subtopic输入订阅消息主题为接下来消息回报做准备

```
pubTopic: '/k09l88hcMxJ/Wechatcontrol/user/Wechatcontrol ', //发布消息的主题  
subTopic: '/k09l88hcMxJ/Wechatcontrol/user/get ', //订阅消息的主题
```

此处订阅消息主题后缀为get，表示获取消息

- 点击订阅后提示订阅成功，改变三色光参数



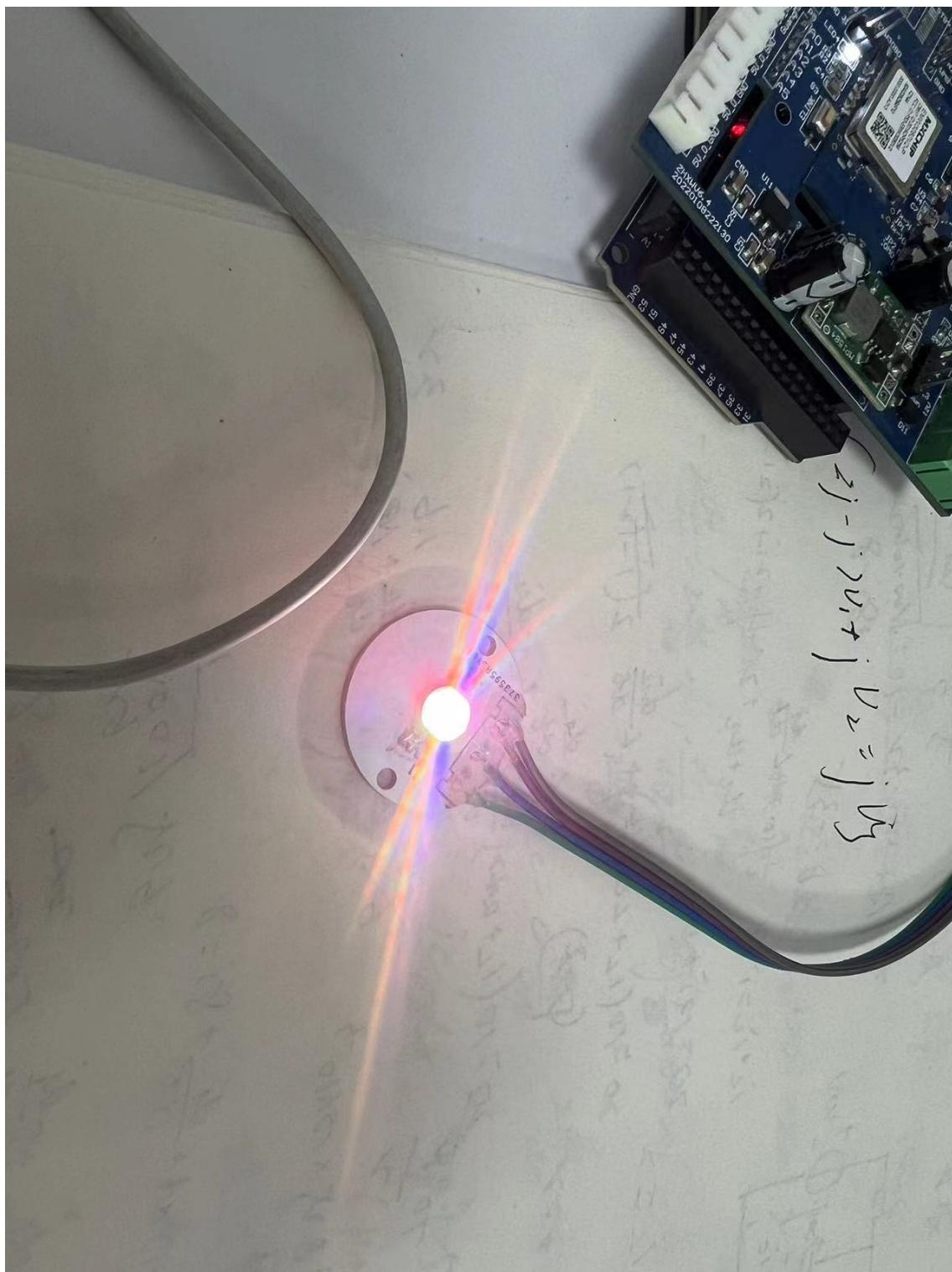
点击开灯后先查看小程序日志

```
/k09l88hcMxJ/Wechatcontrol/user/Wechatcontrol index.js? [br/> {"id":"12233443","version":"1.0","params":{"ColorRed":151,"ColorGreen":112,"ColorBlue":44}} index.js? [br/>>
```

再看aliyun日志

2023/10/31 21:43:00.472	a9fef01c169875978...	1719349...	查看	ArduinoDev	云到设备消息	/sys/k09l88hc...	{"Content":"Pub...	200
2023/10/31 21:43:00.459	a9fef01c169875978...	1719349...	查看筛选相同消息	Wechatcont...	云产品流转	republish	{"Params":"rule...	200
2023/10/31 21:43:00.447	a9fef01c169875978...	1719349...	查看	Wechatcont...	设备到云消息	/k09l88hcMxJ/...	{"Content":"Pub...	200

最后看到实物效果



5. 设备端消息上报，小程序段接收

- 对json.v4文件中上传数据包格式修改为温度和光强数据

```
#define JSON_DATA_PACK_2    "{\"id\":\"110\",\"version\":\"1.0\",\"method\":\"thing.event.property.post\",\"params\":{\"LightDetector\":%d,\"Curtain\":%d,\"Light\":%d,\"Soil\":%d}}\n#define JSON_DATA_PACK_3    "{\"id\":\"110\",\"version\":\"1.0.0\",\"method\":\"sys/%s/thing/event/property/post\",\"params\":{\"temperature\":%d.%02d,\"photores\":%d}}\n#define JSON_DATA_PACK_ALARM \"{\"id\":\"110\",\"version\":\"1.0.0\",\"method\":\"thing.event.GasAlarm.post\",\"params\":{\"GasDetector\":%d}}\\r\"
```

- upload函数已在前面一并修改好，直接上传板子即可
- Arduino串口监视器打印温度和光照强度参数如下

```
13:17:26.267 -> Temperature = 27.63 *C  
13:17:26.267 -> PhotoRes = 609
```

观察阿里云端日志即为数据流转过程

2023/11/01 13:18:10.304	a9fef01f169881589...	1719584...	查看	Wechatcont...	云到设备消息	/k09l88hcMxJ/...	{"Content":"Pub...	200
2023/11/01 13:18:10.293	a9fef01f169881589...	1719584...	查看 筛选相同消息	ArduinoDev	云产品流转	republish	{"Params":"rule...	200
2023/11/01 13:18:10.276	a9fef01f169881589...	1719584...	查看	ArduinoDev	物模型	/sys/k09l88hc...	-	200
2023/11/01 13:18:10.267	a9fef01f169881589...	1719584...	查看	ArduinoDev	设备到云消息	/sys/k09l88hc...	{"Content":"Pub...	200

- 但此时小程序端还没有收到反馈，原因是小程序代码中没有解析消息格式的代码段，因此加入解析格式

```
//接收消息监听  
  
this.data.client.on("message", function (topic, payload) {  
    console.log(" 收到 topic:" + topic + " , payload :" + payload);  
    if(topic=='/k09l88hcMxJ/Wechatcontrol/user/get'){  
        that.setData({  
            //转换成JSON格式的数据进行读取  
            temperature: JSON.parse(payload).items.temperature.value,  
            photores: JSON.parse(payload).items.photores.value,  
        })  
    }  
}
```

- 再次编译小程序端之后发现开始收到下发的日志

```
收到 topic:/k09l88hcMxJ/Wechatcontrol/user/get , payload : index.js? [sm]:71  
{"deviceType": "CustomCategory", "iotId": "SfR9DvJnvbzbbtoaMc1Pk09l80", "requestId": "110", "checkFailedData":  
{}, "productKey": "k09l88hcMxJ", "gmtCreate": "1698816052025", "deviceName": "ArduinoDev", "items": {"photores":  
{"time": 1698816052021, "value": 494}, "temperature": {"time": 1698816052021, "value": 28.19}}}
```

小程序界面反馈，温度与光强会根据环境反馈对应数值

智慧小屋

...



订阅

设置频率



0

设置红色



0

设置蓝色



0

设置绿色



0

开灯

关灯

温度: 28.2°C

光照: 505

注意topic复制过程中很可能出现不必要空格，导致程序无法获取正确数据，注意对topic格式及devicename等反复检查

