

### 1. 设备的wifi连接，联网

```
import network
```

```
def do_connect():
```

```
    sta_if = network.WLAN(network.STA_IF);
```

```
    ap_if = network.WLAN(network.AP_IF);
```

```
    if ap_if.active() == True:
```

```
        ap_if.active(False)
```

```
    sta_if.active(True)
```

```
    sta_if.connect('wifi-名称','wifi密码')
```

```
    while sta_if.isconnected() == False:
```

```
        pass
```

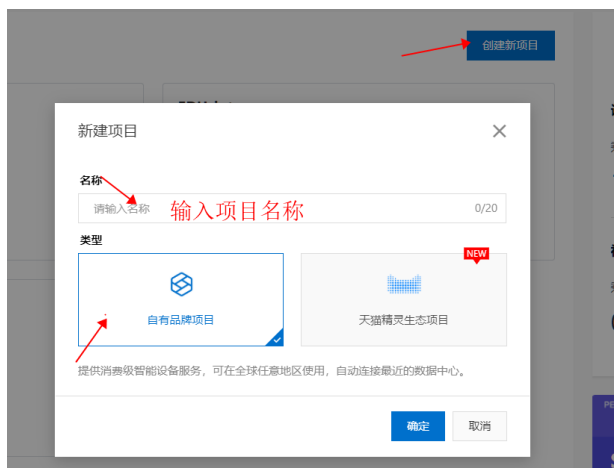
```
    print(sta_if.ifconfig())
```

### 2. 阿里云平台注册/登录，本次实训中不需要付费，如遇到付费情况，请取消操作。

<https://living.aliyun.com/>

可以用淘宝等账户登陆，但是需要授权

### 3. 在阿里云平台创建项目



之后单击进入项目-产品管理

### 4. 创建产品：代表的是一类/一款设备的总称，不是一个实际的设备



新建产品

产品信息

\* 产品名称  
skids\_led

\* 所属品类  
电工照明 / 灯 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

确认 取消

产品管理

请输入产品名称、PK 全部状态 全部连接方式

skids\_led

开发中 WiFi 灯

ProductKey: a1RYOMWrWvX

创建时间: 2020-11-29

单击产品，进入产品配置

选择“下一步”到设备调试，设备调试中选择“乐鑫”

选择认证模组/芯片

模组 芯片

常用

iComm

利尔达

BroadLink

乐鑫/Espresif

欧智通

晶讯

四川爱联科技有限公司

品牌: 上海庆科  
型号: EMW3072 详情

品牌:  
型号:


然后选择模组/芯片

选择认证模组/芯片 ?


模组

芯片


乐鑫/Espressif




品牌: 乐鑫/Espre...  
型号: ESP-WR... [详情](#)




品牌: 乐鑫/Espre...  
型号: ESP32-... [详情](#)



品牌: 乐鑫/Espre...  
型号: ESP-WR... [详情](#)



品牌: 乐鑫/Espre...  
型号: ESP32-... [详情](#)



品牌: 未认证

上一步

下一步

5. 创建设备：是产品的一个实例，是一个真实的设备

上步中“设备调试”中选择“新增测试设备”



品牌: 乐鑫/Espressif  
认证类型: 阿里云IoT技术认证  
[查看详情](#)

型号: ESP32-WROOM-32D  
检测项: 常温射频 (传导), Wi-Fi兼容, 空口...

设备端开发



已选择认证模组  
[下载SDK](#)



嵌入式开发  
[设备端开发指南](#)

测试设备

产品开发阶段允许添加最多50个测试设备，上线发布后将不再限制设备接入数。

已添加设备0/50

在线调试

新增测试设备

华东2(上海)

新加坡

德国(法兰克福)

美国(弗吉尼亚)

DeviceName	状态	最后上线时间	操作
------------	----	--------	----



数据

- 基本
- 所属
- 节点
- 通讯
- 数据
- Prod
- Prod
- Prod
- 认证
- 功能
- 标准
- 自定
- 创建

新增测试设备

DeviceName可以是MAC地址、IMEI号或自定义SN等，须确保产品下唯一，为空将由系统自动颁发，您可以烧录到设备中，并上报到云端进行鉴权认证。

DeviceName ?

skids\_led\_test

确定 取消

完成设备创建，并得到设备参数：

新增测试设备

新增成功

设备证书，请烧录到设备中

ProductKey:	a1RYOMWrWvX	复制
DeviceName:	skids_led_test	复制
DeviceSecret:	5c43c6a2098c9e5888dc3fcc841a00e8	复制

确定

设备参数需要保持，当然也可以才设备管理界面找到相关参数：

a1RYOMWrWvX

skids\_led\_test

5c43c6a2098c9e5888dc3fcc841a00e8

在当前窗口右侧部分找到关键参数secret

**skids\_led**

更新时间：2020-11-29 16:07:18

基本信息 [编辑](#)

所属分类：灯

节点类型：设备

通讯方式：WIFI

数据格式：ICA标准数据格式（推荐）

Product Key: a1RYOMWrWvX

Product Secret: fjZ4QPghRugfz6xv [复制](#) [隐藏](#)

Product Id: 6609891

认证方式：设备密钥

功能定义 [查看](#)

标准功能：9

自定义功能：0

创建时间：2020-11-29

此时我们拥有4个与产品相关的参数。

6. 根据产品、设备的参数，得到skids登陆服务器所需参数。此时skids就代表着所创建的设备

```
1 # todo 1
2 ProductKey = 'a1h68Ao3GZ7'
3 ClientId = "123456|securemode=3,signmethod=hmacsha1|"
4 # todo2
5 DeviceName = 'test1'
6 # todo3
7 DeviceSecret = '1044ed1deba3f7ad33891d337e89f691'
8 #strBroker = ProductKey + ".iot-as-mqtt.cn-shanghai.aliyuncs.com"
9 strBroker = '106.15.83.29'
10 Brokerport = 1883
11 user_name = DeviceName+"&"+ProductKey#用户名
12 # todo4
13 user_password = 'C037F3C23C0F580FE7C5D8E415E00A7783F9C48E'#用户密码
```

如上代码段，需要自行完成todo1~todo4，其中todo1~todo3可从上面参数中得到。  
所以我们要通过其他工具得到password



88413c66e471bec...7.zip  
6.39KB

工具下载解压后得到mqtt签名工具，执行“sign.html”进入配置界面，根据上述参数完成配置

## 填入设备信息：

productKey:   
deviceName:   
deviceSecret:   
timestamp:   
clientId:   
method:

点击这里:



## 签名结果：

password:

## 7. 在skids上利用上述参数，与阿里云平台建立连接，登陆云平台



simple.py  
6.25KB

需要先下载上面的文件到skids上，然后使用如下代码段完成设备与阿里云的连接

```
from simple import MQTTClient
```

```
if __name__ == '__main__':
```

```
    do_connect()
```

```
    client = MQTTClient(CLIENT_ID,SERVER,PORT,username,password,60)    #create  
a mqtt client
```

```
    client.set_callback(sub_cb)                #set callback
```

```
    print("try connect to mqtt")
```

```
    client.connect()                        #connect mqtt
```

```
    print("mqtt connect done")
```

## 8. 云平台与skids进行通信，发送灯的控制命令

### 1. 订阅消息

```
client.subscribe("/sys/" + ProductKey + "/test1/thing/service/property/set")#订阅  
主题
```

### 2. 云平台发布消息

在设备调试界面-》测试设备中选择在线调试



认证类型：阿里云IoT技术认证

检测项：常温射频（传导），Wi-Fi兼容，空口...

[查看详情](#)

### 设备端开发



已选择认证模组

[下载SDK](#)



嵌入式开发

[设备端开发指南](#)

### 测试设备

产品开发阶段允许添加最多50个测试设备，上线发布后将不再限制设备接入数。

已添加设备 1/50

[在线调试](#)

[新增测试设备](#)

[华东2\(上海\)](#)

[新加坡](#)

[德国\(法兰克福\)](#)

[美国\(弗吉尼亚\)](#)

或

DeviceName	状态	最后上线时间	操作
skids_led_test	● 未激活	-	<a href="#">查看</a> <a href="#">调试</a> <a href="#">设备证书</a>

启动skids上的skids\_iot\_test.py程序



simple.py  
6.25KB



skids\_iot\_test.py  
1.86KB

skids与服务器建立连接

Blockly\_iot / skids\_led / 设备调试 / 在线调试

← 在线调试

设备调试

调试设备: skids\_led\_test

调试真实设备

调试虚拟设备

属性调试

服务调用

调试功能: 请选择 方法: 请选择

1

实时日志

● 在线 (真实设备)

自动刷新 ☒

刷新

清屏

类型 / 时间	内容
---------	----

9. skids根据灯的控制命令完成对红色LED灯的控制。

调试设备: skids\_led\_test

调试真实设备

调试虚拟设备

属性调试

服务调用

调试功能: 开关 (powerstate) 方法: 设置

1 {  
2 "powerstate": 0  
3 }

单击发送指令

```
19 #接收信息。接收到的信息是json格式，要进行
20 =def recvMessage(topic,msg):
21     parsed=ujson.loads(msg)
22     str=parsed["params"]
23     print(str)
24     print(type(parsed["params"]))
25     print(str.get("powerstate"))
26     global state
27     state=str.get("powerstate")
28 = if state == 1:
29     print("led on!")
30 = if state == 0:
31     print("led off!")
32

exec(open('/iot_test.py').read(),globals())
wifi connect done
clientid: 123456|securemode=3,signmethod=hmacsha1|
Broker: 106.15.83.29
User Name: skids_led_test&a1RYOMWrWvX
Password: 6F19164AB24898DA49DAD621C7F61590706E0995

mqtt connect done
{'powerstate': 0}
<class 'dict'>
0
led off!
```

在print位置增加对LED的控制即可完成对灯亮灭的控制。