



中国电信天翼 OS 软件开发指南 (MQTT)

V1.0

中国电信集团公司

2019 年 09 月

文档信息

文档名称	中国电信天翼 OS 软件开发指南_MQTT
文件编号	V1.0 （内部版本号 1.0.1.1）
编制人	杜海洋、张海名、王艺等
保密级别	供使用天翼 OS 接入中国电信物联网开放平台的开发者使用

分发范围：终端开发者

目录

概要说明.....	2
使用说明.....	2
前提条件.....	2
操作步骤.....	2
一、 下载天翼 OS 工程文件.....	2
二、 创建产品/设备，属性和服务.....	3
三、 获取设备认证信息和物模型信息.....	4
四、 生成终端 SDK 及 Demo 程序.....	5
五、 开发及编译应用程序.....	6
附录 A：AutoCodeTool 使用说明.....	8
附录 B：其他说明.....	8

概要说明

天翼 OS (ESurfingOS) 是中国电信与华为合作开发的完整、标准化物联网操作系统，具备极致性能、极简开发、云端一体、丰富组件、安全防护等关键能力，支持终端设备快速无缝连接到中国电信物联网开放平台功能，可广泛应用与芯片/模组等生态设备厂家和智慧家庭/智慧城市/工业物联网等各个行业应用领域。

现有天翼 OS MQTT 版本包含功能：动态 JSON 物模型 API 自动生成模块：支持用户定义属性服务自动生成动态 API；透传设备 API 基础模块：初始化，登录，数据上报，指令下发，样例使用程序 Demo；非透传设备 API 基础模块：初始化，登录，数据上报，事件上报，指令下发，指令下发应答，样例使用程序 Demo。

使用说明

前提条件

开发语言：C 语言；环境平台：Windows10 + Keil uVision5+野火 STM32F429IGT6

操作步骤

一、下载天翼 OS 工程文件

工程文件目录如下图所示：

名称	修改日期	类型	大小
arch	2019/9/18 15:48	文件夹	
bin	2019/9/18 15:48	文件夹	
build	2019/9/18 15:48	文件夹	
components	2019/9/18 15:48	文件夹	
demos	2019/9/18 15:48	文件夹	
doc	2019/9/18 15:48	文件夹	
include	2019/9/18 15:48	文件夹	
kernel	2019/9/18 15:48	文件夹	
osdepends	2019/9/18 15:48	文件夹	
targets	2019/9/18 15:48	文件夹	
.travis.yml	2019/9/18 15:48	YML 文件	1 KB
LICENSE	2019/9/18 15:48	文件	2 KB
README.md	2019/9/18 15:48	MD 文件	1 KB

说明：

根目录位置在下文中为了方便描述会以.\代替，其中

bin 中存放的是代码生成工具；

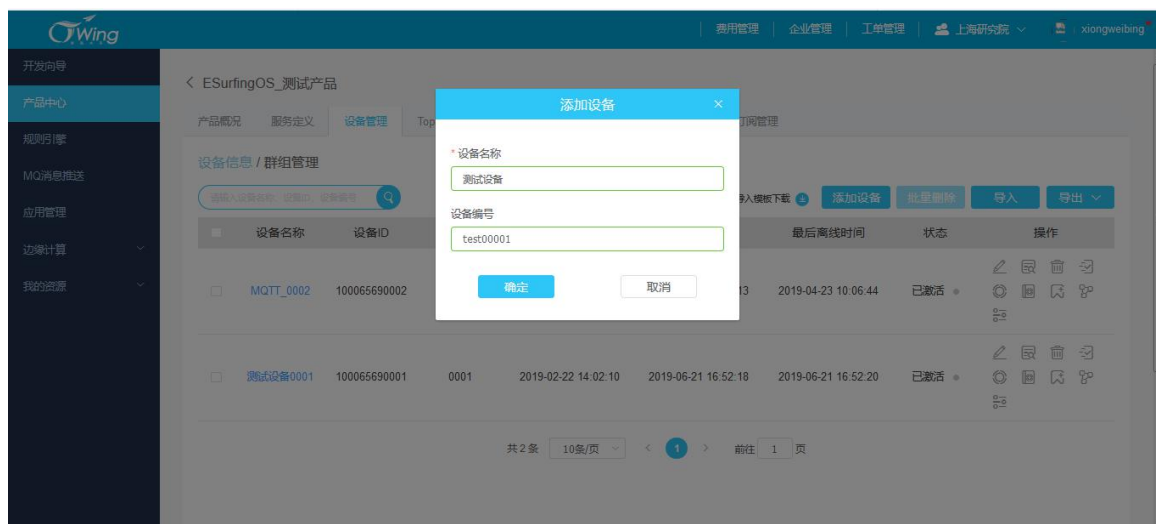
components 下的 connectivity\ctiot\ctiot_mqtt 为接入物联网开放平台的 API 代码；

demos 下的 ctiot_mqtt 存储示例程序；

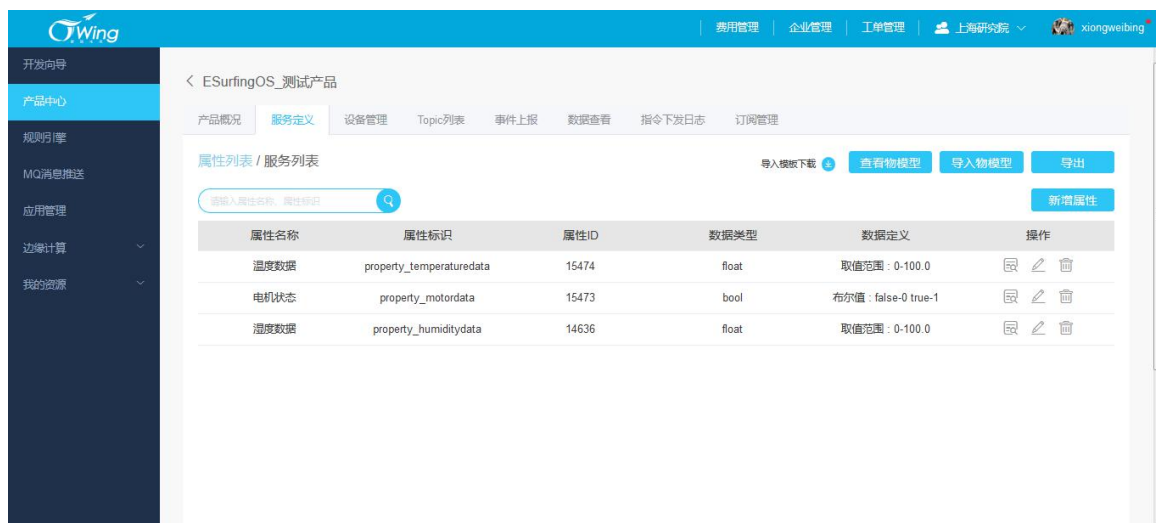
targets 中有用户主程序入口及工程配文件（支持 keil 和 GCC 编译）。

二、创建产品/设备，属性和服务

在平台创建产品和设备：



在平台创建属性和服务（MQTT 透传设备不用设置）：



三、获取设备认证信息和物模型信息

在物联网开放平台的服务定义中查看物模型，将物模型 JSON 内容存为 JSON 文件（编码格式必须为 UTF-8），并根据设备认证信息修改设备信息 JSON 文件。将这两个 JSON 文件放置到\bin 文件中，非透传设备使用）如下所示：



LiteOS_demo633

产品概况 服务定义 设备信息管理 Topic列表 事件上报 数据查看 指令下发日志 订阅管理

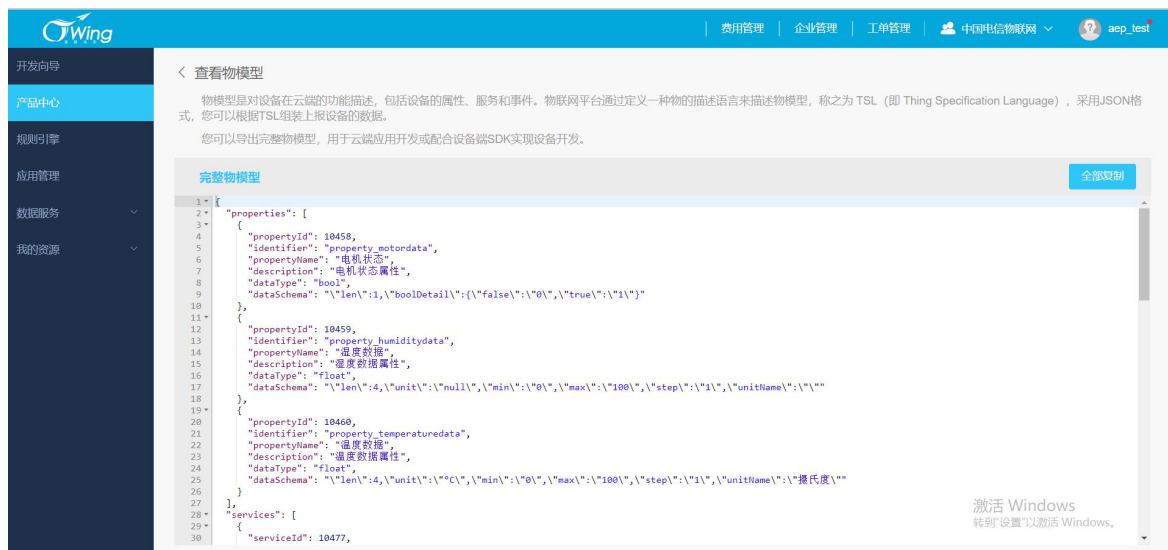
属性列表 / 服务列表

请输入服务名称、服务ID、服务标识 请选择服务类型

服务名称	服务类型	服务ID	服务标识	参数列表	操作
服务名称_指令下发响应	指令下发响应	10480	service_cmdnresponse		眼 编辑 删除
服务名称_指令下发	指令下发	10479	service_cmddn		眼 编辑 删除
服务名称_事件上报	事件上报	10478	service_eventreport		眼 编辑 删除
服务名称_数据上报	数据上报	10477	service_datereport		眼 编辑 删除

共 4 条 10条/页 < 1 > 前往 1 页

查看物模型 新增服务



查看物模型

物模型是对设备在云端的功能描述，包括设备的属性、服务和事件。物联网平台通过定义一种物的描述语言来描述物模型，称之为 TSL（即 Thing Specification Language），采用JSON格式，您可以根据TSL组装上报设备的数据。

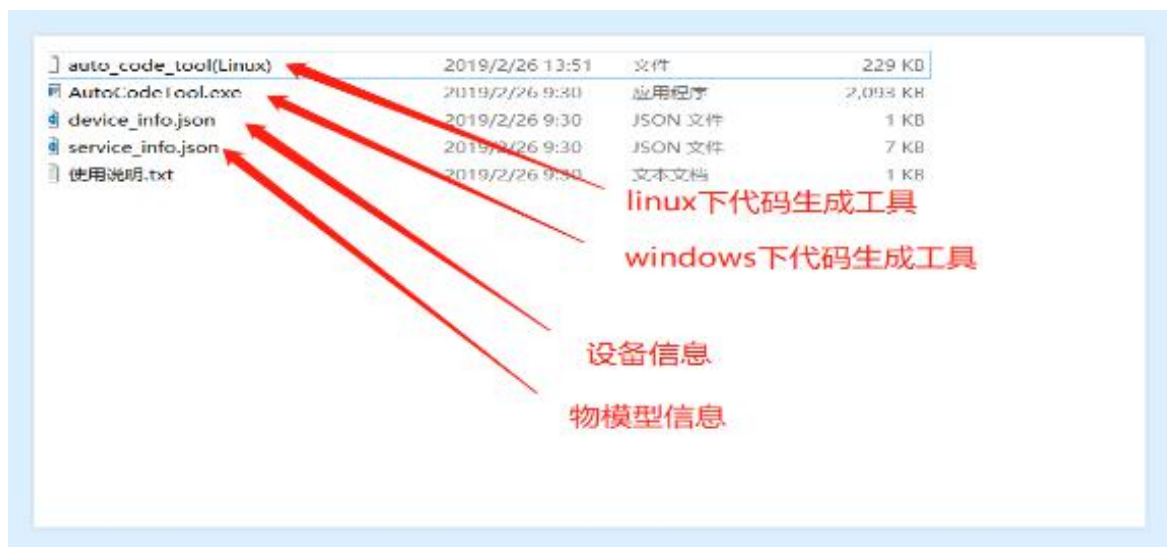
您可以导出完整物模型，用于云端应用开发或配合设备端SDK实现设备开发。

完整物模型

```
1- {
2-   "properties": [
3-     {
4-       "propertyId": 10458,
5-       "identifier": "property_motordata",
6-       "propertyName": "电机状态",
7-       "description": "电机状态属性",
8-       "dataType": "bool",
9-       "dataSchema": "{\"len\":1,\"boolDetail\":{\"false\":\"0\",\"true\":\"1\"}}"
10-     },
11-     {
12-       "propertyId": 10459,
13-       "identifier": "property_humiditydata",
14-       "propertyName": "湿度数据",
15-       "description": "湿度数据属性",
16-       "dataType": "float",
17-       "dataSchema": "{\"len\":4,\"unit\":\"\",\"min\":\"0\",\"max\":\"100\",\"step\":\"1\",\"unitName\":\"\""
18-     },
19-     {
20-       "propertyId": 10460,
21-       "identifier": "property_temperaturedata",
22-       "propertyName": "温度数据",
23-       "description": "温度数据属性",
24-       "dataType": "float",
25-       "dataSchema": "{\"len\":4,\"unit\":\"°C\",\"min\":\"0\",\"max\":\"100\",\"step\":\"1\",\"unitName\":\"摄氏度\""
26-     }
27-   ],
28-   "services": [
29-     {
30-       "serviceId": 10477,
```

全部复制

激活 Windows
转到“设置”以激活 Windows。



四、生成终端 SDK 及 Demo 程序

工程中默认携带 MQTT 透传设备 API 和 Demo 程序，如果用户选择使用 MQTT 透传设备功能，则无需操作此步骤；

当用户选择使用 MQTT 非透传设备集成物模型功能时，使用工具 AutoCodeTool.exe 生成 API 接口、Demo 程序及用户配置信息（工具使用方法参见[附录 A](#)）。

```
C:\Windows\System32\cmd.exe
Microsoft Windows [版本 10.0.18362.295]
(c) 2019 Microsoft Corporation。保留所有权利。

E:\AEP项目GIT\ESurfingOS\bin>AutoCodeTool -p service_info.json -s device_info.json -d ../ -c
Welcome to auto code tool
Current Version:1.0.0
propertyFileName:service_info.json
serviceFileName:device_info.json
destination path:../
Reading transparent file...
transparent:0
begin read property file...
begin write .h file...
H file:../include/ctiot_mqtt/ctiot_mqtt_client.h
begin write .c file...
c file:../components/connectivity/ctiot/ctiot_mqtt/ctiot_mqtt_client.c
process finished...
creating demo files...
creating device info...
H file:../demos/ctiot_mqtt/ctiot_device_info.h
createing demo c file...
demo c file:../demos/ctiot_mqtt/ctiot_mqtt_demo.c
create demo finished...

E:\AEP项目GIT\ESurfingOS\bin>
```

查看生成好的文件是否已放置到指定目录：

ctiot_mqtt_client.c（API 代码）放置在.\components\connectivity\ctiot\ctiot_mqtt目录下；

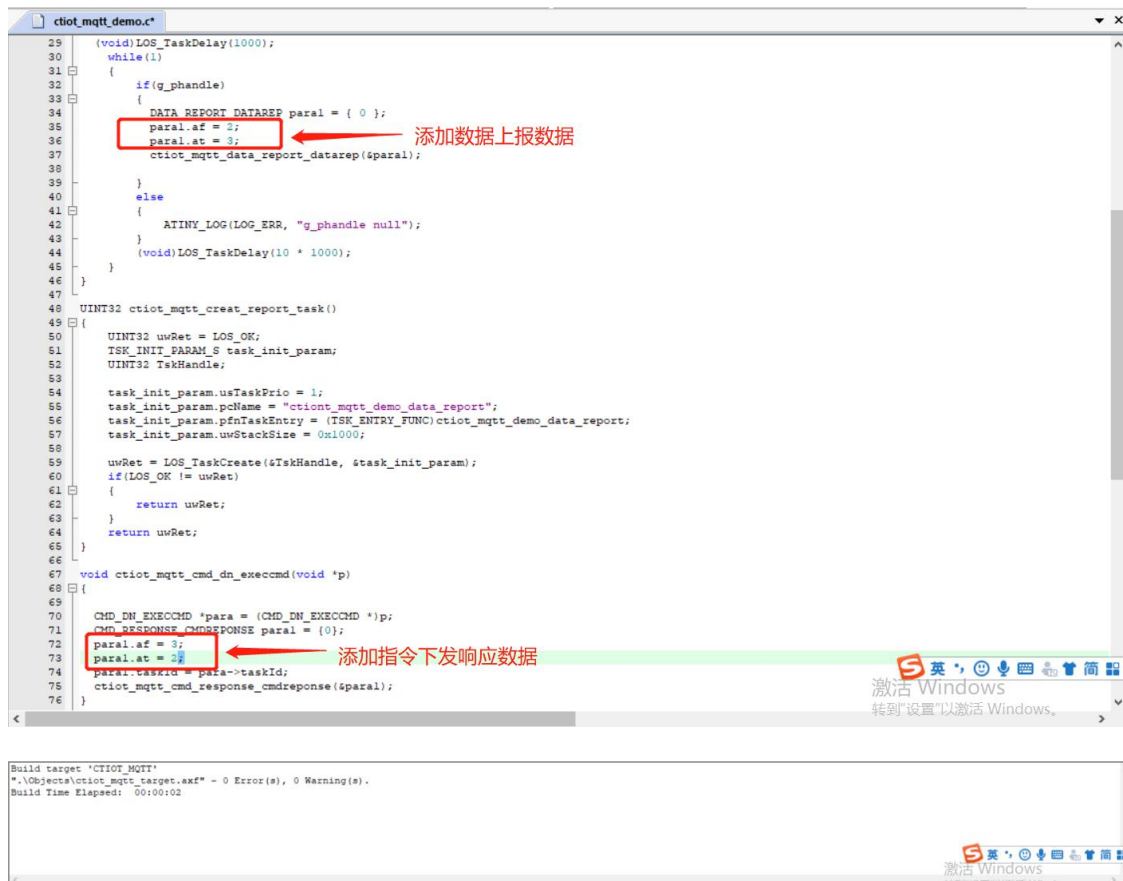
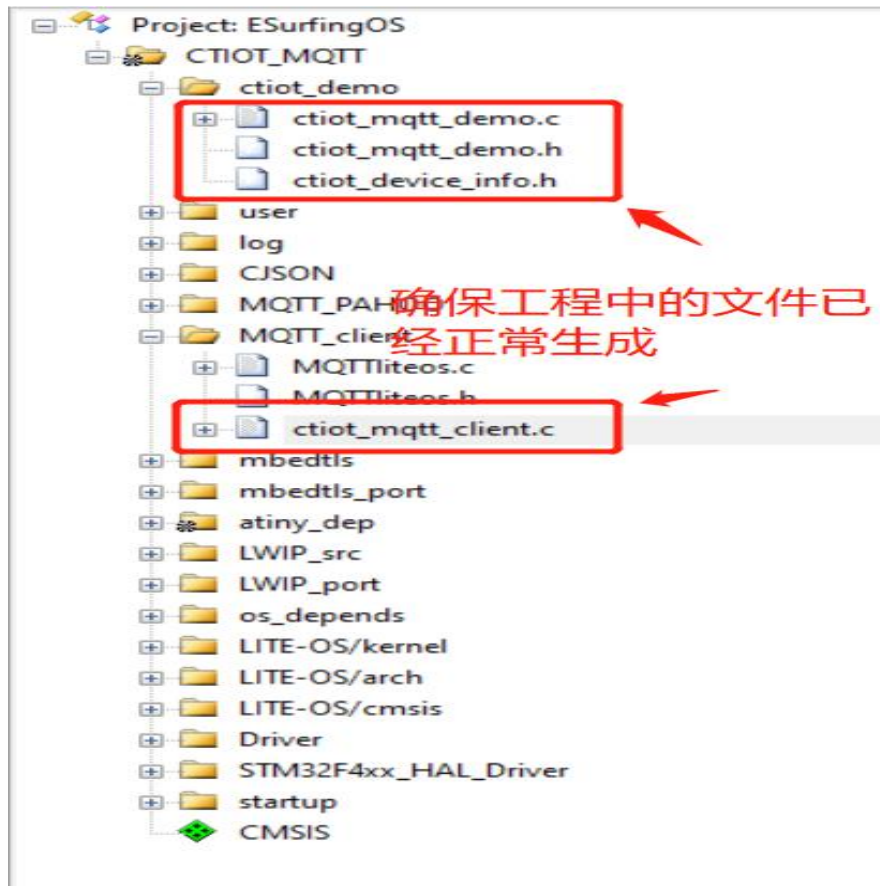
ctiot_mqtt_client.h（API 头文件）放置在.\include\ctiot_mqtt目录下；

ctiot_device_info.h（设备信息）以及 ctiot_mqtt_demo.c（demo 实现代码）放置在.\demos\ctiot_mqtt目录中。

五、开发及编译应用程序

首先使用 Keil 打开工程（.\targets\ctiot_STM32F429IGTx_FIRE\MDK-ARM），确保已将相关文件引入工程，然后根据自身产品场景需求修改工程 demo 程序形成对应的产品业务代码，最后通过 Keil 编译并下载程序调试。

备注：调试验证通过的开发板有野火 STM32F429IGT6。



附录 A: AutoCodeTool 使用说明

AutoCodeTool 工具的使用:

[AutoCodeTool]使用参数说明 AutoCodeTool -p service_info.json -s device_info.json -d ../ -c

-p/P service_info.json 文件

-s/S device_info.json 文件

-d/D 天翼 OS 根目录[目前保持 AutoCodeTool 路径不变 (.bin) 的情况下, 后面的-d ../可加可不加]

-c/C 生成 demo 源文件, 默认不生成

附录 B: 其他说明

device_info.json 参数说明:

```
{
  "DeviceNo": "0001", //设备编号
  "DeviceID": "100065690001", //设备 ID
  "Protocol": "MQTT", //通信协议
  "Authorization": "token", //认证方式
  "Token": "TMsF33Vg0P28kFki93igA1OiGz0LGJvt3g1s29Ow8m0", //特征串
  "Transparent": 0, //是否为透明传输设备 (0 为非透明传输设备, 1 为透明传输设备)
  "TCPAddress": "www.ctwing.cn", //设备接入地址
  "TCPPort": 1883, // 设备接入端口
  "TLSAddress": "www.ctwing.cn", //设备 TLS 接入地址
  "TLSPort": 8883 //设备 TLS 接入端口
}
```

API 帮助文档说明:

需要查看 API 说明, 可以使用 doxygen 工具生成相应的 API 说明文档 (配置文件使用 .\targets\ctiot_STM32F429IGTx_FIRE\doc\Doxyfile)