

基于 AgentTiny SDK 实现 OpenHarmony 对接华为云

1 环境搭建

1.1 硬件环境

BearPi-HM_Nano 开发板、Type-C 数据线、PC

1.2 软件环境

参考如下链接：

<https://gitee.com/openharmony/docs/tree/master/zh-cn/device-dev/quick-start>



图 1

注意

(1) 如果想要使用 DevDevEco Device Tool 参与调试的需要安装 windows 环境

(2) 获取源码：如果不使用 Docker 方式，强烈建议采用获取方式一：从码云仓库获取，但是在最新的 OpenHarmony 3.0，本人还没有发现如何获取到最小系统源码，因此使用其命令也会下载的是全量代码。方式二：本人没有玩转（也有可能是本人环境搭建的没有那么好），方式三：镜像站点速度虽然比较快，但是部分三方库代码还没有合入。方式四：github 实在是下不动。以上纯属个人操作建议。

(3) 在安装开发板中的环境的安装 gcc_riscv32 步骤时，强烈建议直接下载 gcc_riscv32 镜像，否则可能出现某些插件安装失败（下载失败）具体如下图 2 所示。



图 2

2 HuaweiCloud IoTLink SDK 使用

(1) 下载源码 https://gitee.com/xingli_chen/oh3.0_hwcloud_iotlink.git

(2) 将上述源码直接拷贝到 OpenHarmony 源码的 third-party 文件目录下，此处需要注意的是 OpenHarmony 和 HuaweiCloud IoTLink SDK 共同具备的三方库直接使用 OpenHarmony 的库文件，如 cJSON 和 mbedtls。

(3) 在 OpenHarmony 3.0 源码 device\bearpi\bearpi_hm_nano\app\BUILD.gn 中添加如下代码，根据#是否注释来决定具体编译 demo 的选择。

```
# "//third_party/hwcloud_iotlink/demos/test_iotlink:test_iotlink",
"//third_party/hwcloud_iotlink/demos/mqtt_tiny_v5_agriculture_demo:mqtt_tiny_v5_agricul
ture_demo",
```

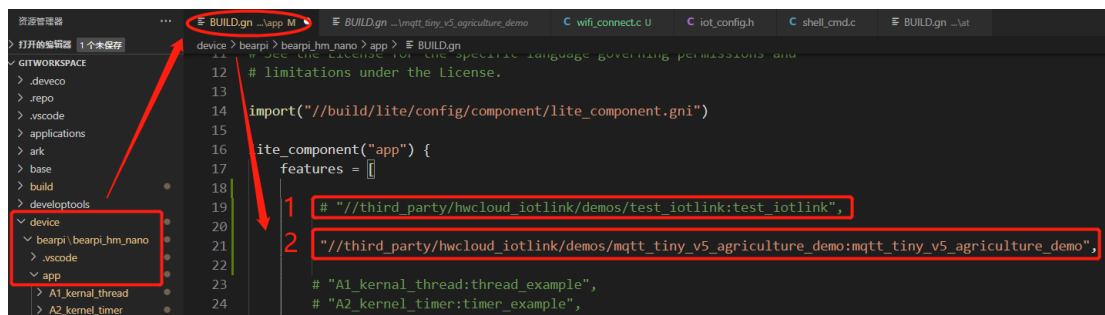


图 3

图 3 中 1 为单元测试代码，可以根据修改 test_main.c 文件中最后一行代码中，APP_FEATURE_INIT()中的入口函数，来进行不同的测试用例的执行，分别测试 OSAL 层不同的组件适配，包括任务、互斥锁、信号量、队列、以及 SDK 本身队列等一系列验证，部分内容可以对比 BearPi-HM_Nano 开发板的官方样例代码进行对比验证程序结果的正确性，具体如图 4。

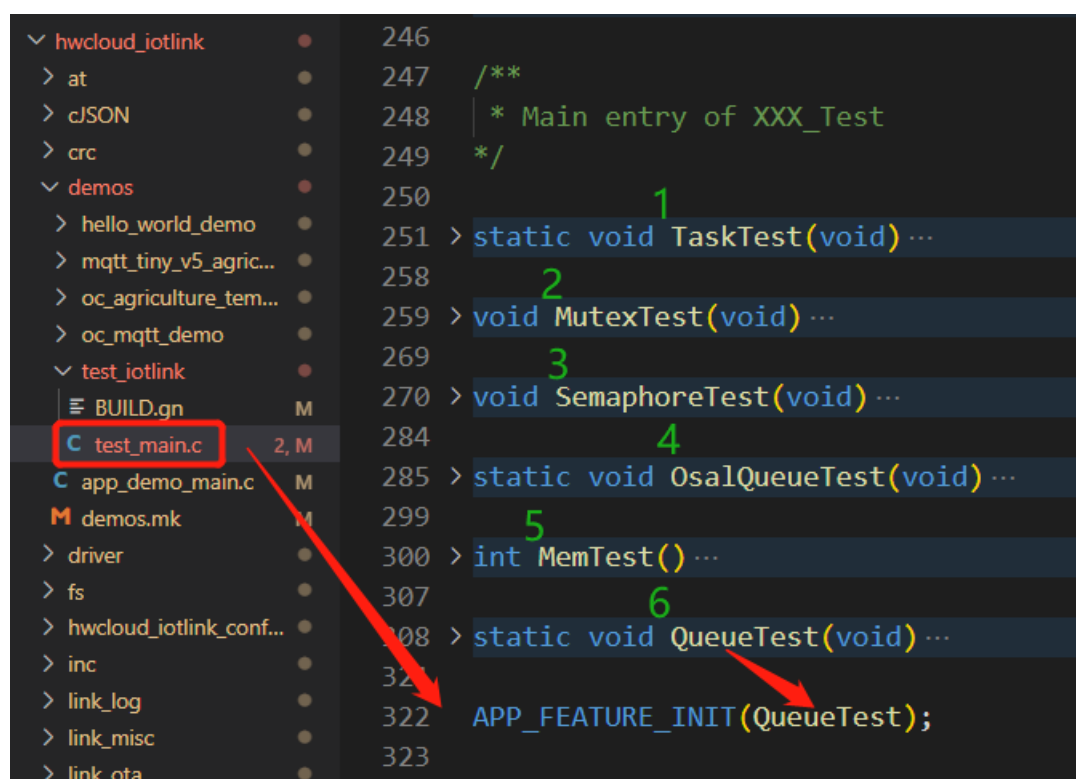


图 4

图 3 中 2 为使用 MQTT 对接华为云的 demo，看一下该 demo 的 BUILD.gn

文件，具体如图 5 所示，其中 A 区域包含开发板硬件和 wifi 相关的库文件，B 区域包括 MQTT 对接华为云必须的一些库文件，如 cJSON，mqtt 相关、osal 相关以及配置库文件，C 区域表示变异该文件时需要编译 hwcloud_iotlink 库，编译是会根据指定路径找到该文件所必要的库以及 C 文件进行编译。

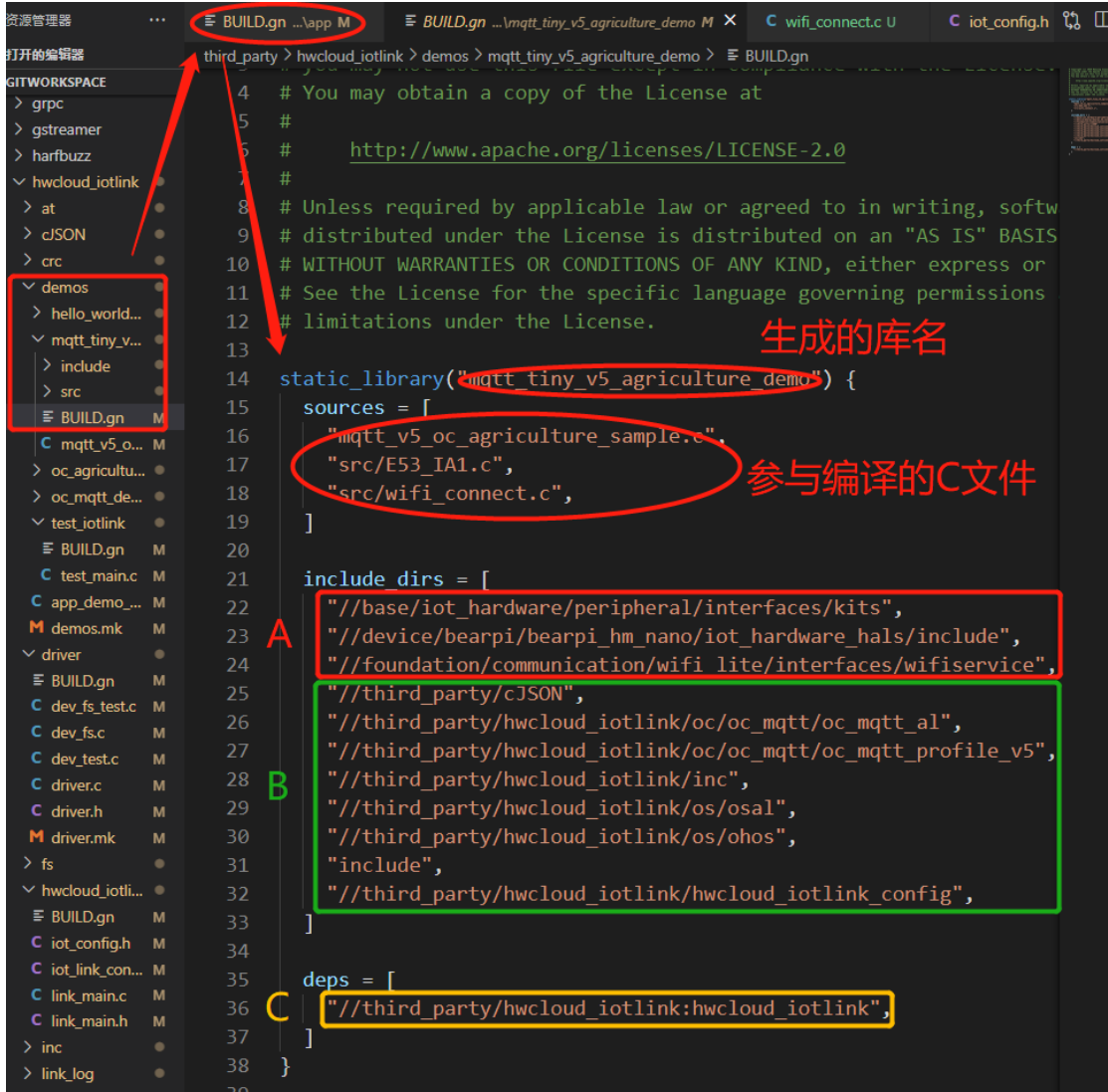


图 5

图 6 中 A 区域表示 MQTT 对接华为云所必须的一些库文件以及 dtls 的库文件，入口函数 `IoTMainTaskEntry()` 中务必先调用 `link_main_task_entry()` 函数，来确保 `osal` 的安装以及其他配置的初始化。

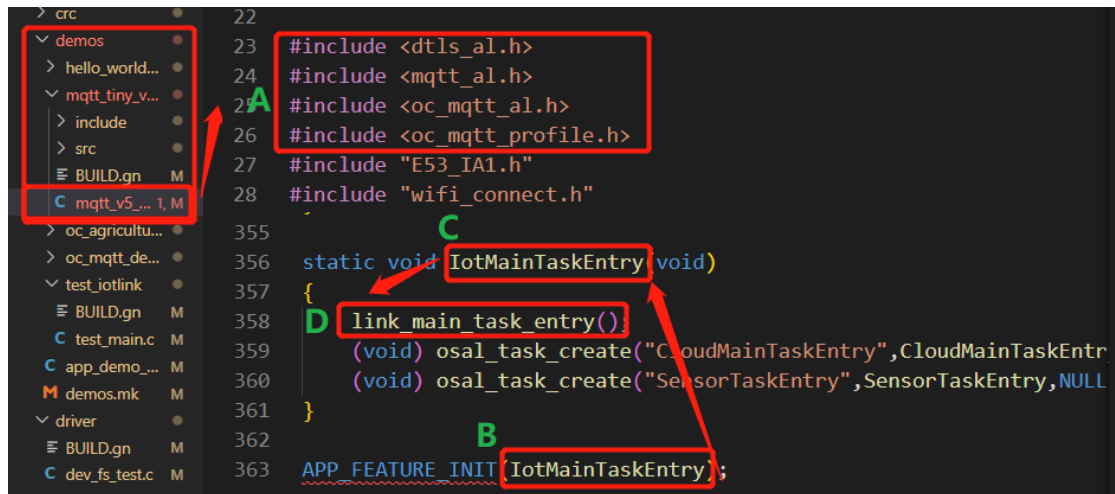


图 6

注意如果需要对接上云需要根据自己的设备修改配网信息，wifi 热点账号以及密码以及自己在云端注册的设备 ID 和设备密钥。



图 7

之后烧录到设备，重启设备（首次使用记得重启两次，第一次烧录的时候可能需要配置一下内部信息），从第二次设备重启之后可以正常对接华为云，具体日志信息如下图 8 所示。

```

[DEBUG][4750][hub_step] hub_step:enter
[DEBUG][4750][dmp_connect] oc_mqtt_connect:server:121.36.42.100 port:1883
[DEBUG][4750][dmp_connect]
oc_mqtt_connect:client_id:60bf519cb86d7b02bc518aa9_202106081919_0_0_1970000100
[DEBUG][4760][dmp_connect] oc_mqtt_connect:user:60bf519cb86d7b02bc518aa9_202106081919
passwd:0d4a2487654978ef1398dc04c77c592bff03940814708036ed0e35194c7ce1fb
[DEBUG][4990][dmp_connect] oc_mqtt_connect:rcode:0 :SUCCESS
[DEBUG][4990][dmp_subscribe] oc_mqtt_subscribe:start
[DEBUG][5000][dmp_subscribe] oc_mqtt_default_subscribe:topic:
$oc/devices/60bf519cb86d7b02bc518aa9_202106081919/sys/commands/#
[DEBUG][5080][dmp_subscribe] oc_mqtt_default_subscribe:retcode:0:success
[DEBUG][5080][dmp_subscribe] oc_mqtt_subscribe:topic:
$oc/devices/60bf519cb86d7b02bc518aa9_202106081919/sys/messages/down
[DEBUG][5170][dmp_subscribe] oc_mqtt_subscribe:retcode:0:success
[DEBUG][5170][dmp_subscribe] oc_mqtt_subscribe:topic:
$oc/devices/60bf519cb86d7b02bc518aa9_202106081919/sys/properties/set/#
[DEBUG][5260][dmp_subscribe] oc_mqtt_subscribe:retcode:0:success
[DEBUG][5260][dmp_subscribe] oc_mqtt_subscribe:topic:
$oc/devices/60bf519cb86d7b02bc518aa9_202106081919/sys/properties/get/#
[DEBUG][5310][dmp_subscribe] oc_mqtt_subscribe:retcode:0:success
[DEBUG][5310][dmp_subscribe] oc_mqtt_subscribe:topic:
$oc/devices/60bf519cb86d7b02bc518aa9_202106081919/sys/shadow/get/response/#
[DEBUG][5390][dmp_subscribe] oc_mqtt_subscribe:retcode:0:success
[DEBUG][5390][dmp_subscribe] oc_mqtt_subscribe:topic:
$oc/devices/60bf519cb86d7b02bc518aa9_202106081919/sys/events/down
[DEBUG][5460][dmp_subscribe] oc_mqtt_subscribe:retcode:0:success
[DEBUG][5460][hub_step] hub_step:ok exit
ret = 0
oc_mqtt_profile_connect_succeed

```

图 8