

空地无人平台各有优劣，形成体系、协同运用可以更好地完成使命任务。课程比赛在此背景下设置赛题。



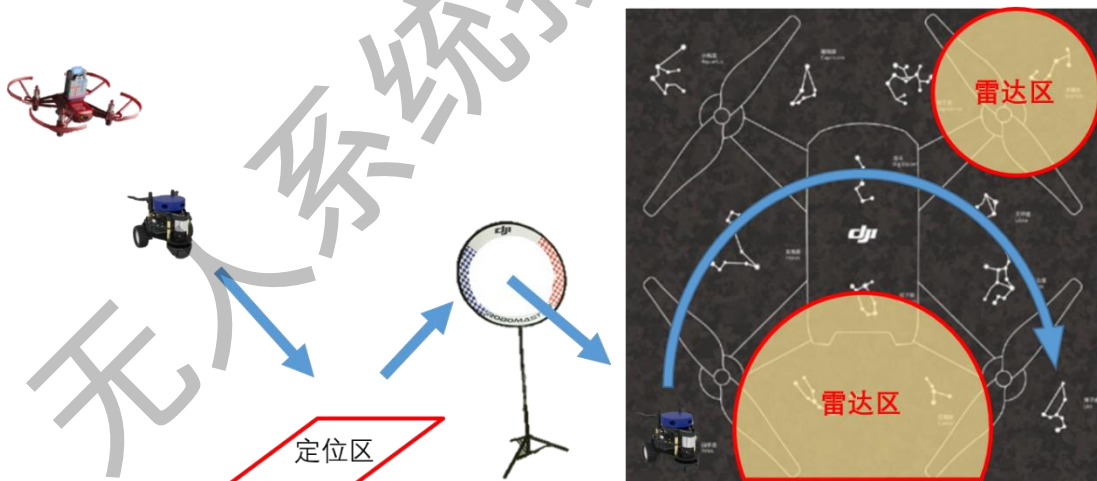
比赛场地如上左图所示，由无人机定位板、航路引导环、无人机 3\*3 定位毯和无人车建图围栏组成。通过裁判系统，无人机会在比赛的阶段通过 LED 模块显示不同的指示灯颜色，以及对应的数字编码。无人车连接纤维板固定引导二维码。

裁判节点 score.py 已经更新在/rmtt\_ros/rmtt\_tracker/src/中。请自行下载更新程序包::

git clone [https://github.com/cavayangtao/rmtt\\_ros.git](https://github.com/cavayangtao/rmtt_ros.git)

或者:

git clone [https://gitee.com/cavayangtao/rmtt\\_ros.git](https://gitee.com/cavayangtao/rmtt_ros.git)



#### 比赛流程:

1. 比赛开始时，需自行开启裁判节点，该节点会进行计时，并控制无人机 LED 模块显示。成功开启该节点，无人机会亮绿色指示灯。
2. 遥控无人车，利用搭载的二维码将无人机引导至定位板上方。无人机成功识别定位板，第一阶段任务完成，LED 模块将显示“1”。
3. 使无人机自行上升旋转，搜索航路引导环，跟踪并穿越该引导环进入定位毯。
4. 进入定位毯后，无人机自主导航至路径规划起始点。无人机成功到达起始点，第二阶段任务完成，LED 模块将显示“2”。
5. 通过路径规划算法，规划路线避开雷达区域，使无人机自主到达任务终点。到达终

点，第三阶段任务完成，LED 模块将显示“3”。此时裁判节点计时将停止：

```
[INFO] [1653036764.924373]: test time: 0:00:46.582298
[INFO] [1653036765.029639]: test time: 0:00:46.582298
[INFO] [1653036765.134041]: test time: 0:00:46.582298
```

注意：一旦无人机进入雷达区，LED 模块将亮黄灯。

附加题：使在路径规划起点等待的无人车跟随无人机行驶至任务终点。

任务参数：

起点：x = -1.2, y = -1.2

终点：x = 1.2, y = 1.2

雷达区 1：x = 0, y = -0.6, r = 0.6

雷达区 2：x = 0.3, y = 1.2, r = 0.5

计分：

1. 比赛名次以得分进行排序，得分相同按时间最短排序，每队 3 次机会。
2. 成功引导无人机到定位板上方，20 分。
3. 无人机成功穿越引导环，30 分。
4. 无人机抵达路径规划起始点，10 分。
5. 无人机抵达路径规划终点，20 分。
6. 无人机路径跟踪过程中黄灯未亮起，20 分。
7. 无人车跟随无人机抵达任务终点，20 分。

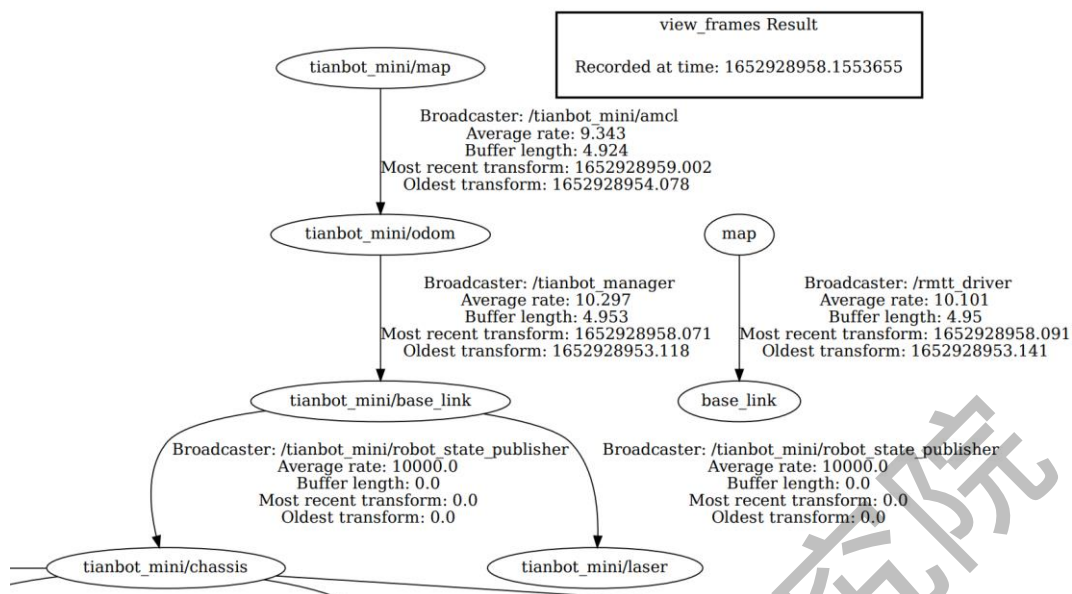
## 空地协同跟踪

请同学们根据提示，尝试实现空地协同跟踪：

1. 将无人机和无人车通过路由（或手机热点，请使用英文热点名）连接至同一电脑。
2. 完成无人车的激光雷达建图（此步亦可在无人车热点模式下提前进行）。建图前请重启无人车，确保里程计清零。无人车放置于 DJI 定位毯(0, 0, 0)位姿处开始建图：



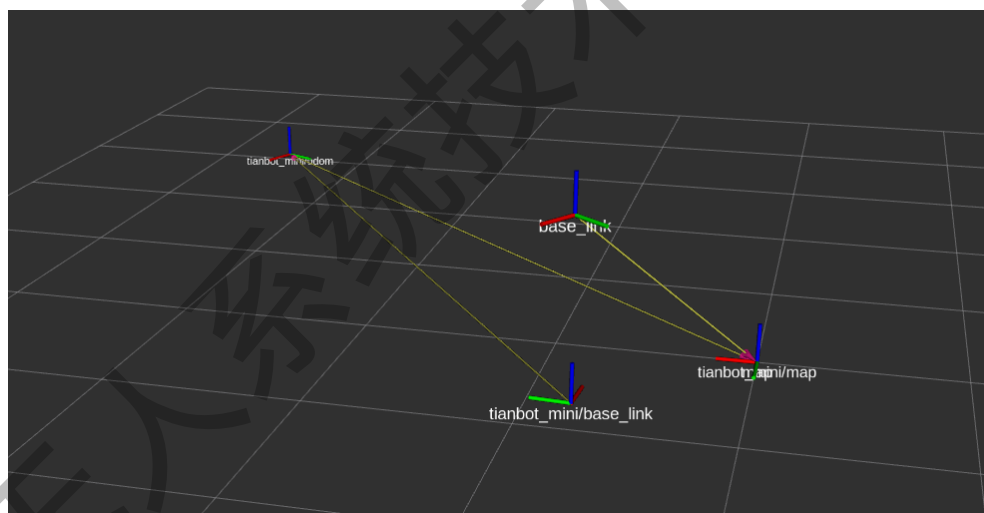
3. 通过 amcl 程序包对无人车进行定位。此时无人车的世界坐标系 frame\_id = tianbot\_mini/map 和无人机世界坐标 frame\_id = map 重合，但未关联。  
如下图所示，通过 rosrun tf2\_tools view\_frames.py 查看 tf 变换关系图，可以看到无人机和无人车的世界坐标系并未关联。



4. 发布静态变换将无人车的世界坐标系和无人机的世界坐标系进行关联。静态变换可以通过在 launch 文件中添加以下语句实现：

```
<node pkg="tf" type="static_transform_publisher" name="map_tf" args="0 0 0 0 0 0 map tianbot_mini/map 100"/>
```

在 rviz 中添加 tf，查看各个坐标系之间的变换关系，可以看到已经获得无人机本体系到无人车本体坐标系的 tf 变换：



5. 还记得第一节的小乌龟跟踪代码吗，由于无人机和无人车的位姿信息已经进行了广播，可直接修改小乌龟跟踪代码中的跟踪节点实现空地跟踪！
6. 遥控无人机看看小车有没有跟随运动。

## Tianbot\_mini 网络配置

首先查看 PC 连接到路由器时的本地 IP：

```
ot@ros2go:~$ ifconfig
flags=4163<IP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.0.208 netmask 255.255.255.0 broadcast 192.168.0.255
    inet6 2409:8a70:b6c:1330:a00:27ff:feb5:4958 prefixlen 64 scopeid
    inet6 fe80::a00:27ff:feb5:4958 prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:b5:49:58 txqueuelen 1000 (以太网)
    RX packets 1475 bytes 565671 (565.6 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 1379 bytes 143928 (143.9 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

打开机器人电源，进入单机模式（机器人会创建 AP 热点：TBMINI-xxx 或 TBMN-xxx）  
系统状态灯为 绿色 或 黄色 闪烁，表示机器人已开机，并进入单机模式。  
打开笔记本 PC，连接 TianbotMini 的 AP 热点（TBMINI-xxx 或 TBMN-xxx）。  
在浏览器中，输入 192.168.1.1/cfg，进入配网页面。

## TIANBOT Config

WiFi SSID(2.4G)	<input type="text" value="TIANBOT-be8"/>
WiFi PASSWORD	<input type="text" value="www.tianbot.com"/>
ROBOT NAME	<input type="text" value="tianbot_mini"/>
<input type="button" value="OK"/>	
设备名称: tianbot_mini	

输入需要连接的 WiFi 网络 SSID（网络名称），PASSWORD（网络密码），配置机器人网络。按下 OK 完成配网，此时机器人会重启。老版固件还需要在 192.168.1.1/cfg 页面中输入 ROS MASTER URI，即本机的 IP 地址。根据以下提示，切换到多机模式（路由器模式）。

### 配置成功，请手动使机器人进入多机器人模式

（请将机器人悬空放置，按住WiFi按钮，同时点击一次重启按钮，看到机器人蓝色指示灯亮起后，松开WiFi按钮）

【机器人仅支持2.4G频段的WiFi网络,多次连接失败，请检查WiFi频段是否为2.4G】

新版固件可随后在 [http://tianbot\\_mini.local/cfg](http://tianbot_mini.local/cfg) 中配置 ROS MASTER URI，此时输入 PC 的 IP 地址，单击确定按钮即可完成配置。机器人指示灯将由红灯转为蓝灯，提示配置成功。

## TIANBOT Config

ROS MASTER URI	<input type="text" value="192.168.0.208"/>
<input type="button" value="OK"/>	

如需更改 WiFi 网络信息，请进入单机模式(AP)

设备名称: tianbot\_mini

以上操作在网络环境不变的情况下，只需要操作一次即可。下次使用时，只需要机器人



进入多机模式即可。机器人会自动连接上次配置过的路由器。

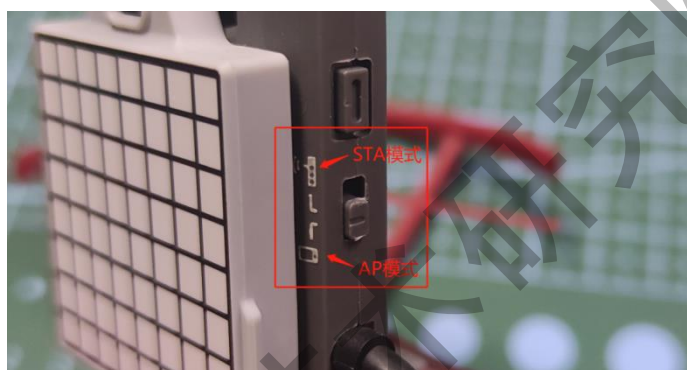
## RoboMaster TT 无人机网络模式配置

TT 无人机具有模式 STA（路由）与 AP（热点）模式。TT 进入 AP 模式并通过本机 wifi 连接后即可进行单机实验（PC 连接到 TT 所创建的 AP 网络：RMTT-xxx）。

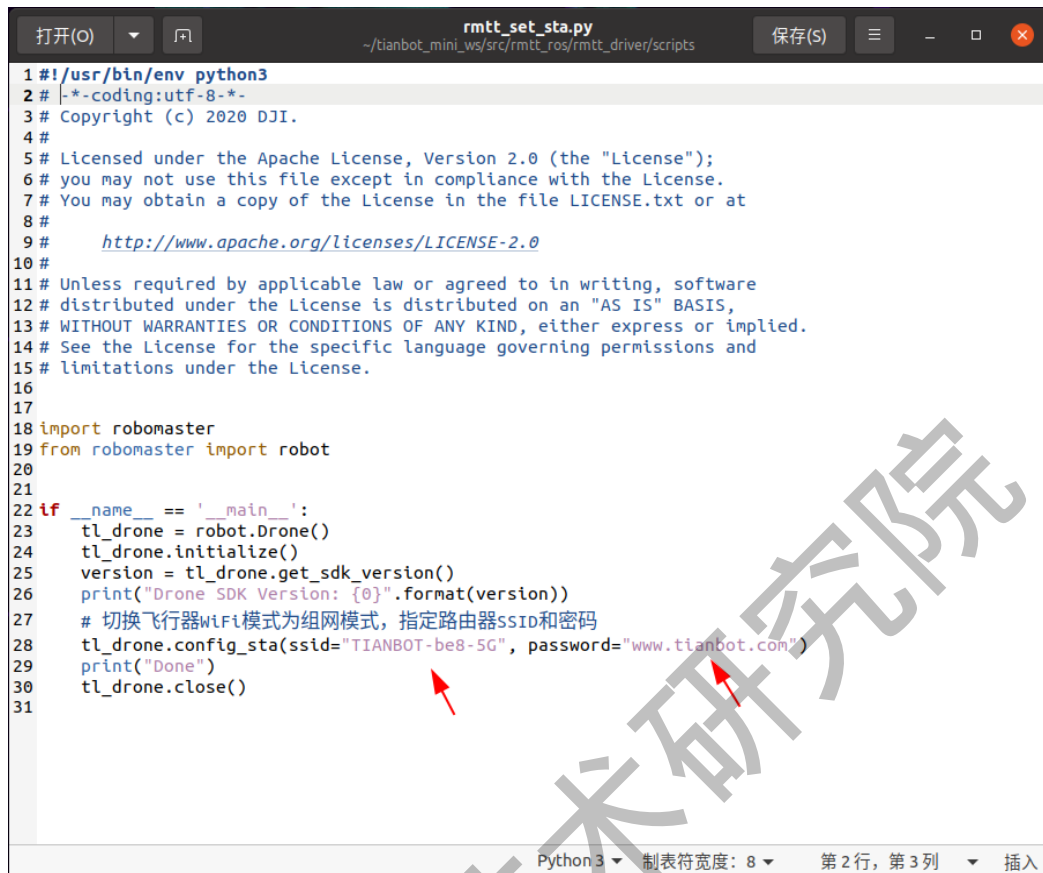
空地协同需要将无人机和无人车连接至同一电脑，在 STA 模式下进行实验，具体流程如下。

TT 进入 AP 模式

将扩展模块上的“模式开关”切换到“AP”短按电源按钮，启动 TT 无人机，稍等片刻（TT 启动后），使用 PC 连接到 RMTT-xxx 网络。

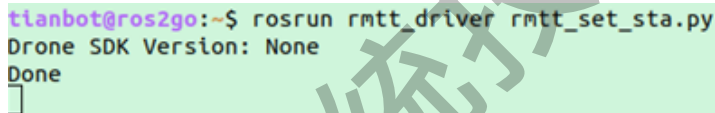


在 PC 端使用文本编辑器打开 `rmtt_ros/rmtt_driver/scripts/set_sta.py`。编辑下图所示的 SSID 与 PSK 字段为要连接的路由器的名称和密码。



```
1#!/usr/bin/env python3
2# -*-coding:utf-8-*-
3# Copyright (c) 2020 DJI.
4#
5# Licensed under the Apache License, Version 2.0 (the "License");
6# you may not use this file except in compliance with the License.
7# You may obtain a copy of the License in the file LICENSE.txt or at
8#
9#     http://www.apache.org/licenses/LICENSE-2.0
10#
11# Unless required by applicable law or agreed to in writing, software
12# distributed under the License is distributed on an "AS IS" BASIS,
13# WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
14# See the License for the specific language governing permissions and
15# limitations under the License.
16
17
18import robomaster
19from robomaster import robot
20
21
22if __name__ == '__main__':
23    tl_drone = robot.Drone()
24    tl_drone.initialize()
25    version = tl_drone.get_sdk_version()
26    print("Drone SDK Version: {}".format(version))
27    # 切换飞行器WiFi模式为组网模式, 指定路由器SSID和密码
28    tl_drone.config_sta(ssid="TIANBOT-be8-5G", password="www.tianbot.com")
29    print("Done")
30    tl_drone.close()
31
```

保存完毕后，打开新的终端，执行 `roslaunch rmtt_driver set_sta.py`



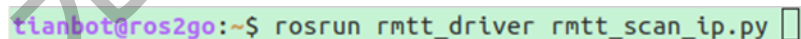
```
tianbot@ros2go:~$ roslaunch rmtt_driver rmtt_set_sta.py
Drone SDK Version: None
Done

```

看到 Done 提示后，直接拨动“模式开关”到 STA 模式，等待 TT 螺旋桨转动。TT 螺旋桨转动，表示 TT 无人机已经正确连接到路由器的网络。TT 螺旋桨如果没有转动，请重复以上步骤，配置 TT 无人机的网络连接。

获取 TT 无人机的 IP 地址

PC 连接到路由器（与 TT 无人机所连接的路由器保持一致），打开一个新的终端，运行 `rmtt_scan_ip.py` 脚本扫描 TT 无人机的 IP 地址，（注意，请确保此时只有 1 台 TT 无人机接入此路由器）。



```
tianbot@ros2go:~$ roslaunch rmtt_driver rmtt_scan_ip.py
```

扫描结束后，可以在脚本的返回结果中看到 TT 无人机的 IP 地址:192.168.8.236。

```
tianbot@ros2go:~$ rosrn rmtt_driver rmtt_scan_ip.py
2021-03-22 18:27:09,216 INFO tool.py:221 [Start_Searching]Searching for 1 available Tello...
2021-03-22 18:27:09,218 INFO tool.py:234 [Still_Searching]Trying to find Tello in subnets...
2021-03-22 18:27:10,513 INFO tool.py:234 [Still_Searching]Trying to find Tello in subnets...
2021-03-22 18:27:13,996 INFO tool.py:234 [Still_Searching]Trying to find Tello in subnets...
2021-03-22 18:27:17,439 INFO tool.py:269 FoundTello: from ip ('192.168.8.236', 8889)_receive_task
, recv msg: b'ok'
2021-03-22 18:27:17,439 INFO tool.py:272 FoundTello: Found Tello.The Tello ip is:192.168.8.236
2021-03-22 18:27:17,439 INFO tool.py:277 FoundTello: has finished, _scan_receive_task quit!
2021-03-22 18:27:17,440 INFO tool.py:301 recv thread start!
2021-03-22 18:27:17,440 INFO rmtt_scan_ip.py:41 send cmd
2021-03-22 18:27:17,552 INFO rmtt_scan_ip.py:54 get host
2021-03-22 18:27:17,578 INFO tool.py:305 recv thread quit!
scan result: host('192.168.8.236', 8889)
tianbot@ros2go:~$
```

记录我们扫描到的这个 IP 地址，这个 IP 将是后续教程中需要用到的 `drone_ip`。至此，我们的所有准备工作就已经完成，接下来我们将在 ROS 中驱动 TT 无人机，并使用它：

```
roslaunch rmtt_driver rmtt_bringup.launch drone_ip:=192.168.8.236 local_ip:=192.168.0.208
```

其中 `local_ip` 是本地 IP，可以缺省，或通过 `ifconfig` 查询。

小贴士：TT 无人机在连接后，螺旋桨会开始低速旋转的待机模式。这是为了帮助 TT 无人机散热，避免因过热保护而关机。待机模式下 TT 可以运行约 40 分钟。此时摇晃 TT 无人机可以使得螺旋桨停止转动。AP 模式下桨叶不会自行转动，无需担心。