

1. 实验名称及目的

1.1 实验名称

时间戳获取实验

1.2 实验目的

通过python接口获取时间戳数据。

注意：本实验只支持Windows下Python环境运行。不支持在WSL以及虚拟机下运行。

1.3 关键知识点

在控制任务中通过使用mav.StartTimeStmplisten接口监听飞机时间戳，该函数绑定本机的20005端口，在子线程中接收CopterSim回传的心跳和时间戳数据，确保终端和CopterSim的连接正常，并不阻塞主线程。其中cpID表示CopterSim的ID。

在数据传输任务中通过使用vis.StartTimeStmplisten接口监听飞机时间戳，该函数绑定本机的20005端口，在子线程中接收CopterSim回传的心跳和时间戳数据，确保终端和CopterSim的连接正常，并不阻塞主线程。其中cpID表示CopterSim的ID。

本实验主要是实现通过Python接口VisionCaptureApi.py（见RflySimAPIs\RflySimSDK\vision目录）获取时间戳数据。关键代码解析如下：

1) 视觉接口使用

```
1 vis = VisionCaptureApi.VisionCaptureApi() \# 创建一个视觉传感器实例
2
3 vis.jsonLoad() \# 加载Config.json中的传感器配置文件
4
5 isSuss = vis.sendReqToUE4() \# 向RflySim3D发送取图请求
6
7 vis.sendImuReqCopterSim() \# 发送请求，从目标飞机CopterSim读取IMU数据,会自动开启数据监听,已经
8
9 vis.startImgCap() \# 开启取图
```

2) 相机数量和参数配置

其中，视觉传感器的初始状态由本文件夹下的Config.json决定，主要包含以下配置项：

```
1 | "SeqID":0：使用自动更新ID的方式，创建了SeqID为0和1的两个视觉传感器
2 |
3 | "TypeID":1：传感器类型为RGB彩色图像
4 |
5 | "TargetCopter":1：相机绑定在1号飞机上
6 |
7 | "SendProtocol":[0,0,0,0,0,0,0,0]：传输模式为0共享内存机制，因此本例程只能运行在Windows环境下。
8 |
9 | "SensorPosXYZ":[ 0.3,0,0]：相机的分布。
```

3) UE控制

接口详细使用方法见：UE4CtrlAPI.py

```
1 | ue = UE4CtrlAPI.UE4CtrlAPI() \# 创建UE控制实例
2 |
3 | ue.sendUE4Cmd('r.setres 720x405w',0) \#发送指令，设置UE4窗口分辨率，注意本窗口仅限于显示，取图
4 |
5 | ue.sendUE4Cmd('t.MaxFPS 200',0) \#发送指令，设置UE4最大刷新频率200Hz，同时也是取图频率
6 |
7 | ue.sendUE4Cmd('stat fps',0) \# 在UE4窗口显示当前刷新频率
```

2.实验效果

本实验通过python接口获取时间戳数据。时间戳内包含校验位、当前飞机ID、仿真起始时间戳、当前时间戳、心跳计数等。

3.文件目录

例程目录：

[\[安装目录\]\RflySimAPIs\8.RflySimVision\0.ApiExps\1-UsageAPI\10.ReadTimeStmp](#)

文件夹/文件名称	说明
ReadTimeStmp.bat	一键仿真启动脚本
ReadTimeStmp.py	Python实验代码

文件夹/文件名称	说明
Python38Run.bat	Python环境启动脚本

4. 运行环境

4.1 软件要求

Windows 10及以上版本；RflySim工具链；VS Code。

①：若使用Pixhawk 6X飞控，平台安装时的编译命令为：px4_fmu-v6x_default，推荐PX4固件版本为：1.12.3。其他配套飞控及编译命令请见：

<https://rflysim.com/doc/zh/1/Hardware.html>

4.2 硬件要求

笔记本/台式电脑① 1台。

①：推荐配置请见：<https://rflysim.com/doc/zh/HowToInstall.pdf>

5. 实验步骤

5.1. 必做实验：Windows取图控制

Step 1: 开启仿真

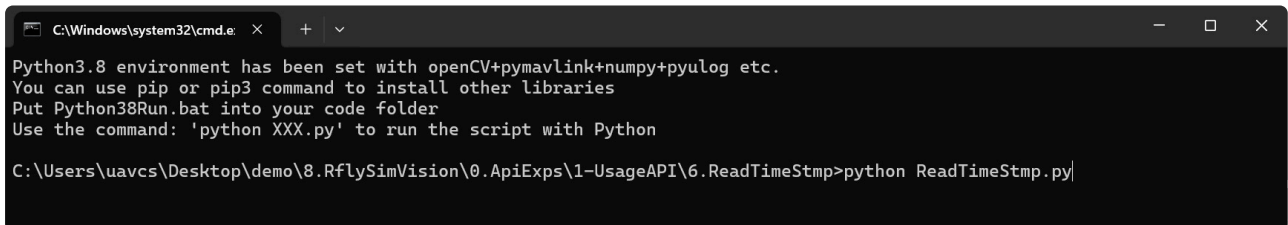
双击运行[ReadTimeStmp.bat](#)，开启一个飞机的软件在环仿真。将会启动1个QGC地面站，1个CopterSim软件且其软件下侧日志栏必须打印出GPS 3D fixed & EKF initialization finished字样代表初始化完成，并且RflySim3D软件内有1架无人机。



Step 2: 运行控制程序

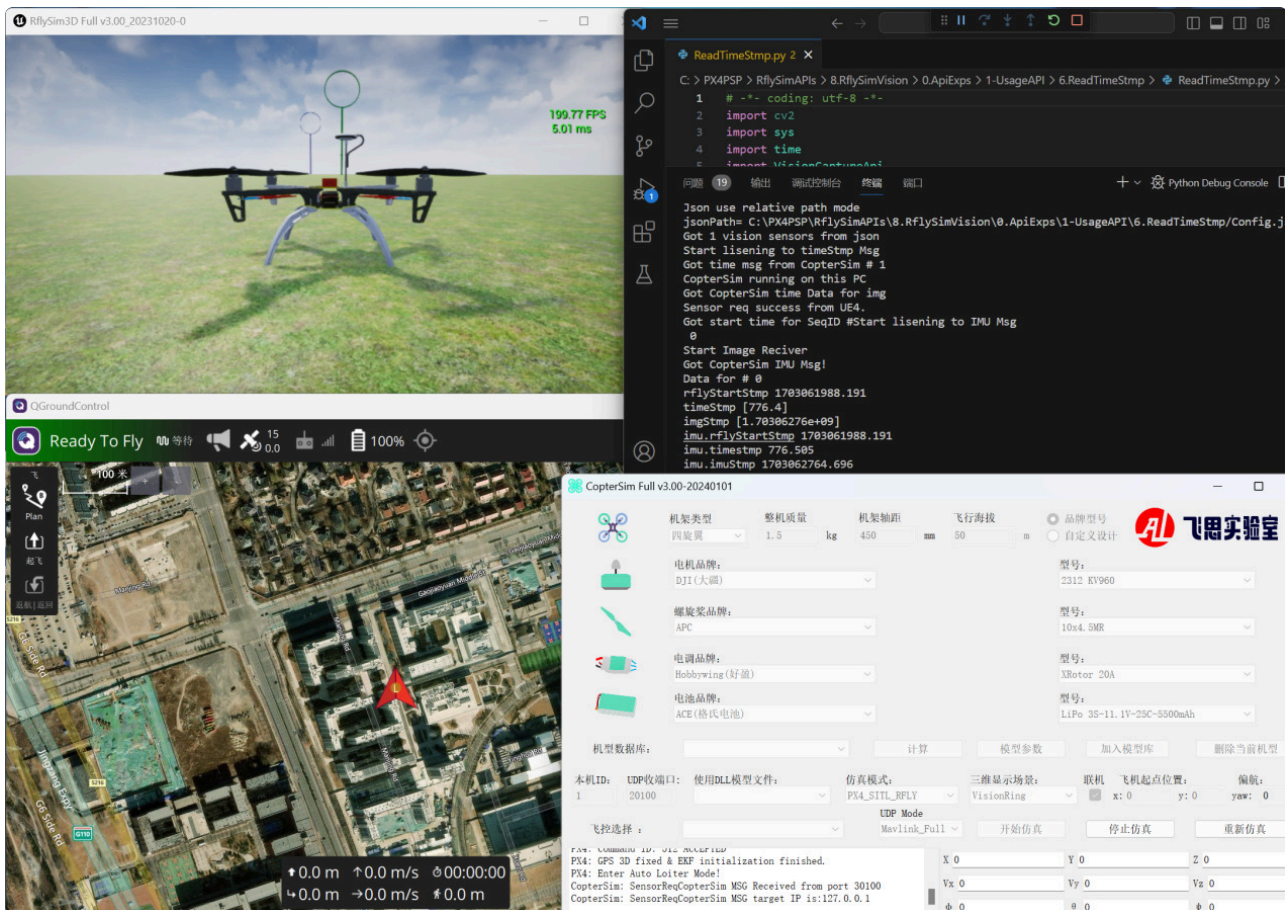
在文件夹下，双击 [Python38Run.bat](#)，打开集成好的python环境，在该环境下运行 [ReadTimeStmp.py](#) 文件，输入

```
python ReadTimeStmp.py
```



Step3: 观察结果

可以订阅得到时间戳，效果如下图所示：



Step 4: 结束仿真

在下图“[ReadTimeStmp.bat](#)”脚本开启的命令提示符CMD窗口中，按下回车键（任意键）就能快速关闭CopterSim、QGC、RflySim3D等所有程序。

5.2. 选作实验（VS Code调试运行）

准备工作

- 先确保已经按 [RflySimAPIs\1.RflySimIntro\2.AdvExps\3.PythonConfig\Readme.pdf](#) 步骤，正确配置VS Code环境。或者配置了自己的Pycharm等自定义Python环境。
- 其他步骤与上文相同，在Step2运行 [ReadTimeStmp.py](#) 时，可使用VS Code（或Pycharm等工具）来打开 [ReadTimeStmp.py](#) 文件，并阅读代码，修改代码，调试执行等。

扩展实验

- 请自行使用VS Code阅读 [ReadTimeStmp.py](#) 源码，通过程序跳转，了解每条代码的执行原理；再通过调试工具，验证每条指令的执行效果。

🔗 VisionCapAPIDemo.py ×

```
> RflySimAPIs > 8.RflySimVision > 0.ApiExps > 1-UsageAPI > 0.VisionSensorAPI > 1.CameraAPI
8   ue = UE4CtrlAPI.UE4CtrlAPI()
9
10  #Create a new MAVLink communication instance, UDP sending
11  mav = PX4MavCtrl.PX4MavCtrl(1)
12
13  # The IP should be specified by the other computer
14  vis = VisionCaptureApi.VisionCaptureApi()
15
16  # Send command to UE4 Window 1 to change resolution
17  ue.sendUE4Cmd('r.setres 1280x720w',0) # 设置UE4窗口分辨率, 同时设置UE4窗口大小
18  ue.sendUE4Cmd('t.MaxFPS 30',0) # 设置UE4最大刷新频率, 同时也设置UE4帧率
19  time.sleep(2)
20
21  # VisionCaptureApi 中的配置函数
22  vis.jsonLoad() # 加载Config.json中的传感器配置文件
--
```

- 请尝试修改代码，实现飞机位置改变、相机姿态角改变、相机参数改变等功能。

6.参考资料

无

7.常见问题

Q1: 无

A1: 无