

这次电赛结束，到今天刚好一个月，实在是太多的感触，就此分四个部分总结一下。对无人机题的认识，学习过程，个人心得，反思和总结

无人机题概况

首先是对无人机题的一个总结。第一周确定分工，我负责 `ros2` 通信和视觉，`ydc` 负责飞控，`lsh` 负责底层硬件和报告。第一周对无人机题有了一个大致的了解，无人机系统一般分为飞控和机载电脑两个部分，飞控部分为运动控制，主要是基本的定高，定点，前进后退等基本动作，以及串口首发数据。本来飞控部分最大的难点就是平稳飞行，需要结合各个传感器通过计算进行 `pid` 调解达到稳定，但是颜工提供的底层 `sdk` 已经为我们封装好了。机载电脑需要处理视觉部分，主要是用 `opencv` 库，以及使用激光雷达进行建图和定位，导航。用深度相机实现定位和追踪（本次未使用），比赛时使用了地面站，也是通过 `ros` 通信。底层硬件部分主要负责对无人机硬件架构的了解，涉及无刷电机，电调，机架，螺旋桨电池，以及各个传感器工作的原理和数据。在赛前我研究了近七年的无人机赛题，任务基本与救援，巡检相关，需要结合视觉和雷达进行定位，这也是近年各种大赛热门的主题，总得来说，无人机是一道综合题，不像其他题特征明显，实际上我们也遇到很多问题，比如电源问题，通信问题，控制问题等，需要队员具有很强的综合能力，分工相对明确并且独立，并且每一个部分都需要完成的很好，最后才能完赛。

学习过程

接下来是备赛过程中我的学习过程。第一周了解开始了解无人机系统的整体架构，机载电脑的任务，开始学习 `ros2`，能够用 `Python` 建立功能包和节点，并发布订阅话题。第二周开始了解 `ros2` 中使用相机节点并结合 `opencv` 库进行视觉处理并发布数据，使用其他节点调用并处理。同时开始了解激光雷达和深度相机使用，以及使用 `gmapping`, `cartographer` 建图和 `ACML` 定位的 `SLAM` 系统和 `navigate2` 导航。第三周开始使用激光雷达进行建图和定位，建立 `tf` 坐标系进行坐标变换，获取无人机相对坐标。同时开始部署无人机赛题所需的 `ros2` 工作空间，提前搭建好节点，与 `mcu` 进行通信。由于承重问题，我们决定放弃中科浩电的成品机，从往年的作品中收集零件制作了一台轴距 `33cm` 的中型无人机，在此过程中了解了电机型号和电调型号的区别，以及无人机的硬件组成，（果然什么东西一定要动手拆才能了解它的组成原理），最后终于在比赛前几天将无人机搭好，但是也因为换机架引发了很多新的问题，比如 `pid` 参数需要重调，光流需要重新校准等，导致比赛开始后这些东西产生了很大的误差，几乎让原来的代码不可用，不过我们应该想到的，换了一套系统原本的代码很难再复现。

个人心得

最后就是我个人的一些心得。经历过这次电赛，由于基本上是自己主导的，才知道要完成一场比赛有多难。也更能体会到当时小唐哥处境。四足狗对他来说也是一个全新的项目，但是在 `RC` 赛初，他一直一个人在帮我们把环境配好，也给我们提供测试框架，我们只需要训练模型然后调用在测试框架上完成即可，可以说他一个人完成了将近 `60%` 的任务量。我们每个人完成了自己的部分，他又进行调整，整合。真的是为比赛尽心尽力了。现在回想起来，当时我不过只是懂一点 `python`，就能够参加这么难的比赛，真的是靠小唐哥带领以及赵老师给的机会。真的非常感激。电赛之后又去回顾了四足狗的开发文档，突然就看懂原来那些用 `ros` 开发的案例，比如深度相机点云图，激光雷达，`ros` 启动文件等等。三个月前那些我一点也看不懂，但是经过这次比赛后，对 `ros2` 有了更清晰的认识。这次比赛虽然没有取得很好的成绩，但是真的收获和成长了很多。首先要感谢赵老师和各位老师还有颜工的指导，

赵老师真的给与我们很大的支持，包括无人机，场地，材料，也经常来关心我们的进度。颜工为我们提供了一套入门的 `sdk`，也亲自来了两趟为我们指导，帮助我们解决了很多实际问题。其次要感谢我们两个队友，这次电赛无憾主要是觉得我们尽力了，大家都很努力，没有人摆烂，即使搬到七楼没有人监督，也是每天 8 点到一直待到晚上 11 点才走，比赛后也是三个人每天睡实验室，晚上到 2, 3 点，早上 5 点多就起床开始调。直到第三天飞控还是不稳真的差点绷不住，直到最后一天 8 点也在努力做最后的尝试。所以我并不觉得遗憾，只是希望下次机会一定要把握住。

反思

最最后做一个反省吧。这次比赛失败有以下几个原因

1. 准备时间太少。准备时间最多 23 天，想要完成这样一个庞大的系统，本身就充满挑战性，从初识无人机，看过颜工的代码，以为都是封装好的，就没去了解更多无人机的其他选择。这也是我认为最大的失误，我觉得过于依赖别人的套件是一件非常危险的事情，这意味着做各种开发都要受困于软件层，而中科浩电提供的软件和硬件无法支持完成电赛，这就导致了飞控部分前期很放松，实际上不管是往年学长的作品还是其他学校，使用的都是集成度更高，更轻便稳定的飞控如拓空者，`pix4`（强烈推荐 `pix4`，被誉为最强大的开源飞控），很少有人电赛使用 `msp432` 做飞控，我们意识到这一点太晚，已经来不及更改了。
2. 各个部分并未深入。先拿我做的 `ros2` 通信来说，`ros2` 的通信机制有四种，但是我却只使用了最简单的一种。我也没有实现用 `ros2` 对无人机的运动进行控制，其次就是激光雷达的建图定位，最后也没有融合 `imu` 实现建图的稳定，如果能够使用稳定的坐标进行导航和路径规划，再用 `ros2` 进行运动控制和避障，或许会获得更好的成绩。飞控部分则没有深入代码的底层，只关注了暴露的函数，以至于实际情况需要光流的数据和进行串口通信花费了很长的时间。同时也没有对其他飞控进一步了解。硬件部分也没有继续关注光流和惯导模块以及飞机的参数，（或许是颜工说了一句我们不需要关注底层），我们在更换硬件的过程中做了无数次尝试，也花费了很长时间。