

嵌入式软件技术方案 STM32 部分

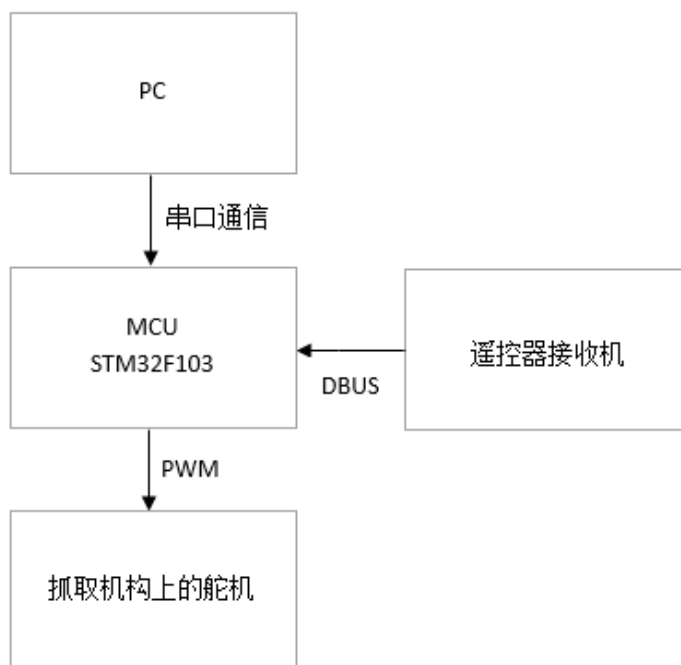
一. 综述

在夏令营项目中，STM32 层面上的嵌入式软件部分可以分为两个部分，即无人机上搭载的机械机构控制部分和地面机器人控制部分。这两个部分的控制技术方案相互独立，两者均只与所属机械结构的机载 PC 进行单对单通信，通过串口通信的方式完成指令的执行和信息的反馈。

二. 无人机机械抓取机构技术方案

无人机上搭载的 STM32 嵌入式控制系统较为简单，其目标是完成与操作者或上层 PC 的通信并执行其控制指令，控制对象是驱动机械爪进行抓取动作的两个舵机。

因此，该系统以搭载有 STM32F103 主控芯片的最小系统板作为主控电路板，外围包括舵机及其电源分配电路、串口转 USB 模块和遥控器接收器。系统控制框图及信号流向如下所示。



该系统上连接有 TD7 遥控器接收机，所以当遥控器左侧拨码开关置于下方时，遥控器可以优先控制机械机构进行抓取动作，这为机械结构的调

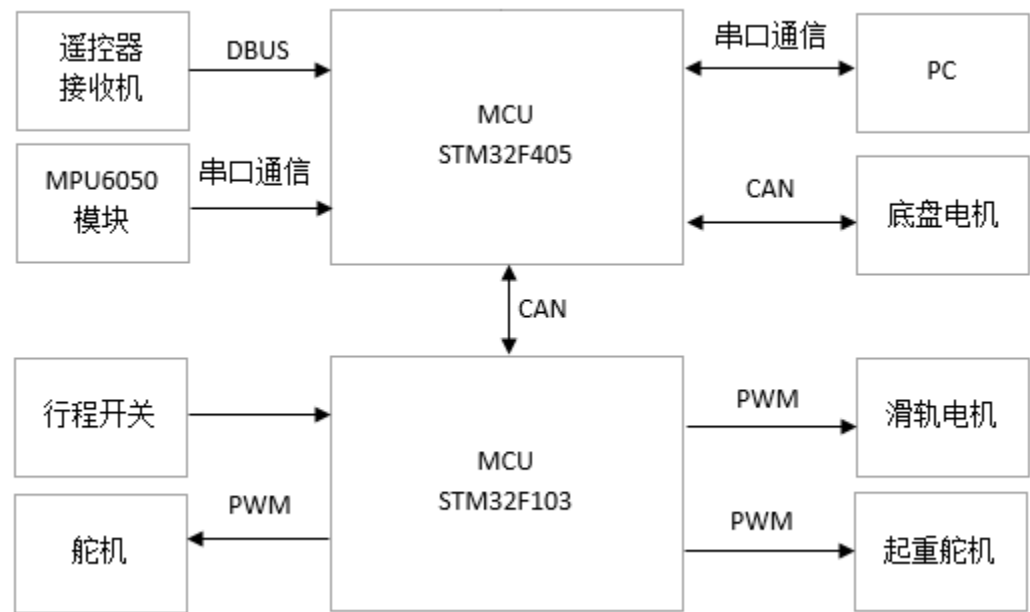
试维护和安全保障提供了便利。

当拨码开关位于其它位置时，机械结构处于受 PC 控制状态。PC 将会在合适的时机向 MCU 发送数据包，MCU 在解算出信息后会控制舵机处于不同的行程，从而驱动机械结构完成抓取不同公仔以及松开的动作。

三. 地面机器人整体控制技术方案

地面机器人上的 STM32 嵌入式控制部分则较为复杂，这一部分的嵌入式系统需要完成信息交互、传感器信息采集、整车运动控制、抓取机构控制和整体逻辑处理等任务。

因此本系统决定采用两块电路板来完成此任务。一块电路板是以 STM32F405 为主的主控板，用于和 PC 进行通信，采集遥控器接收机、MPU6050 的信息，通过 CAN 总线通信闭环驱动底盘电机。另一块电路板是以



STM32F103 为主驱动板，用于驱动直流减速电机和舵机以及采集行程开关的信息。两者之间通过另一条 CAN 总线进行通信。系统控制框图及信号流向如下所示。

该系统有许多控制动作，包括底盘电机的闭环驱动、滑轨电机的闭环驱动、整车姿态角的校准，以及对于舵机、行程开关的控制等等。

对于有反馈信息参与的动作均采用 PID 算法进行闭环控制。整车运动依

靠底盘电机的位置环与速度环的双环控制实现，反馈信息为电调板通过 CAN 总线发出的电机位置信息。而驱动滑轨的直流电机和整车姿态角的校准则只使用了位置环进行控制，前者的反馈通过正交编码器完成，后者的反馈信息包括 MPU6050 经过卡尔曼滤波得出的偏航角数据和 PC 根据摄像头偏转角度得出的偏航角数据。

整车控制指令的下达通过遥控器接收机和与 PC 的串口通信实现，遥控器的拨码开关区分了整车的手动控制模式和自动控制模式。在自动模式下，PC 通过视觉信息控制整车移动到指定位置并进行姿态校准后，整车执行一系列动作完成公仔的抓取，然后整车再由 PC 控制进行公仔的交接。PC 发向嵌入式系统的数据包括整车的速度、摄像头偏航角信息、整车校准状态、行动控制标志，嵌入式系统发向 PC 的数据包括行动完成标志和抓取成功标志。

整车由 SYSTICK 定时中断完成了信息交互和动作控制，在 main 函数 while(1) 循环中依靠状态机保证动作的依序执行和控制参数的正确赋值。同时程序中多处使用 DMA 和看门狗定时器保证了程序的健壮性。

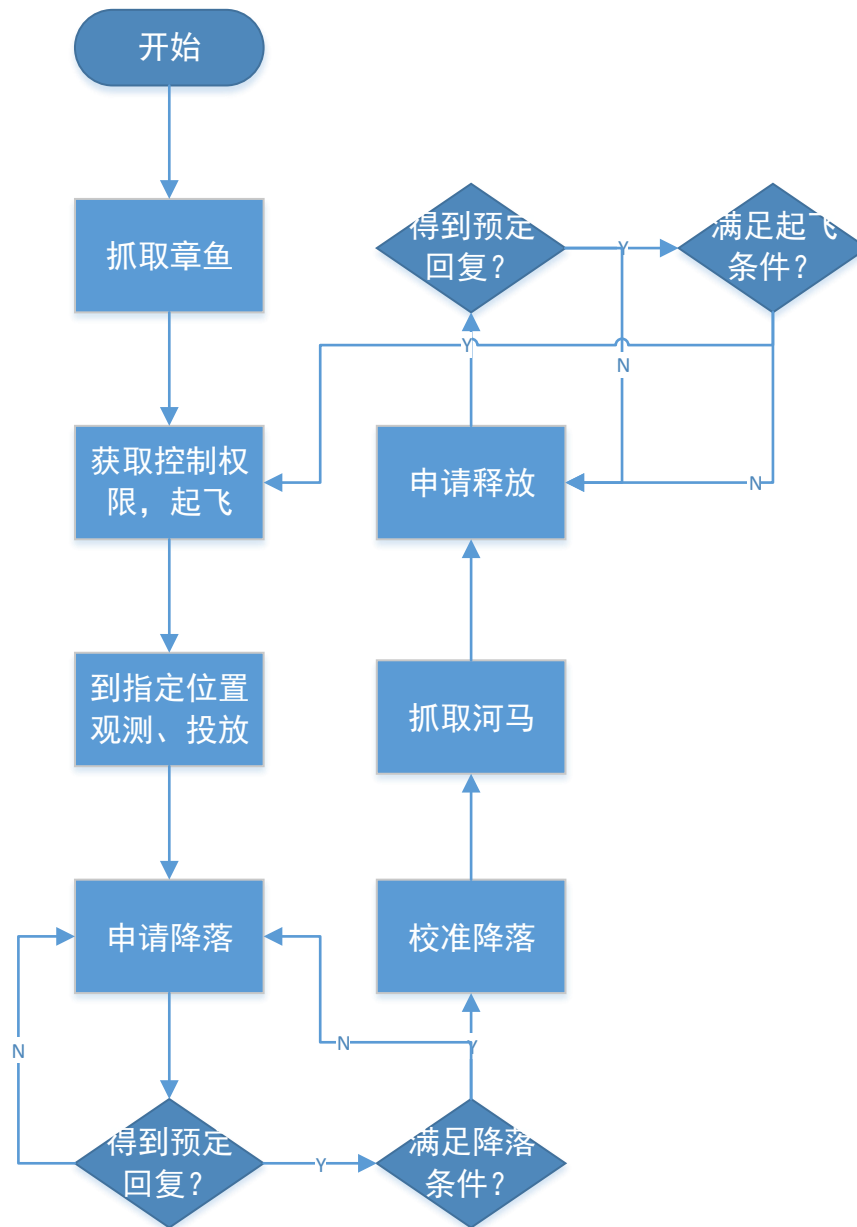
嵌入式软件技术方案 ROS 部分

一综述

在夏令营项目中，ROS 部分的嵌入式软件完成了三件任务，第一项是飞机上的主逻辑节点和控制节点；第二项是小车上的逻辑节点，具体的底层控制是由 stm32 完成的；第三项是完成了小车和无人机的通信，其中飞机是主机，小车是从机。

二无人机部分

在本次比赛中，我组无人机的流程图如下：



其中主要讲述两个功能模块：到指定位置观测、投放、校准降落。

到指定位置观测、投放：

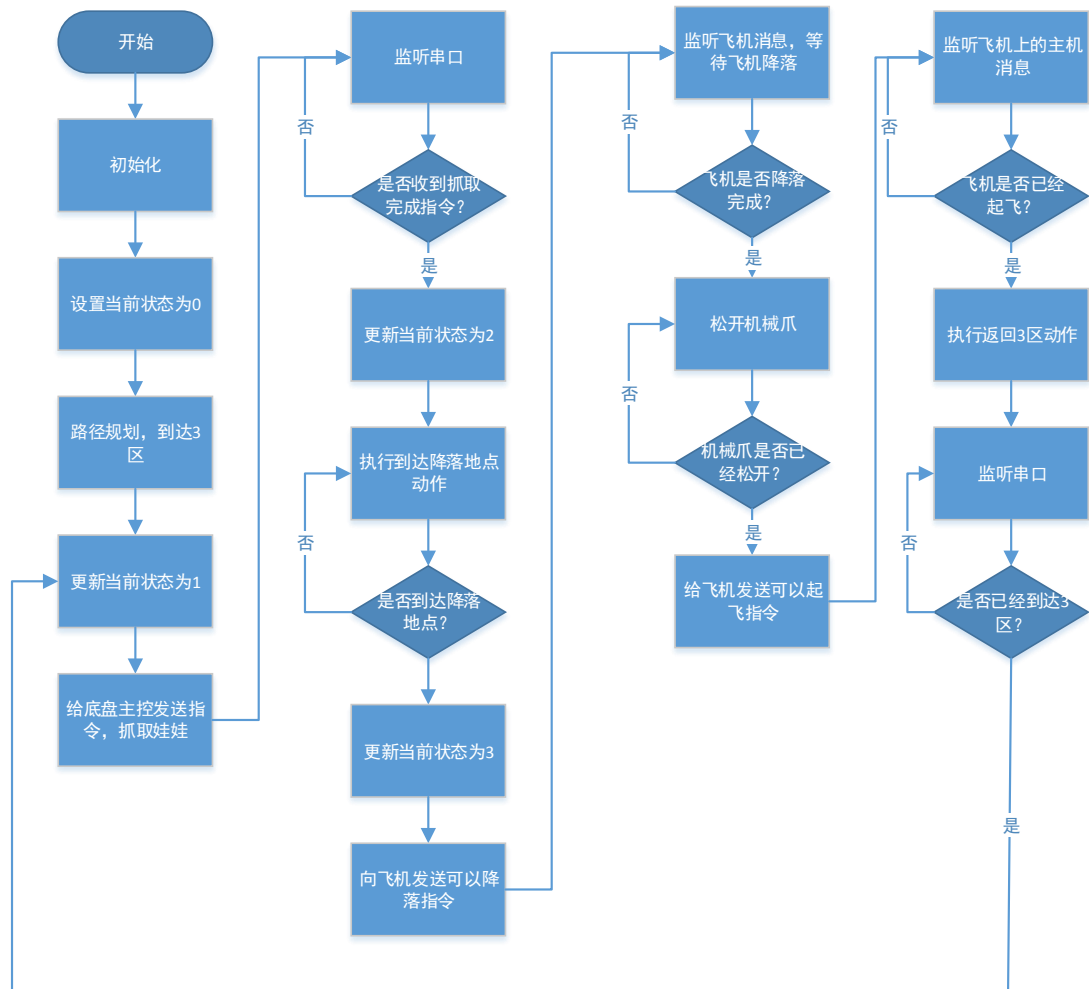
我组飞机的定位采用的是 ORB-SLAM 建图的方式，场地中所有位置都有一个特定坐标。我们事先采集观测点和投放点的坐标，然后采用 PID 控制的方式移动过去。先到观测点，若观测到对应图案，则到投放点投放。

校准降落：

我组飞机投放完一次后，会在原地向小车发送降落请求，得到允许降落的回复之后，会先飞到小车的上方识别车上贴的二维码，然后进行水平方向的校准。当水平方向的误差在 0.023m 之内，开始下降，下降过程中如果水平方向误差超过 0.035m，则停止下降重新进行水平方向的校准。当下降到离二维码只有 0.65m 时，再水平校准一次，最后降落。

三、小车部分

小车部分的流程图如下：



小车的控制分为底盘主控和上层的全场定位和路径规划控制，它们之间利用串口进行通信。首先 NUC 通过外置摄像头利用 ORB_SLAM 进行全场定位，借助 PID 控制器将速度控制量通过串口发给底盘主控，底盘主控通过速度环直接控制电机转速。通过这样的闭环控制，保证了小车的定位精度，避免了底盘主控位置环的控制误差较大且容易受轮子打滑影响的问题。

由于小车要完成到达 3 区、到达交接地点、与飞机交接娃娃等动作，状态比较多，与飞机的配合要求较高，因此我组采用有限状态机的方式严格控制小车的运行状态，通过状态的更新与跳转，保证了小车的动作与飞机的动作具有严格的时序性。

四、通信部分

通信部分，我们采用了无人机与地面机器人协同工作的模式，在两个机器人

中我们采用了 ROS 多机通信的方式，相比传统蓝牙，ZigBee 等通信方式而言，采用 ROS 的多机通讯模式本质上是 TCP 通信的封装，飞机和地面移动机器是两个子系统，两个子系统连接到一个局域网内，并将飞机设置为主机，地面机器人设置为从机。在前期通信实现中，我们采用传统的 Topic 进行节点之间的通信，虽然这样也能实现通信，但是在实验过程中发现采用 Topic 的方式发布者需要发布多次信息接收者才能收到，这样极大的降低了通信效率且会有较大的延迟性，在一个需要实时处理的系统当中是不能被接受的。这种类似于中断请求的方式正好是 ROS 中的 Service，Service 的工作原理与 topic 有些许不同，同一个 service 能被多个节点注册，但是只有最后一个能被其他节点接受。一个节点调用 service 时，通过 `lookupService()` 远程调用在 master 处查找相应 service 的 URI。然后它将通过一个 request 消息调用 service 的提供者，如果成功了，service 提供者将返回一个相应的 response 消息，失败了返回相应错误消息，（所有的消息传输默认都是通过 TCPROS 协议）。采用 ROS service 保证了通信的成功率。