

# 醒狮机器人实验室电控学习指南

首先，再次简单介绍一遍**醒狮战队**，醒狮战队是一支专注于机器人研发的队伍，其成立的目的是参加一年一度的全国大学生机器人大赛**RoboMaster**超级对抗赛，通过不断的研发迭代自己的机器人来实现技术的突破和进步。

其中**电控软件组**的主要任务是通过**控制理论**方面的知识来编写控制代码实现机器人的高性能和高可控性，同时学习不同的控制算法和开发工具，锻炼排查问题和解决问题的能力，而硬件组则是根据比赛规则和机器人需求进行研发，研制出满足队伍需要的硬件模块，同时协助软件组解决问题。队员在队内进行的各种学习和研究都是围绕机器人进行的。

由于比赛对队伍要求的能力越来越强，因而梯队的学习任务也在不断的加重，这样才能适应比赛的节奏，同时涉及的很多东西都是需要实践的，学习时间成本比较高，半途而废非常可惜，因而得再次确认，

**你真的做好心理准备了吗？**

## I、前言

以往的电控组新人培养基本采用统一分配任务的模式，但由于不同批次的新人进梯队的时间不同，学习进度也不一致，导致一旦错过统一培养就需要再由组长逐个分配任务，加重了组长任务的同时还容易有疏漏的地方，组长也无法清楚的知道每个人学到了什么程度，同时由于梯队学习的基础知识跟实际进队后需要掌握的东西相差较大，因而进队后队员仍需要有较长的一段过渡期，无法快速的上手任务，并且由于各兵种的实际情况不同，导致不同兵种成员的储备知识参差，故撰写这份文档，重新规范化整个培养过程，让新人明确整个学习过程，对自己的学习情况能够有清楚的把握。

**电控组分为两个组别，软件组和硬件组**，但从往届的经验来看，硬件组需要的储备知识比软件组的要多得多，在队伍中也充当着比较关键的角色，不仅需要懂硬件，也需要懂软件，帮助各兵种找出问题并解决，同时，软件组的队员也需要懂一部分硬件知识，才能快速排查问题，因而从24赛季开始，前期的培养将不再区分方向，按统一的学习路线学习，软件和硬件的基础知识也将一起传授，等到基础全部学完再区分方向。

培养计划目前分为三年进行：

第一年 (实践)	学习基础的软硬件知识和开发工具的使用，跟随老队员熟悉机器人，维护机器人，学会快速定位机器人出现的问题和解决，思考机器人的不足之处。
第二年 (开发)	通过参考其他学校的开源文档和交流，针对机器人存在的不足之处进行研究和测试，补全缺失的电控知识并加以应用。
第三年 (研发)	根据自身未来发展选择不同的学习方向，学习更加前沿的知识理论，熟悉整个开发流程，并以机器人为平台进行自主研发，包括但不限于ros，视觉。同时撰写相应的文档申请专利。

## II、基础知识与开发软件

---

### 前置知识

- 计算机速成课：

[Crash Course Computer Science](#)

- 从零到一打造一台计算机：

① [【编程前你最好了解的基本硬件和计算机基础知识 - \(模拟电路\)】](#)

② [【编程前你最好了解的基本硬件和计算机基础知识 - \(数字电路\)】](#)

③ [【从0到1设计一台计算机】](#)

务必学完以上课程再开始本教程的学习。

### C语言

- 目标：掌握C语言的基础概念与语法，如 **变量、数据类型、逻辑语句、结构体、指针** 等

#### 科普：什么是 IDE？

集成开发环境（IDE）是一种帮助程序员高效开发软件代码的软件应用程序。它通过将软件编辑、构建、测试和打包等功能结合到一个易于使用的应用程序中，提高了开发人员的工作效率。就像作家使用文本编辑器，会计师使用电子表格一样，软件开发人员使用 IDE 让他们的工作变得更轻松。如下列提到的

[Visual Studio 2022](#) [Keil](#)...均为 IDE

#### 推荐开发软件:

- [Visual Studio 2022](#)

[【vs2022\(Visual Studio 2022\)权威指南&&C语言&&软件工程开发的方向&&技巧要领】](#)

[【VS2022安装教程Visual Studio 2022下载详细安装使用调试教程VS2022使用教程C语言编译器，VS2022下载C语言软件安装C语言软件安装】](#)

- [Visual Studio Code](#)

[【vscode 关于C/C++的环境配置全站最简单易懂！！大学生及初学初学C/C++讲！！！！】](#)

[【用VSCode搭建C/C++环境，下载+安装+配置+使用以及汉化的详细教程，小白也能轻松学会！】](#)

[【Visual Studio和Visual Studio Code区别,vs2022和vscode的区别】](#)

#### 推荐学习视频:

[【浙江大学】C语言入门与进阶 翁恺（全129讲）](#)

# STM32开发初探

## 前期准备

- 烧录器 + stm32开发板

(推荐 j-link烧录器 + STM32F103C6T6/C8T6)

(不建议使用自带的蓝色USB数据线，经常断连，容易搞得自己红温，推荐去合宙买一元一根的，便宜好用)



兼容J-link OB ARM仿真下载

一套 ▾

¥19.5 优惠价¥16.50

×1



32F103C8T6/C6T6开发板

STM32F103C6T6 最小系 ▾

¥8.8 优惠价¥7.80

×1

STM32F103C6T6 最小系统板

- 串口转USB设备 - CH340



天猫 usb转ttl usb转串口下载线  
ch340g模块rs232刷机线板下...

¥4.80 200+人累计付款 广东

注：1、实验室使用的单片机是STM32F4，代码则是STM32标准库配合keil开发。2、Keil版本请先使用5.15 或 5.18 等较低版本的，较高版本调试时会报错（在市的Jlink调试器一般都为非SEGGER公司出售的，故有时会触发非正版BUG，正版的调试器需要前往官网购买，但很贵很贵!!!）3、下载F1和F4系列的芯片包

#### 基础补充：STM32基础理论

推荐上b站看野火教程，其中入门篇要全部会，然后中级篇高级篇中的RCC时钟树、EXIT中断部分、GPIO、I2C、SPI、TIM定时器、ADC、DMA、CAN、USART串口、USB-CDC（选看）

时钟树是一切的基础，也是很多玄学bug的根源。

GPIO常用于机器人按键或限位开关的状态的捕获。

EXTI中断的存在则是为了提高单片机的处理效率，避免轮循模式一直占用CPU资源。

DMA则是更优于中断的处理方式，完美地解决了CPU处理快速和IO操作慢速的冲突，让我们的程序能更少占用CPU资源，多用于加速遥控器串口，裁判系统串口和小电脑串口这有大量数据读写的情况。

I2C和SPI则多见于开发板上传感器的通信，如陀螺仪。

TIM用于计时及产生PWM波控制舵机的运动。

ADC数模转换多用于将超声波，电压等以模拟信号输入的传感器信号转换为单片机可处理的数字信号。

I2C和SPI则多见于开发板上传感器的通信，如陀螺仪。

CAN通信则用于控制使用CAN通信的电机或是两块开发板间的通信。

时钟树是一切的基础，也是很多玄学bug的根源。

USB-CDC主要用于开发板与miniPC的通信。USB-CDC具有优秀的丢包率和通信速率。更重要的是它带来的便捷稳定的硬件连接方式减少了串口转ttl再转ttl转usb的不稳定性。

#### 推荐开发软件:

- [Keil5](#)

① [【避坑】Keil 5下载安装激活到2032年 \(含MDK、C51、STM32单片机\)](#)

② [【Keil+vscode合作，摆脱keil不友好的界面】](#)

③ [【VSCode+Keil5+STM32CubeMX开发环境搭建，一步不跳，一刀不剪，奶奶都能学会版。主要是配置插件Keil assistant，介绍相关使用方法。】](#)

- [J-Link 驱动](#)

① [【JLink】【JFlash】全网最详细JLink烧写流程！含驱动安装流程及烧写步骤](#)

- [J-Scope](#)

① [嵌入式调试神器-虚拟示波器之JScope](#)

#### 推荐学习视频:

① [【单片机】野火STM32F103教学视频\(配套霸道/指南者/MINI\)【全】\(刘火良老师出品\)](#)

② [【正点原子】手把手教你学STM32入门教学视频单片机 嵌入式 之 F103-基于新战舰V3/精英/MINI板](#)

③ [【STM32入门教程-2023版 细致讲解 中文字幕】](#)

④ [【野火 FreeRTOS视频教学 配套书籍《FreeRTOS内核实现与应用开发实战指南》配套例程源码 基于STM32开发板硬件教学 操作系统教学视频】](#)

#### 科普:

开发流: (标准库: Keil5 -> VScode -> jscope 或 HAL库: CubeMX -> VScode -> Ozone)

开发软件可以只使用 keil5 配合 vscode, 或者使用 CubeMX vscode Ozone, 如果学标准库的话建议使用前者, 只需在vscode简单安装一个插件就可以使用vscode阅读和编译代码, 但是调试还是要到keil5;

如果学习HAL库的话, 当然也可以只使用前者的方法(标准库方法+CubeMX), 后者开发方式更高阶而已

jscope和Ozone都为示波器软件, 能将数据图形化, 简单而言就是将变量值在图上打点连线, 最后呈现为图形

硬件的配置是最基础的, 也是我们电控算法实现的基石。ST标准库虽然精简高效, 但是文档式配置法极其耗费精力。

在掌握了必需的标准库配置技能后, 我们可以学习使用已经日渐成熟稳定的CubeMX, 用HAL库/LL库完成底层硬件配置, 在硬件配置层稳定的前提下把更多的时间集中在抽象层的控制算法的优化上。目前较为现代化的开发方式是cubemx+vscode编辑+armgcc编译+ozone调试的方法, 虽然说没有keil的一体集成化ide这么方便地完成开发环境的配置, 但是这一套工具链带来的开发便捷性是质的飞跃。然而, 我们要使用这套方法需要跳出舒适区, 得额外学习cmake, 编写makefile等更深入理解项目编译流程的一系列

列知识。带来的好处是自动高亮变量，自动化格式化代码和丰富的vscode插件的良好生态，如配合git插件带来的优秀的git管理体验。但是也有传闻说23年底Keil将会更新，整体风格更新为vscode现代化集成IDE的样子。

- **C++:**

- ① [【黑马程序员匠心之作】C++教程从0到1入门编程,学习编程不再难](#)
- ② [The Cherno C++](#)
- ③ [【快速学习C和C++，基础语法和优化策略，学了不再怕指针\(南科大计算机系原版\)】](#)

- **Makefile:**

- ① [【Makefile 20分钟入门，简简单单，展示如何使用Makefile管理和编译C++代码】](#)
- ② [【从零开始学Makefile】](#)
- ③ [【GNU Makefile编译C/C++教程 \(Linux系统、VSCODE\)】](#)

- **CMake:**

- ① [【从零开始详细介绍CMake】](#)
- ② [【新坑预警】相信我，我真的可以把CMake讲清楚】](#)
- ③ [【CMake 保姆级教程【C/C++】](#)

- **Ozone:**

- ① [CubeMX+VSCode+Ozone的STM32开发工作流（一）背景知识介绍 - NeoAndrew的文章](#)
- ② [CubeMX+VSCode+Ozone的STM32开发工作流（二）VSCode环境配置 - NeoAndrew的文章](#)
- ③ [CubeMX+VSCode+Ozone的STM32开发工作流（三）利用Ozone进行可视化调试和代码分析 - NeoAndrew的文章](#)

## PCB

无论是硬件方向还是软件方向，都需要有读懂简单电路原理图和设计简单电路的能力，要求使用EDA（电子设计自动化）工具绘制原理图和PCB，并打样焊接。

## 前置知识

- ① [【电控硬件通识part.1】](#)
- ② [【电控硬件通识part.2】](#)
- ③ [【电控硬件通识part.3】](#)
- ④ [【电烙铁焊接教程（包含热风枪）】](#)

**推荐开发软件与学习视频:**

- Altium Designer

- ① [【Altium Designer 23|AD22|AD23新手入门必备课56讲|凡亿教育】\\*](#)
- ② [【Altium Designer 1小时 \(貌似不够\) 速成 \(可能不止一小时~\\* 但我觉得仨小时肯定够了--来自up猪的自信!!\)】\\*](#)
- ③ [\\*【Altium Designer】原理图库和封装库的创建 \(接上一系列的视频\)\\*](#)
- ④ [【Altium Designer Rules】规则逐条详解 \(自封全B站最详细, 画PCB必看\)](#)
- ⑤ [【AD库资源分享】带你认识Altium Designer的原理图库、PCB封装库和集成库](#)

#### ◦ 立创EDA

- ① [【嘉立创EDA-PCB设计零基础入门课程 \(持续更新中.....\)】](#)
- ② [【【教程】零基础入门PCB设计-国一学长带你学立创EDA专业版 全程保姆级教学 中文字幕 \(持续更新中\)】\\*](#)
- ③ [【小白入门-如何使用立创EDA设计一个简单的PCB】\\*](#)

补充: [【分电板AD入门教程】https://www.bilibili.com/video/BV1wu4y137UX/?share\\_source=copy\\_web&vd\\_source=49d05f69dfc6663616a6cfbe19a35edb\)](https://www.bilibili.com/video/BV1wu4y137UX/?share_source=copy_web&vd_source=49d05f69dfc6663616a6cfbe19a35edb)

#### 考核:

【腾讯文档】醒狮电控PCB考核  
<https://docs.qq.com/doc/DU0FxZWl1ZkZid2Jh>

## 集成电路仿真

- Multisim

## 控制仿真

- [MATLAB](#):  
 MATLAB基本语法  
 数据拟合: 学会使用Curving toolbox  
 系统辨识: 黑箱辨识和灰箱辨识  
 频谱分析: 使用FFT (快速傅里叶变换) 将时域数据转换成频域数据来分析云台噪音情况  
 滤波器设计: FDA Tool设计滤波器根据频谱分析发现的高频噪音或低频噪音过滤  
 MATLAB Simulink工具箱: 云台控制框图仿真, 验证电控控制算法  
 MATLAB simscape工具箱: 物理引擎仿真,验证电控控制算法
- [Webots](#): 主要用于电控控制算法仿真验证
- [Gazebo](#): 哨兵路径规划仿真
- [Unity](#): 主要用于视觉仿真验证



## 控制理论

- MATLAB的PID原理讲解:

[https://www.bilibili.com/video/BV1bJ41117kn/?spm\\_id\\_from=333.337.top\\_right\\_bar\\_window\\_history\\_content.click](https://www.bilibili.com/video/BV1bJ41117kn/?spm_id_from=333.337.top_right_bar_window_history_content.click)

- 华南虎战队的PID原理教程:

[https://www.bilibili.com/video/BV1B54y1V7hp/?spm\\_id\\_from=333.999.0.0](https://www.bilibili.com/video/BV1B54y1V7hp/?spm_id_from=333.999.0.0)

- 更好的理解PID【通过推小车的过程】

[https://www.bilibili.com/video/BV1By411h7HL/?share\\_source=copy\\_web&vd\\_source=49d05f69dfc6663616a6cfbe19a35edb](https://www.bilibili.com/video/BV1By411h7HL/?share_source=copy_web&vd_source=49d05f69dfc6663616a6cfbe19a35edb)

### 推荐学习视频:

#### DR.CAN

[【中英字幕】关于控制理论你需要知道的一切 | Everything You Need to Know About Control Theory】](#)

[【Brian Douglas】理解PID控制 | Understanding PID Control】](#)

[【中英字幕】Brian Douglas Control Theory | 自动控制原理】](#)

[【Brian Douglas】现代控制理论 | State Space Control】](#)

[【Brian Douglas】What Is System Identification?】](#)

[【Brian Douglas】线性系统辨识】](#)

[【中英字幕】什么是模糊逻辑 模糊逻辑第1部分 | What Is Fuzzy Logic Fuzzy Logic Part 1.】](#)

[【中英字幕】模糊推理系统 模糊逻辑第2部分 | Fuzzy Inference System Walkthrough Fuzzy Logic Part 2.】](#)

[【中英字幕】模糊逻辑倒立摆示例 模糊逻辑第3部分 | Fuzzy Logic Examples Fuzzy Logic Part 3.】](#)

[【中英字幕】调整模糊逻辑控制器 模糊逻辑第4部分 | Fuzzy Logic Controller Tuning Fuzzy Logic Part 4.】](#)

## Git 代码管理

在往届的实践中，多人代码管理一直是一件非常让人头疼的事情，以各兵种为例，如果一台机器人只由一个人负责代码的编写，那代码管理起来会相对轻松，自己也对自己的代码清楚和了解，往届采用的办法就是直接以压缩包的形式在微信传输，但一旦一台机器人有两个以上的人负责起来的话，那问题就多了，首先要考虑的就是代码同步问题，怎样才能确保这份代码是最新的，怎么确保你修改的代码是最新的，怎么知道别人改了什麼代码，笔者由于这个问题就曾与队员产生过矛盾，也做过这方面的探索，曾经试图采用NAS和同步空间的方式来解决这个问题，让每个人的电脑的那份代码永远是最新的，但使用过程依旧发现了很多的问题，例如无法多人同时编写，需要每个人都配合，无法知晓代码做了什麼改动，不确定因素太多，因而这个方案最终还是放弃了。两个人参与代码的编写就已经那么不容易了，那五个人，十个人如果还是按照那种代码管理方式，那么整个团队的工作将会混乱不堪，在与其他团队的交流沟通中，新赛季决定使用更加科学的代码管理方式git。



### 传统的压缩包管理的缺陷：

- 1、只有熟悉项目代码的人或者通过修改说明文档才能定位修改内容。
- 2、不利于版本控制，会出现队员间的代码不统一或是出现使用远古版本的代码用于调车的情况。
- 3、不利于进度管理和质量控制。

### 使用Git管理的原因：

- 1、便于队伍传承。新人可通过git的历史记录，快速了解规范项目的修改流程是怎样的，了解前辈们的算法改进思路和debug历程。
- 2、便于版本控制。版本回退便捷（当在做了过多修改后出现bug，一键回退到指定历史版本）。
- 3、提升开发效率。git提供了精确的修改记录管理，便于查看修改更新内容（无需浪费口舌说明代码修改位置），方便溯源定位bug。
- 4、便于项目管理。管理者可随时在任意支持访问git远程仓库平台的设备对任意兵种进行代码审查和进度查看，便于队伍项目管理，通过查看更新日志可对队员实现更为精确量化的考勤。

- 【腾讯文档】Git本地配置及GitLab拉取代码教程  
<https://docs.qq.com/doc/DRmhBc2p2Y2R3SkVQ>
- 【腾讯文档】Git标准工作流程  
<https://docs.qq.com/doc/DRlJFSKjNtnRhRlpj>

### 推荐学习视频：

- ① [【尚硅谷Git入门到精通全套教程\(涵盖GitHub\Gitee码云\GitLab\)】](#)
- ② [【【中科大RM电控合集】Git基础连招教学】](#)
- ③ [【嵌入式必备工具Git】](#)

---

## III、RM

看到这里，你们已经基本上掌握了所需的基础的电控知识，虽然这些还只是皮毛，但作为接手机器人的项目的敲门砖已经足够啦，接下来你们面对的将会是大量的实践和调试，这一阶段你们会开始进入到实验室去了解和接触我们的机器人，坚持走下去吧！

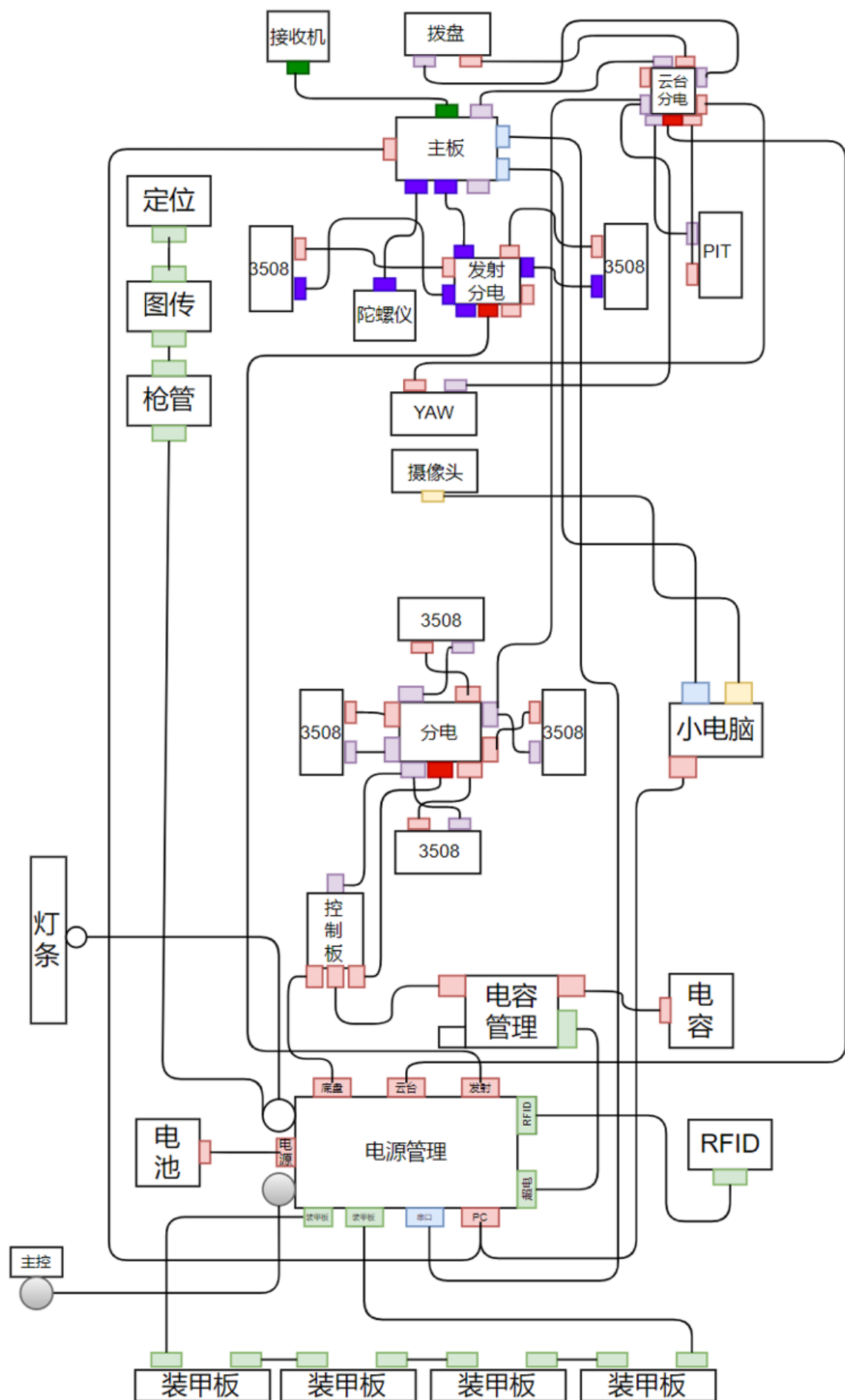
由于步兵机器人的文档资料相较其他兵种更为完善，因而步兵作为电控梯队培养计划的最后一个大型实践项目，同时也是你们作为梯队到正式队员过渡的一个阶段。

这一阶段的学习重心放在《2022第四代步兵入门手册》

- 【腾讯文档】2022第四代步兵入门手册  
<https://docs.qq.com/doc/DU1ZCYmROa1lMS0F1>

### 布线与调试

如何对机器人出现的错误进行快速的定位是每个电控人的基本功，比赛中场只有三分种的维修时间，如果做不到这个程度，那便无法在最短的时间内解决问题，也就没有上场的资格，因而我们需要对机器人的整个框架了如指掌，以下这张图是步兵机器人的整个框架（其他兵种也大差不差），需要做到能够默写的程度。（由于目前机器人在不断的改进，因而会与实际的机器人的走线有所差异，但只要明白了原理就一通百通了）



- 空白框架图训练

<https://www.aliyundrive.com/s/LhLRVw3ZWm3>

提取码: a2j5

• 考核:

这部分的考核的重点是框架图的实践以及调试，理论永远是理论，只有实操过才能印象深刻。

- 【腾讯文档】醒狮电控调试教程文档  
<https://docs.qq.com/doc/DU0NiWHVLa0dPelBX>
- 【腾讯文档】调自瞄的准备工作以及jscope使用教程  
<https://docs.qq.com/doc/DU3h0cFBPbUdxZFJ5>

注：接下来这部分则是考查大家对代码的熟练程度，设置了不同的实践小任务，帮助大家去理解和熟悉代码，以及如何修改代码。

- 【腾讯文档】醒狮电控代码学习情况考核  
<https://docs.qq.com/sheet/DU1pQINjblpVQXJP>

## 服务器与客户端

- 服务器和客户端从官网下载  
[RoboMaster 超级对抗赛](#)
- 【腾讯文档】RM服务器客户端搭建教程  
<https://docs.qq.com/doc/DU3ZWa3IHRFFVckht>
- 【Robomaster裁判系统代码讲解】  
[https://www.bilibili.com/video/BV1V84y1h7iT/?share\\_source=copy\\_web&vd\\_source=49d05f69dfc6663616a6cfbe19a35edb](https://www.bilibili.com/video/BV1V84y1h7iT/?share_source=copy_web&vd_source=49d05f69dfc6663616a6cfbe19a35edb)

• 考核:

单独搭建一次服务器和客户端（两台电脑或者一台电脑都行），用机器人正常开一局比赛。

要求:

- ①会使用路由器后台排查问题
- ②图传能正常显示画面

（若图传一直没画面，且其他都没问题（客户端ID和机器人ID记得选对！），则拔插图传电源（是电源！！不是数据线），拔插几次可能就有画面了，记得，前提是你其他都没问题）

- ③服务器和客户端都正常

## 兵种与组别

各兵种所负责的东西不一样，但内核都是差不多的，本质上就是在规则的约束下进行优化研发。由于各组别的人数要均衡，因而需要和组长商量，由组长根据实际情况分配组别。

硬件组	在目前的基础上继续深入研究硬件方面的知识，例如超级电容的研制等；
步兵组	这三个兵种所运用的知识基本差不多，但由于比赛规则，因而会有略微的差别，步兵组与英雄组主要重心是在云台算法、功率算法；哨兵组需要在步兵英雄重心的基础上加上自主导航算法与自瞄自主决策；
英雄组	---
哨兵组	---
工程组	取矿机构的运动学解算，自定义控制器设计
飞镖组	弹道仿真、自适应调整飞镖架、视觉/惯性制导镖体等
无人机组	N3飞控的使用；云台控制算法的研究，基本上就是把步兵的云台反过来安装在无人机上，但是机架不可避免的高频振荡使无人机云台工况更为复杂。

最后，恭喜你！你已经初步具备了进入赛场的能力，同时你也已经具备了能够加入我们醒狮战队的资格！欢迎你的加入！

## 进阶

- 【腾讯文档】醒狮电控组知识体系  
<https://docs.qq.com/mind/DY0JOY1VITGZ6d2xU>
- 【腾讯文档】RoboMaster 培训体系（220330）  
<https://docs.qq.com/sheet/DUFlaU0FHZk1QS0l1>