

电控组学习路线图

从零基础到机器人工程师 | 24 周培养方案

让代码驱动真实世界，让理论在赛场闭环。

路线定位 面向零基础新成员，将 C 语言、数据结构、计算机原理、模电数电、STM32、HAL、线性代数、自动控制、PID 与 FreeRTOS 串成一条项目主线。新增基础课不独立堆叠，而是在最需要它们的阶段同步学习、立即应用。

适用对象：电控组新成员 / 嵌入式与机器人方向初学者

建议强度：每周 8-12 小时 | 学习周期：24 周 | 培养方式：课程 + 实验 + 评审

版本：2026.08

01 培养目标与使用方式

这不是一份“看完视频就结束”的课程清单，而是一条以可演示成果为节点的工程训练路线。每阶段都遵循：先建立最小知识闭环，再完成一个能测量、能复现、能评审的项目。

完成 24 周后，你应具备

- 编程能力：能用 C 编写模块化、可调试的嵌入式程序，理解指针、内存、数据结构与状态机。
- 硬件能力：能读懂基础原理图，正确处理供电、电平、滤波、采样、驱动与保护问题。
- 单片机能力：能独立完成 STM32 外设配置、驱动编写、通信联调与故障定位。
- 控制能力：能从对象与指标出发设计闭环，完成 PID 的实现、整定、记录和稳定性判断。
- 系统能力：能用 FreeRTOS 组织多任务系统，正确使用通知、队列、互斥与中断接口。
- 工程能力：能用 Git、调试器、示波器/逻辑分析仪和实验记录支撑团队协作与赛场可靠性。

推荐学习闭环

步骤	要做什么	留下什么证据
1. 学	课程只学当前项目需要的章节	知识卡片 / 关键概念清单
2. 做	搭建最小实验，先跑通再扩展	可编译代码 / 接线图 / 参数表
3. 测	用串口、示波器或日志测量结果	波形 / 日志 / 数据曲线
4. 讲	向同伴解释设计取舍与失败原因	5 分钟演示 / 评审记录
5. 改	根据评审修复问题并沉淀模板	版本记录 / README / 复盘

阶段通过原则 能跑不等于掌握。只有当你能解释、能测量、能复现、能交接时，阶段任务才算完成。

02 总路线：六阶段、24 周

阶段	时间	知识交叉点	阶段成果
1 C 语言与程序思维	第 1–4 周	C 基础 + 数据结构 + 计算机原理	命令解析器 + 环形缓冲区
2 电路基础与 STM32	第 5–8 周	模电数电 + CPU/总线 + 裸机外设	传感采集与 PWM 控制台
3 HAL 工程化开发	第 9–11 周	CubeMX/HAL + 分层 + 调试 + Git	可复用驱动小仓库
4 控制理论与 PID	第 12–16 周	线性代数 + 自动控制 + 电机模型	单轴速度/位置闭环
5 FreeRTOS 实时系统	第 17–19 周	调度 + 通信同步 + 实时性	多任务控制系统
6 综合项目与战队接轨	第 20–24 周	需求 + 联调 + 故障处理 + 评审	可上场的最小机器人子系统

依赖关系：C 与内存模型是底座；电路知识决定接口是否可靠；数学与控制原理解释“为什么这样调”；

FreeRTOS 最后引入，用于组织已经跑通的功能，而不是掩盖未解决的问题。

03 第一阶段：C 语言与程序思维

阶段目标 | 第 1-4 周 建立能服务 STM32 开发的 C 语言基础。重点不是语法背诵，而是数据如何表示、存放、传递与被硬件使用。

C 语言基础：一切的起点

- 基本语法：变量、运算符、分支、循环、函数、作用域与头文件。
- 数据类型：整型宽度、有符号/无符号、浮点数、枚举、结构体、联合体、typedef。
- 数组与字符串：边界、长度、缓冲区、格式化输入输出及常见越界风险。
- 指针与内存：地址、指针运算、数组退化、函数指针、const、volatile、static、extern。
- 编译过程：预处理、编译、汇编、链接；理解 .h/.c 分工与重复定义问题。

把数据结构学在“用得上”的地方

- 数组与结构体：组织传感器、遥控器和电机状态；建立“数据对象”意识。
- 链表：理解动态连接思想即可；嵌入式初期优先静态内存，避免无必要的 malloc/free。
- 栈、队列、环形缓冲区：对应函数调用、任务通信和串口收发缓存。
- 查找与排序：掌握时间复杂度的直觉，知道何时线性查找已经足够。
- 有限状态机：用明确状态与事件替代杂乱 if-else；为机器人模式管理打底。

同步补齐的计算机原理

- 二进制、十六进制、补码、位运算、掩码与寄存器字段。
- 栈区、静态区、只读区与堆；变量生命周期、对齐、大小端。
- 函数调用与参数传递；为什么递归和大数组可能耗尽嵌入式栈。
- CPU 执行指令、内存保存数据、外设通过地址被访问的基本图景。

四周任务与验收

周次	训练任务	验收标准
第 1 周	控制台单位换算器；练习分支、循环、函数	无编译警告；非法输入有处理
第 2 周	学生/电机参数管理器；结构体、数组、模块拆分	头源文件分离；接口命名一致
第 3 周	定长环形缓冲区；入队、出队、满/空判断	边界测试通过；不使用动态内存
第 4 周	串口风格命令解析器 + 有限状态机	可解析 set/get/start/stop；异常命令不崩溃

推荐课程

B 站 | 翁恺老师《C 语言程序设计》

<https://www.bilibili.com/video/BV1XZ4y1S7e1/>

通过标志 能在不复制现成代码的情况下实现环形缓冲区与状态机；能画出关键变量的内存关系；能解释 volatile 为什么常用于中断共享变量。

04 第二阶段：电路基础与 STM32 入门

阶段目标 | 第 5–8 周 让代码第一次跳出屏幕。同步学习模电、数电与微型计算机原理，知道信号为什么可靠、外设为什么工作、错误该从软件还是硬件查起。

数字电路：理解单片机看到的 0 和 1

- 逻辑门、真值表、布尔代数；组合逻辑与时序逻辑的区别。
- 高低电平阈值、上拉/下拉、开漏输出、三态、施密特触发与按键消抖。
- 时钟、建立/保持时间、传播延迟；结合 UART、SPI、I²C 时序图阅读。
- 触发器、计数器、分频与寄存器概念；对应 STM32 定时器和 GPIO 寄存器。

模拟电路：让采样、供电与驱动更可信

- 欧姆定律、KCL/KVL、分压、RC 充放电、低通滤波与截止频率。
- 运算放大器的理想模型、跟随/放大/比较；理解传感器信号调理。
- ADC 量化、参考电压、采样时间、源阻抗与噪声；避免把抖动都归咎于软件。
- LDO 与 DC-DC、去耦/旁路、地线与回流、MOSFET 开关、续流与反接保护。
- 电气安全：先核对电压、电流、极性和共地；首次上电使用限流电源。

STM32 与计算机原理同步理解

- CPU、Flash、SRAM、总线和存储器映射；寄存器本质上是有地址的硬件接口。
- 时钟树、复位与启动流程；理解系统频率如何影响定时器和串口波特率。
- 轮询、中断、DMA 的差异；在 CPU 占用、实时性和复杂度之间做取舍。
- GPIO、EXTI、Timer/PWM/Input Capture、ADC、UART、I²C、SPI、CAN 的典型用途。

阶段项目：传感采集与 PWM 控制台

模块	实现内容	必须测量
输入	按键/编码器 + 模拟传感器采集	按键抖动、ADC 原始值与滤波值
输出	LED/蜂鸣器 + PWM 调速或舵机	PWM 频率、占空比与实际响应
通信	UART 命令与状态回传	波特率、丢包/错误输入处理
扩展	任选 I ² C/SPI 传感器	逻辑分析仪时序与数据手册对照

推荐课程

B 站 | 江协科技《STM32 入门教程》

<https://www.bilibili.com/video/BV1th411z7sn/>

通过标志 能依据数据手册接线与配置至少 4 类外设；能用示波器或逻辑分析仪证明时序正确；遇到故障时按供电—时钟—引脚—配置—软件流程逐项定位。

05 第三阶段：STM32 HAL 工程化开发

阶段目标 | 第 9–11 周 从“外设能跑”升级为“代码能复用、能协作、能维护”，接轨战队现有代码体系。

核心知识

- CubeMX：芯片引脚、时钟树、外设参数、DMA 与 NVIC；生成后知道哪些区域可安全修改。
- HAL 调用模型：初始化、阻塞接口、中断接口、DMA 接口与回调函数。
- 软件分层：BSP（板级）—Driver（设备）—Service（业务服务）—App（应用）。
- 非阻塞设计：用状态机、时间戳和回调替代长延时；为 FreeRTOS 做准备。
- 可诊断性：错误码、断言、串口日志、关键变量监视、HardFault 初步分析。
- 协作习惯：Git 小步提交、README、接线表、接口说明和最小复现步骤。

阶段项目：可复用驱动小仓库

驱动	接口要求	验收
UART + DMA	初始化、发送、空闲中断接收、环形缓冲	连续通信 30 分钟无卡死
传感器	init/read/self_test；错误码与超时	断开设备可报告错误并恢复
执行器	enable/set/stop；参数限幅	异常命令不会输出危险占空比
应用层	不直接依赖具体引脚和 HAL 句柄	更换端口仅修改 BSP/配置层

推荐课程

B 站 | 铁头山羊《STM32 HAL 库教程》

<https://www.bilibili.com/video/BV16J4m1w7HB/>

通过标志 能够从零创建 HAL 工程并完成 DMA 通信；驱动有清晰接口、超时与错误返回；代码仓库能由另一名成员按 README 独立复现。

06 第四阶段：数学、自动控制与 PID

阶段目标 | 第 12–16 周 把“凭感觉调参”升级为“从模型、指标和数据出发调参”。线性代数与自动控制原理只学当前闭环需要的部分，并立即映射到电机。

线性代数：为状态与多变量系统准备语言

- 向量与坐标系：位置、速度、姿态和传感器量可以作为状态向量。
- 矩阵运算与线性变换：理解坐标变换、耦合关系和多输入多输出表达。
- 线性方程组、秩与可逆性：判断信息是否足够、方程是否有唯一解。
- 特征值与特征向量：先建立“决定系统自然动态与稳定趋势”的直觉，为 LQR 打底。

自动控制原理：理解闭环为什么有效

- 建模与框图：对象、执行器、传感器、控制器、参考量、扰动与反馈。
- 微分方程、传递函数、极点/零点；关注概念与物理意义，不追求机械推导。
- 稳定性与动态指标：上升时间、超调量、调节时间、稳态误差。
- 频率响应直觉：带宽、相位滞后、噪声放大；理解滤波与响应速度的交换。
- 采样与离散化：采样周期、零阶保持、延时和抖动会改变闭环表现。

PID：机器人的“肌肉记忆”

- P 决定纠错力度，I 消除稳态误差，D 抑制变化趋势；先明白作用再记公式。
- 位置式与增量式 PID；积分限幅/分离、输出限幅、抗饱和、微分先行与低通滤波。
- 前馈 + 反馈、速度环 + 位置环串级控制；内环必须比外环更快。
- 整定顺序：确认方向与量纲 → 固定周期 → 只开 P → 加 D → 最后加 I → 扰动验证。
- 所有调参必须记录采样周期、负载、电源、目标、参数与曲线，禁止只记“手感不错”。

阶段项目：单轴电机速度/位置闭环

里程碑	输出	验收指标示例
M1 数据可信	编码器读取、方向、单位换算、固定周期	静止零漂可解释；周期抖动可测
M2 速度环	目标/反馈曲线；限幅与抗饱和	阶跃稳定；超调与调节时间有记录
M3 位置环	位置外环 + 速度内环	不同目标和负载下均不持续振荡
M4 鲁棒性	扰动、断传感器、堵转等测试	异常进入安全状态；恢复逻辑明确

推荐课程

B 站 | 华南小虎队《PID 从理论到实践》

<https://b23.tv/SB4iQAO>

通过标志 能解释一组曲线背后的超调、稳态误差和噪声；能按固定流程独立完成 PID 整定；能说明采样周期、限幅和滤波为何会影响稳定性。

07 第五阶段：FreeRTOS 实时操作系统

阶段目标 | 第 17–19 周 将已经验证的驱动与控制环组织成可预测的多任务系统。重点攻克任务创建、任务通知和中断到任务的数据链路。

核心知识

- 任务模型：状态、优先级、时间片、阻塞与延时；理解“实时”是按时完成，不是单纯跑得快。
- 栈与内存：静态/动态创建、栈大小、High Water Mark；延续第一阶段的内存模型。
- 任务通知：轻量级事件/计数/数据传递，适合中断唤醒单个任务。
- 队列、信号量、互斥锁、事件组、软件定时器：按通信、同步与资源保护目的选型。
- 中断规则：中断中只做必要工作，使用 FromISR API，及时唤醒高优先级任务。
- 共享资源：临界区、优先级反转与优先级继承；禁止无边界等待。
- 系统可靠性：看门狗、超时、任务心跳、栈监测、CPU 占用和故障降级。

推荐任务架构

任务	建议触发	主要职责	注意事项
Control	固定周期 1–2 ms	采样、控制计算、执行器输出	最高业务优先级；不打印日志
Communication	中断通知 / 队列	遥控、CAN/UART 解析与命令更新	校验、超时、断联保护
Sensor	周期或数据就绪	非关键传感器采集与滤波	DMA；避免长阻塞
Monitor	10–100 ms	状态机、心跳、故障检测、看门狗	汇总而非制造竞争
Logger	低优先级队列	异步输出日志与数据	队列满时有丢弃策略

推荐课程

B 站 | 正点原子《FreeRTOS 手把手教学》

<https://www.bilibili.com/video/BV19g411p7UT/>

通过标志 能画出任务、队列、中断之间的数据流；关键控制周期可测且稳定；能通过日志和任务监测定位死锁、栈不足、阻塞超时或优先级设计问题。

08 第六阶段：综合项目与战队接轨

阶段目标 | 第 20–24 周 用一个完整子系统证明你能把软件、硬件、控制与实时系统连成闭环，并能在团队规范下交付。

建议综合项目（二选一）

项目	核心内容	可展示结果
双轮底盘最小系统	遥控/上位机命令、两电机速度环、CAN/UART、模式管理、失联保护	直线、定速、急停、扰动恢复；实时曲线
单关节云台/机械臂	编码器/IMU、串级 PID、限位、轨迹命令、故障保护	多角度定位、负载变化、软限位与复位

五周项目节奏

周次	工程活动	评审材料
第 20 周	需求冻结、接口表、风险清单、模块分工	一页需求 + 系统框图 + 测试指标
第 21 周	板级点亮、驱动自测、通信联调	接线图 + 驱动测试记录
第 22 周	控制闭环、参数记录、边界条件	曲线 + 参数表 + 调参结论
第 23 周	RTOS 集成、故障注入、连续运行	任务图 + 故障矩阵 + 稳定性记录
第 24 周	演示、代码评审、复盘与交接	仓库 + README + 5 分钟答辩

必须具备的安全与故障处理

- 上电前检查极性、供电范围、共地、短路与机械限位；首次运行限制电流和输出。
- 遥控/通信超时立即进入安全状态；传感器异常、堵转、过温/欠压有检测或保底策略。
- 输出统一限幅；控制器切换模式时处理积分残留和目标突变。
- 每个故障都有：检测条件、记录信息、执行动作、恢复条件，禁止静默失败。

09 24 周执行日历

周	主题	本周必须交付
1	C 语法与函数	单位换算器；0 warning
2	类型、数组、结构体	参数管理器；模块拆分
3	指针、内存、环形队列	环形缓冲区 + 边界测试
4	状态机、编译链接	命令解析器 + 口头讲解
5	数电、电平与 GPIO	按键/LED；消抖测量
6	模电、ADC 与电源	模拟采样 + 滤波曲线
7	定时器、PWM、中断	PWM 控制 + 时序测量
8	UART/I ² C/SPI/CAN	传感采集控制台
9	CubeMX/HAL 与时钟树	从零创建 HAL 工程
10	DMA、回调与非阻塞	UART DMA 稳定通信
11	分层、调试与 Git	可复用驱动仓库评审
12	向量、矩阵与状态	电机状态量与单位表
13	建模、传递函数、指标	闭环框图 + 指标定义
14	采样、滤波与测速	可信速度反馈曲线
15	速度 PID	阶跃响应 + 参数记录
16	串级控制与保护	位置环 + 扰动/堵转测试
17	任务、优先级、栈	任务规划与周期测量
18	通知、队列、互斥	中断到任务数据链
19	监测、看门狗、诊断	多任务系统连续运行
20	项目需求与设计	需求、框图、接口、风险
21	驱动与通信集成	模块自测与联调记录
22	闭环控制集成	控制曲线与参数表
23	故障注入与压力测试	故障矩阵 + 运行记录

周	主题	本周必须交付
24	演示、答辩、交接	仓库、README、复盘

10 评审标准与成长刻度

维度	权重	达到要求	优秀表现
功能正确	20%	需求内功能稳定运行	边界和异常路径同样可靠
理论理解	15%	能解释关键原理与参数	能比较方案并说明取舍
测量验证	20%	有波形、日志、曲线或测试	数据可复现，有指标和结论
代码质量	15%	模块清晰、无明显隐患	接口稳定、错误处理完整
系统可靠性	20%	有超时、限幅和基本保护	完成故障注入与长时运行
协作交付	10%	仓库、README、演示齐全	他人可快速复现和二次开发

建议门槛 总分 75 分且“系统可靠性”不低于 60%。涉及电机、高电流或机械运动时，安全项一票否决。

常见误区

- 只追视频进度，不做实验：课程完成率不能替代工程能力。
- 一上来就用 RTOS：先让单模块稳定，再增加调度复杂度。
- PID 只凭感觉：没有固定周期、单位、曲线和限幅，参数不可复用。
- 硬件问题全靠软件补偿：供电、电平、接地和信号完整性必须先确认。
- 只做正常流程：赛场可靠性来自超时、断联、堵转和传感器异常测试。

11 加入战队后的进阶方向

完成全部学习并通过综合项目后，你将成为电控组正式一员，开始在真实机器人研发中选择更深的技术方向。

方向	前置基础	可进入的项目
MATLAB / Simulink 建模仿真	线性代数、自动控制、实验数据	系统辨识、控制器仿真、参数扫描
LQR 与状态空间控制	矩阵、特征值、状态方程	云台、平衡、底盘多变量控制
机械臂运动学	坐标变换、矩阵、三角函数	正逆运动学、轨迹规划、标定
能量智能分配	电路、电源、优化与实时数据	功率限制、超级电容、策略调度
电子系统自主研发	模电数电、PCB、EMC、测试	驱动板、传感板、功率板迭代

最终目标 不是“学完所有知识”，而是具备持续拆解问题、快速验证、可靠交付的能力。赛场只是最严格、也最诚实的一次系统测试。

12 课程索引

C 语言 | 翁恺老师《C 语言程序设计》

<https://www.bilibili.com/video/BV1XZ4y1S7e1/>

STM32 | 江协科技《STM32 入门教程》

<https://www.bilibili.com/video/BV1th411z7sn/>

STM32 HAL | 铁头山羊《STM32 HAL 库教程》

<https://www.bilibili.com/video/BV16J4m1w7HB/>

PID | 华南小虎队《PID 从理论到实践》

<https://b23.tv/SB4iQAO>

FreeRTOS | 正点原子《FreeRTOS 手把手教学》

<https://www.bilibili.com/video/BV19g411p7UT/>

注：课程链接沿用原路线推荐。模电、数电、线性代数、自动控制原理与计算机原理建议结合校内课程、教材和阶段项目
按需补充；优先掌握本路线列出的应用知识点。