# 电赛控制题知识点

目录

[电赛控制题知识点 1](#_Toc110893626)

[基础篇 2](#_Toc110893627)

[一、C语言基础 2](#_Toc110893628)

[1.变量 2](#_Toc110893629)

[2.表达式 2](#_Toc110893630)

[3.输出函数(前期不重要) 3](#_Toc110893631)

[4.函数 3](#_Toc110893632)

[5.逻辑结构 3](#_Toc110893633)

[6.数组 3](#_Toc110893634)

[7.进制转换 3](#_Toc110893635)

[8.指针 3](#_Toc110893636)

[9.函数的指针传递 4](#_Toc110893637)

[10.宏定义 4](#_Toc110893638)

[11.结构体 4](#_Toc110893639)

[12. 自定义类型名typedef 4](#_Toc110893640)

[13.多文件编程 4](#_Toc110893641)

[二、硬件 4](#_Toc110893642)

[1.元器件了解 4](#_Toc110893643)

[2.电路设计 5](#_Toc110893644)

[3.画单片机底板（后期再学） 5](#_Toc110893645)

[三、单片机学习 5](#_Toc110893646)

[1.入门初期 5](#_Toc110893647)

[2.进阶 6](#_Toc110893648)

[3.高级 6](#_Toc110893649)

[提高篇 6](#_Toc110893650)

[一、硬件 6](#_Toc110893651)

[1.降稳压模块 7](#_Toc110893652)

[2.红外模块与灰度模块 7](#_Toc110893653)

[3.电机及驱动模块 8](#_Toc110893654)

[4.测距模块 8](#_Toc110893655)

[5.编码器 8](#_Toc110893656)

[6.陀螺仪 8](#_Toc110893657)

[7.显示屏 8](#_Toc110893658)

[8.无线通信模块 9](#_Toc110893659)

[9.功率放大模块 9](#_Toc110893660)

[10.其它模块 9](#_Toc110893661)

[二、软件 9](#_Toc110893662)

[1.PID模型（重要） 9](#_Toc110893663)

[2.通信 9](#_Toc110893664)

[3.STM32拓展知识 9](#_Toc110893665)

[4.python基础 10](#_Toc110893666)

[三、图像处理 10](#_Toc110893667)

[1.OpenMV（推荐） 10](#_Toc110893668)

[2. K210 11](#_Toc110893669)

[3.树莓派 11](#_Toc110893670)

[四、补充 11](#_Toc110893671)

# 基础篇

## 一、C语言基础

以下内容包含在平时开发过程中会用到的C语言知识点，都是会遇到且常用的知识点，学习的时候重点关注。

### 1.变量

理解掌握变量的定义、声明、类型、**数值范围（重点）**和每种变量所占的内存，理解全局和局部变量。

### 2.表达式

掌握变量之间的简单运算，掌握逻辑运算符：&&(且) ，||(或) ，!(非) 以及布尔运算：&（与），|（或），^（异或）。了解运算符优先级。

### 3.输出函数(前期不重要)

掌握printf。（32单片机串口通信会用到）

### 4.函数

能调用库函数，掌握自定义函数。重点关注函数的值传递和返回值。

### 5.逻辑结构

选择结构if……else……， if……else if……else…… 。

循环结构for和while。

选择结构switch。

break语句。

### 6.数组

掌握整型数组、字符型数组、浮点型数组的定义和使用。重点是一维数组，二维数组初期用的不多

### 7.进制转换

理解2、10、16进制的相互转换。（理解就可以，计算可以用手机）

### 8.指针

掌握一重指针就可以（重点的重点，经常用到），数组名与指针互换应用。

### 9.函数的指针传递

能灵活使用函数的值传递和指针传递

### 10.宏定义

重要

### 11.结构体

学32单片机的时候会用到，前期稍微理解就好，可以学32单片机的时候再深入

### 12. 自定义类型名typedef

学32单片机的时候会用到，可以后期再学。

### 13.多文件编程

多文件间变量和函数的相互调用。

## 二、硬件

硬件部分较简单，掌握基础就行。

### 1.元器件了解

了解开关、按键、电阻、电容、电感、二极管、三极管并能简单使用。

### 2.电路设计

根据原理图，能用立创画出电路图并布线打板。

### 3.画单片机底板（后期再学）

根据购买的单片机核心板，绘制拓展板，上面预留开关和各种接口。

## 三、单片机学习

### 1.入门初期

选择51单片机入门（较简单且推荐，能很快熟悉Keil的使用及理解单片机编程的原理，学32的时候能很快入门），直接上stm32（入门的时候感觉有点搞心态）。

对于开发环境，有好几个选择，**（个人见解，仅供参考）**首推Keil，理由是对于51和32的开发，网上百分之八十的例程都基于Keil开发，很多例程可以直接拿来用。我们也储备了一些基于Keil开发的代码。

其次是万金油Vscode，安装好相应插件后就可以直接开发。界面美观，代码提示补全功能也比Keil好用，等熟悉Keil开发了之后再使用Vscode开发是个不错的选择。

IAR，听说挺好用，也挺万金油，没有过，就不发表看法了。

掌握引脚输出高低电平，掌握输入功能，实现按键和流水灯。

### 2.进阶

理解中断的概念。理解单片机的时钟（51的优势又体现出来了，功能全面，时钟简单，方便理解），了解时钟周期和机器周期（不重要）。理解外部中断和定时器中断。利用定时器实现精准延时。利用单片机计数器实现外部脉冲个数的计数。利用串口实现单片机与电脑之间通信。

### 3.高级

学习stm32，入门推荐stm32f103c8t6，其它型号也行，但推荐f103系列的。理由：f103系列代码差不多都是通用的，且网络上例程是最丰富的，基本上能找到涵盖学习的所有例程。

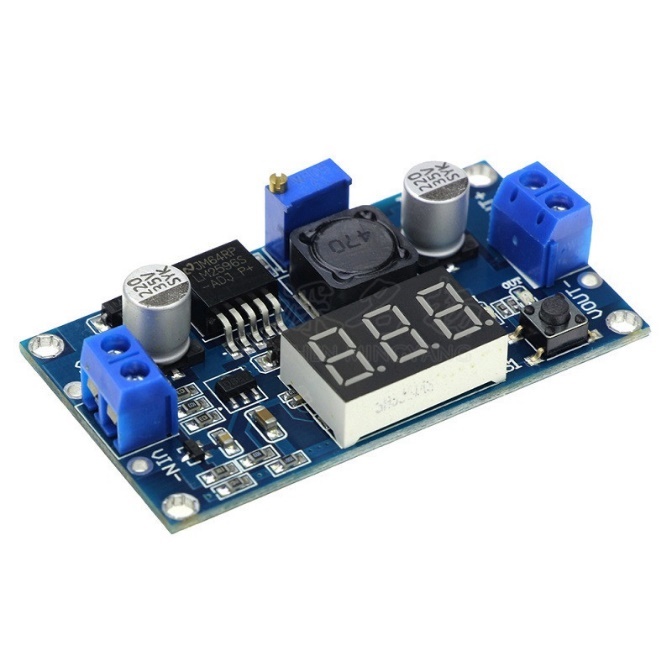
能实现上述**2.进阶**中的所有功能。如果使用固件库开发，最好能自己创建工程模板（非必要），要能实现工程在小、中、大容量间的相互移植。

# 提高篇

## 一、硬件

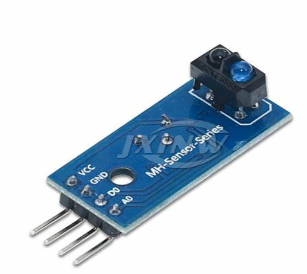
硬件部分主要掌握各个模块的原理和应用。

### 1.降稳压模块



可以自己设计，也可以自己买。主要给单片机供电。

### 2.红外模块与灰度模块

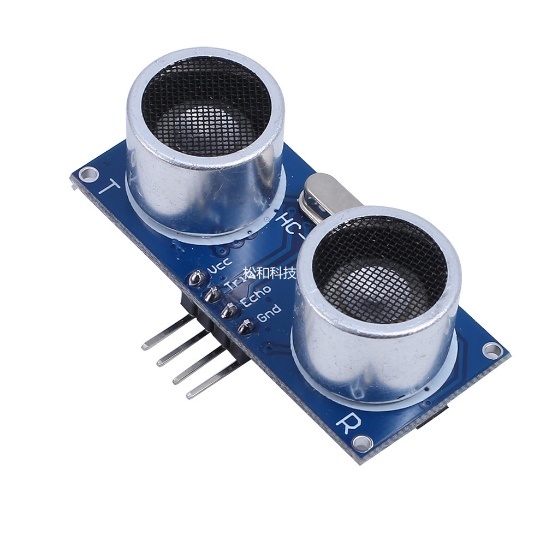


主要用于巡线，能使用就行

### 3.电机及驱动模块

能使用电机驱动模块驱动电机，实现电机的调速和正反转等。电机驱动模块可以自己设计，也可以直接购买。

### 4.测距模块



超声波或激光测距模块的使用

### 5.编码器

掌握编码器的使用，能读取编码器值

### 6.陀螺仪

掌握陀螺仪的使用

### 7.显示屏

OLED与TFT LCD屏，能显示字符和数字。

### 8.无线通信模块

有WiFi、蓝牙、2.4G模块等。（对于WiFi，esp8266与esp32可玩性无疑很高，但连接速度过慢，不推荐使用作为通信模块）。推荐zigbee模块，听说上电即相连。

### 9.功率放大模块

最好能自己做，主要用于扬声器驱动

### 10.其它模块

拿到模块，了解原理，快速应用

## 二、软件

### 1.PID模型（重要）

简单了解位置式pid和增量式pid。能读懂别人的pid模型并根据自己的需求修改应用。可以装一辆循迹小车或两轮平衡小车来练习。

### 2.通信

能使用串口通信、spi通信、IIC通信。串口通信最好能自己写可靠的协议，在使用无线模块时要用到。

### 3.STM32拓展知识

熟练掌握使用输入捕获，中断优先级、ADC、定时器的正交编码器模式、DMA(非必要)。

### 4.python基础

主要用于图像处理方面。如果C语言学的好，这个入门很快。

## 三、图像处理

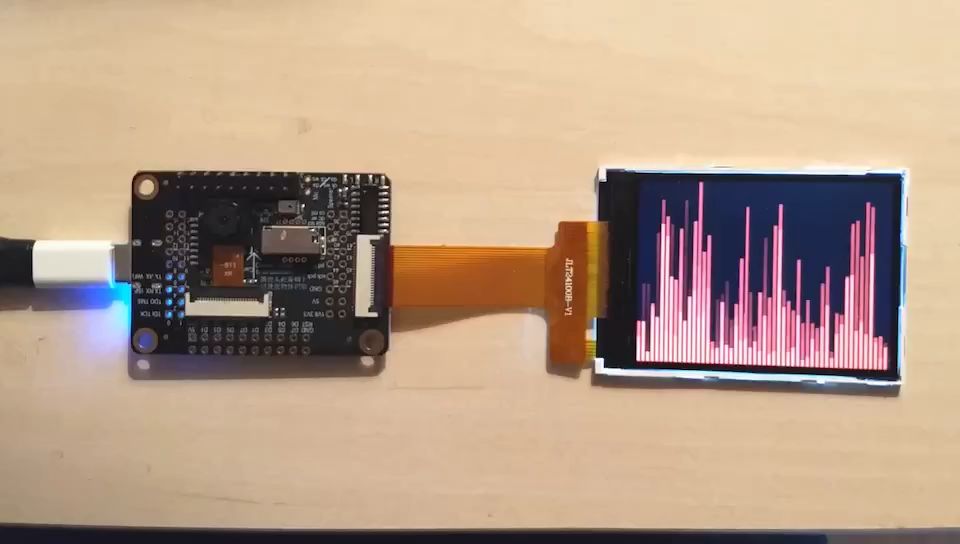
掌握其中任意一款就行

### 1.OpenMV（推荐）



性能够用，例程丰富，入门简单，教程做的好。

### 2. K210



性能强，功能强大，性价比高。

### 3.树莓派

可以跑OpenCV库，图像处理功能最强

## 四、补充

国赛不限主控，推荐提前学习使用stm32大容量型号。如stm32f103RCT6，stm32f103ZET6等。

省赛小车题限制主控为TI公司的主控，推荐：无

个人见解，有很多不足，望补充