## 第五周学习报告

这周开例会被科协的学长学姐训了之后终于开始学习stm32单片机了。

在第四周的时候已经买了野火的开发板“霸道v2”图片包含 游戏机, 电路

描述已自动生成

第一眼看上去和51的开发板相比，真的复杂好多啊，心理压力就上来了。。。。。。

顺带买了野火的stm32库开发指南这本书，挺厚一本，心理压力继续上升。。。。。。

首先就是keil5的安装以及下载stm32芯片包，搞了我两天，因为我一开始不知道野火有资料，在keil的官网上弄了好久，后来从野火的百度网盘里下载了极其详细的资料，终于配置并安装好了keil5和芯片包图片包含 表格

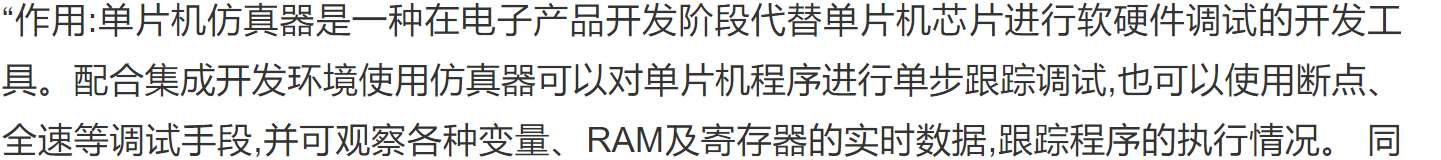
描述已自动生成

文本

低可信度描述已自动生成

资料超级详细，真的感叹现在网络的超能力，所以没有啥东西是学不到的，只有不想学。

也认识了DAP仿真器，也比较清楚仿真器是干嘛的

百度如下：

基础篇第一部分就是认识stm32，即ST公司开发的32位微控制器，也知道32能做什么，比如手环、平衡车、微型四轴飞行器等等

接下来就是寄存器，学51的时候一直没有详细了解啥是寄存器，感觉就和存储器差不多，都是用来存储数据的，学了32才知道寄存器就是有特定功能的内存单元，现在才想起学51的时候控制什么SCON寄存器，原来是这样的哈哈哈哈哈

深入了解啥是寄存器后便开始了实际操作，但操作之前还有对keil工程建立的规范学习和编程配置，不过这方面和学51的时候差不多，只不过多了一些仿真器的配置以及.h/.c文件的包括等等，比51稍稍复杂一点点

接下来就是激动人心的点灯时刻，第一次的点灯是直接控制寄存器来进行的，没有使用固件库（当时还不知道固件库呢，后来才发现固件库是32的精髓）。在这里知道了GPIO的控制输入输出的端口，即stm32可控制的引脚,了解了GPIO的结构图，但是还没学数电模电，所以没怎么看换行（什么P-MOS,M-MOS什么的），知道了输出数据寄存器GPIOX\_ODR(O就是output)，后面还有IDR（输入数据寄存器），还有控制端口高低控制寄存器CRH和CRL，当时感觉这几个寄存器贼乱，后来看野火的视频总结了一下，这三个玩意儿就是让GPIOX初始化的，顺序分别是1、选定具体的GPIO 2、配置GPIO的工作模式（即对CRH和CRL写入数据） 3、控制GPIO输出高低电平（ODR\BRR\BRRS），这样一来我就对这几个玩意儿清晰多了，到后来使用固件库点亮的时候让我对这几个寄存器更加印象深刻，以及背后的基本操作原理更加的熟悉了

然后就到“自己写库----构建库函数雏形”的学习了，知道了啥是stm32函数库，之前学51的时候都没有用到函数库写，是因为51直接写寄存器是比较简单的，而32就不一样了，因为寄存器都不是8位，都是32位了，如果直接写寄存器的话很麻烦每次都要看官方的寄存器说明，所以stm函数库诞生了，我觉得这个有点东西的哈哈哈。知道了直接写入寄存器和用库函数控制寄存器的优缺点文本

描述已自动生成

Ps：这本书真的不错啊，讲的挺好的，和视频配套学习超级棒

然后就按照书上的操作来自己写了一遍函数库（好多宏定义，看的眼花，但是定义后发现真的超级清晰，很舒服的）还有结构体呢，差点忘了C语言这个知识了哈哈哈哈

后来认识了STM32标准库，就是ST公司推出的库模型，还有在keil上新建工程，库函数版，就是工程模板，以后创建工程直接复制就完事了

接下来又来了激动人心的点灯时刻，这次试用固件库来点亮啦，一开始运用固件库的函数还不是很熟悉，按照视频里的敲了一遍代码后灯亮了，是绿色的，后来我自己把代码以及函数都弄清楚后举一反一，把灯变成了蓝色，这个灯还是RBG灯，比51好玩哈哈哈

电子仪器

中度可信度描述已自动生成

文本, 应用程序

描述已自动生成

才发现用宏定义真的很清晰很方便，移植性高，

这就是第五周的学习啦，准备开学了，开学后会继续认真学习了