

嵌入式总结体会(汇总5篇)

当工作或学习进行到一定阶段或告一段落时，需要回过头来对所做的工作认真地分析研究一下，肯定成绩，找出问题，归纳出经验教训，提高认识，明确方向，以便进一步做好工作，并把这些用文字表述出来，就叫做总结。什么样的总结才是有效的呢？下面是小编整理的个人今后的总结范文，欢迎阅读分享，希望对大家有所帮助。

嵌入式总结体会篇一

嵌入式可视电话系统的出现满足了人们在低成本、低功耗的情况下进行音视频一体交流的愿望。接下来就跟着本站小编的脚步一起去看一下关于嵌入式处理学习心得总结吧。

现在的嵌入式工作是越来越盛行，而这方面的人员却是越来越少，所以学习嵌入式是解决找工作难的一个比较有效的途径。因为我开始踏上学习嵌入式的道路。听别人说嵌入式arm相比于其他的嵌入式fpga[dsp等都要难点，而且学习的内容量也是比较多。因此在学习嵌入式arm的初级阶段是随处碰壁，每天都在遭受打击，例如在移植bootloader时，虽然按照网上的一些方法移植成功了，但只知道复制和粘贴要修改的代码，却不知道为何要这么做。我觉得这样的学习态度是不对的，学习不是只求结果，而是要注重解决问题的过程!为了学习嵌入式linux我也在网上查阅了很多的书籍，还下载了很多的视频，但都不是很让人如意。它们大多是讲的很浅显，都是泛泛之谈，没有讲到最核心的东西。到了最后是浪费了我们学习者的时间和精力。之后看见师兄有一本韦东山老师编辑的《嵌入式linux应用开发完全手册》，我就借了几天看了看，发现韦老师对嵌入式linux的极其深度的分析，让我一下子找到了能带我入门的一本书!并且还有配套的视频教学更是讲解地非常到位，将教材和视频相互配合的学习，

更加大大的提供了学习效率，尤其是视频上讲解的内容，都是韦东山老师自己亲自授教，完全没有按照教材上的内容一一读出来，而是根据内核代码分析的教我如何编写自己的代码，尤其是在linux内核中占比例最重的一部分—内核驱动时，更是让我知道用户是如何通过应用程序来访问底层硬件的，解决了我长久以来都无法攻克的难题!!例如在编写lcd驱动时，我对于书上lcd硬件工作原理不是理解的很透彻，通过观看韦东山的第二期视频中的“lcd驱动程序”，老师就像是站在我面前，在屏幕上用电子笔细细给我讲解，写出更加形象的lcd的硬件驱动，重要的是在整个视频中老师都没有按着教材念，而是一个教学经验丰富的老师在给同学们授课一样。编写代码也是亲自编写，没有完全复制别人的代码，编写的同时不断的说明每条语句的作用、编程思想和设计程序的经验，还有阅读大量代码的技巧!在视频中老师教会了许多其他书本中没有提及到的重要知识。

本人大学学的是电子信息科学与技术专业，其实这个专业和嵌入式没有多大的关系，要说有关系就是简单的讲了下c语言语法，学完之后还不知道什么是c语言，只记住了[]char,int,string等占几个字节[]if, for, while循环什么的，感觉也就能算一些小学就会的题目，全国计算机等级二级也就糊涂的过了。我这个专业没学过c++[]一开始不知道什么是面向对象，什么叫面向过程，这可能让大家笑话了。不过这说的是实话，大学的时候比较喜欢动手，喜欢硬件方面的东西，也喜欢看书。在大一的时候就听学长说，单片机很难学，在没事的时候就从图书管借了本关于51的书，一开始还真是不怎么懂，然后硬着头皮去看，当我把51的结构看完了，就不像一开始那样一头雾水了，自己在纸上把51的结构图默画一遍，然后再把各个寄存器功能写一下，各个功能口的结构也画一遍，然后就不觉得51有什么难了。其实有些时候不是这个东西你不会，而是在不会的时候自己先告诉自己这个太难了，自己先放弃了。我们学习51芯片，是为了以后的打基础，知道是怎么通过配置寄存器来实现功能。

大二下学期的时候，我们开了51的课，这个课几乎就没怎么去，因为我认为没有必要去听自己会的东西了。后来听别人说avr的资源会比51的丰富的很多，所以就想试试，看看有什么难的，就从网上买了本马潮老师《avr单片机嵌入式系统原理与应用实践》，一边看书，一边想自己画一个板了，一开始想实现一个点阵屏，实现滚动效果，后来想想做个板也花不少钱呢，为什么不多加一些功能呢，还能多学到些知识，所以就加了温感，红外，铁电存储器等并把所有的扩展口都扩展了出来，备以后用，那时还挺自豪的呢，同学们都很羡慕，不过现在想想其实也没什么了。

大学也听说过嵌入式这个词，不过不太明白是什么意思。20xx年的时候从网上找到中嵌视频，有四张dvd(大约13g)下载下来后，从头到尾的看了一遍，没什么印象，只记得讲一些不知所云的函数，然后莫名其妙的讲如何使用那个44b0的开发板，不过听完了，还是有很大收获的，就有一种想装一个linux的冲动，我是大二下学期才有的自己电脑，不过电脑的配置比较低，1g内存，跑虚拟机就像个蜗牛，所以就想着怎么才能装个linux。自己用xp习惯了，如果全换了又舍不得，那时从书上知道linux的文件系统和windows不一样，只有一个根分区，所以也害怕一装linux全盘就被格式化了，不过最后还是欲望战胜了担心，买了张redhat 9的光盘，小心的装这个系统，里边的英文也不是太难，还是知道大概意思的，所以一步步的装，最后装到我的硬盘的最后空间，重新启动，竟可以选择进入哪一个系统了，从这之后，自己就不害怕怎么装系统了，应了那句言，纸上得来终觉浅，绝知此事要躬行。所以我认为，买一个开发板是完全有必要的，拿到实物操作一下就不会感觉太抽象了。后来也想学学arm，但不知道arm是什么东西，所以就买了本《从51到arm》讲的是关于arm的大体结构，看完后感觉，原来arm就是一个速度更快，接口更丰富的单片机埃 如果想做linux设备驱动的同学应该对arm硬件有一些了解。如果对驱动不感兴趣，完全没有必

要去看。交叉编译器会帮我们做好这些事情，他比我们更了解arm结构。

大三下学期开始准备考研了，所以自己这些兴趣就先放下了，报的北邮的信通学院通信专业，当时就是想一定要上个好学校好专业，也没想太多，结果考了的成绩差了点，没有被录取，也不想调剂一个二流的学校，然后就参加工作了，工作中每天都用的到开发板，当我看到一个小板上还可以跑系统的时候感觉很不可思议，以为要很大工程。后来边工作边学习，每天都要看电脑看到眼发疼，晚上也都差不多十二点时才睡觉。感觉到累，不过也感觉到自己在进步。

闲话有点多了，说了很多都是自己大学那点破事。

听很多的初学者说不知道从哪里开始，学习嵌入都要学哪些东西，从网上搜了一下，一大堆的东西要学，一下就推荐十几本书，说看完这本书怎么怎么样，看完那本书怎么怎么样，一下子就给初学者给吓住了，其实完全没必要把网上的那些东西全部看一遍，那样又费时，又感觉工程太大，让初学者没有信心去把每一种都看完。网上说的些方面都要知道，不一定全部都掌握，我们可以退成求其次，每一个都知道一些，当然会的越多越好。今天学点这个明天学点那，换着样来也不会觉得烦。而且还能感觉到每天都有收获。

下面给嵌入式初学者一些自己的我个人建议。

1嵌入式涉及到很多的基础知识：

c语言，关于c语言要知道基本的语法，什么是结构体，什么是共用体，有什么区别，不用强记一些二级考的东西，因在实际中用的很少。

c++要知道什么是类，怎么定义，继承，接口，要深刻理解，当然基本语法也是必不可少的。

操作系统原理也应该知道基本的原理的，我记得我大学没这个课还是去电信学院蹭别的专业的课呢，后来学习发现很多的东西都是在这个课上学到的。什么时间片啊，任务调度啊，都是在那里知道的。

linux基本命令的使用(vi编辑器必须会用，因为在终端改文件的时候只能用这个)，我们现在很多的人都习惯用鼠标了，很少用到命令，我们学习嵌入linux可要改改这个习惯了□linux用命令操作一是效率高，二是功能非常强大，远非图形界面能比的了的，其实我们用图形界面的操作都要转化为命令传给硬件的。

linux文件系统的管理要清楚，哪个目录存放是什么东西，有什么用，文件权限管理等 shell编程也要了解一些，在这给大家推荐一个教程，周朝剑的 shell编程，也就七八讲吧，讲的很全面。初学者看前4讲就可以了。

linux设备驱动的架构要了解，初学者可以以飞凌的ok6410开发板上的led驱动分析一下，结合原理图和linux设备驱动相关书籍把每一行代码都分析明白。关于驱动给大家推荐一本书□linux设备驱动程序》中国电力出版社出版的，很经典的，初学者看可能看不太明白，不过要坚持看，最次也得把前6章看完了。

2 看书不要希望一次就会，一次就明白是什么意思，也不要死记硬背，看不明白也没关系，以后遇到了知道去哪里去找就可以了，以后遇到的多了，也就记住了，遇到的少的，忘记也就忘记了。还有，不要直接阅读linux内核源代码，看一些自己把头撞破也想不明白的事，会打击初学者的积极性，要循序渐进，今天学个做个led,明天做个串口，后天说不定就做硬件编解码了。

3、你可以不接受培训，但是不能不买开发板。

如果你想学习嵌入式一定要买块学习板，现在的开发板很便宜了，给我们初学者减轻了很多经济负担。看再多的视频，看再多的书，也不如自己动手试一下，看着很简单的东西，做起来就可能遇到很多的问题，遇到问题找到原因，解决问题，只有这样才能学到东西，记得我第一次写led的驱动，想自己试一下，网上也有人写过相关的代码，可是自己一试，就出了好多的问题，那个led弄了我一天才把led灯点亮。所以不要眼高手低。

还有关于培训，现在的培训机构很多，讲的也不错，老师也很好，但是有一个问题，太集中了，而且学生动手的机会也不太多，记得我们公司还招了一个在北京某个培训机构培训半年的员工呢，花了一万多，来了之后也没感觉有多高，问他都学了什么，也就是我前边说的那些基础知识，很多的东西还是不知道的，照着书弄个helloworld模块驱动两天没搞定。不是培训不好，但我个人见解效果并不是很好。记住：只有自己动手做过的东西才是自己的。

错，但是不适合初学者，因为他的很多的东西是不给代码的，用过友善的板子的同学经常有这样的痛苦，想学一个东西，去找他的代码的时候，却找不到，很多友善提供的代码中很多都是封装好的，只提供给客户一个库，初学者想要找代码的时候就不知道怎么办了。

4 不要好高骛远，要脚踏实地

5 多看代码，多写代码

见多识广，熟能生巧。多看代码，要从代码中读出代码的意思来。自己平时也要多写代码，多练习(这一点我做的不好，写的少)。

写完已经晚上两点了，思路有些乱，有些也是个人的见解，在有些人看来可能是错误的，每个人有每个人的学习方法，

但是我们的目的只有一个——希望我们学到更多的知识。

首先，嵌入式系统是以应用为中心，以计算机技术为基础，且软硬件可裁减，适应应用系统对功能、可靠性、成本、体积、功耗有严格要求的专用计算机系统。嵌入式有如下几个突出的特点1. 系统内核小，2专用性强，3系统精简4高实时性5嵌入式系统的应用程序可以没有操作系统直接在芯片上运行，6嵌入式系统需要开发工具和环境。

嵌入式系统是近几年才风靡起来的，但是这个概念并非新近才出现。从20世纪七十年代单片机(最早的单片机是intel公司的 8048，出现在1976年)的出现到今天各式各样的嵌入式微处理器，微控制器的大规模应用，嵌入式系统已经有了近30年的发展历史。经过30年的发展，嵌入式技术已经日趋成熟，在中国嵌入式机一般由嵌入式微处理器、外围硬件设备、嵌入式操作系统以及用户的应用程序等四个部分组成，用于实现对其他设备的控制、监视或管理等功能。

其次，嵌入式系统无疑是当前最热门最有发展前途的it应用领域之一。嵌入式系统用在一些特定专用设备上，通常这些设备的硬件资源(如处理器、存储器等)非常有限，并且对成本很敏感，有时对实时响应要求很高等。特别是随着消费家电的智能化，嵌入式更显重要。

嵌入式系统是软硬结合的学科，通常有两类人会去专研嵌入式：一类是学电子工程、通信工程等偏硬件专业的，另一类是学软件、计算机专业出身的人，主要从事嵌入式操作系统和应用软件的开发。我是软件专业的学习嵌入式，越是智能设备越是复杂系统，软件越起关键作用，而且这是目前的趋势。嵌入式应用在很多方面，具体的可以分为工业控制，交通管理，信息家电，家庭智能管理系统□pos网络及电子商务，环境工程与自然，机器人，机电产品方面应用等方面。其中目前在服务领域，如远程点菜器等已经体现了嵌入式系统的优势。例外还有很多地方用到了嵌入式系统，最近飞利浦

和arm共同推出32位risc嵌入式控制器，适用于工业控制，采用最先进的0.18微米cmos嵌入式闪存处理技术，操作电压可以低至1.2伏，它还能降低25%到30%的制造成本，在工业领域中对最终用户而言是一套极具成本效益的解决方案。

嵌入式总结体会篇二

16年我第二件大事情就是买房，关注过近几年房价的人可能还记得，16年一线城市猛涨的情景。其实当时15年底，上海市中心和学区房已经开始上涨，我15年底听同事开始讨论上涨的房价，我心里开始有了买房的打算，大约16春节（2月份吧，具体记不得了），我回老家探望父母，同时跟他们提出了买房的打算。

我的父亲是一个“央视新闻爱好者”，爱好看狼咸平xx刀，xx檀的节目，大家懂了吧，父亲说上海房价太高了，都是泡沫，不要买。

这个时候我已经不是菜鸟了，我想起我总结的第一条经验（见上文），我开始收集往年的房价数据，中央历年的房价政策，在复盘15年的经济政策时我发现，当年有5次降息降准，提升公积金贷款额度。

放松贷款要求于是我判定房价一定会继续涨，涨到一个幅度各地才会出台各种限购政策，并且房价在城市中是按内环往外涨的于是我开始第一次在人生大事上反对父母，我坚决表态要买房。

父亲还是不太同意，他说年底吧，先看看情况（实际是年底母亲的退休公积金可以拿出来大概十几万吧，另外未来丈母娘的公积金也能拿出来大概比这多些）。

我还是不同意，父亲最终拗不过我，终于松口，于是我们拿

着双方家庭凑的 50w 现金开始买房，后来上海的房价大家都看到了。这件事也是我做的不多的正确的事情之一。

但是最可笑的是，我研究房价的同时居然犯下了一个匪夷所思的错误，我居然没有研究买房子最重要的因素是什么，我们当时一心想买一手房（现在想想真是脑子进水），最后买了一套松江区交通不便的房子，这第一套房子的地理位置也为我后来第二次离职埋下了隐患，这个后面会说。

一线或者准一线城市能买尽量买，不要听信房产崩溃论，如果买不起，那可以在有潜力的城市群里用父母的名义先买一套，毕竟大多数人的财富其实是涨不过通货膨胀的。另外买房最重要的三个要素是，地段，地段，地段。

买房的那天上午和女朋友领的证，话说当时居然把身份证写错了三次。。。

这下我终于算是有个家了，交完首付那个时候身上真的是身无分文了。航天的基层员工的收入真的是不高，我记得我当时作为组长，每月到手大概也就 7k-8k 的样子，另外有少量的奖金，但是总数仍然不高，好在公积金比较多，我日常也没什么消费欲望，房贷到是压力不大。

买完房子之后，我心里想，这下真的是把双方家庭都掏空了（我们双方家庭都比较普通，我的收入也在知乎垫底，没办法）万一有个意外怎么办，我想来想去，于是在我下一个月发工资之后，做了一个我至今也不知道是对是错的举动，我利用当月的工资，给全家人家人买了保险保险，各种重疾，意外都配好了。

但是为什么我至今也不知道对错呢，因为后来老丈人，我母亲都遭遇病魔，但是两次保险公司都拒赔，找出的理由我真是哑口无言，谁叫我近视呢。另外真的是要感谢国家，亲人重病之后，最终还是走了医保，赔偿了部分，不然真的是一

笔不小的负担。

嵌入式总结体会篇三

嵌入式系统无疑是当前最热门最有发展前途的it应用领域之一。以下本站小编为大家带来嵌入式系统学习心得总结，希望对大家有所帮助！

嵌入式系统用在一些特定专用设备上，通常这些设备的硬件资源(如处理器、存储器等)非常有限，并且对成本很敏感，有时对实时响应要求很高等。特别是随着消费家电的智能化，嵌入式更显重要。像我们平常常见到的手机[]pda[]电子字典、可视电话[]vcd/dvd/mp3 player[]数字相机(dc)[]数字摄像机(dv)[]u-disk[]机顶盒(set top box)[]高清电视(hdtv)[]游戏机、智能玩具、交换机、路由器、数控设备或仪表、汽车电子、家电控制系统、医疗仪器、航天航空设备等等都是典型的嵌入式系统。

看到了嵌入式系统的范围，你也应该知道嵌入式系统工程师是做什么的，就是开发上面的那些电子产品的工程师。

关于如何学习嵌入式？

学习嵌入式，该学习什么基本的知识呢？

首先c语言，这个是毋庸置疑的，不管是做嵌入式软件还是硬件开发的人员，对c语言的掌握这个是必需的，特别是对于以后致力于嵌入式软件开发的人，现在绝大部分都是用c语言，你说不掌握它可以吗？至于如何学习c语言，我想这些基础的知识每个人都有自己的方法，关键要去学习，看书也好，网上找些视频看也好。很多人会问[]c语言要学到怎么样，我觉得这没有标准的答案。我想至少你在明白了一些基础的概念后，就该写代码了，动手才是最重要的，当你动手了，遇到

问题了，再反过来学习，反过来查查课本，那时的收获就不是你死看书能得到的。

其次，应该对操作系统有所了解，这对你对硬件和软件的理解，绝对有很大的帮助。应该把系统的管理理解一下，比如进程、线程，系统如何来分配资源的，系统如何来管理硬件的，当然，不是看书就能把这些理解透，如果不是一时能理解，没关系，多看看，结合以后的项目经验，会有更好的理解的。

还有应该学习嵌入式系统，如linux或者wince下的编程，这些对以后做应用的编程很有帮助，当然，如果做手机的话，那可以学习mtk[]塞班[]android等操作系统[]android是以后发展的趋势，现在很热门[]android也是基于linux系统封装的，所以建议先学习下linux[]

还有，应该学习下单片机或者arm或者mips[]很多人说我没有单片机的经验，直接学arm可以吗?我觉得那完全没有问题的，当然如果你学习过单片机，那最好不过了，以后学习arm就更简单了。

最后如果你把以上的知识都有所了解后，就该去阅读阅读一些优秀的代码，比如结合arm芯片手册学习去学习下uboot的源代码，了解下最小的系统开发，那对你整个嵌入式开发的非常有帮助的，可以的话，还可以学习下linux的源代码，当然如果你直接阅读2.6的代码，我想你会很痛苦的，可以先看看linux代码早期的版本，比如0.12的代码等等，麻雀虽小，五脏俱全，如果你全看完了，那我想你就是一名很成功的嵌入式工程师。

至于上面说的知识如何学习呢?每个人都应该去找自己最好的方法，所谓的最好的方法就是最适合自己的方法。很多人看不进书，那就去看看视频吧。视频也应该要选择，不是随便

看看，应该要有所选择，有所挑选。我也是这样一路走过来的，不过我这人比较懒，我看不进书，一看书我就想睡觉，所以这几年我都是看视频过来的，现在也算学有所成，但也不敢说是高手吧。

我在网上看到，有人把自己学习嵌入式的视频总结出来，嵌入式系列实践和视频教程，大家可以看看网址我仔细看了觉得很不错，我本来也想整理下，可是奈何我自己的电脑硬盘空间才40g□没有办法，时间也是不够，心有余而力不足，我不是在帮他做广告，虽然要花钱，我觉得算不错，至少自己不用去整理和花大量的时间去找资源，当然如果网上有的下，你网速和硬盘都够的话，我建议大家自己网上下也好。想想我自己该开始学的时候，那时视频资源少之又少，我买书和买视频至少花了几千块，但是我觉得那很值，至少我现在赚回来了很多知识。

最后，希望这篇文章对热爱嵌入式，想致力于嵌入式开发的朋友有所帮助。送给大家一句简单而又朴素的话，坚持就是胜利，贵在坚持，也可以看看如何坚持嵌入式学习！

首先我声明，我是基于嵌入式系统平台级设计的，硬件这个方向我相对来讲比较有发言权，如果是其它方面所要具备的基本技能还要和我们培训中心其它专业级讲师沟通。他们的方面上我只能说是知道些，但不是太多，初级的问题也可以问我。

对于硬件来讲有几个方向，就单纯信号来分为数字和模拟，模拟比较难搞，一般需要很长的经验积累，单单一个阻值或容值的精度不够就可能使信号偏差很大。因此年轻人搞的较少，随着技术的发展，出现了模拟电路数字化，比如手机的modem射频模块，都采用成熟的套片，而当年国际上只有两家公司有此技术，自我感觉模拟功能不太强的人，不太适合搞这个，如果真能搞定到手机的射频模块，只要达到一般程度可能月薪都在15k以上。

另一类就是数字部分了，在大方向上又可分为51/arm的单片机类、dsp类，fpga类，国内fpga的工程师大多是在ic设计公司从事ip核的前端验证，这部分不搞到门级，前途不太明朗，即使做个ic前端验证工程师，也要搞上几年才能胜任。dsp硬件接口比较定型，如果不向驱动或是算法上靠拢，前途也不会太大。而arm单片机类的内容就较多，业界产品占用量大，应用人群广，因此就业空间极大，而硬件设计最体现水平和水准的就是接口设计这块，这是各个高级硬件工程师相互pk、判定水平高低的依据。而接口设计这块最关键的是看时序，而不是简单的连接，比如pxa255处理器i2c要求速度在100kbps。如果把一个i2c外围器件，最高还达不到100kbps的与它相接，必然会导致设计的失败。这样的情况有很多，比如51单片机可以在总线接lcd，但为什么这种lcd就不能挂在arm的总线上，还有arm7总线上可以外接个winbond的sd卡控制器，但为什么这种控制器接不到arm9或是xscale处理器上，这些都是问题。因此接口并不是一种简单的连接，要看时序，要看参数。一个优秀的硬件工程师应该能够在没有参考方案的前提下设计出一个在成本和性能上更加优秀的产品，靠现有的方案，也要进行适当的可行性裁剪，但不是胡乱的来，我遇到一个工程师把方案中的5v变1.8v的dc芯片，直接更换成ldo，有时就会把cpu烧上几个。前几天还有人希望我帮忙把他们以前基于pxa255平台的手持gps设备做下程序优化，我问了一下情况，地图是存在sd卡中的，而sd卡与pxa255的mmc控制器间采用的spi接口，因此导致地图读取速度十分的慢，这种情况是设计中严重的缺陷，而不是程序的问题，因此我提了几条建议，让他们更新试下再说。因此想成为一个优秀的工程师，需要对系统整体性的把握和对已有电路的理解，换句话说，给你一套电路图你终究能看明白多少，看不明白80%以上的话，说明你离优秀的工程师还差得远哪。其次是电路的调试能力和审图能力，但最最基本的能力还是原理图设计、pcb绘制，逻辑设计这块。这是指的硬件设计工程师，从上面的硬件设计工程师中还可以分出ecad工

程师，就是专业的画pcb板的工程师，和emc设计工程师，帮人家解决emc的问题。硬件工程师再往上就是板级测试工程师，就是c语功底很好的硬件工程师，在电路板调试过程中能通过自己编写的测试程序对硬件功能进行验证。然后再交给基于操作系统级的驱动开发人员。

总之，硬件的内容很多很杂，硬件那方面练成了都会成为一个高手，我时常会给人家做下方案评估，很多高级硬件工程师设计的东西，经常被我一句话否定，因此工程师做到我这种地步，也会得罪些人，但硬件的确会有很多不为人知的东西，让很多高级硬件工程师也摸不到头脑。

那么高级硬件工程师技术技能都要具备那些东西哪，首先要掌握eda设计的辅助工具类

如protel\orcad\powperpcb\maplux2\ise\vdhl语言，要能用到这些工具画图画板做逻辑设计，再有就是接口设计审图能力，再者就是调试能力，如果能走到总体方案设计这块，那就基本上快成为资深工程师了。

硬件是要靠经验，也要靠积累的，十年磨一剑，百年磨一针。

把一个月前想写的东西，今天终于用一上午的进间整理完了，希望对喜爱嵌入式系统开发的工程师和学生们有所帮助。

共2页，当前第1页12

嵌入式总结体会篇四

先不说这些，在18年时妻子做完了手术，手术居然很成功。休息完之后我们19年初开始备孕了，这次真的是上天保佑，运气不错，很快就怀上了。这段时间，我虽然每天做地铁小时到公司上班，经受高强度的工作，我心里每天还是乐滋滋的。

但是，突然有一天（pl华为小组长）根我说（lm 需要派人去杭研所支持工作，我是最合适人选，让我有个心里准备。当时我是不想去的，这个时候妻子是最需要关怀的时候，我向lm 表达了我的意愿，并且我也知道如果去了杭州年底绩效考评肯定不高。

过程不多说了，反正结果是我去了杭州。

同时利用华为的现金流在绍、甬不限购地区购买一些房子，我没有炒房的想法，只是防止被通货膨胀侵蚀而已，不过后来结果证明我貌似又蒙对了啊，我自己的看法是：

杭绍甬在经济层面会连成紧密的一片，在行政区上杭州兼并绍 部分区域的概率其实不大，行政区的扩展应该是先兼并自身的下级代管城市。

嵌入式总结体会篇五

本人大学学的是电子信息科学与技术专业，其实这个专业和嵌入式没有多大的关系，要说有关系就是简单的讲了下c语言语法，学完之后还不知道什么是c语言，只记住了char,int,string等占几个字节if, for, while循环什么的，感觉也就能算一些小学就会的题目，全国计算机等级二级也就糊涂的过了。我这个专业没学过c++一开始不知道什么是面向对象，什么叫面向过程，这可能让大家笑话了。不过这说的是实话，大学的时候比较喜欢动手，喜欢硬件方面的东西，也喜欢看书。在大一的时候就听学长说，单片机很难学，在没事的时候就从图书管借了本关于51的书，一开始还真是不怎么懂，然后硬着头皮去看，当我把51的结构看完了，就不像一开始那样一头雾水了，自己在纸上把51的结构图默画一遍，然后再把各个寄存器功能写一下，各个功能口的结构也画一遍，然后就不觉得51有什么难了。其实有些时候不是这个东西你不会，而是在不会的时候自己先告诉自己这个太难

了，自己先放弃了。我们学习51芯片，是为了以后的打基础，知道是怎么通过配置寄存器来实现功能。

大二下学期的时候，我们开了51的课，这个课几乎就没怎么去，因为我认为没有必要去听自己会的东西了。后来听别人说avr的资源会比51的丰富的很多，所以就想试试，看看有什么难的，就从网上买了本马潮老师《avr单片机嵌入式系统原理与应用实践》，一边看书，一边想自己画一个板了，一开始想实现一个点阵屏，实现滚动效果，后来想想做个板也花不少钱呢，为什么不多加一些功能呢，还能多学到些知识，所以就加了温感，红外，铁电存储器等并把所有的扩展口都扩展了出来，备以后用，那时还挺自豪的呢，同学们都很羡慕，不过现在想想其实也没什么了。

大学也听说过嵌入式这个词，不过不太明白是什么意思。20xx年的时候从网上找到中嵌视频，有四张dvd(大约13g)下载下来后，从头到尾的看了一遍，没什么印象，只记得讲一些不知所云的函数，然后莫名其妙的讲如何使用那个44b0的开发板，不过听完了，还是有很大收获的，就有一种想装一个linux的冲动，我是大二下学期才有的自己电脑，不过电脑的配置比较低(1g内存，跑虚拟机就像个蜗牛，所以就想着怎么才能装个linux)自己用xp习惯了，如果全换了又舍不得，那时从书上知道linux的文件系统和windows不一样，只有一个根分区，所以也害怕一装linux全盘就被格式化了，不过最后还是欲望战胜了担心，买了张redhat 9的光盘，小心的装这个系统，里边的英文也不是太难，还是知道大概意思的，所以一步步的装，最后装到我的硬盘的最后空间，重新启动，竟可以选择进入哪一个系统了，从这之后，自己就不害怕怎么装系统了，应了那句言，纸上得来终觉浅，绝知此事要躬行。所以我认为，买一个开发板是完全有必要的，拿到实物操作一下就不会感觉太抽象了。后来也想学学arm但不知道arm是什么东西，所以就买了本《从51到arm》讲的是关于arm的大体结构，看完后感觉，原来arm就是一个速度更快，

接口更丰富的单片机埃 如果想做linux设备驱动的同学应该对arm硬件有一些了解。如果对驱动不感兴趣，完全没有必要去看。交叉编译器会帮我们做好这些事情，他比我们更了解arm结构。

大三下学期开始准备考研了，所以自己这些兴趣就先放下了，报的北邮的信通学院通信专业，当时就是想一定要上个好学校好专业，也没想太多，结果考了的成绩差了点，没有被录取，也不想调剂一个二流的学校，然后就参加工作了，工作中每天都用的到开发板，当我看到一个小板上还可以跑系统的时候感觉很不可思议，以为要很大工程。后来边工作边学习，每天都要看电脑看到眼发疼，晚上也都差不多十二点时才睡觉。感觉到累，不过也感觉到自己在进步。

闲话有点多了，说了很多都是自己大学那点破事。

听很多的初学者说不知道从哪里开始，学习嵌入都要学哪些东西，从网上搜了一下，一大堆的东西要学，一下就推荐十几本书，说看完这本书怎么怎么样，看完那本书怎么怎么样，一下子就给初学者给吓住了，其实完全没必要把网上的那些东西全部看一遍，那样又费时，又感觉工程太大，让初学者没有信心去把每一种都看完。网上说的些方面都要知道，不一定全部都掌握，我们可以退成求其次，每一个都知道一些，当然会的越多越好。今天学点这个明天学点那，换着样来也不会觉得烦。而且还能感觉到每天都有收获。

下面给嵌入式初学者一些自己的我个人建议。

1嵌入式涉及到很多的基础知识：

c语言，关于c语言要知道基本的语法，什么是结构体，什么是共用体，有什么区别，不用强记一些二级考的东西，因在实际中用的很少。

c++要知道什么是类，怎么定义，继承，接口，要深刻理解，当然基本语法也是必不可少的。

操作系统原理也应该知道基本的原理的，我记得我大学没这个课还是去电信学院蹭别的专业的课呢，后来学习发现很多的东西都是在这个课上学到的。什么时间片啊，任务调度啊，都是在那里知道的。

linux基本命令的使用(vi编辑器必须会用，因为在终端改文件的时候只能用这个)，我们现在很多的人都习惯用鼠标了，很少用到命令，我们学习嵌入linux可要改改这个习惯了□linux用命令操作一是效率高，二是功能非常强大，远非图形界面能比的了的，其实我们用图形界面的操作都要转化为命令传给硬件的。

linux文件系统的管理要清楚，哪个目录存放是什么东西，有什么用，文件权限管理等 shell编程也要了解一些，在这给大家推荐一个教程，周朝剑的 shell编程，也就七八讲吧，讲的很全面。初学者看前4讲就可以了。

linux设备驱动的架构要了解，初学者可以以飞凌的ok6410开发板上的led驱动分析一下，结合原理图和linux设备驱动相关书籍把每一行代码都分析明白。关于驱动给大家推荐一本书□linux社备驱动程序》中国电力出版社出版的，很经典的，初学者看可能看不太明白，不过要坚持看，最次也得把前6章看完了。

2 看书不要希望一次就会，一次就明白是什么意思，也不要死记硬背，看不明白也没关系，以后遇到了知道去哪里去找就可以了，以后遇到的多了，也就记住了，遇到的少的，忘记也就忘记了。还有，不要直接阅读linux内核源代码，看一些自己把头撞破也想不明白的事，会打击初学者的积极性，要循序渐进，今天学个做个led,明天做个串口，后天说不定就

做硬件编解码了。

3、你可以不接受培训，但是不能不买开发板。

如果你想学习嵌入式一定要买块学习板，现在的开发板很便宜了，给我们初学者减轻了很多经济负担。看再多的视频，看再多的书，也不如自己动手试一下，看着很简单的东西，做起来就可能遇到很多的问题，遇到问题找到原因，解决问题，只有这样才能学到东西，记得我第一次写led的驱动，想自己试一下，网上也有人写过相关的代码，可是自己一试，就出了好多的问题，那个led弄了我一天才把led灯点亮。所以不要眼高手低。

还有关于培训，现在的培训机构很多，讲的也不错，老师也很好，但是有一个问题，太集中了，而且学生动手的机会也不太多，记得我们公司还招了一个在北京某个培训机构培训半年的员工呢，花了一万多，来了之后也没感觉有多高，问他都学了什么，也就是我前边说的那些基础知识，很多的东西还是不知道的，照着书弄个helloworld模块驱动两天没搞定。不是培训不好，但我个人见解效果并不是很好。记住：只有自己动手做过的东西才是自己的。

错，但是不适合初学者，因为他的很多的东西是不给代码的，用过友善的板子的同学经常有这样的痛苦，想学一个东西，去找他的代码的时候，却找不到，很多友善提供的代码中很多的都是封装好的，只提供给客户一个库，初学者想要找代码的时候就不知道怎么办了。

4 不要好高骛远，要脚踏实地

5 多看代码，多写代码

见多识广，熟能生巧。多看代码，要从代码中读出代码的意思来。自己平时也要多写代码，多练习(这一点我做的不好，

写的少)。

写完已经晚上两点了，思路有些乱，有些也是个人的见解，在有些人看来可能是错误的，每个人有每个人的学习方法，但是我们的目的只有一个——希望我们学到更多的知识。