



要不要从单片机转Linux？进来看看大神怎么说



高级谢顶工程师

+ 关注他

90 人赞同了该文章

究竟要不要从单片机转Linux？创客学院带你[了解嵌入式行业发展跟现状分析](#)

从面向工资的角度来说，假设题主不升学历的话，推荐优先选择序列如下：

- 1、嵌入式（ARM+Linux）：可对接工业机器人、物联网、高端3C产品设计，但要去补课，主要是补充信号处理方面的知识，可不追求精深，但要了解。工资水平根据选择企业和行业特点决定，相比3、4要好点。
- 2、嵌入式（FPGA、CPLD等集成电路设计）：这条路基本上走信号处理的道路，对数学、信号、模电理论等有很深的功底要求，一般能顺利工作的起点多为硕士，但若有兴趣喜欢做，本科能做好的也有。理论上，薪水是第一位，但工作难度也是第一位。
- 3、单片机：相关产品举例——电饭锅、豆浆机、智能化程度不高的家用电器之类的，同质化严重，能干的人也不少，竞争相对激烈，产品利润低，导致薪水以及未来薪水涨幅都不是太好。但比PLC强一点。
- 4、PLC：悲催的PLC，最不推荐做的一个方向。搞PLC的一般都在各地的中小自动化公司或大公司的设备部，加班是常事，薪水不会太高，因为中专水平开始就可以抢你饭碗，技术附加值现在很低，跳槽也没啥好发展。

究竟是否要从单片机转嵌入式Linux？

著名嵌入式工程师火哥这样说：（来自嵌入式老鸟的职场之道）

尽管火哥目前从事Linux/Android方面的嵌入式开发工作，但是读书的时候也有5年左右单片机裸机和RTOS经验，之前面试也拿过一些做单片机stm32开发的offer,所以今天分享一些关于单片机是否要转嵌入式Linux的一些观点。

一、你真的决定要转嵌入式Linux吗？

要不要从单片机转嵌入式Linux是一个影响到职业发展的严肃决策，火哥不能帮你拿主意，只能帮你列出利弊，你需要根据自身的各种环境(比如所在城市Linux嵌入式相关岗位多吗？是否决定背井离乡去北上深发展等等)做出最有利的决策。火哥的观点并不是Linux嵌入式绝对比单片机开发职业发展好，只是根据自身的一些经历，给出一些看法。

1. 单片机开发相对于Linux嵌入式最大的劣势在于基层的打工岗位平均薪资偏低。我想这是大部分单片机工程师想转嵌入式Linux的最主要的原因。技术本身各有优劣，但是就基层打工而言，在同一座城市，就打工而言，单片机的薪资比嵌入式Linux低30%到50%。也许很多有会拿某些做单片机的高薪的老板(比如野火的火哥，周立功等)或者一些公司高管的薪资和做嵌入式Linux打工的工程师对比，证明单片机也可以拿高收入。但是他们都犯了田忌赛马的错误，拿别人的上等马和你的中等马对比，火哥认为这种田忌赛马的比较非常不切实际，基层打工的就应该和基层打工的这个水平的对比。目前火哥调查到情况来看，在深圳的基层就业市场，单片机(含RTOS)开发的起步薪资在6K-7K，一般人到15K就是高薪，很难突破。20K以上的单片机offer也有，但是很少，最多也见过25K的单片机offer,但是那都是有名企，名校或者其他业务背景换来的，一般人真拿不到。嵌入式Linux/Android这块，薪资相对高不少，起步价12K很正常，工作2年一般都能突破15K，3-5年突破20K甚至到25K也是很多的，5年以上去一线土豪大厂(华为、OPPO、VIVO等)可以到30K，如果在手机大厂做到模块owner专家，突破40K也是可以有的，但是在往上就有点难了。整体来看，同档次的基层打工者，嵌入式Linux薪资比单片机开发还是高不少，单片机15K就会遇到瓶颈，突破20K就很难了，嵌入式Linux突破20K并不是难事，到25K以上才遇到瓶颈。

2. 你所在的城市嵌入式Linux岗位多吗？我觉得这是影响你做决定的另一个大问题。我们学嵌入式Linux这门技术，绝大部分人是为了从事相关的工作，而不是陶冶情操。但是根据火哥统计来看，嵌入式Linux的普遍薪资虽然高于单片机，但是就业岗位并没有单片机那么多。深圳目测是嵌入式Linux就业机会最多的城市，但是单片机机会看起来似乎更多，在深圳，单片机和Linux岗位数量对比大概6比4。从嵌入式Linux城市的岗位数量来看，依次是深圳 > 上海 > 北京 > 成都 = 杭州 = 苏州 > 东莞(华为加持)>= 珠海 >= 南京 = 广州 = 武汉 = 西安 = 福州 = 厦门 > 佛山 = 长沙 = 天津 = 合肥 > 其它，在其它城市我就很少了解到做嵌入式Linux开发相关的岗位。所以如果你要考虑转行，首先考虑是否愿意背景离乡去以上几个城市工作。否则哪怕你学了牛逼的Linux技术，可能也会在当地找不到对口不得不继续从事单片机开发。

3. 单片机虽然待遇较低，但是就业机会更多，对大龄求职者相对友好。这点其实并不矛盾，做单片机各行各业的小公司特别多(当然像格力，美的这种传统家电大厂也有，但是华为、商汤等新兴高科技大厂很

少招单片机), 业务多种多样, 需要的人更多, 但是一般小厂可能资金并不充裕, 给不起人才高价, 加上单片机开发门槛较低(不考虑产品稳定性, EMC等有工作经验自然会知道的概念, 仅指学习使用单片机干活编程的培养门槛), 培养周期比较短(大二学生一个暑假的功夫), 所以导致中低端能干活的人还是挺多, 拉低平均薪资。但这就像富士康一样, 招人多, 但抱怨缺人, 老板就是不想把待遇提上去, 要控制成本, 这是缺人但是待遇低之间一种永远存在的矛盾。但是对大龄求职者来说, 由于单片机岗位实在是多, 很多小厂真的难以高的性价比招到优秀的人, 在小厂也没有HR插手做人力资源规划, 年龄梯队建设, 反而在年龄上卡的不严, 大龄求职者只要薪资不挑, 也能找到一份工作, 所以相对于做Linux的大厂, 对年龄问题比较宽容一些(可能由于单片机有点历史了, 大龄从业者也多)。

4. 单片机也有一些较高薪职位。在单片机工程师眼中, 20k以上就算是高薪, 以这个为标准, 火哥也见过一些较高薪的职位, 但是这里面有些特殊的要求。某家做智能门锁STM32 RTOS的公司给火哥开过20K的offer, 主要是因为公司希望招的技术人员都有985 211以上学历背景, 这样沟通起来可能观念比较一致, 所以薪资能给到20k。另外还有一家明星独角兽创业公司给过25K的单片机offer, 但是前提要求是能在Linux环境下开发单片机, 而且面试难度较高, 本来25K要价超出他们预期了, 他们不愿意给这么多, 后来火哥吊着他们一个星期, 谎称拿到华为同等价格的offer, 他们才最终答应给25K的offer。剩下火哥还了解到有年薪80w的单片机高级工程师(掌握某些行业特殊的认证标准)。但是整体来说, 单片机较高薪的offer确实不容易拿到, 而且拿到也不仅仅是因为你的单片机技术好, 解bug能力强, 而是因为你的背景, 学历, 有其它offer抬价, 掌握特殊的业务竞争力等等(关于嵌入式业务竞争力可以读读火哥上一篇关于嵌入式竞争力的文章), 这些特殊业务和文凭, 背景都不是能靠当下自己砸时间努力看书学习写代码就能轻易获取的。如果能单单靠自己砸时间努力就能获取的技术就不是啥门槛性的技术。列举了以上这几点, 要不要从单片机开发转嵌入式Linux相信你也应该可以下决定了。

二、单片机和嵌入式Linux开发到底有什么异同

1. 单片机转嵌入式Linux保留的优势主要在于熟练的C语言编程能力以及丰富的底层软件与硬件接口的调试经验。因为Linux内核本身就是C语言编写的, Linux底层应用程序大部分也是C语言的, 一般有单片机开发经验的, C语言应该不是问题, 所以在切换的过程中, 不需要考虑语言的切换, 当然如果没有系统学过数据结构(火哥认为数据结构是熟悉C语言必不可少的内容, 哪怕单片机也要懂数据结构), 可能需要补补, 另外有uCOS FreeRTOS这类的实时操作系统开发经验就更好, 至少在看大规模C语言代码的时候不害怕(其实懂uCOS内核也不代表能马上掌握Linux内核, Linux内核真的复杂太多, 设计理念也很不一样)。另外丰富的单片机底层调试与寄存器配置经验, 可能帮助你在学习Linux嵌入式开发的时候能快速定位底层的一些问题, 节约一些时间。

2. 单片机转Linux嵌入式需要熟悉一套风格迥异的开发环境。对于大部分单片机工程师而言, 都是在Windows上用IDE环境开发单片机程序为主(不排除有些公司已经使用Linux开发单片机了), Ubuntu这类Linux系统接触比较少, 所以你们要越过的第一关并不是Linux内核源码, 而是如何使用Ubuntu这种Linux系统。因为在Debian Ubuntu CentOS 等各种Linux系统做软件开发, 主要都是通过命令行操作, 而不是

鼠标界面点击。并且Linux系统里面的应用软件不像Windows里面的IDE这样给你集成好了，点击就能用，Linux里面很多Makefile之类的编译脚本以及Samba, SSH Server等各种服务，arm-linux-gcc等各种编译链接工具，就像IDE软件里面的各个组件，需要自己重新组装使用，并且用的时候可能有各种环境甚至编译问题(开源软件猛糙快就是这样)，需要自己折腾花费不少时间，所以对于Linux入门者，哪怕你有丰富的单片机C语言经验，折腾起来也挺耗时间，需要克服心理的障碍，Linux系统开发环境就像一匹不听话的烈马，驯服他才能发挥出它的价值，而Windows系统开发环境更像一匹听话的普通马。

3. 嵌入式Linux开发代码的掌控程度远低于单片机开发，单片机转Linux需要适应如何在这种低掌控度的不安全感中进行开发。做单片机开发，包括RTOS，一般代码量最多几万行，哪怕不是每行代码都读过，也基本能够精确掌控每个模块，大概哪里出了bug，作为开发者很容易定位。而做嵌入式Linux开发，单独Linux内核就有几百万甚至上千万行，还不包括各种你不熟悉的应用层的开源库，这导致大部分代码都不可能掌控。嵌入式Linux的开发模式就是在这种大部分代码都不是你开发的，你也掌控不了大部分代码的情况下开发驱动或者应用，你经常会不熟悉的东西需要各种搜索问人，这种开发模式，会把人置于一种不安之谷当中，很多你使用的函数你并不知道它的实现细节，可能工作机制都只是略懂。在这种模式下开发需要很好的搜索，沟通，团队协作能力，不能再像单片机那样一个人全局掌控，蒙着头开发了，这种不可掌控的不安全感是单片机工程师往Linux嵌入式工程师发展过程中最需要适应的。

4. 嵌入式Linux开发需要有大局观，不用迷失在细节的丛林找不到方向和出口。很多单片机工程师有个思维习惯，编程时喜欢细抠到每个寄存器的工作原理，每个功能的实现细节逻辑，每个if else都不放过。这在Linux嵌入式学习中将非常耗费时间，事倍功半的做法。注重细节固然好，但是当系统庞大到一种程度的时候，过于注重细节的人，往往很难驾驭这个系统。我看到很多初学者，每个寄存器，链接脚本的每种写法都要去纠结很久，芯片启动方式更上生搬硬套把2440的启动流程作为芯片启动的唯一真理，处处往其他芯片套。殊不知很多东西都是人为约定的一些习惯性做法，每家公司的芯片有自己的特点，流程是死的，人是活的。掌握大局观，让自己迅速熟悉整个系统的知识，很多细节上习惯性的东西自然而然就明白了，对整个系统的掌控力度就高了，遇到具体阻碍你前进的细节问题，再去尝试处理。掌控并不意味着你知道每行代码每个寄存器的意思，但是你能让整个系统按照你的想法去运行。

三、 单片机转嵌入式Linux大概需要学哪些基本的知识

吹了这么多水，到了真正上干货的时候了，从单片机转Linux嵌入式到底需要学些啥。如果没有这一段，火哥真怕你下定转行决心之后，马上拿起一本Linux内核架构与实现的书，以为像学uCOS等RTOS系统一样，都是C语言代码，然后看得云里雾里，从入门到放弃。其实火哥在刚学习Linux嵌入式的，也犯过类似的错误，因而总结了一些经验教训。

1. 你需要花一定时间熟悉如何使用Linux系统进行编程开发。相信很多童鞋都是从小使用Windows电脑长大的，在学计算机编程之前，应该没有接触过Ubuntu等Linux系统，而这个系统又不像Windows那样点点

鼠标就听话，需要各种命令行操作。而且系统里面有各种服务和应用工具需要你自己按照需求去配置。所以学习嵌入式Linux开发不是急于钻进Linux内核代码的海洋中，而是先用好Linux系统。你可以参考《鸟哥的Linux私房菜》等Linux系统如何安装使用等书籍，学会使用命令行。但是火哥认为学习要有目标性，如果对着《鸟哥的Linux私房菜》一页一页每个命令学下去，看几天就会枯草乏味了，所以火哥建议由单片机开发经验的童鞋，第一个目标就是在Linux系统上如何搭建你的单片机开发板交叉环境，编译出一个裸机的Led点灯的程序(不需要跑Linux内核),并通过开发板产商提供的工具烧写。这件事情并不难，网上已经有很多如何在Linux系统下交叉编译单片机程序的文章，尤其是stm32系列的，你可以参考别人的文章做一遍，在这过程中你将会熟悉各种常用的命令，shell, arm-linux-gcc交叉编译工具链，Makefile(可以单独参考陈皓的文章)等Linux编程环境相关的知识，从而开始进入Linux的世界。

2. 你需要知道嵌入式Linux系统是怎么跑起来的，一般由哪些组成部分。相信大部分人转行嵌入式Linux开发并不是为了在Linux系统上开发单片机程序，而是想做Linux驱动或者应用开发。在走完入门第一步之后，就不要太纠结那些编译，链接脚本那些文件格式怎么写，而是尽快把精力专注到Linux系统开发。学习嵌入式Linux系统开发，首先就得知道一个Linux系统如何跑起来，如何把一个Linux系统环境在开发板上搭起来。这牵涉到bootrom, bootloader, uboot, dts, Linux内核, cmdline, rootfs等一系列组件，以及nand启动，nor启动等各种不同的方式，知道一个Linux嵌入式系统如何工作，再进一步修改添加自己的驱动和应用。这里有很多实践性的东西，可能需要视频资料带你跑一遍，才能快速入门，火哥之前看的是韦东山老师的嵌入式Linux视频，总共有二三四期等，有免费试看，也有付费的，直接X宝联系卖家可以了解，火哥就不贴广告链接了。另外还有其它视频，火哥没有看过的，就不随便推荐了，在嵌入式Linux学习这块，理论性的知识需要慢慢看书，但是实践性的知识还是需要视频资料带你操作快速入门，看书和实践循环渐进。

3. 你可以尝试做一些简单的Linux应用和驱动开发了。走完了前面两个步骤，相信你对嵌入式Linux开发有一定专业性认识了。在嵌入式Linux学习这块，火哥更倾向于在做中学，60%实践+40%理论。因为嵌入式Linux里面很多系统环境相关的东西，并不是书本上那种严格的理论公式，通过看书很难找到规律，而且在Linux内核里面有一整套GNU(GNU是啥意思自己搜)世界的行话，那些写Linux内核，驱动书籍的技术专家不可能所有行话都给你解释，所有在没有使用和开发经验的前提下就直接打开《Unix环境高级编程》、《Linux设备驱动程序》等大牛的经典书籍，容易让你云里雾里。火哥建议是，跟着一个视频教程，比如韦东山老师的嵌入式Linux视频，从0开始写一个简单驱动和应用，让自己写的驱动和应用代码跑起来，点亮1个Led灯，先不用管Linux驱动中你调用的那些初始化，注册等框架函数如何实现。通过实践的过程，熟悉整个代码的运行环境以及Linux开发中的各种行话(系统调用，vfs等等)。这也是让你一步步取得学习正反馈，提升成就感和学习信心的方法。其实Linux驱动开发本身并不难(难度不会超过你复变函数课本上的公式)，也并不神秘，只不过它一整套行话术语，容易让初学者云里雾里，熟悉这套行话，摆脱恐惧心理，一个智商正常的普通本科生应该是能够掌握的。

4. 你需要补充一些计算机的理论知识。据火哥了解，大部分单片机工程师都是电子、通信、自动化、机械等专业出身，很少计算机专业的。这些专业的童鞋，相对会缺乏计算机专业的基本理论知识，如数据结

构，操作系统，计算机组成原理，计算机网络，算法，编译链接的基本原理，数据库等等。补充理论知识是一个长期的过程(可能需要2-3年)，并不是要等到计算机的专业理论都学完才可以去找工作，你可以一边补充基础知识一边面试一边找工作，同时检验自己的基础知识。这些基础知识不但能提高你的技术功底，也能帮你通过笔试面试，决定你能否在一线城市突破20K的薪水。在有了计算机科班的基础知识，以及有一定Linux驱动开发经验之后，Linux内核的学习是有必要的，但是入门的初学者不用急于进入Linux内核源码，内核这块东西还是挺多的挺艰深的，得花时间边看书边看代码慢慢啃，没法速成，但是Linux内核功底还是能够提高你的薪资竞争力。

四、给有工作经验的转行者的建议

1. 有工作的转行者，你的优势是有保底的工作和收入，劣势是没有足够的学习时间自由。有了当前的工作饭碗，不用急着找新工作，你可以保持一个良好的心态，不骄不躁，可以边学习边骑驴找马，找到合适的工作。当然有工作的人，时间上并不自由，如果你想学的新东西和你当前工作内容并不直接相关，火哥建议你选择一份加班少的工作，这样可以腾出下班后的业余时间来学习嵌入式Linux的新知识。有工作的人，有了一些小积蓄，缺的是时间，那么在学习上可以适当买一些性价比高的付费视频，加快入门和学习进度，节约宝贵的时间，这也是用金钱换时间。一般情况下，不需要学精通才出去找工作，学到可以self correct(这个词来源于一个TED关于如何快速自学的演讲)，能够干一些活的状态，就可以出去面试找相关的工作了，Linux嵌入式学习以通用基础知识为主，驱动里面的音视频，通信协议相关的业务上的知识可以找到相关的工作再补充。

2. 如何在只有单片机经验，没有Linux开发经验的情况下，如何通过社招嵌入式Linux开发工作？这个对于社招看中相关工作背景的求职者是个头疼的问题，火哥有以下几点建议：

首先，看看公司是否有Linux相关的部门和开发计划，可以主动尝试内部转岗。

第二，看看能不能建议公司技术负责人把单片机开发的环境迁移到Linux系统上，在Linux系统环境上开发单片机，这样至少你在工作中有使用Linux系统的机会。

第三，可以先尝试面一些在Linux环境下开发单片机的公司，面试后询问面试官是否在Linux环境下开发单片机，如果是，可以先加入用Linux开发单片机的公司继续做一段时间单片机，顺便再工作中熟悉一下Linux。

第四，尝试一下面试真正做Linux系统开发的公司一些Bootloader或者固件开发相关的岗位，因为Bootloader和存储控制器，电管管理等固件代码和单片机裸机rtos开发有很多相关性，甚至ARM SOC里面就有单片机核以及相关的固件，用于控制wifi,存储，电源休眠唤醒等相关的功能，这样能保证你能够在新工作中有一定输出贡献，而不是完全的学习者。

第四，尝试面试一些流动性高，离职率高的真正做Linux公司。这样的公司一个词形容：“缺人”。火哥之前在珠海某IC原厂工作，由于珠海相关的从业者基数少，应届毕业生很多也不愿意来小城市发展，再加上公司近年业绩不好，离职率高，导致公司缺人，并且很难招到人，所以后来放开招人标准，愿意培养只有单片机经验，没有Linux经验社招者。当然现在火哥在的深圳知名大厂，由于从业人口基数大，公司品牌

和待遇都很有竞争力，竞争较为激烈，虽然号称缺人(其实是招人太挑剔了)，但是一般不会给没有Linux经验的求职者机会。

最后，在学习Linux嵌入式的过程中，遇到的问题和写的代码，最好有技术博客和GitHub记录下来，把相关链接贴在简历上，这也是像面试官证明你真的对嵌入式Linux有一定了解，面试官看完你的博客和GitHub，才能对你的技术水平有进一步了解，决定是否给你机会。

五、给在校生转行者的建议

对在校生来说，你多的是时间上的自由，缺的是金钱以及没有一份保底的工作。

当然时间这东西就像卫生纸，看起来多，但是用着用着就没有了，所以即使有充裕的时间，也要好好规划，尽量多学一些东西。

在校生既然缺钱，火哥就不建议花好几万培训费去参加线下的嵌入式培训，实际上培训的内容都是嵌入式入门级知识，质量上可能还比不过几百块的韦东山嵌入式Linux视频。而且视频教程在时间上可能比培训更加灵活。

在校生的优势在于大家都没有行业相关经验，校招的时候，不会考察你行业相关的知识，而更加注重计算机基础知识的考察(火哥甚至是凭着操作系统，C语言，数据结构的笔试成绩，基本没有Linux开发经验的情况下拿到IC原厂Linux的offer，当然校招之后还有大半年，买了韦东山付费视频狂补相关知识)，所以在校生应该乘着宝贵的时间，把火哥上一节提到的第4点基础知识打牢固，这样工作之后就可以专注业务知识学习，更快升职加薪，不用因为基础知识不牢，回炉重造。

另外，在校生可能有各种单片机嵌入式的竞赛，在竞赛过程中，可能可以拿到很多奖，得到老师同学的夸赞。但是切记戒骄戒躁，不要因为比普通的同学多掌握2个技术就自以为了不起(你们以后面试的竞争对手根本不是这些完全没有技术同学)，不要因为几个奖状就飘了，而不能静下心来学习数据结构，操作系统原理等基础知识。如果不是在知名期刊发表高水平论文，或者国际上认可的ACM, Kaggle, ISLVR C图像竞赛中拿到那种高质量有分数和名次比赛结果的奖，其它国内竞赛的奖在面试与实际工作中并没有太多说服力，面试官反而喜欢基本功扎实，可塑性强的学生，而不是国内各种奖一堆，结果时间复杂度，链表堆栈，mmu虚拟地址空间等概念一问三不知的学生。

那么其他专业该能转嵌入式吗，如何转？

前几天，有朋友在后台问，他是一个机械专业的学生，想转到嵌入式方向，该怎么学？今天我们特意做了这样这个话题，希望能帮到那些想转嵌入式领域的朋友们！

当局者迷，旁观者清，关于转行这件事，先听听其他人的看法：

机械到嵌入式，我不觉得是转行，算是自己专业领域的拓展，毕竟嵌入式软件不是纯理论的东西，绝大部分是通过机械来实现其功能的。举一个极端的例子，你觉得设计一个机器人都需要哪些知识？仅仅的机械设计和嵌入式软件就够了吗？远远不止。学习嵌入式软件设计并不意味着放弃机械设计。能有这个想法很好，纯粹地做机械，不是太好，以后的趋势肯定是要复合型的。首先从个人发展来说，软件行业（嵌入式/通信/互联网）整体薪资待遇都比传统的机械行业要好，如果有毅力和决心，并且能主动学习，为了提高自己的生活水平，换行业未尝不可。从行业发展来说，据我所了解的，国内的高精度数控机床的研发一直是很薄弱，这方面的研究需要跨行业的人才，如果能有这样的机遇是相当不错的；另外机器人领域的研究（自动化生产）也在兴起，有跨行业的技术背景也是相当不错的。嵌入式软件开发以后会比较火，而且应用范围比较广。但不太建议完全放弃原来的工作领域，转入嵌入式。最好是能找到一些交叉点，这样你的职业生涯起伏会比较小。

我是学机械的，但同时对于电子与软件非常喜欢。我学习单片机和画电路图，所以，在这方面是有一些心得的。如果纯机械的人，想转到电子或软件，要进行的转变比较大，最好要有人带。软件要学得比较深，所以，要向这个方面发展的话，需要学习的很多很多。我想，如果自己喜欢，而且各方面的条件比较合适的话，可以向这方面发展呀。如果觉得你在机械设计这个行业的专长还没有发挥到极致，建议不要转行。如果你有能力只是感觉工作困难就不要转了。这年头做哪个行业都会遇到困难，看你在困难面前能坚持多久。综合分析下，如果你觉得自己的能力没办法再坚持下去，建议马上转，不要犹豫。当然跨行还会遇到困难。

当然，上面的建议只是参考，如果你真的想好了，要转做嵌入式工程师，下面的建议可以看看：

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，什么时间片啊，任务调度啊，都是需要知道的。

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。

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，弄了一天才把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。

4. 不要好高骛远，要脚踏实地

对于初学者，还有一点小建议，我们拿到一个开发板，很兴奋也很激动，想法自然也很多，想用开发板实现自己的这个想法，实现那个想法，然后又不知道从何下手，然后就去论坛中发一个贴，“那个什么什么怎么实现啊，哪位大侠能详细说一下啊，跪求！！”然后等着别人去回复。见没人回复就抱怨。有想法是好的，我们要实现自己的想法也是有前提的，首先，我们对这部分知识了解吗？如果不了解就去google一下（最好别用百度，不太好用），了解了一些，也就大概知道要怎么做。如果再遇到问题，再去发贴问，这样也显得咱们有水平。比如自己连串口是什么都不太清楚呢，就问串口通信怎么做啊，就是人家把

代码给咱，咱也不明白是怎么回事啊。


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。

发布于 2019-07-08 17:28

单片机 Linux

▲ 赞同 90 ▼ ● 8 条评论 ↗ 分享 ♥ 喜欢 ★ 收藏 📄 申请转载 ...

 评论千万条，友善第一条

- 8 条评论 默认 最新
-  赵智Wei

🤖想转硬件，但是门槛相当高，还是linux靠谱

2020-02-27

● 回复 👍 1
-  JUCAT

确实说中了挺多

2019-07-17

● 回复 👍 2
-  我真的不知道

详细，实用

2020-01-26

● 回复 👍 1
-  改名保平安

明灯指路

2022-07-23

● 回复 👍 赞
-  letw

求教为啥要学信号处理方面的知识

2022-07-18

● 回复 👍 赞



李洪祥

...

看了老哥的详细描述，真是🤖️救了老命

2022-02-18

🗨️ 回复 👍 赞



ffgggggggyh

...

公司面试对学历有硬要求吗？单片机自学成才

2022-01-30

🗨️ 回复 👍 赞



fairtale

...

写得真好，很全面👍

2021-11-28

🗨️ 回复 👍 赞

文章被以下专栏收录



嵌入式物联网

推荐阅读



Linux基础入门：单片机和Linux有什么不同吗

沐歌爱编程



单片机知识是Linux驱动开发的基础之一

韦东山嵌入式



是否要从单片机转嵌入式Linux?

向晨

发表于嵌入式老鸟...



是否要从单片机转为嵌入式Linux?

韦东山嵌入式



