

推荐书籍:

Keil Cx51 V7.0单..用实践

深入浅出 ARM7

ARM 嵌入式系统基础教程

嵌入式 Linux 应用开发..手册

Windows CE 嵌入式系统

Windows CE 嵌入式高级编程及其实例详解

限制字数, 没法说清楚了, 书籍见以上, 都是我看过的

ARM 入门最好的文章 (转自 21ic)

一 首先说说 ARM 的发展

可以用一片大好来形容, 翻开各个公司的网站, 招聘里面嵌入式占据了大半工程师职位。

广义的嵌入式无非几种: 传统的什么 51、AVR、PIC 称做嵌入式微控制器; ARM 是嵌入式微处理器; DSP; FPGA。

客观的讲, 工作需求量上 DSP 的需求比 ARM 要多, 而 ARM 和 FPGA 差不多。

DSP 因为数字处理与通信领域的空前发展而火爆, 小到 MP3 射象头, 大到我们军品里的控制器, 应用面很广。

FPGA 的兄弟一般做 ANSIC (特殊芯片设计, 好象是这么翻译的)。而 ARM 单纯说来并不比一个单片机强多少, 但是它的独特就在于不断下降的价格和提升的性能。这完全依靠于 ARM 公司的战略, 厉害!! 很佩服他们的战略眼光!!

值得注意的是: 在找工作中, 企业 (著名的, 小的不算) 对单纯的 ARM 硬件开发工程师并不比单片机重视, 很少有大企业的职位里写“从事过 ARM 开发优先”。写的多的是什么? “嵌入式 LINUX”到这相信大家看出来了吧, 需要的是硬件中的软件。

二 ARM 是硬件还是软件

很难说, ARM 是硬件, LINUX 是软件。ARM 的硬件多半已经模块化了, 像我这样把板子改成这样的就算动的多的了, 这同样是 ARM 公司的战略, 再次佩服。

实际中的 LINUX 的开发工作更多, 更耗时。从这方面说 ARM 应该算是软件了。

在找工作中更是这样, 举个例子, 联想里和 ARM 最接近的是“BIOS 工程师”是软件, MOTO 里接近的是嵌入式 LINUX 工程师是软件。而其他很多公司把嵌入式产品开发归为硬件。所以, 不要讨论这个, 好好玩转自己的板子才是关键。实在不爽你就把自己叫“嵌入式开发工程师”

三 要不要买开发板 买哪家

我的答案是“在你个人的学习方法”, 但是如果看家是需要看这骗笔记的水平, 个人推荐还是买现成的。

1 买

买板子可以把注意力集中在软件开发上, 软件开发 (尤其是驱动) 可以不必担心自己硬件上的问题, 我就是以便调试一边写驱动和程序, 每次写驱动前就要先确认硬件没问题。另外, 买板子更省钱和时间, 我自己做的板子, 原理图 PCB 花了 2 周以上! 制版又 15 天, 回来以后焊接 44B0 160 个脚! 那叫一个麻烦~~花了多少钱呢? 2 层板, 制版费就 300 块! 当然 我把接口都外引了, 还做了个 20X18 的 LCD 背板, 板子比较大。总体下来 元件+LCD 屏+PCB=11XX 块! 够 2410 的了。

再有就是买的资料相对来说比较全, 但是不要指望有技术支持! 都是骗人的, 卖你之后就不会理你。

2 做

自己做可以更了解底层硬件, 可以按照自己的要求加东西, 比如我就加了 GPS 模块、GPRS 模块、SD 卡模块, 扩了个 IIC 的 35 个键子的键盘、把 LCD 接口按照买的 LCD

改装了, 可以用 **FPC** 线直接连接。做的很爽的。玩一把吗。

当然, 你可以有策略的做, 比如像我一样, 把 **RAM** 和 **ROM**, 网络都保持和某现成的板子一样, 这样他们的资料你就可以拿过来直接用, 给自己留个退路。其他的如 **SD** 了什么的自己做。都达到了~~就是费钱, 费时间。

再有就是给做的朋友几点建议: 尽量拿到现成的板子, 尽量多搜集其他板子的全套资料, 一定要拿到一张没问题的原理图。

网上流传的原理图多数是龚俊 03 年画的, 再这里对龚俊表达一下我的敬意!! 牛人! 但是那个图有个小 **BUG**, 我指的是 03 版的, 后来的没这问题了。8019 那地址线和地址有问题。还有人仿照他的 **PDF** 图画的 **SCH**, 更是漏洞百出! 谴责! 顺便谴责把龚俊板子偷卖的人。

3 买哪家

个人感觉分 3 类吧

1) 首先是 **ZLG** 的, 资料非常的全, 感觉他是真正想教你怎么开发 **ARM**, 而不是像有的公司自己技术都没做好就做个板子出来卖钱。但是最大的不利就是价格太贵! 而且主要是 **P HILIP** 的, 货源比较麻烦~~可能有人说 **21XX** 系列的不贵啊, 那是总线不外扩的, 只能跑 **U COS**, 不能跑 **UCLINUX**。但是说是话, **21XX** 系列才是 **ARM7** 的价格性能结合点。**ARM7** 最适合做工业控制, **ARM** 普及, 销量都是怎么来的? 都是 **ARM7** 来的, 而 **44B0** 是典型的商业片子。但是, 这里如果你看中的是为工作做准备, 还是选能跑 **UCLINUX** 的吧。但是仍然作为第一个推荐, 因为菜鸟时期, 合适的资料太重要了!! 在这里被 **ZLG** 的务实精神感动! 你看人家那代码写的。

2) 感觉立宇泰的 **44B0** 不错

硬件没别的, 就是资料比较全的说, 不像有些家, 原理图直接拿人家的, 还错的~~

3) 找个最便宜的

好象最便宜的有卖 **350** 的吧? 也是没别的, 就是即省了钱 还省时间搜集资料, 至于资料不全, 别计较了~~硬件肯定好使就行吧。

四 要不要有 51 AVR 等单片机基础

有更好，但没有也无所谓。

两个月以前，我只是看别人做，耳濡目染~~，本科学过单片机，从来没做过。我们这的技术主干做 AVR 和 51，我就跟他们调过 C 语言程序。你看出来了？我是个不折不扣的菜鸟吧？

但是做这个之前我特意找了 ZLG 的两本书，看了里面的例子和原理图，这很重要。例程里有汇编有 C，都看懂了就 OK，不用自己现写

五 开发都需要学习哪些软件

总结起来最主要的有以下几个吧

1 ADS 调试用

确切的说是 ADS+AXD。ADS 里包含 AXD。原来都用 SDT 后来 ARM 公司停止对 SDT 支持了，改支持 ADS 了，还是用 ADS 吧。

有的人的程序发布的仍然是 SDT 版本的，但基本都可以找到相应 ADS 的，新人在这里不要发蒙。ADS 是编译器，AXD 是调试器。便宜成 AXF 以后再在 ARM 的 RAM 里调试。

2 PLASHPGM

FLASH 烧写的软件。AXD 在 RAM 里调试，掉电就没有了，方便程序修改。调试好的程序再下到 FLASH 里，上电直接运行。

同类的软件还有很多，什么 FLUTED 了、FLSHP 了都是，但 PLASHPGM 最好，要是有人还问 FLASH 不支持 BIN 格式文件的问题就要看我写的 PLASHPGM 使用了。

3 BANYANT 调试代理（不知道名对不，起这么个难记的，我一般都叫它“半羊”因为知道它那几天刚吃了烤羊）

调试代理就是用它帮你使用更简单的 JTAG（便宜啊）来实现原本 1K 才卖的 JTAG 仿真器

的大部分功能。JTAG 调试原理看我另一篇笔记。简单的就可以把他理解为你自己做的 JTAG 的驱动就行了。

调试代理还有很多种, 什么 H-JTAG 了、ARM7 了 (不知道具体叫什么, 就记得可执行文件叫 ARM7.EXE) 都是, BANYANT 比较好。

需要注意的是, 没种调试代理安装方法虽然都简单 但都不一样, 需要看说明。而且 AXD 调试之前都要运行。省钱了, 就别怕麻烦了。

4 ARM-ELF-TOOLS 工具链

里面是 UCLINUX 开发用的工具比如 ARM-ELF-GCC 之类的。工具链就是把很多工具打包在一起发布的方便你开发的工具。具体安装方法看我另一篇笔记。

另外如果你开发 LINUX 就要用 ARM-LINUX-TOOLS, 不一样, 不通用。

5 U-BOOT

大名鼎鼎的 BOOTLOADER 生成工具, 同类的好象还有 VIVI (名字很暧昧~~)

生成的 BOOTLOADER 烧到 FLASH 里, 然后就可以用 BOOTLOADER 下载 烧写其他了。有了 BOOTLOADER 才能下 UCLINUX。BOOTLOADER 就像电脑上的 BIOS。当然 UCOS 的不用这个, 用什么我不知道:)

最新版本是 1.1.4 具体使用方法看我另一篇笔记吧。

6 UCLINUX 包

UCLINUX 的源码包, 不用多说了吧? 建议大家用现成的先体会一下, 然后再自己编译, 裁剪。因为单独 UCLINUX 的编辑技术上比较简单, 但涉及的方面还是比较广的。

7 VMWARE

老牌的虚拟机软件, 在一个机器上虚拟出一个机器装 LINUX (PC 上用的), 省得你来回开

关机了。记得装 VMWARE-TOOLS, 安装方法在我另一篇笔记里。

六 有哪些书推荐

主要推荐 ZLG 的三本书, 有这 3 本再加网络就不用别的书了, 其中主要推荐前 2 本, 3 是介绍体系结构的, 也不错

1 ARM 嵌入式 LINUX 系统构建与驱动开发, 北航版

驱动写的很详细, 前期开发介绍也很好

2 ARM 嵌入式系统实验教程(二), 北航版

有 ZLG2200 的原理图, 实验程序和注释, 了解人家是怎么开发的。

3 ARM 嵌入式系统基础教程

最重要的是体系结构, 汇编部分介绍, 看看吧, 增加理论素质。

4 44B0 数据手册

写在这里是强调它的重要。

不推荐 ZLG 早期出的红色皮的 ARM 什么体系结构~只类的

写的太多, 看烦了。

其他的书没了几本, 没看见好的, 大家有看过的推荐吧。

七 选 UCOS? UCLINUX? LINUX? VXWORKS? 还是当单片机用

1 搞开发, 工程设计, 用 UCOS, 小巧, 多进程, 简单, 体现了 ARM7 的精髓。

2 LINUX 正根的嵌入式系统, LINUX 消费与通信领域用的比较多, 但缺点是必须 ARM 9 才能跑, ARM9 的板子自己做就不行了, 6 层板太贵。买开发板 1K 达底吧。但是还是推

荐选 ARM9+LINUX

3 UCLINUX 算投机嵌入式系统:) ARM7 上可以跑, 由有 LINUX 近亲, 学好好可以比较方便的转向 LINUX, 像我一样的穷鬼用吧, 呵呵

4 VXWORKS

学好后可以找到高薪工作, 但工作机会本身并不比 LINUX 好找。

5 当单片机用

不推荐, 虽然我景仰的 ZLG 一直号召这么干。因为对于学习来说多进程的系统设计才是 ARM7 的玩头, 至少你也要整个 UCOS 啊:)

有朋友问我先学这个当基础不行吗? 那当然可以~~但我当时就用了 1 周搞定, 就是我先的 ADS 在 RAM 中调试的笔记。后来我清空 FLASH 用的小段程序还是当时写的 LED 闪烁程序呢。另外 1 周里其中还因为自己过于菜, 没把 OM 设置好导致晶阵不阵哈哈。这个是必须做的, 但一带而过。目标放的远一点。

八 学习要用多长时间

我从菜鸟到现在两个多月, 中间被打扰数次。仍然搞定了 PCB 设计与制版, RAM 调试, FLASH 下载, UBOOT 移植和下载 UCLINUX (没剪裁, 用现成的), LCD 驱动。

再次坦诚的说: 我确实比较菜, 说这些不是让大家羡慕, 只是告诉你们“你们应该比这个更好”

我相信良好的态度+合适的方法+及时的总结=成绩

九 ARM 都玩什么

就几点

1 硬件设计

2 系统移植

3 驱动开发

4 应用开发

没了。其中推荐把注意力放在后两个上, 3 有难度, 4 比较灵活, 最重要的是都是找工作的

砧码。

十 我用的什么板子

很多人问我这个问题，我特意写过一个的。再说一次吧。

我是自己做的，RAM、ROM、8019 等比较麻烦而当时的自己不理解的都按照龚俊的做的。

保持其他不便的情况下做了如下改动。

1 加了串口通信的 GPS 模块、 GPRS 模块

一个 UART0，一个 UART1，当与要用超级终端时用跳线切换。

2 SD 卡，和 7843 触摸屏

同时挂在 SPI 总线上，因为 SPI 只能挂一个设备，同时还做了 I/O 模拟的准备，把两个设备又通过跳线挂在 4 个 IO 上。

3 扩了个 IIC 的 35 个键子的键盘

用的 ZLG 的 7290，不占用 CPU，最大支持 64 个键子，只上了 35 个，其他留接口。

4 把 LCD 接口按照买的 LCD 改装了，可以用 FPC 线直接连接。

所以改动虽然多，但真正很移植了，初期调试有关的并没动，就是为了现在不那么菜的时候做准备：)

十一 开发流程

这里是我的开发流程

1 设计原理图（含 WIGGLER 的 JTAG 仿真器）

2 设计 PCB（含 WIGGLER 的 JTAG 仿真器）

3 制版

4 焊接电压，确定电压没问题

5 焊接 44B0+JTAG 电路+WIGGLER 的 JTAG 仿真器+周边电路，用 BANYANT+仿真器连接，BANYANT 提示错误，但可以显示 44B0 编号，好像是 0X0F0F0F0F 只类的，说明 44B0 没问题

6 焊接晶体+RAM+ROM+周边电路，用 BANYANT+仿真器连接，可以显示正确的 44B0 了

7 用 BANYANT+仿真器连接, 开 AXD, 在命令行窗口操作 RAM, 看可不可以修改, 可以的话 (用内存窗口看 RAM 地址) RAM 就没问题

可以用这个命令“setmem 0xc000000,0xffffffff,32”

我的 RAM 挂在 BANK6 上 所以地址是 0XC000000, 你要是改了就得改。

8 用我写的《自己写了个 C 工程模板又写了个使用说明》里的方法调试程序吧, 写个简单的, 我当时写个 LED 闪烁的。看好使不。

9 把程序按照《FLASH 烧写总结》里的烧到 FLASH 里, 测试一下

10 按照《UBOOT 移植操作》移植 UBOOT

11 按照《UCLINUX 下载简单说明》下个 UCLINUX

12 看看驱动开发的方法, 自己写个驱动看看

13 还有很多事东西着你玩 呵呵

十二 关于 JTAG 仿真器

JTAG 仿真器现在用的多是筒板的, 一个 244 那种, 用的没什么不好。

按照并口定义不同分几种, 建议选 WIFFLER 定义的, 因为支持的软件多。这个网上多的是, 不多说了。

值得提的是有的 JTAG 仿真器原理图上有跳线用来使能复位信号, 这个一般不跳上。就是不用复位信号, 因为 JTAG 协议里本身也可以控制 44B0 复位。

当然, 板子上的复位信号跳线也不用跳。

十三 哪个公司的 ARM

这个问题比较简单。

1 ARM7 主要就几个公司的

三星, PHILIP, ATMEL 的

ATMEL 的有比较便宜的 AT91SAM7S32 和 AT91SAM7S64 专为 8 位应用产品量身定做, 价格很便宜好象《3 刀吧。和 PHILIP 的 21XX 差不多, 资料太少, 项目中选还不错。

其他两家上面说了的不多说了吧

2 ARM9

这个玩 2410 和 2440 的多吧, 现在 2440 还比较火啊 可以考虑买个了

但是 2410 就比较便宜了, 作为学习来说反到和不错, 推荐整一个:)

其实 ARM9 用的最多的领域应该是消费电子, 比如手机 PDA, 而这上面用的多的应该是 INTEL 的和 TI 的吧。

但是 INTEL 的 TI 的入门材料少, 价格高, 自己看情况定吧。

[此贴被 dove188 在 2008-01-08 17:30 重新编辑]