

1. 三周从 STM32 入门 Linux
2. 三周掌握 Linux 驱动基本技能
3. 四周掌握输入子系统_Framebuffer_APP 基本编程
4. 入门之后

123 部分如果全天学习的话，最快的学习时间是 10 周；你可以给自己加点余量，争取 3、4 个月搞定。

1. 三周从 STM32 入门 Linux

面向对象：掌握 C 语言、STM32 单片机开发的人

目标：快速入门 Linux 驱动开发，掌握简单的 LED、按键驱动开发

时长：3 周

过程：

- ① 掌握 ubuntu 常用操作
- ② 掌握在 ubuntu 下开发单片机程序，跟 STM32 的开发有一个对比
- ③ 掌握 u-boot 的使用(无需了解代码，但是要会编译、烧写 u-boot)
- ④ 掌握 Linux 内核的配置、编译、烧写
- ⑤ 掌握文件系统的结构，会烧写我们构建好的文件系统(不要求会自己编译)
- ⑥ 编写 LED 驱动、编写按键驱动

1.1 基于 Ubuntu 开发裸机(目标①②)(学习时间：1 周)

学习新 1 期视频，下载地址：<http://wiki.100ask.net/>，存放位置为：

005_ARM裸机1期加强版(156节，23节免费，已完结)

只学习前面 9 课：

- ☐ 第001课_不要再用老方法学习单片机和ARM(3节, 免费)
- ☐ 第002课__ubuntu环境搭建和ubuntu图形界面操作(8节, 免费)
- ☐ 第003课__linux入门命令(17节)
- ☐ 第004课_vi编辑器(7节)
- ☐ 第005课_linux进阶命令(6节)
- ☐ 第006课_开发板熟悉与体验(6节, 免费)
- ☐ 第007课_裸机开发步骤和工具使用(3节, 免费))
- ☐ 第008课_第1个ARM裸板程序及引申(10节, 前三节免费)
- ☐ 第009课_gcc和arm-linux-gcc和Makefile(8节)

1.2 ARM 板系统的基本使用(目标③④⑤)(学习时间: 1 周)

学习以下衔接课程, 下载地址: <http://wiki.100ask.net/>, 存放位置为:

06_u-boot_内核_根文件系统(新1期_2期间的衔接, 29节, 已完结, 全部免费)

注意 1: 这部分视频中涉及 u-boot 的源码分析, 因为未学习第 1 期中的代码重定位等知识, 所以**不要看源码分析**, 只要学会: **编译、烧写、使用 u-boot** 就可以。

注意 2: 这部分视频中涉及内核的部分源码分析, 不要求掌握这些源码, 只要学会: 大概**了解内核启动过程**做了什么事、会**编译、烧写、使用 uImage** 就可以。

注意 3: 这部分视频中涉及根文件系统之 busybox 的编译, 照着操作就可以了。

只学习其中的 1、3、4 课(搭建开发环境时, 需要阅读 JZ2440 开发板使用手册):

- ☐ 第001课_U-boot分析与使用(6节, 免费)
 - ☐ 第001节_u-boot分析之编译体验.zip ✓
 - ☐ 第002节_u-boot分析之Makefile结构分析.zip ✗
 - ☐ 第003节_u-boot分析之源码第1阶段.zip ✗
 - ☐ 第004节_u-boot分析之源码第2阶段.zip ✗
 - ☐ 第005节_u-boot分析之u-boot命令实现.zip ✗
 - ☐ 第006节_u-boot分析_u-boot启动内核.zip ✗
- ~~☐ 第002课_从0写bootloader_更深入了解bootloader(4节, 免费)~~
- ☐ 第003课_内核配置裁剪及启动流程(4节, 免费)
 - ☐ 第001节_内核启动流程分析之编译体验.zip ✓
 - ☐ 第002节_内核启动流程分析之配置.zip ✓
 - ☐ 第003节_内核启动流程分析之Makefile.zip ✗
 - ☐ 第004节_内核启动流程分析之内核启动.zip ✗
- ☐ 第004课_根文件系统(4节, 免费)
 - ☐ 第001节_构建根文件系统之启动第1个程序.zip ✓
 - ☐ 第002节_构建根文件系统之init进程分析.zip ✓
 - ☐ 第003节_构建根文件系统之busybox.zip ✓
 - ☐ 第004节_构建根文件系统之构建根文件系统.zip ✓
- ~~☐ 第005课_字符设备驱动(11节, 免费)~~

1.3 Linux 基础驱动程序(目标⑥)(学习时间: 1 周)

学习以下衔接课程, 下载地址: <http://wiki.100ask.net/>, 存放位置为:

006_u-boot_内核_根文件系统(新1期_2期间的衔接, 29节, 已完结, 全部免费)

只学习其中的第 5 课, 基于快速入门的目的, 第 5 课不需要全部学完, 只需要学习前 5 节视频:

- ☐  第001课_U-boot分析与使用(6节, 免费)
- ☐  第002课_从0写bootloader_更深刻理解bootloader(4节, 免费)
- ☐  第003课_内核配置裁剪及启动流程(4节, 免费)
- ☐  第004课_根文件系统(4节, 免费)
- ☐  第005课_字符设备驱动(11节, 免费)
 - ☐  第001节_字符设备驱动程序之概念介绍.zip ✓
 - ☐  第002节_字符设备驱动程序之LED驱动程序_编写编译.zip ✓
 - ☐  第003节_字符设备驱动程序之LED驱动程序_测试改进.zip ✓
 - ☐  第004节_字符设备驱动程序之LED驱动程序_操作LED.zip ✓
 - ☐  第005节_字符设备驱动程序之查询方式的按键驱动程序.zip ✓
 - ☐  第006节_字符设备驱动程序之中断方式的按键驱动_Linux异常处理结构.zip
 - ☐  第007节_字符设备驱动程序之中断方式的按键驱动_Linux中断处理结构.zip
 - ☐  第008节_字符设备驱动程序之中断方式的按键驱动_编写代码.zip
 - ☐  第009节_字符设备驱动程序之poll机制.zip
 - ☐  第010节_字符设备驱动程序之异步通知.zip
 - ☐  第011节_字符设备驱动程序之同步互斥阻塞.zip

2. 三周掌握 Linux 驱动基本技能

面向对象：已经会编译第 1 个 LED 裸板程序、第 1 个 LED 驱动的人

目标：快速入门 Linux 驱动开发，掌握简单的 LED、按键驱动开发

时长：3 周

过程：

- ① 掌握裸机重定位（这有助于后续理解 u-boot）
- ② 掌握裸机中断处理流程
- ③ 掌握 Linux 异常、中断处理体系
- ④ 编写 Linux 下的按键驱动程序
- ⑤ 掌握 Linux 中驱动和应用程序的交互方式：查询、中断-唤醒、poll、异步通知

2.1 裸机核心知识(目标①②)(学习时间：1 周)

学习新 1 期视频，下载地址：<http://wiki.100ask.net/>，存放位置为：

005_ARM裸机1期加强版(156节，23节免费，已完结)

学习第 010 课到第 014 课：

 第010课_掌握ARM芯片时钟体系(2节)

 第011课_串口(UART)的使用(5节)

 第012课_内存控制器与SDRAM(4节)

 第013课_代码重定位(6节)

 第014课_异常与中断(8节)

2.2 Linux 内核中断体系及 APP 与内核交互(目标③④⑤)(学习时间: 2 周)

学习以下衔接课程, 下载地址: <http://wiki.100ask.net/> , 存放位置为:

006_u-boot_内核_根文件系统(新1期_2期间的衔接, 29节, 已完结, 全部免费)

只学习其中的第 5 课的部分内容, 学习第 005 课中的第 006 ~ 第 011 节:

- ☐  第001课_U-boot分析与使用(6节, 免费)
- ☐  第002课_从0写bootloader_更深刻理解bootloader(4节, 免费)
- ☐  第003课_内核配置裁剪及启动流程(4节, 免费)
- ☐  第004课_根文件系统(4节, 免费)
- ☐  第005课_字符设备驱动(11节, 免费)
 -  第001节_字符设备驱动程序之概念介绍.zip
 -  第002节_字符设备驱动程序之LED驱动程序_编写编译.zip
 -  第003节_字符设备驱动程序之LED驱动程序_测试改进.zip
 -  第004节_字符设备驱动程序之LED驱动程序_操作LED.zip
 -  第005节_字符设备驱动程序之查询方式的按键驱动程序.zip
 -  第006节_字符设备驱动程序之中断方式的按键驱动_Linux异常处理结构.zip
 -  第007节_字符设备驱动程序之中断方式的按键驱动_Linux中断处理结构.zip
 -  第008节_字符设备驱动程序之中断方式的按键驱动_编写代码.zip
 -  第009节_字符设备驱动程序之poll机制.zip
 -  第010节_字符设备驱动程序之异步通知.zip
 -  第011节_字符设备驱动程序之同步互斥阻塞.zip

3. 四周掌握输入子系统_Framebuffer_APP 基本编程

面向对象：具备了基本驱动开发能力并且希望驱动和应用兼顾的人

目标：掌握输入子系统驱动开发、掌握 Framebuffer 驱动开发、掌握基本 APP 编程能力

时长：4 周

过程：

- ① 学习裸机触摸屏编程（只要求理解原理，不要求能写出来）
- ② 编写 Linux 输入子系统驱动开发：按键、触摸屏
- ③ 学习裸机 LCD 编程（只要求理解原理，不要求能写出来）
- ④ 编写 Linux 下的 LCD Framebuffer 驱动
- ⑤ 练习 Linux 下 APP 编程

有同学提问：

老师我看您在这一阶段目标中非常侧重 LCD 学习，
但感觉还是有很多项目开发用不到 LCD 和触摸屏。

答：

这阶段的学习涉及触摸屏、LCD，

请注意：不是让你去关注触摸屏、LCD 的硬件操作，
而是输入子系统、LCD 的驱动跟之前写的 LED、按键驱动不一样，
它们借助于内核现有的代码来实现，
你要体会这种“基于某个子系统编写驱动”的思路。

而 APP 编程里，也不是让你去关注硬件的操作，

而是要关注很多接口函数：比如文件读写、多线程、网络，poll。

当然如果你本身对 APP 编程已经有经验了，那没必要学习本阶段的 APP 编程。

3.1 Linux 输入子系统的驱动开发(目标①②)(学习时间: 1 周)

学习新 1 期视频, 下载地址: <http://wiki.100ask.net/>, 存放位置为:

005_ARM裸机1期加强版(156节, 23节免费, 已完结)

学习第 018 课(能理解硬件操作即可, 不要求能自己写出来):

☐  第018课_ADC和触摸屏(12节)

然后学习第 2 期中第 002、003、005 课:

008_韦东山第2期驱动大全视频(81节, 6节免费, 已完结)

☐  002_输入子系统驱动(2节)

☐  003_分层分离驱动(1节)

☐  ~~004_LCD驱动(7节,免费)~~

☐  005_触摸屏驱动(3节)

3.2 LCD Framebuffer 驱动开发(目标③④)(学习时间: 1 周)

学习新 1 期视频, 下载地址: <http://wiki.100ask.net/>, 存放位置为:

005_ARM裸机1期加强版(156节, 23节免费, 已完结)

学习第 017 课, 掌握 LCD 的硬件操作原理, 不要求能写出裸机程序:

☐  第017课_LCD(11节)

然后学习第 2 期中第 004 课, 最好自己写一下驱动:

008_韦东山第2期驱动大全视频(81节, 6节免费, 已完结)

☐  004_LCD驱动(7节,免费)

3.3 Linux 下 APP 编程(目标⑤)(学习时间: 2 周)

第 3 期的第 1 个项目, 涉及大量的 Linux APP 编程。里面含有 Makefile、多线程、网络等很多知识。

对于初学者, 可以先不管 Makefile。

对于初学者, 也可以先不管过于复杂的“数码相框项目”。

① 前期准备:

第 3 期不再使用 2.6.22.6 内核, 而是使用 linux 3.4.2;

第 3 期使用工具链 arm-linux-gcc 4.3.2;

第 3 期使用新的文件系统: fs_mini_mdev_new。

请注意上述 3 点。

下载地址: <http://wiki.100ask.net/>, 存放位置为:

010_韦东山Linux_第3期视频_项目实战(适用任意Linux板, 111节, 6节免费, 已完结)

3期视频(含相应的文档与源码)

项目1_文件浏览器_数码相框(33节, 2节免费)

资料

数码相框固件.rar

下载解压“数码相框固件.rar”, 可以使用其中的:

数码相框文件系统

3.4.2内核

映象文件: u-boot.bin, uImage_digitpic, fs_mini_mdev_new_digitpic.yaffs2

工具链也在“数码相框固件.rar”里。

② 视频学习:

在 PC 上的实验都可以不做, 以前之所以录制在 PC 做实验是为了方便, 但是后来发现并不方便。特别是 64 位 ubuntu 系统不支持那些实验。

Makefile 的知识挺复杂, 可以只使用我们提供的 Makefile, 不需要深入理解。

数码相框的代码过于复杂, 理解“面向对象编程”的思想即可, 不需要去钻研里面的代码。

这部分视频的学习, 是为了让你掌握基本的 Linux APP 编程。

视频下载地址: <http://wiki.100ask.net/> , 存放位置为:

010_韦东山Linux_第3期视频_项目实战(适用任意Linux板, 111节, 6节免费, 已完结)

3期视频(含相应的文档与源码)

项目1_文件浏览器_数码相框(33节, 2节免费)

视频

- ☐  第1课第1节_数码相框之系统框架(免费).zip
- ☐  第1课第2.1节_数码相框_字符的编码方式_P.rar
- ☐  第1课第2.2节_数码相框_字符的点阵显示_P.rar
- ☐  第1课第2.3.1节_数码相框_freetype理论介绍_P.rar
- ☐  第1课第2.3.2节_数码相框_在PC上测试freetype_P.rar
- ☐  第1课第2.3.3节_数码相框_在LCD上显示一个矢量字体_P.rar
- ☐  第1课第2.3.4节_数码相框_在LCD上显示多行文字_P.rar
- ☐  第1课第3.1节_数码相框_额外项目_电子书之效果及框架_P.rar
- ☐  第1课第3.2节_数码相框_额外项目_电子书之代码阅读及编写_P.rar
- ☐  第1课第3.3节_数码相框_额外项目_电子书之指针的操作_P.rar
- ☐  第1课第3.4节_数码相框_额外项目_电子书之链表的操作_P.rar
- ☐  第1课第3.5节_数码相框_额外项目_电子书之在PC上显示_P.rar
- ☐  第1课第4节_数码相框_编写通用的Makefile_P.rar
- ☐  第1课第5.1节_数码相框_使用多种方法支持多输入_框架及轮询方式_P.rar
- ☐  第1课第5.2节_数码相框_使用多种方法支持多输入_select方式_P.rar
- ☐  第1课第5.3节_数码相框_使用多种方法支持多输入_多线程_P.rar
- ☐  第1课第6.1节_一小时学会网络编程_两个简单例子_tcp_udp(免费).zip
- ☐  第1课第6.2节_修改电子书源码支持远程打印1_P.rar
- ☐  第1课第6.2节_修改电子书源码支持远程打印2_P.rar

4. 入门之后

入门之后，请按需学习。

4.1 驱动程序的学习

4.1.1 按需学习

学习完某个硬件的裸机视频，就学习对应的 Linux 驱动视频。比如学习完裸机的 I2C 视频，就可以学习 Linux I2C 驱动视频。不需要把裸机视频全部学习了，再去开始学习驱动。我们的驱动视频非常多，以第 2 期为核心来学习。第 3 期中的摄像头、ALSA 声卡驱动特别庞大复杂，你工作中



4.1.2 检验学习成果：移植到新的内核上去

视频里大部分驱动是基于 linux 2.6.22.6 来录制的，但是所讲的知识可以在新内核上无缝使用。

你可以尝试把视频中的驱动移植到 linux 3.4.2 去，我们有示例视频：

009_UBOOT移植_LINUX移植_驱动移植(19节，全部免费，已完结)

- ☐  ~~毕业班第2课_移植UBOOT-2012.04.1(11节，免费)~~
- ☐  ~~毕业班第2课_移植3.4.2内核(4节，免费)~~
- ☐  毕业班第4课_移植2期驱动到 Linux3.4.2内核(4节，免费) 
- ☐  毕业班第1课去哪了.docx
- ☐  源码文档图片.zip

4.1.3 设备树

有一种错误的观点，说“新驱动都是用设备树来写了”。

设备树不可能用来写驱动。请想想，要操作硬件就需要去操作复杂的寄存器，如果设备树可以操作寄存器，那么它就是“驱动”，它就一样很复杂。

设备树只是用来给内核里的驱动程序，提供硬件的信息。比如 LED 驱动，在内核的驱动程序里去操作寄存器，但是操作哪一个引脚？这由设备树指定。

我们的驱动视频是基于 linux 2.6.22.6 录制的，但是那些知识在最新的内核上也完全没有变。

我们录制了设备树的视频，使用 linux 4.19 版本的内核：

018_设备树详解(29节，4节免费，已完结)

- ☐  018_设备树详解(29节，4节免费，已完结)

4.2 u-boot 的学习

u-boot 是很复杂的，它是一个裸机程序的集大成者，如果你的工作不涉及 u-boot 开发，可以**先不学习它，会使用就可以了**。

复杂在于它的启动流程，在于它支持很多的设备，比如支持网卡、SD 卡、Flash。
但是，我们要记住它的核心功能：启动内核。

所以，学习 u-boot 时，要掌握 2 点：

1. u-boot 本身怎么启动？重定位、把自己从 Flash 中读到内存里，需要理解这些知识
2. 怎么去读出 Flash 上的内核：
这会涉及 u-boot 中 Flash 的驱动，
但是我们可以只关注 u-boot 提供的命令，比如使用"nand read ..."命令来读；
而不需要关注"nand read"命令的实现
3. 怎么启动内核。

怎么学习 u-boot？

4.2.1 先学习怎么使用

学习以下课程，下载地址：<http://wiki.100ask.net/>，存放位置为：

06_u-boot_内核_根文件系统(新1期_2期间的衔接，29节，已完结，全部免费)

- ☐ 第001课_U-boot分析与使用(6节，免费)
- ☐ 第001节_u-boot分析之编译体验.zip ✓
- ☐ 第002节_u-boot分析之Makefile结构分析.zip ✗
- ☐ 第003节_u-boot分析之源码第1阶段.zip ✗
- ☐ 第004节_u-boot分析之源码第2阶段.zip ✗
- ☐ 第005节_u-boot分析之u-boot命令实现.zip ✗
- ☐ 第006节_u-boot分析_uboot启动内核.zip ✗

4.2.2 u-boot 过于复杂，我们先学习自己写一个 bootloader

4.2.2.1 先学习重定位

学习新 1 期视频，下载地址：<http://wiki.100ask.net/>，存放位置为：

005_ARM裸机1期加强版(156节，23节免费，已完结)

- ☐  第013课_代码重定位(6节)
 - ☐  第001节_段的概念_重定位的引入_P.zip
 - ☐  第002节_链接脚本的引入与简单测试_P.zip
 - ☐  第003节_链接脚本的解析_P.zip
 - ☐  第004节_拷贝代码和链接脚本的改进_P.zip
 - ☐  第005节_代码重定位与位置无关码_P.zip
 - ☐  第006节_重定位_清除BSS段的C函数实现_P.zip

4.2.2.2 再学习 Nand Flash 操作

学习新 1 期视频，下载地址：<http://wiki.100ask.net/>，存放位置为：

005_ARM裸机1期加强版(156节，23节免费，已完结)

- ☐  第016课_Nand Flash(5节)
 - ☐  第001节_NAND_FLASH操作原理_P.zip
 - ☐  第002节_NandFlash时序及初始化_P.zip
 - ☐  第003节_NandFlash的芯片id读取_P.zip
 - ☐  第004节_NAND_FLASH数据读取_P.zip
 - ☐  第005节_NandFlash的擦除与烧写_P.zip

4.2.2.3 现在，从 0 写 bootloader

学习视频，下载地址：<http://wiki.100ask.net/>，存放位置为：

006_u-boot_内核_根文件系统(新1期_2期间的衔接，29节，已完结，全部免费)

- ☐  ~~第001课_U-boot分析与使用(6节, 免费)~~
- ☐  第002课_从0写bootloader_更深刻理解bootloader(4节, 免费) 
- ☐  ~~第003课_内核配置裁剪及启动流程(4节, 免费)~~
- ☐  ~~第004课_根文件系统(4节, 免费)~~
- ☐  ~~第005课_字符设备驱动(11节, 免费)~~

4.2.3 可以研究 u-boot 1.1.6 了

学习视频, 下载地址: <http://wiki.100ask.net/>, 存放位置为:

006_u-boot_内核_根文件系统(新1期_2期间的衔接, 29节, 已完结, 全部免费)

- ☐  第001课_U-boot分析与使用(6节, 免费) 
- ☐  ~~第002课_从0写bootloader_更深刻理解bootloader(4节, 免费)~~
- ☐  ~~第003课_内核配置裁剪及启动流程(4节, 免费)~~
- ☐  ~~第004课_根文件系统(4节, 免费)~~
- ☐  ~~第005课_字符设备驱动(11节, 免费)~~

4.2.4 试试自己移植

学习视频, 下载地址: <http://wiki.100ask.net/>, 存放位置为:

009_UBOOT移植_LINUX移植_驱动移植(19节, 全部免费, 已完结)

- ☐  毕业班第2课_移植UBOOT 2012.04.1(11节, 免费) 
- ☐  ~~毕业班第3课_移植2.4.2内核(4节, 免费)~~
- ☐  ~~毕业班第4课_移植2期驱动到Linux3.4.2内核(4节, 免费)~~
- ☐  毕业班第1课去哪了.docx
- ☐  源码文档图片.zip

4.3 Linux 内核的学习

我目前的视频里, 对内核的讲解并不多。

我认为内核就是各类驱动的集合。

但是, 你可以理解一下内核的启动流程, 尝试移植一下内核。

4.3.1 理解内核启动流程

学习视频，下载地址：<http://wiki.100ask.net/>，存放位置为：

006_u-boot_内核_根文件系统(新1期_2期间的衔接，29节，已完结，全部免费)

- ☐  ~~第001课_u-boot分析与使用(6节，免费)~~
- ☐  ~~第002课_从0写bootloader_更深刻理解bootloader(4节，免费)~~
- ☐  第003课_内核配置裁剪及启动流程(4节,免费) 
- ☐  ~~第004课_根文件系统(4节,免费)~~
- ☐  ~~第005课_字符设备驱动(11节，免费)~~

4.3.2 移植新内核

学习视频，下载地址：<http://wiki.100ask.net/>，存放位置为：

009_UBOOT移植_LINUX移植_驱动移植(19节，全部免费，已完结)

- ☐  ~~毕业班第2课_移植UBOOT 2012.04.1(11节，免费)~~
- ☐  毕业班第3课_移植3.4.2内核(4节，免费) 
- ☐  ~~毕业班第4课_移植2期驱动到Linux3.4.2内核(4节，免费)~~
- ☐  毕业班第1课去哪了.docx
- ☐  源码文档图片.zip

4.4 应用开发的学习

第3期的第1个项目，建议观看其中的**全部视频**。

1. 掌握各类基础技能：多线程、网络等等
2. 掌握面向对象的编程思想
3. 不要陷于细节，比如“数码相框”中对图像着处理、坐标的变换，这些很复杂，不用管它。

“数码相框”名字比较老土了，但是里面的知识还是APP开发的基础。

在这个项目里，我们还实现了**电子书**、**文件浏览器**。文件浏览器就比较高大上了。

第3期的第2个项目中的内容非常多，有摄像头、ALSA声卡、WIFI网卡使用、3G上网卡等等，按需学习。

可以看看里面的**摄像头应用编程**，至于其中的摄像头驱动太复杂了，你用到时再看。

- ☐  010_韦东山Linux_第3期视频_项目实战(适用任意Linux板, 111节, 6节免费, 已完结)
- ☐  3期视频(含相应的文档与源码)
- ☐  项目2_摄像头_WIFI_3G_ALSA驱动_视频监控(64节,3节免费)
- ☐  第2课第1.1_11节_在LCD上显示摄像头图像1_效果_框架_准备工作_P.rar
- ☐  第2课第1.1_11节_在LCD上显示摄像头图像2_实现摄像头模块_P.rar
- ☐  第2课第1.1_11节_在LCD上显示摄像头图像3_实现转换模块_1_P.rar
- ☐  第2课第1.1_11节_在LCD上显示摄像头图像4_整合代码_P.rar
- ☐  第2课第1.1_11节_在LCD上显示摄像头图像5_调试测试_P.rar
- ☐  ~~第2课第1.1_11节_在LCD上显示摄像头图像6_在PC上显示_P.rar~~ (批电脑,不学)

第3期的第3个项目是电源管理, 理解一下概念即可, 不用看。