[Android高手必须掌握的28大项内容和10个建议](http://blog.csdn.net/linux_loajie/article/details/7661712)

分类： [Android博文](http://blog.csdn.net/linux_loajie/article/category/1071087)2012-06-14 08:57 1779人阅读 [评论](http://blog.csdn.net/linux_loajie/article/details/7661712#comments)(2) [收藏](javascript:void(0);) [举报](http://blog.csdn.net/linux_loajie/article/details/7661712#report)

[android](http://www.csdn.net/tag/android)[linux内核](http://www.csdn.net/tag/linux%e5%86%85%e6%a0%b8)[manager](http://www.csdn.net/tag/manager)[java](http://www.csdn.net/tag/java)[jni](http://www.csdn.net/tag/jni)[代码分析](http://www.csdn.net/tag/%e4%bb%a3%e7%a0%81%e5%88%86%e6%9e%90)

（一）成为Android高手必须掌握的8项基本要求

【1】 Android操作系统概述  
1. Android系统架构。  
           2. Android利用设计理念。  
           3. Android 开源知识。  
           4. Android 参考网站与权威信息。  
【2】 Android SDK及其开发环境搭建  
           1. Android SDK的版本发布。  
           2. ADT插件的安装和更新。  
           3. Android利用程序架构。  
           4. 建立第一个Android项目(HelloAndroid！)。  
           5. 通过Eclipse导入一个开源项目  
           6. 可视化的界面开发工具  
           7. 布置利用程序到Android模仿器或手机  
【3】Android源代码开发环境搭建与源码结构分析  
           1.开发系统主机环境搭建。  
          2. 安装软件包Required Packages。  
           3.Ubuntu 6.06 (Dapper)。  
          4. Ubuntu 8.04。  
          5. Ubuntu 7.10。  
           6.Install javva。  
           7.源码结构分析Android source (kernel / platform/application)。  
          8. Android build system。  
          9. Android bring up。  
 【4】 Android 利用程序设计模型  
           1. drawable。  
           2. layout 。  
           3. values（strings.xml、colors.xml、style.xml 、arrays.xml等）。  
           4. R.java分析及应用 。  
           5. AndroidManifest.xml。  
           6. Android.mk。  
           7. Activity 。  
           8. Intent。  
           9. Service 。  
           10.ContentProvider。  
【5】 Android 利用程序界面(一)  
           1. TextView的应用。  
           2. drawable的应用 。  
           3. 引用Drawable色彩常数及背风景。  
           4. CharSequence数据类型与ResourceID利用。  
           5. DisplayMetrics的应用。  
           6. Style样式的定义。  
           7. Button事件处理。  
           8. setContentView的利用。  
           9. Intent对象的应用。  
           10. Bundle对象的实现。  
           11. startActivityForResult法子。  
           12. AlertDialog窗口。  
           13. Button与TextView的交互。  
           14. Typeface对象应用。  
           15. Gallery对象。  
           16. 多按钮利用。  
           17. Menu功效菜单程序设计。  
           18. ProgressDialog与线程利用。  
           19. 动态产生按钮并最大化。  
           20. 选择功效的对话框。  
           21. Android主题(Theme)实现。  
           22. 支撑多种分手率的屏幕。  
【6】 Android利用程序界面（二）  
           1. EditText与setOnKeyListener事件。  
           2.ImageButton的焦点及事件处理。  
           3. Toast对象的应用。  
           4. CheckBox的isChecked属性。  
           5. 多选项CheckBox的利用。  
           6. RadioGroup组与onCheckedChanged事件。  
           7. ImageView的堆栈利用。  
           8. Spinner与setDropDownViewResource。  
           9. ArrayList与Widget的依附性。  
           10. Gallery与衍生BaseAdapter容器。  
           11. JavaI/O的利用。  
           12.ImageButton选择特效。  
           13. AutoCompleteTextView与数组。  
           14. AnalogClock与DigitalClock的原理。  
           15. DatePicker与TimePicker利用。  
           16. ImageView点击事件与透明度处理。  
           17. ProgressBar与Handler的整合利用。  
           18. GridView与ArrayAdapter设计。  
           19. ListView的布局。  
           20. ListActivity与Menu整合技术。  
           21. JavaI/O与ListActivity的联合。  
           22. 应用decodeFile法子。  
           23. 运用Matrix对象来缩放图文件。  
           24. Bitmap与Matrix旋转ImageView。  
           25. RadioButtonID。  
           26. 对话窗口上的ICON图标。  
【7】Android 利用交互与手机模块把持  
           1. PendingIntent与AlarmManager。  
           2. PendingIntent 与 SmsManager 。  
           3. Intent 与 Email。  
           4. 数据存储法子。  
           5. NotificationManager与Notification。  
           6. Adapter对象与Cursor。  
           7. TelephonyManager的利用。  
           8. RunningTaskInfo的利用。  
           9. Provider.Contact的应用。  
           10.Provider.Calendar的应用。  
           11. Service与Runnable整合并用。  
           12.BroadcastReceiver与Intent 。  
           13. receiver与intent-filter。  
           14. Android网络编程。  
           15. HTTP恳求与连接。  
           16. WebView.loadUrl。  
           17. Webkit 与 WebCore。  
           18. HttpURLConnection与URLConnection和运行线程。  
           19. Runnable混搭SurfaceView。  
           20. Google ApI。  
【8】 Android 系统服务  
           1. Activity Manager。  
           2. Windows Manager。  
           3. Surface Manager(surface flinger)。  
           4. Package Manager。  
           5. Telephony Manager。  
           6. Content Provider。  
           7.Resource Manager。  
           8. AssetManager。  
           9. View System。  
           10.Location Manager。  
           11. Notification Manager。  
           12.AlarmManager。  
           13. BatteryManager。  
           14. HardwareManage。  
           15. 如何添加一个新系统Service并开放API。  
           16. 要害服务。  
           17. Android系统启动流程。

（二）成为Android高手必须掌握的20项进阶要求

ARM平台 Android 移植与驱动核心开发  
       Android嵌入式智能操作系统是基于Linux内核和驱动的，对于HTC、华为等公司开发Android操作系统时，需要专门将Android移植到特定硬件平台下，同时将必要的驱动进行编写及开发。本课程旨在让学员成为Android的核心层开发者，让学员在这种嵌入式设备厂商中谋得核心开发人员的职位。  
       1. Android系统体系及移植相关工具  
           1.1Android操作系统体系结构介绍。  
               Android Linux内核移植介绍。  
               文件系统,toolbox等移植。  
              存储驱动移植。  
             Bootloader相应方案与移植。  
               Android虚拟器Dalvik移植介绍。  
           1.2ARM处理器介绍。  
               ARMv5TJS。  
               ARM中断处理。  
               ARM体系结构与ARM处理器的体系结构。  
               IP核、芯片与开发板。  
               ARM处理器结构处理器模式。  
                  . 寄放器。  
                  . 流水线结构。  
               C语言内嵌汇编款式与编程实例。  
           1.3ARM指令系统。  
                .算术逻辑指令。  
                .指令对状态寄放器的影响。  
                .对比指令。  
                .分支指令。  
                .移位。  
                .指令位图。  
                .辨认机器指令。  
           1.4内存造访与内存把持。  
               .数据总线与地址总线的应用与实例。  
               .STR/LDR 。  
           1.5ARM历程调用标准。  
               .LDM/STM。  
               .反汇编。  
               .APCS。  
               .C语言函数与汇编的互相调用。  
           【案例】应用Android Linux patch进行Linux内核的构建  
       2. AndroidLinux内核移植与核心开发  
           2.1ARM处理器的Android系统构建。  
           2.2输入输出设备驱动移植。  
           2.3存储设备nand flash，MMC/SD卡驱动移植。  
           2.4Framebuffer移植。  
           2.5 Touchscreen移植 。  
           2.6 Open Binder IPC移植。  
       【案例】  
               2.7在ARM处理器上移植Android实验 。  
            2.8编写Nand flash驱动。  
             2.9移植Framebuffer驱动。  
             2.10硬件相关部分（键盘、触摸屏、LCD等）。  
       3. Android相关环境库、以及Bionic系统移植  
           3.1libc移植。  
           3.2libm移植。  
           3.3libdl移植。  
           3.4libm移植。  
           3.5 busybox, toolbox移植。  
       【案例】  
               1.移植glibc。  
              2.移植busybox和toolbox。  
       4. Android Dalvik移植一  
         4.1 JNI Call Bridge相关原理 。  
           4.2 dx, dalvitvm原理和应用。  
          4.3 Dalvik指令款式。  
           4.4 Dalvik履行款式dex 。  
           4.5 Java字节码款式。  
       5. Android Dalvik移植二  
           5.1 Dalvik Library原理 。  
           5.2 dalvit libcore核心库移植。  
           5.3 dalvit/vm/native移植。  
           5.4 Dalvik Interpreter原理 。  
           5.5 Dalvik Mterp。  
       6. ARMAT&T汇编  
           6.1AT&T汇编语法详解。  
           6.2gas工具的应用。  
           6.3label标签的作用。  
           6.4数据与代码的散播。  
       7. 链接脚本ld-script  
          7.1 段地址的断定。  
          7.2 编写ld-script。  
          7.3 ELF款式与libc函数库的关系。  
       8. 时序电路根基  
           8.1电平信号。  
          8.2 电路图分析。  
           8.3时序图分析。  
           8.4datasheet查看。  
           8.5s3c2440的外围器件结构。  
           8.6GPIO操作。  
       9. AMBA总线标准  
          9.1 AMBA总线标准 。  
          9.2 FCLK/HCLK/PCKL频率分配与设置。  
          9.3 Watchdog计时操作。  
          9.4 UART操作。  
          9.5 ARM中断体系。  
          9.6 异常向量表。  
          9.7 ARM模式切换。  
           9.8 ARM中断把持器。  
           9.9 软中断。  
           9.10 Linux系统调用的实现。  
       10. Nand Flash把持器  
           10.1 Flash原理。  
           10.2 Nand把持器原理及时序分板。  
           10.3k9f1208时序分析。  
       11. 内存把持器  
            11.1 s3c2440物理地址散播 。  
            11.2 SDRAM原理与时序。  
            11.3 内存把持器的物理连接。  
            11.4 地址对齐。  
            11.5 SRAM造访。  
       12. 内存管理单元  
            12.1 MMU的原理。  
             12.2ARM协处理器指令。  
             12.3 section模式。  
             12.4 small page模式 。  
             12.5 I & D TLBs 。  
             12.6 I & DCache。  
             12.7 哈佛结构的原理与实现。  
       13. DMA 直接内存造访的机制与应用  
           13.1 DMA的状态机 。  
            13.2 DMA的把持。  
       14. 触摸屏接口与ADC接口 与实例  
            14.1 ADC原理。  
            14.1 触摸屏接口与把持。  
       15. LCD把持器  
             15.1 图形显示原理。  
             15.2 LCD把持器。  
             15.3 LCD时序。  
             15.4 LCD驱动相关的框架与体系 。  
        16. I2S音频总线  
              16.1音频硬件原理。  
              16.2 I2S时序。  
              16.3 s3c2440上I2S的操作。  
              16.4音频设备的裸驱动讨论。  
       17. CS8900网卡把持  
              17.1 逻辑链路层与MAC层。  
              17.2 CS8900原理与时序分析。  
              17.3后续网卡裸驱动讨论与体系结构。  
       18. TCP/IP协议栈移植，与网络体系结构  
             18.1 LwIP TCP/IP协议栈简介 。  
             18.2 LwIP配置与编译。  
             18.3 tftp配置与编译。  
       19. Bootloader原理与移植  
              19.1u-boot的配置、编译。  
              19.2u-boot启动历程分析。  
              19.3ARM-Linux的启动请求。  
              19.4 ARM-Linux kernel的ld-sript分析 。  
              19.5 u-boot到ARM-Linux的跳转代码分析 。  
              19.6应用mkimage制作启动镜像文件。  
       20. ARM-Linux原理与移植  
              20.1ARM-Linux的配置、编译。  
              20.2ARM-Linux启动代码分析。  
              20.3ARM-Linux 地址映射的关系。  
              20.4 Busybox的配置、编译 。

**（三）成为Android高手的十个建议**

**1、Q：如何成为高手?**

　　A：成为一名真正的Android高手必须掌握和遵循的一些准则：

　　1)学会懒惰

　　aDon"t Reinvent the Wheel(不要重复发明轮子)。

　　bInventing the Wheel(发明轮子)。

　　cDon"t Reinvent the Wheel(不要重复发明轮子)。

　　“轮子理论”，也即“不要重复发明轮子”，这是西方国家的一句谚语，原话是：Don"t Reinvent the Wheel。“不要重复发明轮子 ”意思是企业中任何一项工作实际上都有人做过，我们所需要做的就是找到做过这件事情的人。拿到软件领域中就是指有的项目或功能，别人已经做过，我们需要用的时候，直接拿来用即可，而不要重新制造。

　　2)精通Android体系架构、MVC、常见的设计模式、控制反转(IoC)

　　3)编写可重用、可扩展、可维护、灵活性高的代码

　　4)高效的编写高效的代码

　　5)学会至少一门服务器端开发技术

**2、Q：JNI是什么?**

　　A：JNI是Java Native Interface的缩写，中文为JAVA本地调用。从Java1.1开始，Java Native Interface(JNI)标准成为java平台的一部分，它允许Java代码和其他语言写的代码进行交互。JNI一开始是为了本地已编译语言，尤其是 C和C++而设计的，但是它并不妨碍你使用其他语言，只要调用约定受支持就可以了。

　　使用java与本地已编译的代码交互，通常会丧失平台可移植性。但是，有些情况下这样做是可以接受的，甚至是必须的，比如，使用一些旧的库，与硬件、操作系统进行交互，或者为了提高程序的性能。JNI标准至少保证本地代码能工作在任何Java 虚拟机实现下。

**3、Q：Android和Linux的区别?**

　　A：Android是一个专门针对移动设备的软件集，包括一个操作系统，中间件和一些重要的应用程序。Android SDK 提供了在Android平台使用java语言进行android 应用程序开发必须的工具和API接口。Android 系统架构除了Linux2.6内核之外，还提供了丰富的lib和适用于java的运行时环境(很重要的一点是Dalvik虚拟机，类似于JVM)、应用程序框架和核心应用。在此基础之上可以快速开发应用程序。

　　linux内核像是核心技术支持，而google 将其商业化于移动设备上。android上开发应用程序和Windows MFC开发有些类似。

**4、Q：SDK升级了，如何更新SDK?**

　　A：更新SDK是非常棘手的。当一个新的SDK发布，必须是plugin也发布。更新容易出现的错误是两个版本都存在，而且都不正常。最终不得不卸载了它们并且重新安装最新的一个。然后那个最新的SDK工作正常了。建议任何面对SDK或者plugin升级的人都采用相同的过程。简单的卸载老版本，然后安装新版本。不要升级。

**5、Q：如何进入Recovery模式**

　　A：每部Android设备进入Recovery模式的方法不同。以Milestone为例：若bootloader为90.78，按住键盘的“X”键，再按电源键开机，看到moto的经典logo即可放开此2键。等待出现一个三角形图标，然后按住音量向上键+轻按拍照键，会出现四个选项。放开按键，可以用方向键选择一个选项。作用分别为重启，应用update.zip，清除所有数据至出厂状态、清除cache。

**6、Q：Android bionic移植需要注意的事项**

　　A：下面通过一个例子来说明移植binic需要做的事情(BIONIC使用的内核头文件从2.6.29 升级到 2.6.31)：

　　1)修改之前，先要备份 android/bionic

　　2)按照下面方法复制linux kernel v2.6.31 头文件:

　　复制 include/asm-generic 到 bionic/libc/kernel/original folder

　　复制 include/linux 到 bionic/libc/kernel/original folder

　　复制 include/mtd 到 bionic/libc/kernel/original folder

　　3)运行 tools/update\_all.py 脚本，生成新的头文件，方法如下.

　　$ cd android/bionic/libc/kernel

　　$ tools/update\_all.py

　　以上的操作将复制新的头文件到 bionic/libc/kernel/common 目录

　　4)删除 bionic/libc/kernel/original 目录

　　5)复制平台相关的头文件：

　　复制 linux v2.6.31 平台相关的 asm 目录到指定的路径，方法如下：

　　复制 arch/arm/include/asm 到 android/bionic/libc/kernel/arch-arm/

　　复制 arch/x86/include/asm 到 android/bionic/libc/kernel/arch-x86/

　　6)从 v2.6.29 bionic/libc/kernel/common/linux 复制一下的头文件到新的 android/bionic/libc/kernel/common/linux 目录：

　　android-alarm.h

　　android-pmem.h

　　android\_power.h

　　ashmem.h

　　binder.h

　　msm\_adsp.h

　　msm\_audio.h

　　msm\_mdp.h

　　keychord.h

　　7)编译

　　$ cd android

　　$ make -j4

**7、Q：Android源码下vendor目录下文件的作用?**

　　A：指的是某些公司的产品型号，主要是一些针对硬件配置的东西,比如radio 方面的,若是采用TI 的,这里面会有TI提供的一些RIL方面的接口等。

**8、Q：编译Android源码和编译Android内核有什么区别?**

　　A：其实没什么区别，就多了几个android特有的驱动，最重要的是binder

**9、Q：如何获取手机和存储卡上的图片?**

　　A：

　　protected String getAbsoluteImagePath(Uri uri)

　　{

　　// can post image

　　String [] proj={MediaStore.Images.Media.DATA};

　　Cursor cursor = managedQuery( uri,

　　proj, // Which columns to return

　　null, // WHERE clause; which rows to return (all rows)

　　null, // WHERE clause selection arguments (none)

　　null); // Order-by clause (ascending by name)

　　int column\_index = cursor.getColumnIndexOrThrow(MediaStore.Images.Media.DATA);

　　cursor.moveToFirst();

　　return cursor.getString(column\_index);

　　}

　　复制代码

**10、Q：如何使用双缓冲?**

　　A：系统自身已经实现了双缓冲，也就是说为了避免重影，必须连续onDraw两次，或者是对上一次onDraw里面画脏的部分进行修复。

　　执行Bitmap.createBitmap函数一般要花销100-300ms的时间，要想提高它的效率，只能祈祷google能把android的代码写得更高效一些了。

[**成为Android高手必须掌握的28大项内容和10个建议**](http://blog.sina.com.cn/s/blog_a752817501014no0.html)

[**71道经典Android面试题和答案--重要知识点都涉及到了**](http://blog.sina.com.cn/s/blog_a752817501014nng.html)

[**安卓Android和Java语言的异同、关系和区别**](http://blog.sina.com.cn/s/blog_a752817501014nlc.html)

[**高校应届毕业生“IT业”收入最高，Android技术最热门**](http://blog.sina.com.cn/s/blog_a752817501014nmv.html)

[贡献15本pdf完整版经典Android教程和**海量Android源码**](http://huangwen200301930.blog.163.com/blog/static/484849812011111141638960/)

[**给将成为“Android高手”的10个建议**](http://blog.sina.com.cn/s/blog_a752817501014nou.html)

[成为Java高手的25个学习目标--非常经典](http://blog.sina.com.cn/s/blog_a752817501014mht.html)

[**成为“Android高手”需要经过的六个阶段和6个境界**](http://blog.sina.com.cn/s/blog_a752817501014noi.html)

[**Android中设置146种颜色相对应的xml值**](http://blog.sina.com.cn/s/blog_a752817501014njh.html)

[**Android SDK 目录和作用的分析详解**](http://blog.sina.com.cn/s/blog_a752817501014nhd.html)

[**Android中拍照、图片、录音、视频和音频功能的方法和代码**](http://blog.sina.com.cn/s/blog_a752817501014nkz.html)

[老程序员的忠告：不要做浮躁的软件工程师](http://blog.sina.com.cn/s/blog_a752817501014mgn.html)

[为程序开发人员量身定制的12个目标](http://blog.sina.com.cn/s/blog_a752817501014mgl.html)

[2011年3月最新“211工程”大学名单(共112所)](http://blog.sina.com.cn/s/blog_a752817501014mdh.html)

[技术开发人员为什么会无奈苦逼？](http://blog.sina.com.cn/s/blog_a752817501014mhz.html)

[印度和中国程序员的区别](http://blog.sina.com.cn/s/blog_a752817501014mhd.html)

[中国最高等级实验室——“国家实验室”名单（仅20所）](http://blog.sina.com.cn/s/blog_a752817501014mbb.html)

[技术精英谈IT程序员的出路在哪里——很经典](http://blog.sina.com.cn/s/blog_a752817501014m82.html)

[让IT人远离慢性疲劳，长时间操作电脑需要养成的几个好习惯](http://blog.sina.com.cn/s/blog_a752817501014mhv.html)

[成为Java高手的25个学习目标--非常经典](http://blog.sina.com.cn/s/blog_a752817501014mht.html)

[周立功谈嵌入式：我的25年嵌入式生涯](http://blog.sina.com.cn/s/blog_a752817501014mho.html)

[技术开发人员应该多吃的15种健康食品](http://blog.sina.com.cn/s/blog_a752817501014mh3.html)

[IT技术开发人员获得成功的六大步骤](http://blog.sina.com.cn/s/blog_a752817501014mgy.html)

[开发人员的7大坏毛病](http://blog.sina.com.cn/s/blog_a752817501014mh0.html)http://simg.sinajs.cn/blog7style/images/common/sg_trans.gif

[IT技术开发人员35岁之前应该做的十件事](http://blog.sina.com.cn/s/blog_a752817501014mgj.html)

[2011年中国百万开发者大调查：程序员的薪水不错](http://blog.sina.com.cn/s/blog_a752817501014mge.html)http://simg.sinajs.cn/blog7style/images/common/sg_trans.gif