1. 安卓源码下载

在线阅读：<http://androidxref.com/>

清华大学镜像：<https://mirrors.tuna.tsinghua.edu.cn/help/AOSP/>

1. 源码目录结构

|  |  |
| --- | --- |
| Android源码根目录 | 描述 |
| abi | 应用程序二进制接口 |
| art | 全新的ART运行环境 |
| bionic | 系统C库 |
| bootable | 启动引导相关代码 |
| build | 存放系统编译规则及generic等基础开发包配置 |
| cts | Android兼容性测试套件标准 |
| dalvik | dalvik虚拟机 |
| developers | 开发者目录 |
| development | 应用程序开发相关 |
| device | 设备相关配置 |
| docs | 参考文档目录 |
| external | 开源模组相关文件 |
| frameworks | 应用程序框架，Android系统核心部分，由Java和C++编写 |
| hardware | 主要是硬件抽象层的代码 |
| libcore | 核心库相关文件 |
| libnativehelper | 动态库，实现JNI库的基础 |
| ndk | NDK相关代码，帮助开发人员在应用程序中嵌入C/C++代码 |
| out | 编译完成后代码输出在此目录 |
| packages | 应用程序包 |
| pdk | Plug Development Kit 的缩写，本地开发套件 |
| platform\_testing | 平台测试 |
| prebuilts | x86和arm架构下预编译的一些资源 |
| sdk | sdk和模拟器 |
| system | 底层文件系统库、应用和组件 |
| toolchain | 工具链文件 |
| tools | 工具文件 |
| Makefile | 全局Makefile文件，用来定义编译规则 |
| 表1 | |

以上系统源码分类清晰，并且内容庞大且复杂。接下来分析packages中的内容，也就是应用层部分。

1. 应用层：

应用层位于整个Android系统的最上层，开发者开发的应用程序以及系统内置的应用程序都是在应用层。源码根目录中的packages目录对应着系统应用层。它的目录结构如下所示

|  |  |
| --- | --- |
| packages目录 | 描述 |
| apps | 核心应用程序 |
| experimental | 第三方应用程序 |
| inputmethods | 输入法目录 |
| providers | 内容提供者目录 |
| screensavers | 屏幕保护 |
| services | 通信服务 |
| wallpapers | 墙纸 |
| 表2 | |

从目录结构可以发现，packages目录存放着系统核心应用程序、第三方的应用程序和输入法等等，这些应用都是运行在系统应用层的，因此packages目录对应着系统的应用层.

1. 应用框架层（FrameWork）

应用框架层是系统的核心部分，一方面向上提供接口给应用层调用，另一方面向下与C/C++程序库以及硬件抽象层等进行衔接。 应用框架层的主要实现代码在/frameworks/base和/frameworks/av目录下，其中/frameworks/base目录结构如下表所示

|  |  |  |  |
| --- | --- | --- | --- |
| /frameworks/base目录 | 描述 | /frameworks/base目录 | 描述 |
| api | 定义API | cmds | 重要命令：am、app\_proce等 |
| core | 核心库 | data | 字体和声音等数据文件 |
| docs | 文档 | graphics | 图形图像相关 |
| include | 头文件 | keystore | 和数据签名证书相关 |
| libs | 库 | location | 地理位置相关库 |
| media | 多媒体相关库 | native | 本地库 |
| nfc-extras | NFC相关 | obex | 蓝牙传输 |
| opengl | 2D/3D 图形API | packages | 设置、TTS、VPN程序 |
| sax | XML解析器 | services | 系统服务 |
| telephony | 电话通讯管理 | test-runner | 测试工具相关 |
| tests | 测试相关 | tools | 工具 |
| wifi | wifi无线网络 |  |  |
| 表3 | | | |

1. 系统运行库层（Native）
   1. C/C++程序库部分

系统运行库层（Native)中的 C/C++程序库的类型繁多，功能强大，C/C++程序库并不完全在一个目录中，这里给出几个常用且比较重要的C/C++程序库所在的目录位置

|  |  |
| --- | --- |
| 目录位置 | 描述 |
| bionic/ | Google开发的系统C库，以BSD许可形式开源 |
| /frameworks/av/media | 系统媒体库 |
| /frameworks/native/opengl | 第三方图形渲染库 |
| /frameworks/native/services/surfaceflinger | 图形显示库，主要负责图形的渲染、叠加和绘制等功能 |
| external/sqlite | 轻量型关系数据库SQLite的C++实现 |
| 表4 | |

* 1. 安卓运行时库部分

Android运行时库的代码放在art/目录中（表1）。

1. 硬件抽象层部分

硬件抽象层的代码在hardware/目录中（表1），这一部分是手机厂商改动最大的一部分，根据手机终端所采用的硬件平台会有不同的实现。