美化目录  
一，手机端美化必备软件  
  
  
二，任务栏美化  
①状态栏透明   
②下拉栏透明   
③下拉栏通知颜色及大小修改  
④图标替换  
⑤添加网速  
⑥时间居中及显秒  
⑦添加删除运营商  
⑧防魅族状态栏浅谈  
⑨状态栏图标位置修改  
⑩下拉栏动态背景  
  
  
三，手机系统全局美化  
①全局背景修改  
②全局透明  
③全局字体颜色修改  
④进度条颜色丶大小修改  
⑤手机切换特效  
⑥系统电量百分比显示  
⑦锁屏界面时间字体修改  
⑧全局字体呈斜体  
⑨去分割线  
⑩加入crt关屏特效  
  
  
四，反编译汇总  
  
  
  
五，其他美化

学前必看  
  
  
  
  
1、只修改里面图片，就不要反编它，直接用压缩工具打开替换图片  
  
  
2、修改代码时要反编framework-res.apk，但不要改图片，不然回编时很容易出错，谨记第一点  
  
  
3、修改framework-res.apk中，很多修改都改动了resources.arsc中的东西，所以要修改必须一次性修改，比如你要全局透明，又要改变状态栏图标位置，你必须同时修改好再回编，不能改好透明后再反编改其他的!  
  
  
4、改透明你可以直接改它们的颜色代码！  
  
  
5、替换系统软件时，请用个移动速度快的管理器，不然一卡你特么就玩完了！  
  
  
6、关于回编译后替换文件的，我一直没这样做，而是把原apk中的签名文件复制到回编译后的apk里，前提是你已经应用了系统核心破解！不然就要刷机阿鲁！！！谨记这一点，所以，我每次刷机后，第一件事就是系统核心破解！关于这个，去下载个幸运破解器！  
  
  
7、此教程中的代码，可能因为某些原因缺少空格！请大家自己添加，大多数代码都在每段代码中 android 前面加一个空格！  
  
  
8、玩机需谨慎，备好刷机包！

一、手机端美化必备软件  
  
  
re管理器/xplore管理器 by 替换图片软件等等。  
  
  
图片编辑器 by 制作透明，图标等等  
  
  
反编译apktool4.4 by 反编译软件  
  
  
920文本编辑器/jota文本编辑器 by 编辑xml等  
  
  
点九编辑器 by 制作 9.png  
手机端差不多就需要这些软件，管理器，文本编辑器任选一种，推荐用我说的。。  
  
  
没有re管理器，自己找去吧，系统不一样。

二—任务栏美化  
￣  
①状态栏透明  
  
  
  
  
反编译systemUi.apk  
  
  
res/layout/statu\_bar.xml  
  
  
打开它，第一段代码可能如下  
注意！是第一段代码！！！  
  
  
<?xml version="1.0" encoding="utf-8"?>  
<com.android.systemui.statusbar.StatusBarView android:orientation="vertical" android:background="#ff000000" android:focusable="true" android:descendantFocusability="afterDescendants"   
  
  
看到了没  
android:background="#ff000000"   
  
  
这个代码就是状态栏背景了哦！我把它指向黑色了！大家想要透明可以把他修改为   
android:background="#00000000"   
这样就透明了哦！  
  
  
  
  
如果透明后，会出现重叠现象的请接着看以下教程↓  
  
  
  
  
同样是要修改systemui.apk。  
反编译全部后找到smali\com\android\systemui\statusbar\StatusBarService.smali  
  
  
找到  
new-instance v0, Landroid/view/WindowManager$LayoutParams;  
const/4 v1, -0x1  
const/16 v3, 0x7d0  
const/16 v4, 0x48  
const/4 v5, 0x2  
  
  
修改为  
new-instance v0,   
Landroid/view/WindowManager$LayoutParams;  
const/4 v1, -0x1  
const/16 v3, 0x7d0  
const/16 v4, 0x48  
const/4 v5, -0x3  
（其实只改了后面的几个代码！）  
回编替换classes.dex  
因为我们直接修改的颜色代码，所以就不用换图片了，这样就可以啦！！

②下拉栏透明  
  
  
接下来就是下拉栏透明！这个下拉栏也可以替换图片来透明，systemui.apk里的drawable\_hdpi下应该只有几个前面带statu的图片吧！你把它们全部替换成透明图片。  
  
  
下拉栏可能也会有点透明，因为手机不一样所以替换背景图片之后透明的也不一样，以前我替换过背景图片来透明下拉栏，但还是有许多处不是透明的，比如快捷设置，通知，正在进行的背景都没透明！  
  
  
而现在，通过研究xml，我把这些全都给透明了，修改xml代码来透明下拉栏具体都在res/layout下，  
statu\_bar.xml,  
statu\_bar\_expanded.xml,  
statu\_bar\_tracking.xml  
安卓手机大概都有这么几个，我的手机里还有一个statu\_bar\_latest\_event,不知道大家有木有，而这个就是下拉栏中正在进行，通知的xml了。  
还有status\_bar\_easy\_setting.xml，这个就是快捷设置的xml，  
打开xml可以看到  
android:background="@drawable/easy\_setting\_loading\_bg" 这种代码吧！  
引号里的drawable/easy\_setting\_loading\_bg  
这句代码表示此背景指向drawable等目录下名称为easy\_setting\_loading\_bg的图片  
  
  
而我们想改透明，你只要把背景指向透明即可，如下  
android:background="#00000000"   
这样这个背景就是透明的了！  
但到底是哪里的背景就需要你们的火眼金睛来仔细观察这些xml下的代码了！  
如果你找不到需要透明的背景，我教你一个简单的方法，那就是可以找到res/values/string  
这个里面就是你通知栏里的字了，我复制了一段我状态栏里的，如下  
↓↓  
<string name="status\_bar\_no\_notifications\_title">无通知</string>  
<string name="status\_bar\_ongoing\_events\_title">正在进行的</string>  
<string name="status\_bar\_latest\_events\_title">通知</string>   
  
  
大家看到了么，那个汉字，比如“通知”前面有name= "status\_bar\_latest\_events\_title"那就可以找与这个与status\_bar\_latest\_events\_title相关英文字的xml进行修改。

③下拉栏通知颜色及大小修改  
  
  
  
  
首先反编译framework-res.apk  
找到values/styles.xml  
  
  
找到  
<style   
name="TextAppearance.StatusBar.EventContent.Title"   
parent="@style/TextAppearance.StatusBar.EventContent">  
  
  
修改下三行：  
<item   
name="textSize">16.0sp</item>  
  
  
<item name="textStyle">bold</item>  
  
  
<item   
name="textColor">?textColorPrimaryInverse</item>  
  
  
  
  
第一行是字体大小，数字越大字体越大  
第二行是样式，比如bold就是粗体  
第三行是颜色，把?textColorPrimaryInverse修改为颜色  
比如：#ffffffff这个是颜色代码，自己改成喜欢的颜色。

④图标替换  
  
  
这个也就是修改你状态栏上的信号，蓝牙，wifi，静音，电量，下拉栏上的图标等等了！  
  
  
替换图标就要用到xplore管理器了，当然你也可以用re管理器把它们解压，但是用x管理器替换图标图片更方便！  
用xplore管理器长按systemui.apk，以zip方式打开apk  
，你就可以看到里面的文件目录了，是不是比解压更方便呢！找到res/drawable\_hdpi（高分辨率，通常我们替换这里面的就可以）  
是不是看到许多熟悉的图片，找到你需要修改的图片，把你准备的素材图片复制或者剪切进去，然后覆盖，注意素材尺寸要和原图片大小差不多哦！  
然后长按systemui打包zip为apk，这样就ok了！  
ps：关于电量图标的，2.+替换framework-res.apk里的才可以，4.0+要替换systemui.apk的。

⑤添加网速  
  
  
在res/layout中找到  
status\_bar.xml，然后打开它  
把这代码   
<com.android.systemui.statusbar.policy.Traffic android:textAppearance="@\*android:style/TextAppearance.StatusBar.Icon" and没他:-Oroid:gravity="left|center" android:id="@id/traffic" android:layout\_width="wrap\_content" android:layout\_height="fill\_parent" android:singleLine="true" />  
添加到  
<LinearLayout android:gravity="center\_vertical" android:orientation="horizontal" android:id="@id/statusIcons" android:paddingRight="1.0dip" android:layout\_width="wrap\_content" android:layout\_height="fill\_parent" android:layout\_alignParentRight="true" />  
  
  
前面  
  
  
  
  
打开res/values/ids.xml  
  
  
修改添加  
<item type="id" name="traffic">false</item>  
  
  
打开res/values/styles.xml  
修改添加  
<style name="TextAppearance.StatusBar.Traffic" parent="@\*android:style/TextAppearance">  
<item name="android:textSize">16.0dip</item>  
<item name="android:textStyle">normal</item>  
<item name="android:textColor">#ffffffff</item>  
</style>  
  
  
最后在相应位置添加附件  
反编译，然后替换修改的文件，这样带网速的状态栏就好了替换就ok  
ps：添加代码时，注意空格哦，可能有些代码改变了位置

⑥时间居中及显秒  
  
  
时间居中  
  
  
还是SystemUI.apk找到res\layout\status\_bar.xml  
打开status\_bar.xml（内容不一定会完全一样，但方法一样，安卓通用）   
  
  
找到以下代码，把它们全部删除  
<com.android.systemui.statusbar.Clock  
android:textAppearance="@android:style/TextAppearance.StatusBar.Icon"  
android:gravity="left|center"android:paddingRight="4.0dip"android:layout\_width="wrap\_content"  
android:layout\_height="fill\_parent"  
android:singleLine="true"/>  
  
  
再次找到这一句：xmlns:android="[http://schemas.android.com/apk/res/android](http://jump.bdimg.com/safecheck/index?url=x+Z5mMbGPAsNL8eEDgn5GE69IR1W66oayijCOBiuw/KmtlHlg6hC9MTaOXkcb31xhXugxMXOJ8uk1qQkA4PbQtS5buMAyaaPl6zBuBRArcjlwsonEkMPiRhH1M/GL7XEy96wyuuynd4zR9+N34FbkHgQbCWoaNDidj2oeHoEzTI=)">  
  
  
在这句的下面添加以下代码   
<com.android.systemui.statusbar.Clock  
android:layout\_gravity="center\_horizontal"  
android:paddingTop="4.0dip"android:textAppearance="@android:style/TextAppearance.StatusBar.Icon"  
android:layout\_width="wrap\_content"  
android:layout\_height="fill\_parent"  
android:singleLine="true"/>  
  
  
最后，重编译，替换status\_bar.xml后再替换到手机，重启！ok  
再最后！如果时间没有垂直居中的话请调整这句：  
android:paddingTop="4.0dip"离顶部的距离  
  
  
时间显示到秒且居中  
找到res\layout\status\_bar.xml  
打开status\_bar.xml  
找到以下代码并将它们删除！  
<com.android.systemui.statusbar.Clock  
android:textAppearance="@android:style/TextAppearance.StatusBar.Icon"  
android:gravity="left|center"android:paddingRight="4.0dip"android:layout\_width="wrap\_content"  
android:layout\_height="fill\_parent"  
android:singleLine="true"/>  
  
  
再次找到这一句：  
xmlns:android="[http://schemas.android.com/apk/res/android](http://jump.bdimg.com/safecheck/index?url=x+Z5mMbGPAsNL8eEDgn5GE69IR1W66oayijCOBiuw/KmtlHlg6hC9MTaOXkcb31xhXugxMXOJ8uk1qQkA4PbQtS5buMAyaaPl6zBuBRArcjlwsonEkMPiRhH1M/GL7XEy96wyuuynd4zR9+N34FbkHgQbCWoaNDidj2oeHoEzTI=)">  
在这句的下面添加以下代码  
<DigitalClock  
android:textAppearance="@android:style/TextAppearance.StatusBar.Icon"  
android:layout\_gravity="cente  
r\_horizontal"android:id="@+id/digitalClock"android:paddingTop="4.0dip"android:layout\_width="wrap\_content"  
android:layout\_height="fill\_parent"/>  
  
  
最后，重编译，替换status\_bar.xml后再替换到手机，重启！ok！  
再最后！ 如果时间没有垂直居中的话请调整这句：  
android:paddingTop="4.0dip"离顶部的距离。

⑨状态栏图标位置修改  
  
  
反编译framework-res.apk，  
打开\res\values\arrays.xml文件，  
  
  
搜索"config\_statusBarIcons  
  
  
除了clock之外，其余图标位置上下互换，状态栏显示位置则左右互换。  
以信号与电池为例，array，phone\_signal在battery上面。上下位置互换后，在状态栏中显示的位置左右对调。其余同理  
回编framework-res.apk，替换resources.arsc，存储方式为存储，替换framework-res.apk，重启，OK（注意权限）  
4如果无法回编，表示你修改的位置它无法出现，多次尝试，你会找到你要的位置

---下面是手机全局美化---

③全局字体颜色修改  
  
  
关于这字体颜色修改，我也不多说什么，一些教程上什么一级字体二级字体的代码，我特么在color里都没找到！手机不同，所以color.xml里的一级二级字体代码也不一样！  
  
  
在这里说一下，我修改字体颜色的经验！  
反编译 framework-res.apk  
res/color  
目录下会有几个xml，是不是看到里面有几个primary开头的xml，这个就是全局一级字体了！  
secondary开头的xml是全局二级字体！  
  
  
我复制了我手机xml下的一小部分↓  
  
  
android:color="@color/dim\_foreground\_dark\_disabled" />  
  
  
看到没，android:color后指向了颜色，  
指向了color中dim\_foreground\_dark\_disabled的颜色  
而这个dim\_foreground\_dark\_disabled 就在res/values/color/中！  
你可以直接在res/color中改！也可以在values/color中改！  
我建议在values/color下改，因为res/color下有好几个一级二级字体xml，它们都指向values/color中的代码！  
你在values/color中改了颜色代码，它们也会变得！  
  
  
还有！在values/color中，我建议几个不改的地方如下↓  
前几句就可以看到  
color name="darker\_gray">#ffaaaaaa</color>  
<color name="white">#ffffffff</color>  
<color name="black">#ff000000</color>  
<color name="transparent">#00000000</color>   
  
  
这些都是系统指向颜色，  
  
  
dark\_gray为灰色  
white为白色  
black为黑色  
transparent为透明  
  
  
这几个你别闲的蛋疼给改了！！！！！  
  
  
路径：  
framework-res.apk/res/values/colors.xml  
  
  
找到  
  
  
<color name="bright\_foreground\_dark">#ffc8c8c8</color>  
设置界面 以及对话框顶栏字体颜色  
  
  
<color name="bright\_foreground\_dark\_disabled">#80ffffff</color>  
未启用功能字体颜色（暗色）  
  
  
<color name="dim\_foreground\_dark">#ffbebebe</color>  
关于手机里的 型号 版本号字体颜色  
  
  
<color name="dim\_foreground\_dark\_disabled">#80bebebe</color>  
关于手机里的 型号 版本号以及二级目录字体暗颜色  
  
  
<color name="hint\_foreground\_dark">#ff808080</color>  
输入框内提示文本颜色  
  
  
<color name="dim\_foreground\_light">#ff323232</color>  
关机界面 飞行模式等下的二级文本颜色  
  
  
<color name="bright\_foreground\_light">#ff000000</color>  
选择列表字体颜色  
  
  
把以上的颜色代码 修改为你喜欢的颜色 然后保存并退出 （颜色代码可在百度上搜索，文中"#xxxxxxxx" #后两位为透明度，倒数六位为颜色代码）

②全局透明  
  
  
  
  
安卓2.3  
  
  
打开res/values/styles.xml   
查找<style name="Theme">，  
该主题下找到<item name="windowBackground">，将这一行修改为<item name="windowBackground">@color/background</item>  
该主题下找到<item name="windowShowWallpaper">，将这一行修改为<item name="windowShowWallpaper">true</item>  
查找<style name="Theme.NoTitleBar" parent="@style/Theme">  
在<item name="windowNoTitle">true</item>下一行添加  
<item name="windowShowWallpaper">false</item>  
<item name="windowBackground">@drawable/holo\_dark\_background</item>   
查找<style name="Theme.Black" parent="@style/Theme">，  
修改<item name="colorBackground">这一行为<item name="colorBackground">@color/transparent</item>  
查找<style name="Theme.Dialog" parent="@style/Theme">，该主题下添加<item name="windowShowWallpaper">false</item>  
查找<style name="Theme.Light" parent="@style/Theme">，该主题下添加<item name="windowShowWallpaper">false</item>  
查找<style name="Theme.Translucent" parent="@style/Theme">，该主题下添加<item name="windowShowWallpaper">false</item>  
查找<style name="Widget.PopupWindow" parent="@style/Widget">，该主题下添加<item name="windowShowWallpaper">false</item>  
查找<style name="Widget.KeyboardView" parent="@style/Widget">，该主题下添加<item name="windowShowWallpaper">false</item>  
查找<style name="Theme.Panel" parent="@style/Theme">，该主题下添加<item name="windowShowWallpaper">false</item>  
查找<style name="Widget.QuickContactBadge" parent="@style/Widget">，该主题下添加<item name="windowShowWallpaper">false</item>  
查找<style name="Widget.QuickContactBadgeSmall" parent="@style/Widget">，该主题下添加<item name="windowShowWallpaper">false</item>  
查找<style name="Theme.ExpandedMenu" parent="@style/Theme">，该主题下添加<item name="windowShowWallpaper">false</item>  
打开res/values/colors.xml，添加<color name="background">#99000000</color>  
选择一张图片命名为holo\_dark\_background.png，放在res/drawable-nodpi下面。  
【全局背景】  
打开res/values/styles.xml  
查找<style name="Theme">，   
修改<item name="windowBackground">这一行  
  
  
为<item name="windowBackground">@drawable/poxiao\_dark\_background</item>  
  
  
选择一张图片命名为holo\_dark\_background.png，放在res/drawable-hdpi下面。  
当然，这张poxiao\_dark\_background可以随便命名。  
  
  
  
  
  
  
4.1全局透明  
  
  
4.1可以用 其他版本自己测试  
  
  
1.反编译framework-res.apk  
2.打开res/values里的styles.xml  
3.找到这个：<style name="Theme">（直接搜索就可以了）  
4.找到：<item name="colorBackground">@color/background\_dark</item>  
5.修改为：<item name="colorBackground">8@color/transparent</item>  
6.查找<item name="windowBackground">@drawable/screen\_background\_dark</item>  
7.”screen\_background\_dark“这个自己命名一下，比如：image\_background  
8.找到<item name="windowShowWallpaper">false</item>  
9.修改为<item name="windowShowWallpaper">true</item>  
10.找到<style name="Theme.NoTitleBar"parent="@style/Theme">  
11.最后添加<item name="windowShowWallpaper">false</item>  
12.找到<style name="Theme.Black"parent="@style/Theme">  
13.最后添加<item name="windowShowWallpaper">false</item>  
14.找到<item name="windowBackground">@color/black</item>  
15.修改为<item name="windowBackground">@drawable/image\_background</item>  
16.找到<style name="Theme.Dialog"parent="@style/Theme">  
17.最后添加<item name="windowShowWallpaper">false</item>  
18.找到<style name="Theme.Light"parent="@style/Theme">  
19.最后添加<item name="windowShowWallpaper">false</item>  
20.找到<style name="Theme.Translucent"parent="@style/Theme">  
21.最后添加<item name="windowShowWallpaper">false</item>  
22.找到<style name="Theme.NoDisplay"parent="@style/Theme">  
23.最后添加<item name="windowShowWallpaper">false</item>  
24.找到<style name="Theme.Panel"parent="@style/Theme">  
25.最后添加<item name="windowShowWallpaper">false</item>  
26.修改完成，保存  
27.打开/res/values/drawable.xml  
28.找到<item8 type="drawable"name="screen\_background\_dark">#ff000000</item>  
29.修改为<item type="drawable"name="screen\_background\_dark">#00000000</item>（通讯录拨号背景）  
30.把image\_background.png放到反编译的文件夹里的res\drawable-hdpi  
30.把image\_background.png放到反编译的文件夹里的res\drawable-hdpi  
31.回编译  
32.把resources.arsc替换原来的apk  
33.把image\_background.png放到apk里的res\drawable-hdpi

⑤手机切换特效【转】  
  
  
转自miui论坛！  
  
  
这些特效,例如我只要个锁屏的特效或者只要个系统内部切换的特效,都是可以的  
  
  
可以自由选择,自由搭配~~适合任何机型,只要是安卓系统!  
  
  
  
  
MIUI~V4系统需要打开如下设置才可以看到效果.  
设置--开发人员选项--窗口动画缩放X1 过度动画缩放X1  
  
  
MIUI系统的APP如果没有精简的话,应该可用的在3-4MB之间吧~  
所以未进行过精简的,建议先精简系统,删除掉不用的内置程序,扩大空间.  
  
  
  
  
1: 不要把人家发的未注明版本,机型的framework-res.APK拿来直接替换!  
你需要替换的仅仅是framework-res\res\anim 里面的脚本文件!  
不要嫌麻烦,如果人家提供的framework-res.APK和你的机子不符合就会出现各种BUG!  
  
  
我们要做的就是把有特效的脚本放进去替换掉默认的就行了.  
  
  
  
替换的时候,要先把framework-res.APK 拉到system文件夹下更改权限为1247,再拉到framework文件夹替换!  
不要想着放到framework里面后再改权限,如果你这么做,100%会无法开机滴!  
  
  
  
  
framework-res.APK在手机中的位置: system / framework / framework-res.APK   
------------------------------------------------------------------------------------  
  
  
  
  
MIUI除了framework-res.apk  
  
  
还多了一个framework-miui-res.apk 这个大家不用动它.仍然修改framework-res.apk   
  
  
大家只要用RE管理器把system/framework/framework-res.apk提取出来  
  
  
然后把下面的附件解压后,添加进framework-res.apk/res/anim中即可，  
  
  
-------------------------------------------------------------------------------------  
  
  
接下来我说说方法  
  
  
提取别人的特效文件 或者 下载我附件中的特效文件:  
  
  
1-1: 得到别人的framework-res.APK,  
解压!! 对,解压,,, APK就是个压缩文件的一种格式  
  
  
1-2:解压可以到framework-res\res文件夹 找到 anim 文件夹  
把anim文件夹,拖出来,保存起来吧~~以后也可以用  
解压出来的其他的东西都不需要了,删掉吧~~  
只要有了anim文件夹就可以了里面全部都是动态特效脚本文件  
  
  
转到下面第3步.  
  
  
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
  
  
2:或者直接下载附件中你喜欢的特效,解压后得到anim文件夹.  
  
  
3:接着,把你自己手机里的framework-res.APK 复制到SD卡,再拿到电脑上  
这次不用解压, 用压缩工具打开 是打开 不是解压!!!  
到framework-res\res\anim 可以看到全部都是脚本文件  
  
  
5:然后把特效的anim里面的脚本文件,全选,  
拖进压缩工具里显示的framework-res\res\anim 文件夹里面替换.  
  
  
6:替换完毕后,关闭压缩工具界面,就可以了  
  
  
7:最后就是放回SD卡,再用RE管理器,进去改权限,替换,然后重启就好了!   
-------------------------------------------  
接着,我再说说一些关于特效脚本的东西  
  
  
anim里面的XML文件，它们被编译进逐帧动画（frame by frame animation）或补间动画（tweened animation）对象.  
  
  
例如:  
  
  
options\_panel\_enter.xml  
options\_panel\_exit.xml 这2个是控制弹出菜单效果的.  
  
  
dialog\_enter.xml  
dialog\_exit.xml 这2个是控制对话框效果的.  
  
  
activity\_close\_enter.xml  
activity\_close\_exit.xml  
activity\_open\_enter.xml  
activity\_open\_exit.xml 这4个是控制内部切换效果的.  
  
  
  
  
input\_method\_enter.xml  
input\_method\_exit.xml  
input\_method\_extract\_enter.xml  
input\_method\_extract\_exit.xml  
input\_method\_fancy\_enter.xml  
input\_method\_fancy\_exit.xml 这6个是控制输入法效果的.  
  
  
  
  
status\_bar\_enter.xml   
status\_bar\_exit.xml 这2个是控制通知栏效果的.  
  
  
  
  
lock\_screen\_behind\_enter.xml  
lock\_screen\_exit.xml 这2个是控制解锁效果的.  
  
  
  
  
  
现在大家知道了这些后,接着,以后看到什么好看的,想要的窗口效果,就去提取出来吧~~  
  
  
得到后放进自己的framework-res.apk的res/anim里面进行替换就行咯,然后改权限,放回手机~重启~就看到效果了~~

⑥系统电量百分比显示  
  
  
4.x系统的教程  
  
  
4.x以后我们可以直接修改“systemui.apk”来增加百分比电量  
  
  
反编译systemui.apk  
  
  
要修改的两个xml文件  
  
  
\res\drawable  
“stat\_sys\_battery.xml”和“stat\_sys\_battery\_charge.xml”，  
  
  
前者是根据当前电量显示图标，后者是根据充电电量显示图标。  
因为已经有前辈为我们写好，所以可以直接拿来替换掉原版里的文件。然后将你选择的200个图标（100个当前电量+100个充电电量）复制到“\res\drawabledrawable-hdpi”目录（480x800分辨率可以使用这个hdpi目录，手机分辨率低的可能要复制到mdpi下）。  
  
  
图标名称要参照上述2个文件。  
  
  
反编译后将生成apk中的“resources.arsc”和“res目录”以"储存"方式替换到原systemui.apk中，这样就基本完成了。  
  
  
附件将打包！  
  
  
2.x系统  
  
  
反编译 framework-res  
  
  
res/drawable/ stat\_sys\_battery.xml和 stat\_sys\_battery\_charge.xml   
  
  
这个就是安卓2.x的电量了  
  
  
方法同上面的4.x！  
  
  
不过xml是我手机提取出来的，网上没找到2.x的，但我的xml充电只显示到28！所以各位友友凑合用吧，看4.x的充电图标xml就知道这需要多少代码了。。。

⑦锁屏界面时间字体修改  
  
  
这个不用反编译！  
  
  
系统根目录/system/font/  
在这里你可以看到好几个ttf的字体包，因为机型不同，所以你可以直接找到一个带clock的字体包，这个就是锁屏界面的字体了！原生锁屏哈！！你准备的ttf不用太大，内含数字和英文字母就行！

⑧全局字体呈斜体  
  
  
此教程为原创，仅在本人手机上亲测成功！  
  
  
反编译framework-res.apk  
  
  
打开res/values/styles.xml  
  
  
直接搜索 normal 可以将它改成 italic(斜体)，bold\_italic(粗斜体)  
注意哦，此处normal必须在android:textSize= 后面才可以改的！

⑨去分割线  
  
  
配合全局背景或者全局透明！  
  
  
反编译framework-res   
res/values/styles.xml  
  
  
找到  
<style name="Widget.TextView.ListSeparator" parent="@style/Widget.TextView">  
<item name="textSize">15.0sp</item>  
<item name="textStyle">bold</item>  
<item name="textColor">?textColorPrimary</item>  
<item name="gravity">center\_vertical</item>  
<item name="background">@drawable/dark\_header\_dither</item>  
<item name="paddingLeft">5.0sp</item>  
<item name="layout\_width">fill\_parent</item>  
<item name="layout\_height">25.0dip</item>  
  
  
  
  
修改其中:<item name="background">@drawable/dark\_header\_dither</item>  
  
  
为  
<item name="background">@null</item>  
  
  
然后再找到  
<style name="Widget.TextView.ListSeparator.SamsungSettings"（看到没，这个是三星设置，有木有，我靠，老子的没有这东西！所以安卓通用你个妹！所以我只改了上面那个！） parent="@style/Widget.TextView.ListSeparator">  
<item name="textStyle">bold</item>  
<item name="textColor">#ffd6d6d6</item>  
<item name="background">@drawable/zzz\_dark\_header\_dither</item>  
  
  
  
  
修改其中： <item name="background">@drawable/zzz\_dark\_header\_dither</item>  
  
为   
<item name="background">@null</item>  
  
  
然后回编。提取resources.arsc到原生apk 再把附件图片全部拉进去frmaework-res.apk!re管理器替换即可，注意该权限！！

⑩加入CRT关屏特效   
  
  
这种特效类似于以前黑白电视关闭的效果，一闪就灭了，闪瞎我的眼！  
  
  
  
  
  
  
  
  
1.反编译framework-res  
2.找到framework-res\res\values\bools.xml  
3.搜索<bool name="config\_animateScreenLights">true</bool>  
4.改为<bool name="config\_animateScreenLights">false</bool>  
5.回编译得到的resources.arsc替换原文件  
6.得到新的farmework-res.apk  
7.改权限放进手机system/farmework文件夹替换原文件  
  
  
有可能不行

四，反编译汇总【部分转】  
  
注意事项；反编译不能在中文文件夹下进行，当然软件的名字也不能有中文，也不能删除中文的东西。  
把apktool数据包放在SD卡跟目录(当然也可以随便放，不过还是建议放在跟目录)  
1；长按是编辑菜单，有些文件夹提示无权访问时可以长按，点击获取访问权限。  
2；点击apk弹出操作菜单，反编译全部是同时反编译dex和资源(res)；  
反编译dex就是反编译apk里面的class,dex；  
反编译资源就是反编译res文件。  
dex文件解压出来也可以单独编译，但是res不能解压出来编译。一般我们美化修改只需要反编译资源就可以满足需求。  
  
  
3；同样是操作菜单，提取odex文件就是提取class.dex文件，虽然它们是同一个文件，但是他们是不能通用的，也就是你把后缀odex改成dex是没有用的，同理，dex改成odex也是没用的。简单介绍一下odex，它可以提高软件的响应速度，不过提取了odex的apk是不能直接安装的，因为apk里少了class,dex文件。删除dex文件就是删除apk里面的class,dex文件，一般是配合提取odex文件使用的，因为此提取odex不会删除dex文件的。  
  
  
4；合并odex，此软件不能合并其它手机的系统odex文件，只能合并自己手机的系统odex文件。点击需要合并的odex文件，反编译，等待之后出现一个文件夹，如果没有出现就是不能反编译，点击生成的文件夹，回编译，等待之后出现一个\*\*\*.dex文件，点击，选择打包到同名apk或jar，成功之后会出现一个class.dex文件，至此合并odex文件就成功了。  
小贴士： dex是安卓系统的可以执行文件,odex就是相当于预读dex,会占用内存,但是程序运行速度会加快   
  
  
5；签名，点击apk，选择签名，等待之后会出现一个\*\*\*\_signed.apk，签名成功。  
回编译之后的软件是没有签名的，必须要签名之后才可以使用。  
  
  
系统软件是不能更改签名的，否则不能用，所以我们必须用到提取签名和添加签名，点击需要编译的apk，选择提取签名，会出现一个签名文件夹，然后点击更改过得的apk，选择添加签名,那么我们刚刚提取的签名就添加进去了。  
  
  
一个文件夹里只能存在一个签名文件夹，添加另一个是需删除前一个。删除签名就是删除apk的签名文件夹。  
  
  
6；导入框架，反编译其它手机的系统文件时需要导入其它手机的框架。导入其它手机的框架之后无法反编译自己手机的系统文件，所以我们要删除导入的框架。导入的框架的文件夹在系统跟目录/data/data/pey\*\*\*/?，删除?文件夹即可。(pey\*\*\*是pey开头的一个文件夹，名字记不住那么长)  
  
  
小提示；反编译失败的原因有很多。反编译出现了文件夹不一定是成功，有可能照样不能回编译，出现这种情况一般是没有导入框架或者改过了图片。  
导致回编译不成功的原因有很多，不过一般照着教程应该不会有问题。  
回编译成功的软件不一定全都可以用，也有可能出现应用程序强制关闭，这就是反编译出错了，但是文本格式没有出错，所以也可以回编译。  
  
  
  
  
解疑如下：  
  
  
首先，busybox仅在首次运行时充当解压软件的作用，之后除了解压打包boot.img,recovery.img还会用到外，busybox不再需要。所以，如果你不修改bootimg,recovery.那么根本不需要人工安装busybox。只需要确保apktool运行时/lix存在。  
如果/lix不是文件夹，那么可以确定是busybox不适合你的手机。  
  
  
其次，出现了类似 Input file (/sdcard/360/??T3g.apk) was not found等带问号的，问号部分属于“非法字符”，你特么把它放在中文目录下了吧！速度放在一英文目录下。。  
  
  
接着，出现brut.directory.PathNotExist: apktool.yml,说明反编译就出错了，回编译绝对不可能成功。  
  
  
然后，出现sh: /sdcard/apktool/\*\*\*.sh:No such file or directory 检查apktool目录是否位于/sdcard/apktool,有些包含内置存储卡的机器会有/mnt/sdcard,/mnt/sdcard2,极有可能你放错了位置。  
  
  
出现libpng error:Not a PNG file.这句是说后面的文件不是合法的png文件，可能是你将jpg直接改名为png，这是行不通的。小贴士：如果你只替换图片图标就别反编译了，直接解压替换就ok！  
  
  
最后一点，提示缺少liblog.so，这是十分奇怪的。java根本没有依赖这个库，东方大神也无能为力。  
  
  
  
  
  
  
  
  
错误信息：  
  
  
  
segmentation fault,请使用3.7版本。  
armel还是armhf?   
armhf的换armel,armel的放弃。   
  
  
  
  
  
  
  
  
  
/lix/jvm/java-7-openjdk-armel/jre/bin/java: error while loading shared libraries: liblog.so: cannot open shared object file: No such file or directory  
与root授权软件有关！  
  
  
/sdcard/apktool/apktool.sh[2]: /lix/jvm/java-7-openjdk-armel/jre/bin/java: not found  
把软件root权限去掉，重新允许试试   
  
  
  
  
  
  
/data/data/per.pqy.apktool/mydata/apktool.sh[4]: /data/data/per.pqy.apktool/lix/jvm/java-7-openjdk-armhf/jre/bin/java: No such file or directory  
卸载重装！  
  
  
/sdcard/apktool/baksmali.sh: /lix/jvm/java-7-openjdk-armel/jre/bin/java: not found  
升级版本  
  
  
  
/data/data/per.pqy.apktool/mydata/apktool.sh[4]: /data/data/per.pqy.apktool/lix/jvm/java-7-openjdk-armel/jre/bin/java: not found  
更换可用的busybox到apktool数据目录！   
  
  
  
  
  
  
  
  
/lix/jvm/java-7-openjdk-armhf/jre/bin/java: error while loading shared libraries: libjli.so: cannot open shared object file: No such file or directory  
使用re将/lix/jvm/java-7-openjdk-armhf/jre/lib/arm/jli/libjli.so链接到/lix   
  
  
  
  
  
  
关键字：  
W:Could not decode attr value,using undecoded value instead:  
缺少系统框架，将/system/framework/framework-res.apk作为框架导入（可能有多个框架文件）   
  
  
  
  
  
  
cpio: can't create symlink from sbin/ueventd to ../init: Operation not permitted  
1585 blocks  
不能在内存卡上解，在data/a下   
  
  
sh：applet not found  
用apktool目录下的busybox替换系统的！  
如果替换后又出现如下这种情况  
sh: <stdin>[1]: busybox: cannot execute - Permission denied   
你特么没有改权限。。。把权限给改了就ok  
  
  
反编后的.xml要utf-8编码来修改哦,不能用RE。  
编辑反编译后的.xml文件推荐使用:Jota+文本编辑器,设置打开为utf-8编码,保存为utf-8编码,设置换行为linux/android方式,最好关闭自动换行功能!  
  
  
加速编译:把编译对象改为英文名,复制到根目录/cache或者/data下(系统磁盘读写速度快),然后用apktool进入/cache或者/data下,  
如果无法进入,可以长按目录,在弹出菜单中选择"获取访问权限"即可进入,然后选择目标apk进行反编,你会发现速度变得超快!  
小贴士：我就是这样做的^v^，可是内存小的容易卡！注意哦！  
  
  
修改.xml必须是utf-8编码,修改完成后,要删除编辑后的备份文件,比如Jota编辑的文件会生成"原文件名.bak"这样的备份,要删除,回编会输出一些提示,如果提示只有五四句则成功回编,如果有"more","VS","don't"等关键字则出错!  
  
  
【回编出错常见错误】  
①输出信息中有"vs"字符,出现原因很简单,大多数是在定义id时重复,修改/style/public.xml中的对应问题即可!  
②出现"don't":这个问题可能是你删除图片时没有删除/style/public.xml!还可能是添加了图片却没有在style/public.xml添加id//,其实很多回编出错都和public.xml有关!  
③反编译apk时0秒完成，出现"notfound"。可能是因为busybox版本不合而无法建立资源链接。根目录中没有lix文件夹软链接。需要用RE进入/data/data然后用RE搜索apktool，，会搜到一个文件夹，进入后会看到lix这个文件夹，长按lix文件夹，在弹出菜单中选择"建立链接"，然后返回系统根目录，最后选择链接到此即可！   
  
  
还有一点，手机端反编译后一定会有apktool.yml，如果没有，则反编译失败，回编译肯定Game over。。。

五，其他美化  
  
  
  
  
  
  
以下教程，我没亲测，手机不同，可能也不一样！  
  
  
  
  
①拨号键盘背景修改  
  
  
话说我改过，到现在还没回编译。。。Q\_Q。。一直放哪的。。没时间。。  
  
  
  
  
反编译 Contacts.apk  
  
  
res/layout-long-finger/twelve\_key\_dialer.xml  
打开它，我的xml里只有一个android:background  
  
  
有个android:background="#ff24282e"   
这里分号注释的是颜色代码 大家可以图片编辑软件查这种颜色代码 这里我把他改为黑色 黑色的颜色代码为#ffoooooo  
透明为#00000000  
改好后保存   
从编译出来的Contacts.apk 文件里替换twelve\_key\_dialer.xml 到原文件Contacts.apk 里 压缩方式选择存储  
  
  
  
  
  
  
②短信背景的修改  
  
  
这个短信背景。。我不怎么好说。。需要反编译短信软件，注意！是原厂自带的短信！不是第三方！想要第三方的自己去网上撸。。  
  
  
  
  
还有因为机型不一样，所以短信apk也不一样！比如我的手机短信还特么是韩语！勒个去，想自己搞搞，但看到4M多，我就放弃了。。花时间太多了。。嘎嘎。。  
还有这个应该是2.3的。。  
  
  
反编译 mms.apk  
  
  
\res\layout\compose\_message\_activity.xml  
  
  
打开它  
  
  
1.搜索  
<LinearLayout   
androidrientation="vertical" android:id="@id/recipients\_subject\_linear"  
在下面找到   
背景颜色代码 android:background="#ffcecece" 修改这个代码  
比如改为 #ff000000 黑色  
  
  
2.搜索  
<LinearLayout android:gravity="bottom"   
androidrientation="vertical"   
<item android:id="@id/progress">  
在下面找到   
背景颜色代码 android:background="#ffcecece" 修改这个代码  
比如改为 #ff000000 黑色  
  
  
3.搜索  
<view   
android:scrollbarAlwaysDrawVerticalTrack="true"   
android:scrollbarStyle="insideOverlay" android:id="@id/history"  
在下面找到   
android:listSelector="#00000000" 修改这个代码  
如果已经是 #00000000 就不用改  
然后编译  
  
  
编译好了找到compose\_message\_activity.xml8替换到原来的MMS.APK里！

好了，就这些了，教程有什么错误请及时私我！！！！！！  
  
  
最后！玩机需谨慎！  
  
  
祝大家玩出自己的一片天！  
  
  
  
  
以上教程仅供参考，如有事故，概不负责！