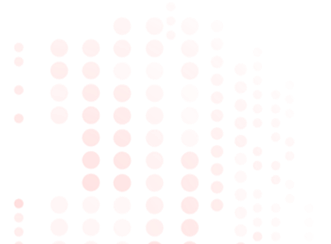


**ANDROID预置APK**

**SMART Module**



**SIMCom Wireless Solutions Limited**

Building B, SIM Technology Building, No.633, Jinzhong Road

Changning District, Shanghai P.R. China

Tel: 86-21-31575100

support@simcom.com

www.simcom.com

|  |  |
| --- | --- |
| **Document Title:** | ANDROID预置APK |
| **Version:** | 1.00 |
| **Date:** | 2021.10.10 |
| **Status:** | Release/Confidential |

**GENERAL NOTES**

SIMCOM OFFERS THIS INFORMATION AS A SERVICE TO ITS CUSTOMERS, TO SUPPORT APPLICATION AND ENGINEERING EFFORTS THAT USE THE PRODUCTS DESIGNED BY SIMCOM. THE INFORMATION PROVIDED IS BASED UPON REQUIREMENTS SPECIFICALLY PROVIDED TO SIMCOM BY THE CUSTOMERS. SIMCOM HAS NOT UNDERTAKEN ANY INDEPENDENT SEARCH FOR ADDITIONAL RELEVANT INFORMATION, INCLUDING ANY INFORMATION THAT MAY BE IN THE CUSTOMER’S POSSESSION. FURTHERMORE, SYSTEM VALIDATION OF THIS PRODUCT DESIGNED BY SIMCOM WITHIN A LARGER ELECTRONIC SYSTEM REMAINS THE RESPONSIBILITY OF THE CUSTOMER OR THE CUSTOMER’S SYSTEM INTEGRATOR. ALL SPECIFICATIONS SUPPLIED HEREIN ARE SUBJECT TO CHANGE.

**COPYRIGHT**

THIS DOCUMENT CONTAINS PROPRIETARY TECHNICAL INFORMATION WHICH IS THE PROPERTY OF SIMCOM WIRELESS SOLUTIONS LIMITED COPYING, TO OTHERS AND USING THIS DOCUMENT, ARE FORBIDDEN WITHOUT EXPRESS AUTHORITY BY SIMCOM. OFFENDERS ARE LIABLE TO THE PAYMENT OF INDEMNIFICATIONS. ALL RIGHTS RESERVED BY SIMCOM IN THE PROPRIETARY TECHNICAL INFORMATION ，INCLUDING BUT NOT LIMITED TO REGISTRATION GRANTING OF A PATENT , A UTILITY MODEL OR DESIGN. ALL SPECIFICATION SUPPLIED HEREIN ARE SUBJECT TO CHANGE WITHOUT NOTICE AT ANY TIME.

**SIMCom Wireless Solutions Limited**

Building B, SIM Technology Building, No.633 Jinzhong Road, Changning District, Shanghai P.R. China

Tel: +86 21 31575100

Email: [simcom@simcom.com](mailto:simcom@simcom.com)

**For more information, please visit:**

<https://www.simcom.com/download/list-863-en.html>

**For technical support, or to report documentation errors, please visit:**

https://www.simcom.com/ask/ or email to: [support@simcom.com](mailto:support@simcom.com)

***Copyright © 2020 SIMCom Wireless Solutions Limited All Rights Reserved.***

# About Document

## Document Information

|  |  |
| --- | --- |
| **Document** | |
| Title | Android预置APK |
| Version | 1.00 |
| Document Type | Application Note |
| Document Status | Released/Confidential |

## Version History

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date** | **Owner** | **Status / Comments** |
| 1.00 | [October](C:/Users/mingmin.liu/AppData/Local/Youdao/dict/Application/8.9.8.0/resultui/html/index.html#/javascript:;),11, 2021 | Mingmin Liu | First Release. |

## Scope

**This document applies to the following products**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Type** | **Size(mm)** | **Comments** |
| ALL | Smart Module | N/A | N/A |

# Contents

[About Document 3](#_Toc29049)

[Document Information 3](#_Toc758)

[Version History 3](#_Toc27400)

[Scope 3](#_Toc23395)

[Contents 4](#_Toc22026)

[1. 文档目的 5](#_Toc29343)

[2. Android平台预置APK流程 5](#_Toc26681)

[一．预置无源码的APK进系统 5](#_Toc4749)

[二．预置带源码的APK进系统 6](#_Toc27286)

[三．增加APP默认权限 6](#_Toc13338)

# 文档目的

#### 一． Android 系统稳定性描述

本文介绍了ANDROID平台预置APK的相关知识，可以帮助开发人员快速熟悉此模块在系统层面的实现框架，从而提高问题处理效率。

# Android平台预置APK流程

#### 一． Android 系统稳定性描述

目前在工作过程中发现客户有预置app的需求，现将预置app的流程介绍一下，以便查阅。

预置apk的方法：

1.不带源码只有apk。

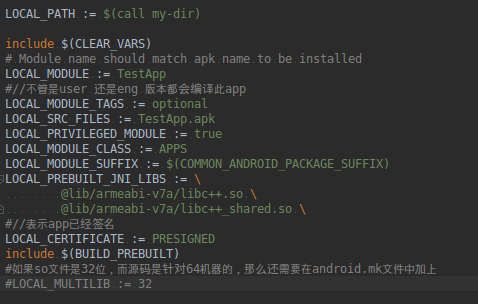
2.带源码。

## 一．预置无源码的APK进系统

1.在源码目录中packages/apps下创建一个新文件夹并以要预置的apk来命名，以TestApp为例：packages/apps/TestApp；

2.将要预置的apk(TestApp.apk) 放到 packages/apps/TestApp目录下；

3.在 packages/apps/TestApp 目录下创建文件Android.mk，文件内容如下：



若无so，删除LOCAL\_PREBUILT\_JNI\_LIBS这一项。  
若有so，则将app复制一份，将后缀改为zip后缀(TestApp.zip)并解压,将压缩包中的lib文件夹提取出来并放到packages/apps/TestApp目录下，使用LOCAL\_PREBUILT\_JNI\_LIBS列出所有so的路径，不要忘记使用@。@标识符会将apk中的so抽离出来build进apk同级目录下的lib文件夹中

若apk支持不同cpu类型的so，针对so的部分的处理:

Ifeq ($(TARGET\_ARCH),arm)

LOCAL\_PREBUILT\_JNI\_LIBS := \

@lib/armeabi-v7a/xxx.so\

@ lib/armeabi-v7a/xxxx.so

else ifeq ($(TARGET\_ARCH),x86)

LOCAL\_PREBUILT\_JNI\_LIBS := \

@lib/x86/xxx.so

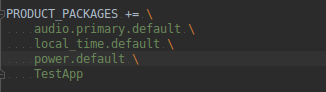
else ifeq ($(TARGET\_ARCH),arm64)

LOCAL\_PREBUILT\_JNI\_LIBS := \

@lib/armeabi-v8a/xxx.so

即将和TARGET\_ARCH对应的so抽离出来

4.打开文件 device\sprd\common\device.mk将 TestApp 添加到 PRODUCT\_PACKAGES 里面。



5.重新 build 整个工程

如果App使用System Level的permission，需要預置到/system/priv-app底下 (原在/system/app)。  
修改Android.mk，增加LOCAL\_PRIVILEGED\_MODULE := true，以声明app需要放在/system/priv-app下。

## 二．预置带源码的APK进系统

1.在 packages/apps 下面以需要预置的 APK的 名字创建一个新文件夹，以预置名为TestApp的APK 为例

2.将 Test APK的Source code 拷贝到 TestApp 文件夹下，删除 /bin 和 /gen 目录

3.在 Test 目录下创建一个名为 Android.mk的文件，内容如下：

LOCAL\_PATH:= $(call my-dir)

include $(CLEAR\_VARS)

LOCAL\_MODULE\_TAGS := optional

LOCAL\_SRC\_FILES := $(call all-subdir-java-files)

LOCAL\_PACKAGE\_NAME := TestApp

include $(BUILD\_PACKAGE)

4.打开文件 device\qcom\common\device.mk或base.mk，将 TestApp 添加到 PRODUCT\_PACKAGES 里面。

PRODUCT\_PACKAGES += TestApp

1. 重新 build 整个工程

## 三．预置可卸载APP,恢复出厂后不可恢复

1.在 packages/apps 下面以需要预置的 APK的 名字创建一个新文件夹，以预置名为TestApp的APK 为例

2.将 Test APK的Source code 拷贝到 TestApp 文件夹下，删除 /bin 和 /gen 目录

3.在 Test 目录下创建一个名为 Android.mk的文件，内容如下：

4.打开文件 device\qcom\common\device.mk 或base.mk，将TestApp添加到PRODUCT\_PACKAGES里面。

PRODUCT\_PACKAGES += TestApp

1. 重新 build 整个工程

 注意：这个比不能卸载的多了一句：

LOCAL\_MODULE\_PATH := $(TARGET\_OUT\_DATA\_APPS)

## 四．增加APP默认权限

预置进系统的app在首次进入的时候会弹出选择权限的界面。此时需要在代码中增加相应的代码来默认打开权限。

文件位置：frameworks/base/services/core/java/com/android/server/pm/DefaultPermissionGrantPolicy.java

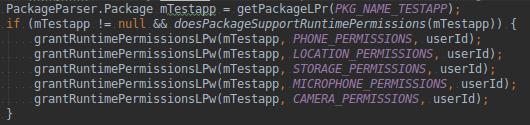
修改方法：

1.定义包名，包名即APK的包名:

IMG_256

2.增加权限：

在grantDefaultSystemHandlerPermissions(int userId)方中增加以下代码：



图中为示例，具体权限可根据需要增加。