**安卓原生framework源码：**

<https://cs.android.com/android/platform/superproject/+/master:;bpv=0;bpt=0>

**安卓原生framework自带的某个测试样例：**

<https://cs.android.com/android/platform/superproject/+/master:cts/tests/AlarmManager/>

**我们的系统测试的主要工作内容：**

参照安卓原生framework自带的测试样例，对我们的framework中的改动过的类编写测试。

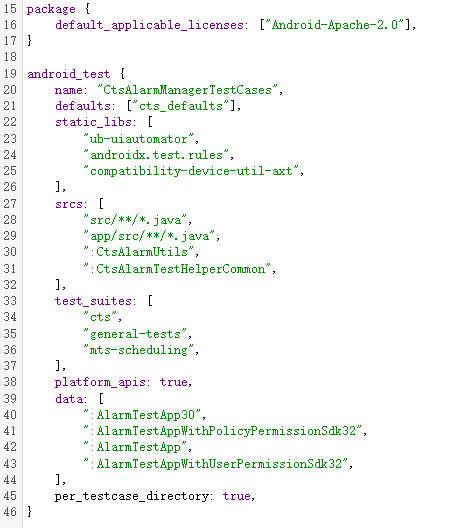
**Android.bp文件：**

新建的测试模块必须要有的一个配置文件，用于配置构建系统。

里面的简单配置可参考文档：

<https://source.android.com/docs/core/tests/development/blueprints>

**Android.bp文件大致内容：**



**其中的android\_test：**

模块类型为android\_test，表示构建完后会生成一个.apk文件。这么设计可能是因为有一些待测试条目，如广播等，需要借助于一个具体的app才能发出。

**其中的name:**

每个模块都必须有一个name，并且name的值在所有的Android.bp文件中必须是唯一的。对于android\_test模块类型，其name字段代表最终构建生成的.apk文件名。

**其中的srcs:**

1. srcs 属性以字符串列表的形式指定用于编译模块的源文件。您可使用模块引用语法 “:” 来引用生成源文件的其余模块的输出，如 genrule 或 filegroup。

参考：<https://blog.csdn.net/liujun3512159/article/details/124601811>

1. 官方定义：

list of source files used to compile the C/C++ module. May be .c, .cpp, or .S files. srcs may reference the outputs of other modules that produce source files like genrule or filegroup using the syntax ":module".

参考：<https://note.qidong.name/demo/soong_build/>

1. 举例：





**其中的static\_libs:**

1. 编译所依赖的静态库，相似于Android.mk中的LOCAL\_STATIC\_LIBRARIES。
2. static\_libs将指示构建系统将命名模块的内容合并到当前模块的最终结果 apk 中。这意味着每个命名模块都应生成一个.jar文件，其内容将用于在编译期间解析类路径引用，并合并到生成的 apk 中。
3. 官方原文：<https://source.android.com/docs/core/tests/development/blueprints>

The static\_libs setting instructs the build system to incorporate the contents of the named modules into the resulting apk of current module. This means that each named module is expected to produce a .jar file, and its content will be used for resolving classpath references during compile time, as well as incorporated into the resulting apk.

**其中的test\_suits：**

1. test\_suites: ["device-tests"],  
   该test\_suites设置使 Trade Federation 测试工具可以轻松发现测试。可以在此处添加其他套件，例如 CTS，以便可以共享此测试。

参考：<https://source.android.com/docs/core/tests/development/blueprints>

1. For a test to be a part of VTS, it must have the following setting in Android.bp.

参考：<https://source.android.com/docs/core/tests/vts/setup11>

1. list of compatibility suites (for example "cts", "vts") that the module should be installed into.

参考：<https://note.qidong.name/demo/soong_build/>

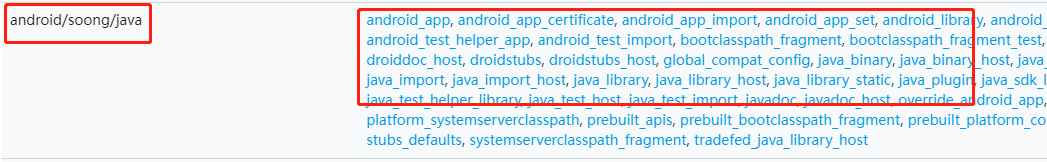
**其中的data：**

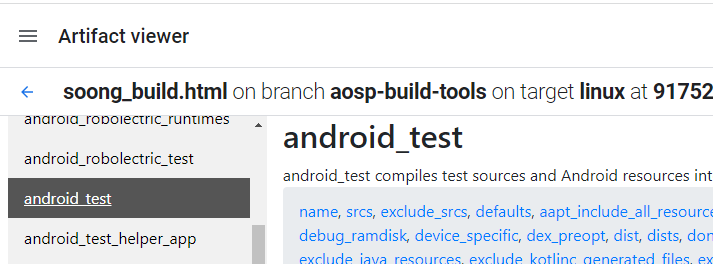
list of files or filegroup modules that provide data that should be installed alongside the test.

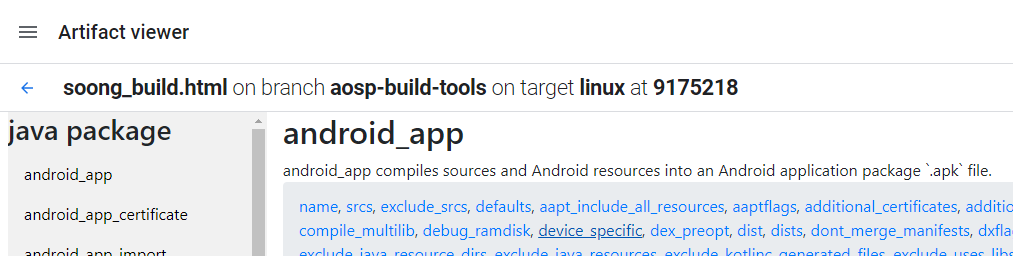
参考：<https://note.qidong.name/demo/soong_build/>

**Soong关于.bp文件配置的官方文档（非常全）**

<https://ci.android.com/builds/submitted/9175218/linux/latest/view/soong_build.html>







**关于Android.bp和Soong构建系统**

1. Android.bp是专用于Soong构建系统的。Soong构建系统致力于取代传统的Make构建系统。

参考：<https://source.android.com/docs/setup/build>

1. android编译系统soong集成了Blueprint，Blueprint可将我们编写的android.bp解析生成一个ninja构建文件。我们在编译一个模块时，只需要将这个模块的android.bp文件配置好，编译系统会自动为这个模块生成ninja清单，最终使用ninja来调用gcc、clang、java、dex、aapt2等等命令来构建模块。

参考：<https://www.jianshu.com/p/f69d1c381182>

1. ninja is the new build tools for aosp, which using \*.ninja to work;
2. for the legacy Android.mk, kati is applied to transform Androd.mk to \*.ninja;
3. google is now using Android.bp file to describe module compile details, which is also transformed to \*.ninja by blueprint/soong
4. after all Android.mk is rewrite to Android.bp, kati will be removed, and we will have blueprint/soong/ninja only

参考：<https://stackoverflow.com/a/51923815/17576647>

1. Soong build system is equivalent to legacy "Android make build system". Therefore Soong is collection of program, which are recipes "how to build hundreds of android modules" with lots of small tools.
2. Ninja is a process to really handle the final build manifest(Build.ninja) which is made by Blueprint or Kati (translate Android.mk to Android.bp).
3. Blueprint is like GNU Make grammar, so called "framework for build system". it made by Go language, hence it better provides multi-processing. Blueprint translate "hand-writing build manifest"(\*.bp) to final manifest(build.ninja) which is ninja format.

参考：<https://stackoverflow.com/a/50920255/17576647>

1. Soong官方文档：

<https://android.googlesource.com/platform/build/soong/+/master/README.md>

**Android.bp文件里面， 模块类型android\_test与android\_test\_helper\_app的区别**

1. android\_test\_helper\_app compiles sources and Android resources into an Android application package `.apk` file that will be used by tests, but does not produce an `AndroidTest.xml` file so the module will not be run directly as a test.
2. android\_test compiles test sources and Android resources into an Android application package `.apk` file and creates an `AndroidTest.xml` file to allow running the test with `atest` or a `TEST\_MAPPING` file.
3. 总结一下，android\_test\_helper\_app和android\_test都会编译出apk，不同的是android\_test会额外编译出一个AndroidTest.xml，这是cts测试包的配置文件，从而可以使apk作为载体运行cts测试。

参考soong框架部分源码：

<https://android.googlesource.com/platform/build/soong/+/master/java/app.go>