

Tina

System init 使用说明文档 V2.0

文档履历

版本号	日期	制/修订人	制/修订记录
V1.0	2017/12/05		初始版本
V2.0	2018/03/28		



目 录

1. 概述.....	4
1.1. 编写目的.....	4
1.2. 适用范围.....	4
1.3. 相关人员.....	4
2. System init 简介.....	5
3. system init 选择.....	5
4. Busybox init 配置使用说明.....	6
4.1. 添加自启动脚本.....	6
4.2. log 系统与内核模块自加载.....	7
4.3. 系统启动调用顺序.....	7
4.4. 添加平台配置文件到根文件系统.....	8
5. Declaration.....	8



1. 概述

1.1. 编写目的

帮助熟悉使用 tina system init。

1.2. 适用范围

Allwinner 软件平台 Tina v3.0 版本以上
Allwinner 硬件平台 R6 R11 R16 R18 R30 R40

1.3. 相关人员

适用 Tina 平台的 system init 相关人员。



2. System init 简介

所谓 system init 这里指的是内核启动之后调用的第一个应用进程，也称为用户空间守护进程、1 号进程，用户空间的所有其他进程都由该进程派生。

目前 tina 系统支持的 system init 有两种方式。一种是 procd init，另一种是 busybox init。其中 procd init 为 tina 原生支持（tina 基于 openwrt），适用于所有平台；而 busybox init，理论上也适用于所有平台，但是目前投入使用的主要为 R6, R11 系列平台。

本文侧重于使用 busybox init 的配置方法。

3. system init 选择

目前大部分方案默认使用 procd init，其中 R6 与 R11 大部分方案使用 busybox init。

在编译之前，可执行 make menuconfig 进行配置（选择使用 procd init 还是 busybox init）：

```
make menuconfig
└─> System init (busybox-init) --->
    └─> ( ) procd-init
    └─> (X) busybox-init
```



4. Busybox init 配置使用说明

关于 procd 配置使用说明这里不做过多说明，重点说明使用 busybox init 方式启动的注意事项。

```
make menuconfig ---请已经保证按照小节 3 选上 busybox-init
└─> Base system --->
    *- busybox-init-base-files..... Busybox init base system --->
    [ ] Customize busybox init base files options
    [*] Use the rc.log #详见①
    [*] Auto load the script in /etc/rc.d #详见②
    [*] Use the rc.modules #详见③
```

①用于配置记录系统启动 log 信息。

启动脚本位于: <tina>/package/busybox-init-base-files/files/etc/init.d/rc.log, 最终将会拷贝到小机端的/etc/init.d/目录下。

②用于启动加载/etc/rc.d 下面的脚本（默认只调用 adb, 所有要启动 adb 需要把该选项选上）

③用于加载内核模块。

启动脚本位于<tina>/package/busybox-init-base-files/files/etc/init.d/rc.modules, 最终将会拷贝到小机端的/etc/init.d/目录下。

4.1. 添加自启动脚本

Busybox init 添加自启动脚本的方式主要有以下三种方式。

(1) 方式 1: rc.preboot

如下路径创建 rc.preboot 脚本，脚本写入需要配置内容。

```
<tina>/target/allwinner/<platform>/busybox-init-base-files/etc/init.d/rc.preboot
```

rc.preboot 脚本将会被系统最先调用，目的是满足开机快速启动的需求。因此，想要开机最先启动的应用可以在 rc.preboot 脚本中配置。

另外，需要注意一个问题，由于该脚本最先被调用，因此系统的一些配置信息都还为来得及初始化，比如内核模块加载，UDISK 挂载，系统 log 系统，热插拔使能，以及/dev 下驱动节点等等。

(2) 方式 2: procd 与 sys 式启动脚本

这类的脚本与平时的脚本格式有些区别，该类脚本主要目的在于某个单独应用的自启动配置。脚本往往存放在某个 package 下，编译的时候拷贝到根文件系统的/etc/init.d/下，同时编译系统会自动将其链接到根文件/etc/rc.d/下。

下面 smartlinkd 应用的启动脚本为例子，

脚本路径:<tina>/package/allwinner/smartlinkd/files/smartlinkd.init。

A. Procd 式

```
#!/bin/sh /etc/rc.common #本质为 script 脚本,以#!开头,之后执行/etc/rc.common
START=98 #开机启动优先级(序列) [数值越小,越先启动]
STOP=98 #关机停止优先级(序列) [数值越小,越先关闭]
USE_PROCD=1 #procd 与 sys 脚本的区别体现在这个变量上
PROG=smartlinkd
start_service() { #启动函数
    procd_open_instance
    procd_set_param command $PROG -d
    procd_close_instance
}
shutdown() {
    echo shutdown
}
```

B.Sys 式

```
#!/bin/sh /etc/rc.common
START=98
STOP=98
```

```
PROG=smartlinkd
```

```
start() {  
    smartlinkd -d &  
}
```

完成上述脚本的内容后，需要拷贝一份

```
<tina>/package/busybox-init-base-files/files/etc/init.d/load_script.conf
```

到如下目录，并在 load_script.conf 中换行添加 smartlinkd

```
<tina>/target/allwinner/<platform>/busybox-init-base-files/etc/init.d/
```

根文件系统/etc/rc.d/会有很多包的配置脚本。有时候并不需要启动所有的启动脚本，而 load_script.conf 配置文件就是用于控制/etc/rc.d/下那些脚本需要被启动。

总结：使用方式二适用于单个 package 应用自启动配置，同时这种脚本的格式既能兼容 procd init 也能兼容 busybox init。

关于此类脚本的方式，详情可参考，以下网址：

```
https://wiki.openwrt.org/inbox/procd-init-scripts
```

```
https://wiki.openwrt.org/doc/techref/initscripts
```

(3) 方式 3: rc.final

如下路径创建 rc.final 脚本，脚本写入配置内容。

```
<tina>/target/allwinner/<platform>/busybox-init-base-files/etc/init.d/rc.final
```

rc.preboot 脚本将会被系统最后调用，在调用该脚本前，系统的基本配置已经完成。

4.2. log 系统与内核模块自加载

默认情况下，系统使用如下 rc.log 脚本用于配置 log 系统。

```
<tina>/package/busybox-init-base-files/files/etc/init.d/rc.log
```

使用如下 rc.modules 用于配置内核模块自加载。

```
<tina>/package/busybox-init-base-files/files/etc/init.d/rc.modules
```

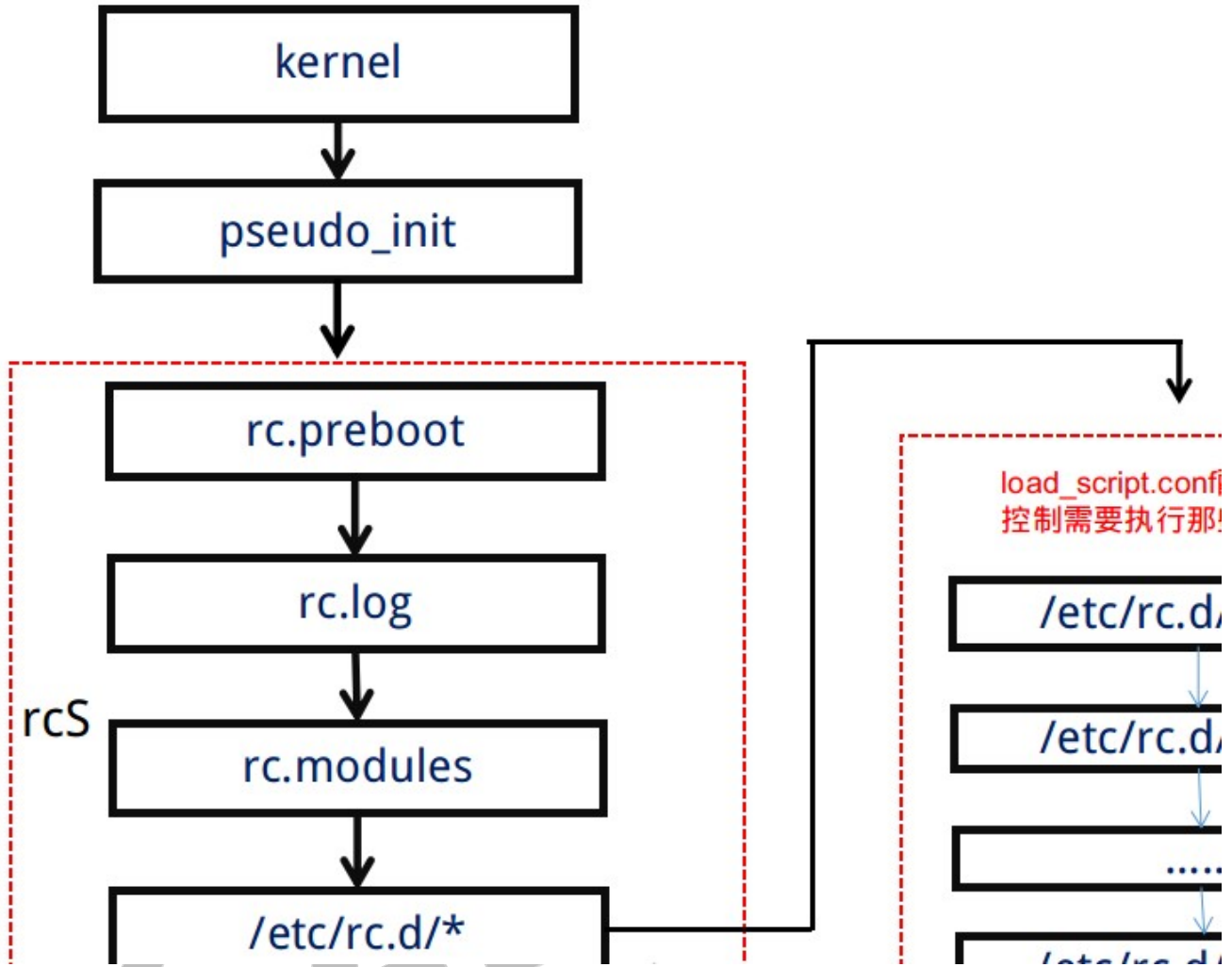
对于以上默认的 rc.log,以及 rc.modules，如果某个平台下不想使用这两个脚本去配置 log 系统和内核模块自加载。用户可重新定义两个配置脚本，在以下路径创建 rc.log, rc.modules 脚本。

```
<tina>/target/allwinner/<platform>/busybox-init-base-files/etc/init.d/
```

编译的时候系统会把默认的配置脚本 rc.log,rc.modules 覆盖掉。

4.3. 系统启动调用顺序

以下是使用 busybox init 系统启动调用顺序。



4.4. 添加平台配置文件到根文件系统

不同的平台文件系统具有其共性与特殊性。tina/package/busybox-init-base-files/files 下提供了所有平台根文件系统的基础文件。而在 tina/target/allwinner/<platform>/busybox-init-base-files 下存放的是平台特性文件。同时其优先级高于前者,即如有前者目录和后者目录存在有相同文件时,以后者为准。如有以下两个文件:

文件 A:

```
<tina>/target/allwinner/banjo-R11_pref1/busybox-init-base-files/etc/banner
```

文件 B:

```
<tina>/package/busybox-init-base-files/files/etc/banner
```

最终拷贝到根文件系统的为文件 A。

由于 pseudo_init 与 rcS 文件中存在很多平台共性的代码,避免系统充斥大量冗余代码,以及方便基础文件的维护和开发所以不允许在特定平台下自定义 pseudo_init、rcS 文件。

(必须使用 tina/package/busybox-init-base-files/files 下的 pseudo_init、rcS)。

5. Declaration

This document is the original work and copyrighted property of Allwinner Technology ("Allwinner"). Reproduction in whole or in part must obtain the written approval of Allwinner and give clear acknowledgment to the copyright owner.



外部公开

The information furnished by Allwinner is believed to be accurate and reliable. Allwinner reserves the right to make changes in circuit design and/or specifications at any time without notice. Allwinner does not assume any responsibility and liability for its use. Nor for any infringements of patents or other rights of the third parties which may result from its use. No license is granted by implication or otherwise under any patent or patent rights of Allwinner. This datasheet neither states nor implies warranty of any kind, including fitness for any particular application.

