

# Tina

Uboot 阶段修改系统配置 V1.3

# 文档履历

版本号	日期	制/修订人	制/修订记录
V1.0	2016/8/19		初始版本
V1.1	2016/10/27		R40 方案第二次 review
V1.2	2017/11/08		Tina v2.5
V1.3	2018/04/13		Tina v3.0



# 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. 系统配置文件.....	4
2.1. sys_config.fex.....	4
2.2. 设备树.....	4
2.3. Uboot 配置头文件.....	5
2.4. env.cfg/sys_partition.fex/sys_partition_nor.fex.....	6
3. sys_config.fex 文件解析.....	7
3.1. 量产和烧录相关属性.....	7
3.1.1. eraseflag.....	7
3.1.2. burn_key.....	8
3.2. Boot 阶段电源管理.....	8
3.2.1. power_start.....	8
3.3. 串口调试.....	9
3.3.1. debug_mode.....	9
3.3.2. uart_debug_port/uart_debug_tx/uart_debug_rx.....	9
3.4. 其余模块配置.....	9
4. uboot 配置头文件.....	10
4.1. 地址相关 (base、size、offset).....	10
4.2. uboot 命令相关.....	10
4.3. 模块开关宏.....	10
5. 环境变量和分区配置.....	11
5.1. env.cfg.....	11
5.2. sys_partition.fex/sys_partition_nor.fex.....	11
6. uboot 编译.....	13
7. Declaration.....	14

## 1. 概述

### 1.1. 编写目的

本文主要服务于使用 Tina 软件平台和 Allwinner 智能硬件平台的广大客户，以冀帮助客户方软件相关人员熟练掌握如何在 Tina 平台修改系统配置以适应不同开发和调试需求。

### 1.2. 适用范围

Allwinner 软件平台 Tina

### 1.3. 相关人员

适用 Tina 平台的广大客户和相关技术人员。

## 2. 系统配置文件

在 Tina 中，软件开发人员可以修改的系统配置文件主要有以下几类：

### 2.1. sys\_config.fex

**sys\_config.fex** 是最主要的**系统配置文件**，用户基本可以在其中配置所有**系统需求**。

文件路径为：

```
tina/target/allwinner/xxx/configs/sys_config.fex
```

sys\_config.fex 是**高度模块化的**，每个模块都有 **section 名**（称为主键，main name），以“**[ ]**”括起来单独一行作为**模块头部**，接着每行会以**键对值形式**列出该**模块的属性**（称为次键，sub name）和值（value），例如重要的模块有[product]、[platform]、[target]、[dram\_para]等。

示例：

```
[target]
storage_type = -1
burn_key    = 1
```

在软件代码中可以通过接口函数 **script\_parser\_fetch** 去**获取对应模块的属性值**；在打包固件时，也会根据 sys\_config.fex **更新固件头部信息**。sys\_config.fex 文件将所有模块的配置**集中管理、统一接口、高度可视化、可读性强、易于修改、配置灵活、uboot 和内核通用、修改之后直接打包固件无需重新编译**等特性和优点，很好的满足了用户定制软件系统和打包固件的需求。

### 2.2. 设备树

**dts 文件编译生成的 dtb 文件**是内核设备驱动所依赖的硬件设备树信息表。

对于 linux-3.4，不使用设备树文件。

对于 linux-3.10 和 linux-4.4 文件路径为：

```
tina/lichee/linux-3.10/arch/arm/boot/dts/平台代号-soc.dts
```

```
tina/lichee/linux-4.4/arch/arm64/boot/dts/sunxi/平台代号-soc.dts
```

平台代号如下

芯片	平台代号
R6	sun3iw1p1
R16	sun8iw5p1
R18	sun50iw1p1
R40	sun8iw11p1
R11	sun8iw8p1
R30	sun50iw3p1

传统的 linux 需要用户在修改该文件上花费较多时间精力，但 Tina 软件平台 sys\_config.fex 文件和系统的存在减轻了这部分负担，因为 sys\_config.fex 会覆盖 dtb 文件相应的属性，即 **sys\_config.fex 拥有较高优先级**。

另外，sys\_config.fex 和 dtb 文件中的属性值存储在 flash 上，如果需要在**烧写固件之后修改相应属性**，可以进入 uboot 命令行（在 env bootdelay 不为 0 时可以按按键盘任意键进入）执行 fdt 命令：

```
fdt - flattened device tree utility commands
Usage:
fdt addr [-c] <addr> [<length>] - Set the [control] fdt location to <addr>
fdt move <fdt> <newaddr> <length> - Copy the fdt to <addr> and make it active
fdt resize - Resize fdt to size + padding to 4k addr
fdt print <path> [<prop>] - Recursive print starting at <path>
fdt list <path> [<prop>] - Print one level starting at <path>
fdt get value <var> <path> <prop> - Get <property> and store in <var>
fdt get name <var> <path> <index> - Get name of node <index> and store in <var>
fdt get addr <var> <path> <prop> - Get start address of <property> and store in <var>
fdt get size <var> [<path>] [<prop>] - Get size of [<property>] or num nodes and store in <var>
fdt set <path> <prop> [<val>] - Set <property> [to <val>]
fdt mknnode <path> <node> - Create a new node after <path>
fdt rm <path> [<prop>] - Delete the node or <property>
fdt header - Display header info
fdt bootcpu <id> - Set boot cpuid
fdt memory <addr> <size> - Add/Update memory node
fdt rsvmem print - Show current mem reserves
fdt rsvmem add <addr> <size> - Add a mem reserve
fdt rsvmem delete <index> - Delete a mem reserves
fdt chosen [<start> <end>] - Add/update the /chosen branch in the tree
    <start>/<end> - initrd start/end addr
fdt save - write fdt to flash
```

常用的命令如：

```
fdt print --打印整棵设备树
fdt print /soc/vdevice --打印“/soc/vdevice”路径下的配置信息
fdt set /soc/vdevice status "disabled" --设置“/soc/vdevice”下 status 属性的属性值
```

如在 sys\_config.fex 下的 [platform] 属性，要查看其值，需要写对**路径/soc/platform**（不能直接写 platform），即执行

```
fdt print /soc/platform
```

另外，fdt set 之后需要执行 **fdt save** 才能真正写入 flash 保存。

## 2.3. Uboot 配置头文件

Uboot 对于具体每个平台，由一个头文件，即平台代号.h 来进行配置。

对于 nor flash 存储介质，需要使用到单独的头文件，平台代号\_nor.h。

对于 uboot2011 文件路径为：

```
tina/lichee/brandy/u-boot-2011.09/include/configs/平台代号.h
tina/lichee/brandy/u-boot-2011.09/include/configs/平台代号_nor.h
```

对于 uboot2014 文件路径为：

```
tina/lichee/brandy/u-boot-2014.07/include/configs/平台代号.h
tina/lichee/brandy/u-boot-2014.07/include/configs/平台代号_nor.h
```

编译 uboot 时会将目标 board 的头文件（平台代号.h 或平台代号\_nor.h）include 到 config.h，而 config.h 是 autoconf.mk 的依赖文件，autoconf.mk 会把其中的**宏定义转换成环境变量的形式 include 到编译环境中**，从而**子目录的 Makefile 可以获取到这些配置决定编译哪些源文件**，而源文件中也可得到这些环境变量决定**编译哪些代码**，进而实现 uboot 的**可配置可裁剪**。

一些重要的宏，如

```
CONFIG_TARGET_NAME 决定了 uboot 固件名
CONFIG_SYS_TEXT_BASE 决定 uboot 运行地址
```

## 2.4. env.cfg/sys\_partition.fex/sys\_partition\_nor.fex

env.cfg 根据使用内核版本 (linux3.4 或 linux3.10 或 linux4.4) 分为 env-3.4.cfg, env-3.10.cfg, env-4.4.cfg, 用于环境变量。sys\_partition.fex(当使用 nor flash 存储介质时为 sys\_partition\_nor.fex) 用于分区配置。

Tina 下的配置文件可能有几个路径。

Tina 默认配置文件路径:

```
tina/target/allwinner/generic/configs/env.cfg  
tina/target/allwinner/generic/configs/sys_partition.fex
```

芯片默认配置文件路径:

```
tina/target/allwinner/xxx-common/configs/env.cfg  
tina/target/allwinner/xxx-common/configs/sys_partition.fex
```

具体方案配置文件路径:

```
tina/target/allwinner/xxx-xxx/configs/env.cfg  
tina/target/allwinner/xxx-xxx/configs/sys_partition.fex
```

优先级依次递增, 即优先使用具体方案下的配置文件, 没有方案配置, 则使用芯片默认配置文件, 没有方案配置和芯片配置, 才使用 tina 默认配置文件。

实际开发中, dts 文件很少需要修改, 下面主要讲解 sys\_config 和配置头文件以及环境变量和分区修改。



### 3. sys\_config.fex 文件解析

在 sys\_config.fex 文件中对大部分属性都做了说明。下面主要从**功能需求方面**介绍一些常用属性项。

#### 3.1. 量产和烧录相关属性

##### 3.1.1. eraseflag

```
[platform]
eraseflag = 0
```

当 `eraseflag=1` 时烧录固件时会**格式化除了 private 分区之外的所有分区** (private 分区因为有备份恢复所以不受影响)，在使用 PhoenixSuit 或 LiveSuit 烧录固件时会弹出如下对话框：



图 3-1 PhoenixSuit 烧录对话框

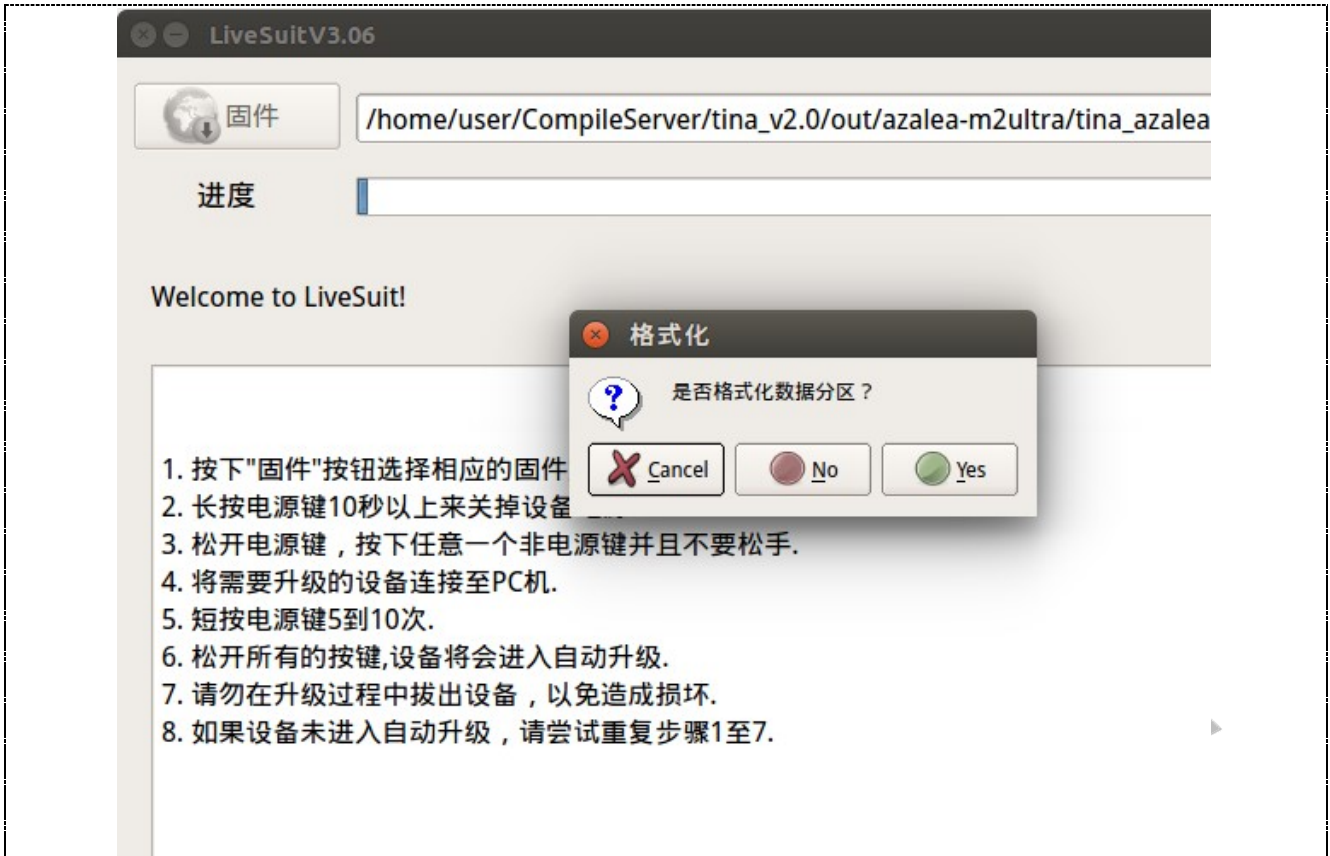


图 2 LiveSuit 烧录对话框

如果选择“是”，`eraseflag` 会自动修改为 1。

当 `eraseflag=0` 时即在烧写之前不会擦除分区；另外，当 `eraseflag=0x11` 时会强制擦除 `private` 分区。一般量产过程，建议置 `eraseflag=1`。

### 3.1.2. burn\_key

```
[target]
burn_key = 1
```

当用 `DragonSN` 或 `DragonKey` 等烧写 SN、MAC 工具进行烧号时，`burn_key` 标志一定要置为 1，另外一定要有 `private` 分区。

## 3.2. Boot 阶段电源管理

### 3.2.1. power\_start

```
[charger0]
power_start = 0
```

目标板供电方式一般有 `battery`、`dcin`、`vbus3` 种，`power_start` 意义如下：

值	意义
0	不允许插 <code>dcin</code> 启动,启动需要满足条件为: 长按 <code>power</code> 键, 前状态为系统状态, 电池电量充足
1	允许插 <code>dcin</code> 启动, 前提是电池电量充足
2	不允许插 <code>dcin</code> 启动, 启动需要满足条件为: 长按 <code>power</code> 键, 前状态为系统状态, 忽略电池电量情况
3	允许插 <code>dcin</code> 启动, 忽略电池电量情况

#### 注意

`power_start` 只有在电池存在的情况下才有意义，在没有电池情况下会直接启动系统



```
[regulator0]
```

AXP 电源管理芯片提供的电源引脚，可为 dram、wifi 等各个模块供电。

### 3.3. 串口调试

#### 3.3.1. debug\_mode

```
[platform]  
debug_mode = 0
```

打印信息开关，0 为关闭，1 为打开。

#### 3.3.2. uart\_debug\_port/uart\_debug\_tx/uart\_debug\_rx

```
[uart_para]  
uart_debug_port = 0  
uart_debug_tx = port:PB22<2><1><default><default>  
uart_debug_rx = port:PB23<2><1><default><default>
```

调试 uboot 时的打印信息可从 **uart0** 或 **card0** 输出，默认是从 **uart0** 输出，若想从 **card0** 口查看打印信息需要执行：

```
pack -d
```

此时 pack 脚本会执行 `uart_switch`，即修改 `sys_config` 中 `[uart_para]` 的 `uart_debug_tx` 和 `uart_debug_rx` 引脚为 `PF02` 和 `PF04`，而且由于这两个引脚被 `sd0` 和 `uart0` 复用，`sd0` 会被设置为 `disable`，`[uart0]` 的 `uart_tx`、`uart_rx` 也会相应修改为 `PF02` 和 `PF04`。

### 3.4. 其余模块配置

其余模块如 `dram`、`usb`、`gmac`、`wlan`、`bt`、`lcd`、`csi`、`pwm`、`hdmi`、`nand`、`mmc`、`spi` 等都有相应配置，可参考 `System Configuration` 相关说明文档，根据需要修改。

## 4. uboot 配置头文件

### 4.1. 地址相关 (base、size、offset)

如 CONFIG\_SYS\_SRAM\_BASE、CONFIG\_SYS\_SDRAM\_BASE、CONFIG\_SYS\_TEXT\_BASE 等基地址，许多其它的 XXX\_ADDR 地址宏都是基于这些基地址，如 boot0、fes 运行在 SRAM 的地址，uboot 运行在 SDRAM 的地址。

另外，还有些 buffer 大小、分区大小、堆栈大小等 size 相关的宏，可根据目标板的硬件特性作相应调整，例如小内存时可以调整 CONFIG\_SYS\_MALLOC\_LEN 大小。

### 4.2. uboot 命令相关

在 uboot 命令行执行：

```
sunxi# ?           #打印出所有可用的命令
sunxi# printenv   #打印所有环境变量
```

在 uboot 代码中都是以 U\_BOOT\_CMD 宏声明和定义。配置头文件中相应的 CONFIG\_CMD\_XXX 宏可以裁剪或是增加 uboot 命令，如增加新的文件系统支持。

### 4.3. 模块开关宏

前面已经提到过，Tina 平台的 uboot 的配置头文件一般有两份，以 R6 为例，有 sun3iw1p1.h 与 sun3iw1p1\_nor.h，这是为了达到同一份固件自适应不同存储介质（emmc、nand、spi nor 等）的目的而设计的，emmc、nand 一般都是大容量，与 spi nor 小容量系统有着不同的配置需求，Tina 在每次 mboot 编译 uboot 时都会先执行：

```
“make sun3iw1p1_config; make”
```

然后执行：

```
“make sun3iw1p1_nor_config; make”
```

第一次 make 会使用配置头文件 sun3iw1p1.h，第二次 make 会使用 sun3iw1p1\_nor.h。

CONFIG\_SUNXI\_MODULE\_XXX 决定开关模块，如 NAND、SDMMC、SPINOR、AXP、SPRITE 等，烧录固件时一定要定义 CONFIG\_SUNXI\_MODULE\_SPRITE。

## 5. 环境变量和分区配置

### 5.1. env.cfg

有三份环境变量，`env-3.4.cfg`，`env-3.10.cfg`，`env-4.4.cfg`。分别为不同版本的 linux 内核准备的。存放路径请参考本文档 2.4 节说明。这些环境变量存在 `env` 分区。

下面对 `env` 中重要的配置项作说明：

配置项	含义
<code>bootdelay</code>	串口选择是否进入 <code>uboot</code> 命令行的等待时间，单位秒。例如：为 0 时自动加载内核，为 3 则等待 3 秒，期间按任何按键都可进入 <code>uboot</code> 命令行
<code>bootcmd</code>	默认为 <code>run setargs_nand boot normal</code> ，但 <code>uboot</code> 会根据实际介质正确修改 <code>setargs_nand</code> ，称为 " <code>update_bootcmd</code> "。另外，在烧录固件时 <code>bootcmd</code> 会被修改成 <code>run sunxi_sprite_test</code> ，即此时不会去加载内核，而是去执行烧录固件命令
<code>setargs_xxx</code>	会去设置 <code>bootargs</code> 、 <code>console</code> 、 <code>root</code> 、 <code>init</code> 、 <code>loglevel</code> 、 <code>partitions</code> ，这些都是内核需要用到的环境变量。其中 <code>partitions</code> 会根据分区进行自适应
<code>boot_normal</code>	正常启动加载内核
<code>boot_recovery</code>	正常启动加载恢复系统
<code>boot_fastboot</code>	正常启动加载 fastboot
<code>console</code>	设置内核的串口
<code>loglevel</code>	设置内核的 log 级别
<code>verify</code>	设置是否校验内核

进入 `uboot` 命令行执行 `env` 相关命令可以查看，修改，保存 `env` 环境变量。

常用的命令如：

```
env print          --打印所有环境变量
env set bootdelay 1  --设置 bootdelay 为 1
env save          --保存环境变量
```

`env set` 之后需要执行 `env save` 才能真正写入 flash 保存。

### 5.2. sys\_partition.fex/sys\_partition\_nor.fex

用户在使用过程中需要调整分区大小或者增减分区时需要修改 `sys_partition.fex/sys_partition_nor.fex`，一个典型的分区如下：

[partition]			
	<code>name</code>	= <code>boot</code>	(详见①)
	<code>size</code>	= <code>8192</code>	(详见②)
	<code>downloadfile</code>	= " <code>boot.fex</code> "	(详见③)
	<code>user_type</code>	= <code>0x8000</code>	

详注：

① 分区名

② 大小为扇区个数，一个扇区大小为 512Bytes，因此 `boot` 分区大小为 `8192*512=4MB`。修改分区大小的时候，尽量对齐到所用介质的块大小。

③ `downloadfile` 决定打包固件时分区的镜像文件，通常该镜像（例 `boot.fex`）已经被格式化成文件系统（如 `fat`），这样在加载镜像时可以使用 `fatload` 去加载分区里面的镜像文件

典型的分区有：

分区	作用
boot-res	存 boot_logo
env	存环境变量
boot	存内核
rootfs	存根文件系统
rootfs_data	存用户数据
recovery	存恢复系统（详见 OTA 相关文档）
private	存 SN、MAC 等敏感信息，烧录固件时不会丢失
extend	存/usr 目录（详见 OTA 相关文档）
misc	存系统状态和命令（详见 OTA 相关文档）
UDISK	大容量存储

需要注意的是所有这些分区大小总和不能超过实际介质的大小。



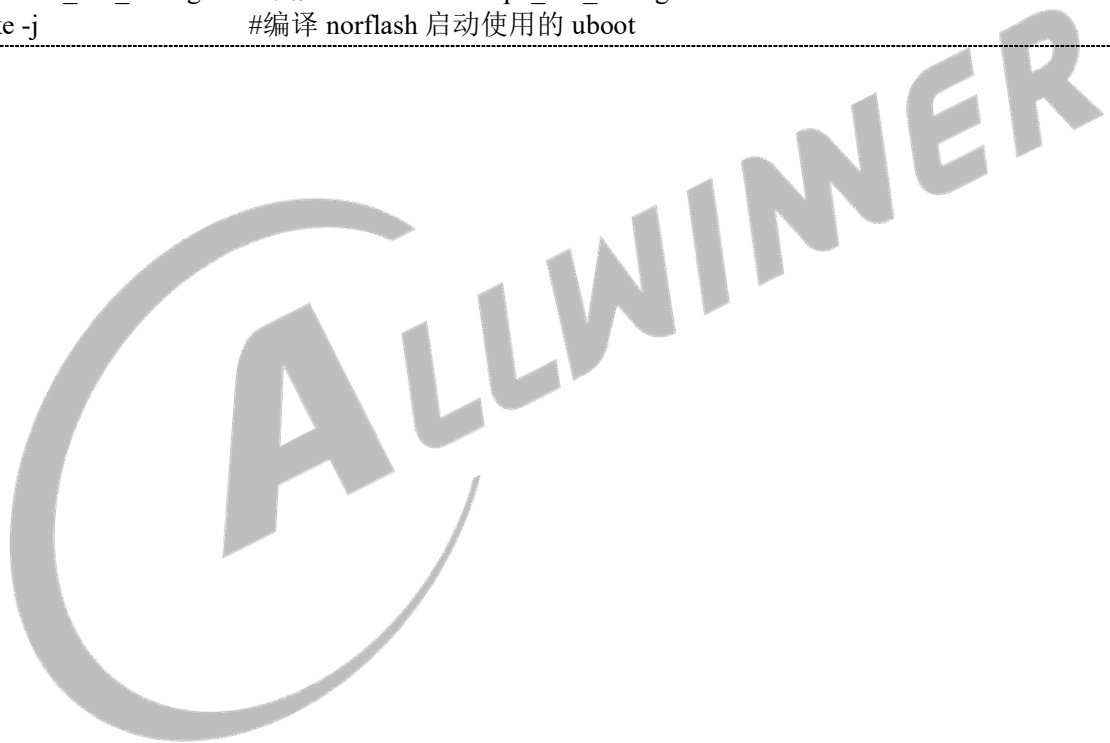
## 6. uboot 编译

在 tina 目录下即可编译 uboot

```
source build/envsetup.sh
lunch xxx
muboot
```

另一种编译方式是，进入 uboot 目录编译

```
source build/envsetup.sh
lunch xxx
cboot #跳转到 uboot 目录，也可以使用 cd lichee/brandy/u-boot-xxx/
make xxx_config #例如 make sun3iw1p1_config
make -j #编译 uboot
make xxx_nor_config #例如 make sun3iw1p1_nor_config
make -j #编译 norflash 启动使用的 uboot
```



## 7. 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.

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.

