



Linux TVE 开发指南

版本号: 1.0
发布日期: 2021.2.03

版本历史

版本号	日期	制/修订人	内容描述
1.0	2021.2.03		1. 创建该文档



目 录

1 概述	1
1.1 编写目的	1
1.2 使用范围	1
1.3 相关人员	1
2 模块描述	2
2.1 TV-OUT 基本知识	2
2.2 VGA 接口 (Video Graphics Array)	2
2.3 复合视频接口 (或称 RCA/AV/CVBS 接口)	2
2.4 S 端子接口 (也称 Separate Video)	3
2.5 色差分量接口	4
2.6 AllWinner 的 tv_out 的硬件架构	5
3 驱动初始化	7
3.1 tv_top	7
3.2 tve:	7
4 设备树配置	9
4.1 tve 节点配置	9
4.2 设置启动后自动使能 cvbs 输出	10
5 移植调试常见问题	11
5.1 时钟问题	11
5.2 probe 阶段 get alias failed	12
5.3 de_feat.h 配置	13
5.4 IP 选择 (Makefile 配置)	13
5.5 驱动内部配置	13
5.6 EXTCON 配置	13
5.7 Audio 输出	14
5.8 背景绿色问题	14
5.9 颜色偏白	14
6 调试手段	15

插 图

图 2-1	D-Sub 接口	2
图 2-2	AV 接口	2
图 2-3	AV 接口	3
图 2-4	S 端子接口	3
图 2-5	色差分量接口	4
图 2-6	硬件架构图	5
图 2-7	对外引脚图 (V40)	5
图 2-8	对外引脚图 (V40)	6
图 2-9	对外引脚图 (V40)	6
图 2-10	对外引脚图 (V40)	6
图 2-11	对外引脚图 (V40)	6
图 2-12	GPIO	6
图 2-13	GPIO	6
图 4-1	S 端子接口	10
图 5-1	数据源选择	11
图 5-2	时钟源选择	11
图 5-3	相关代码	12

1 概述

1.1 编写目的

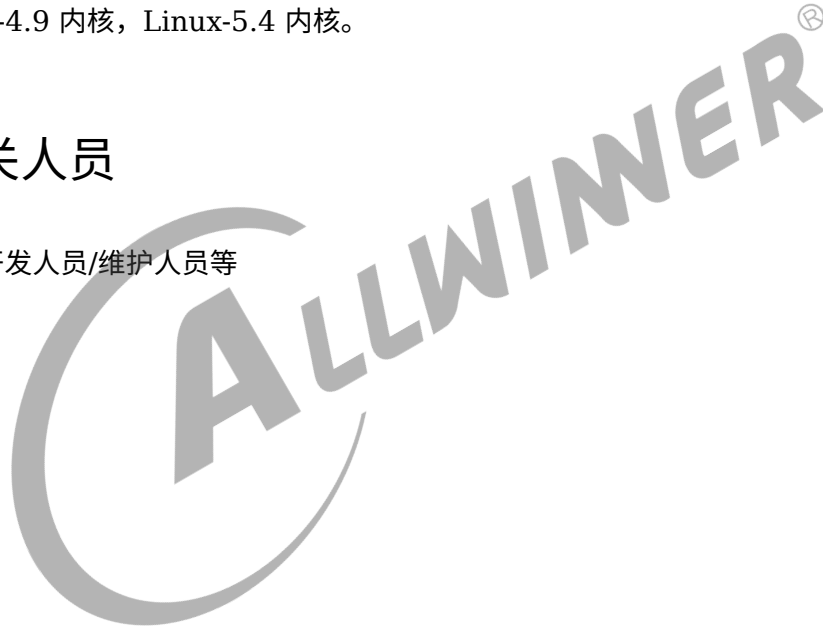
介绍 TVE 模块接口原理与使用方法。

1.2 使用范围

适用于 Linux-4.9 内核，Linux-5.4 内核。

1.3 相关人员

TVE 驱动的开发人员/维护人员等



2 模块描述

2.1 TV-OUT 基本知识

- 定义

将模拟视频信号从显卡输出到电视的相关接口。

2.2 VGA 接口 (Video Graphics Array)[®]



图 2-1: D-Sub 接口

VGA 接口，即 D-Sub 接口。有些显卡虽然只有 DVI 口，但是可以通过 DVI 转 VGA 转接头转换成 VGA 信号

特点：

- 常用与计算机设备与外部显示设备相连接；
- R、G、B 彩色分量信号和扫描同步信号 HSYNC、VSYNC，这五个信号量分离传输；

2.3 复合视频接口 (或称 RCA/AV/CVBS 接口)



图 2-2: AV 接口



图 2-3: AV 接口

特点：

- 实现视频和音频分离传输，避免了因为音/视频混合干扰而导致的图像质量下降；

缺点：

- 该接口传输的信号是一种亮度/色度（Y/C）混合的视频信号，仍然需要显示设备对其进行亮/色分离和色度解码才能成像，这种先混合再分离的过程必然会造成色彩信号的损失，色度信号和亮度信号也会有很大的机会相互干扰，从而影响最终输出的图像质量。

2.4 S 端子接口（也称 Separate Video）



图 2-4: S 端子接口

特点：

- 实现光亮度与色度信号分离传输；

缺点：

- 该接口将两路色差信号混合为一路色度信号 C 进行传输，然后再在显示设备内解码进行处理，这样多少仍会带来一定信号损失而产生失真（这种失真很小），而且由于混合导致色度信号的带宽也有一定的限制。S-Video 虽不是最好的，但考虑到目前的市场状况和综合成本等其它因素，它还是应用最普遍的视频接口。

2.5 色差分量接口



图 2-5: 色差分量接口

特点：

色差分量（Component）接口采用 YPbPr 和 YCbCr 两种标识，前者表示逐行扫描色差输出，后者表示隔行扫描色差输出。色差分量接口一般利用 3 根信号线分别传送亮色和两路色差信号。这 3 组信号分别是：亮度以 Y 标注，以及从三原色信号中的两种——蓝色和红色——去掉亮度信号后的色彩差异信号，分别标注为 Pb 和 Pr，或者 Cb 和 Cr，在三条线的接头处分别用绿、蓝、红色进行区别。这三条线如果相互之间插错了，可能会显示不出画面，或者显示出奇怪的色彩来。色差分量接口是模拟接口，支持传送 480i/480p/576p/720p/1080i/1080p 等格式的视音频信号，本身不传输音频信号。

2.6 AllWinner 的 tv_out 的硬件架构

Figure 1-1 shows a block diagram of the TVE_TOP.

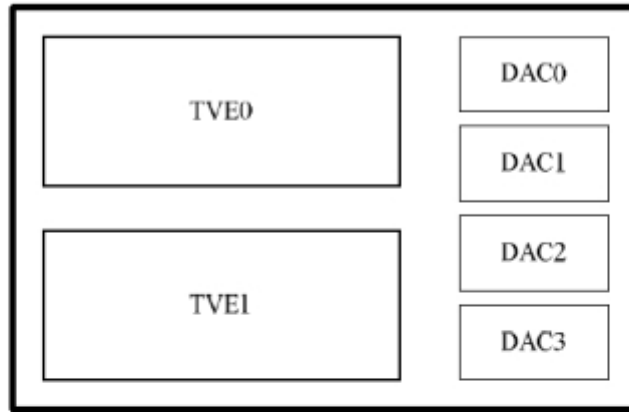


Figure 1-1. TVE Block Diagram

Figure 1-2 shows a block diagram of detail TVE.

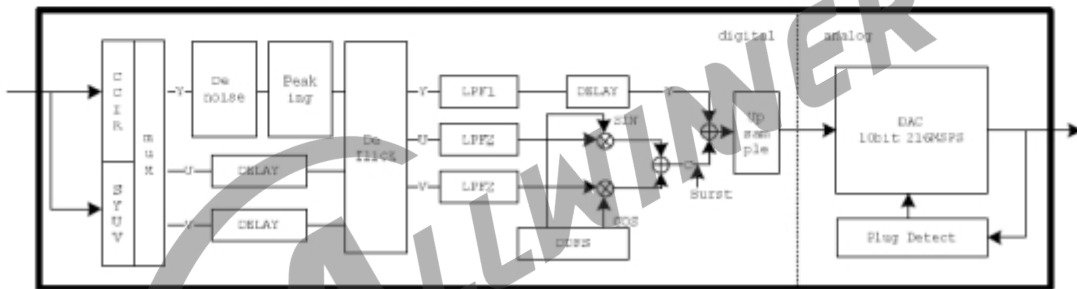


Figure 1-2. TVE Block Diagram

图 2-6: 硬件架构图

硬件框架分析：

1. 从 spec 给出的框架图来看，整个 tve 系统包含三个设备：tve_top、tve、dac
2. 注意，一个比较容易被混淆的地方是每个 tve 输出到 dac 的通道有 1~3 个（不同的 tve 有不同的输出通道数），如 V40，tve0 的输出通道只有一个 dac_src0，而 tve1 就有三个输出通道 dac0、dac1、dac2。tve 和 dac 之间可以任意组合。



图 2-7: 对外引脚图 (V40)



图 2-8: 对外引脚图 (V40)

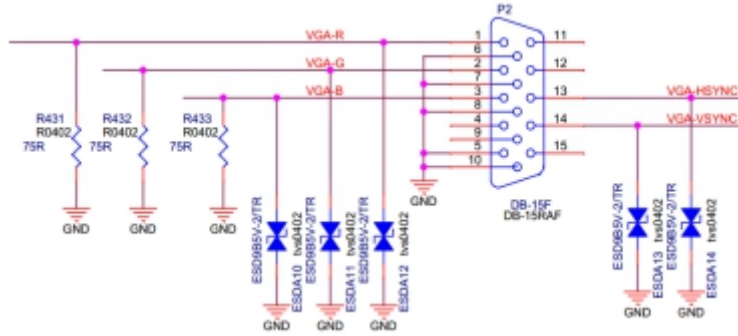


图 2-9: 对外引脚图 (V40)

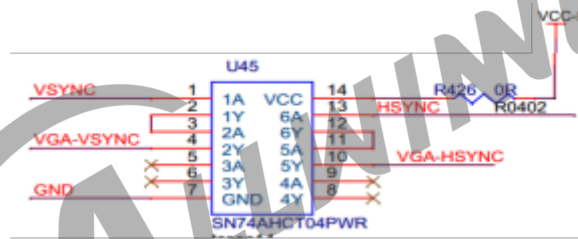


图 2-10: 对外引脚图 (V40)



图 2-11: 对外引脚图 (V40)



图 2-12: GPIO



图 2-13: GPIO

3 驱动初始化

由于 tv 模块在 dts 中可以配置多个节点（如 tv0, tv1, ...），但是这些节点都使用的是同一个 compatible。因此 tve 驱动的 probe 函数会进入多次。

为了区分解析后的不同 tv 节点的配置，使用了 of_alias_get_id 来获取不同 tv 节点 id 来区分，根据 of_alias_get_id 的源码可以知道，id 的值就等于 tv 节点所携带的阿拉伯数字的值（如 tv0 节点，获取到的 id 就是去字符串“tv0”的‘0’字符，再将其转换为 int 类型）。

3.1 tv_top

初始化内容：

获取时钟，不需要指定父时钟（在使用的时候设置开关，不用设置频率）；

获取地址映射；

特点：一般在一个平台中，tve 可以有很多个，但 tv_top 设备只有一个，因此其相关的配置在驱动中不需用 id 进行标识；

3.2 tve:

注意：一般在一个平台中，tve 可以有很多个，因此其相关的配置在驱动中必须用 id 进行标识；

初始化内容：

- (1) 地址映射、时钟获取（需要指定父时钟）；
- (2) 从 sys_config 的配置中获取输出接口类型（interface）、dac 的数据源及其映射的 dac；
- (3) 注册供热插拔上报的 switch；
- (4) 使能 tv_top 的时钟；
- (5) 使能 tv_out 的 regulator 电源
- (6) 根据驱动里面配置的 DAC_COUNT 去轮询本 dts 节点中配置的 dac 输出通道和 dac 输出类型；

根据 dac 通道获取 cali_value 和 offset（这两个的用处????）；

(6) 检查 boot 部分，通过 boot_disp 获取 uboot 时配置的视频输出类型和视频格式，如果是 tv 或者 VGA 输出，则配置驱动里面的 enable 标志置 1，tv_mode 变量置为 uboot 时的视频格式，否则 enable 标志保持置 0，tv_mode=DISP_TV_MOD_PAL

(7) 根据 tv_mode 去设置 tve 的 clk 的频率，并使能时钟

首先要通过查表获取 tv_mode 对应的 pixel clock rate，将此频率作为 tve 的时钟频率；

然后判断一下在此时钟频率下，tve clk 的父时钟能否取到合适的频率，如果取不到，则设置为最高频率 984MHz；

然后判断一下，在目前的父时钟频率下能否取到合适的 tve 的 clk 的频率，如果取不到则重新设置父时钟的频率。

(8) tve 底层初始化：

注意：底层的代码中所有的函数都兼容 tve0、tve1 等，并没有出现有某个 tve 单独抽出来的情况，这是 tve 的底层代码编写的基本思想；

从其全局变量来看，tve 的底层只需知道 tve 和 dac 即可。

参数：tve 对象（表明是要配置 tve0 还是 tve1）、dac_src 及其对应的 dac、dac 的 cali 和 offset 值、dac_type、dac_src 的个数。

动作流程：

1. 配置每个 dac_src 对应的 tve 和 dac_map;
2. REF_EXT_SEL 置成 enable 状态、REF2_SEL 置成 0.4v、REF1_SEL 置成 1.0v;
3. DAC clock invert 置 1，设置 low_bias，BIAS_EXT_SEL 置 1；
4. 配置每一个 dac_src 输出的视频信号类型；

(9) 开启 dac 的插拔自动检测。

1. 开启 tv clk 的 gating，设置检测的脉冲数、脉冲开始计数值、脉冲的周期，每个 dac 的 de_bounce_time 设置成 9，每个 dac 的自动检测使能；
2. 创建监控线程。

4 设备树配置

4.1 tve 节点配置

需要在 board.dts 中添加 tv0/1 节点，并设置以下关键属性：

(1) interface: interface type,1<->cvbs,2<->YPBPR,4<->SVIDEO tve 接口类型，具体详见模块描述；

(2) dac_src0/dac_src1/dac_src2/dac_src3: 此 TVE 将数据输出到哪个 dac 接口

(3) dac_type0/dac_type1/dac_type2/dac_type3: 设置相应的 dac 输出什么样的数据类型，这些数据类型有，

0: composite, rgb 交叉混合信号，用于一转三或者三对三的 cvbs (av) 线输出；



1: luma, 亮度分量输出，用于 S-Video 类型接口

2: chroma, 色度分量输出，用于 S-Video 类型接口



图 4-1: S 端子接口

3: reserved

4: y/green: 用于 YPbPr 类型接口的 Y 分量的输出；或者用于三对三 CVBS(AV) 线的 green 分量的输出

5: u/pb/blue: 用于 YPbPr 类型接口的 Pb 分量的输出；或者用于三对三 CVBS(AV) 线的 blue 分量的输出

6: v/pr/red: 用于 YPbPr 类型接口的 Pr 分量的输出；或者用于三对三 CVBS(AV) 线的 red 分量的输出

4.2 设置启动后自动使能 cvbs 输出

在 board.dts 中设置 disp 节点中设置下面的属性：

screen0/1_output_type

screen0/1_output_mode

5 移植调试常见问题

5.1 时钟问题

(1) 注意 tcon_tv 的 clock 的源是可选的，可以通过设置 disp_top 模块来实现：

•6.1. TCON_TV Clock Select And RGB Source Select Register (Default Value: 0x0000_0000).

Offset: 0x0000			Register Name: TV_CLK_SRC_RGB_SRC
Bit	Read/Write	Default/Hex	Description
31:29	l	l	l
28	R/W	0x0	RGB3_SRC_SEL RGB3 Source Select 0:LCD3 1:TV3
27:25	l	l	l

Copyright © Allwinner Technology Co., Ltd. All Rights Reserved.

Page 8

图 5-1: 数据源选择

3:1	l	l	l: clock from CVBSOUT1
0	R/W	0x0	TCON_TV0_CLK_SRC TCON TV0 Clock Source 0: clock from CCMU 1: clock from CVBSOUT0

图 5-2: 时钟源选择

当 tcon_tv 的时钟源选择 cvbsout0 时，在设置时钟频率时只需要设置 cvbs 的时钟频率即可，tcon_tv clk 只需要进行使能、bus clk 使能、reset dessert 即可。

```

101
102 /* @sel: the index of timing controller
103 * @en: enable clock or not
104 */
105 s32 tcon1_tv_clk_enable(u32 sel, u32 en)
106 {
107     unsigned int tcon_index = -1;
108     if (sel >= DEVICE_NUM)
109         return -1;
110
111     if (!de_feat_get_tcon_type(sel))
112         return -1;
113
114     tcon_index = de_feat_get_tcon_index(sel);
115     if (tcon_index < 0)
116         return -1;
117
118     if (tcon_index == 0) {
119         lcd_top[0]->tcon_tv_setup.bits.tv0_clk_src = TV_CLK_F_TVE;
120         lcd_top[0]->tcon_clk_gate.bits.tv0_clk_gate = en;
121     } else if (tcon_index == 1) {
122         lcd_top[0]->tcon_tv_setup.bits.tv1_clk_src = TV_CLK_F_TVE;
123         lcd_top[0]->tcon_clk_gate.bits.tv1_clk_gate = en;
124     }
125
126     return 0;
127 }

```

图 5-3: 相关代码

(2)tve 时钟频率与 hdmi 时钟频率的区别

hdmi 的图像数据优于是数字化的，因此其像素时钟频率的计算方式为 $v_total * h_total * fps$ ，例如 720x480p60 为 27MHZ；

但 tve 的像素时钟主要左右再 DAC 模块上，因此不能用数字化的公式来进行计算，必须有 DAC 模块来决定。

5.2 probe 阶段 get alias failed

对应打印：

```
pdev->id = of_alias_get_id(pdev->dev.of_node, "tv");
```

```
if (pdev->id < 0) {
```

```
    TV_DBG("failed to get alias id");
```

```
    return -EINVAL;
```

```
}
```

原因：dts 中，alias {} 中没有加入 tv0/1

5.3 de_feat.h 配置

要在 `drivers/video/fbdev/sunxi/disp2/disp/de/lowlevel_v2x/de_feat.h` 中对应位置添加 `SUPPORT_TV`，否则 `display2` 驱动在设置 `cvbs` 输出时会发生错误，设置无法到达 `disp_tv.c` 和 `drv_tv.c`。

5.4 IP 选择 (Makefile 配置)

需要查询 `spec` 文档才能确定在 `drivers/video/fbdev/sunxi/disp2/tv/Makefile` 中选择哪一个 IP

5.5 驱动内部配置

需要根据具体情况设置：`drivers/video/fbdev/sunxi/disp2/tv/de_tvec.h`

```
#define TVE_TOP_SUPPORT //是否有 tve top 模块
#define TVE_DEVICE_NUM 1 //一共有几个 tv out 输出口
#define TVE_TOP_DEVIVE_NUM 1 //一共有几个 tve top 模块
#define TVE_DAC_NUM 1 //DAC 个数
```

5.6 EXTCON 配置

作用：`tve` 模块利用 `extcon` 机制来上报热插拔事件，可以在 `/sys/class/extcon` 中手动查询 `cvbs` 的插拔状态。

(1) `./buildsh menuconfig`(在 `tina` 中为 `make kernel_menuconfig`) 打开 `CONFIG_EXTCON`

(2) `include/linux/extcon.h` 中添加

```
#define EXTCON_DISP_CVBS 46
#define EXTCON_DISP_TVD 47
#define EXTCON_DISP_EDP 48
```

5.7 Audio 输出

不支持

5.8 背景绿色问题

关闭 CEU，即关闭 tcon 内部 color 转换。

drivers/video/fbdev/sunxi/disp2/disp/de/lowlevel_v2x/de_feat.h

打开 BYPASS_TCON_CEU

关键寄存器：TV_CEU_CTL_RE0

5.9 颜色偏白

DE0 和 DE1 不能同时选中一个 tcon。如果 DE0 连接到 TCON_TV0，则 DE1 就不能连接到 TCON_TV0，否则 TVE 或者 HDMI 的输出可能会变白。

6 调试手段

(1) 检测 sys 节点，重点检查帧率

```
cat /sys/class/disp/disp/attr/sys
```

(2) 检测 clk

进入 debugfs 目录

```
cat clk/clk_summary
```

(3) 检查 tcon_tv/tve top/tve 的寄存器






著作权声明

版权所有 © 2022 珠海全志科技股份有限公司。保留一切权利。

本档及内容受著作权法保护，其著作权由珠海全志科技股份有限公司（“全志”）拥有并保留一切权利。

本档是全志的原创作品和版权财产，未经全志书面许可，任何单位和个人不得擅自摘抄、复制、修改、发表或传播本档内容的部分或全部，且不得以任何形式传播。

商标声明

、、**全志科技**、（不完全列举）均为珠海全志科技股份有限公司的商标或者注册商标。在本档描述的产品中出现的其它商标，产品名称，和服务名称，均由其各自所有人拥有。

免责声明

您购买的产品、服务或特性应受您与珠海全志科技股份有限公司（“全志”）之间签署的商业合同和条款的约束。本档中描述的全部或部分产品、服务或特性可能不在您所购买或使用的范围内。使用前请认真阅读合同条款和相关说明，并严格遵循本档的使用说明。您将自行承担任何不当使用行为（包括但不限于如超压，超频，超温使用）造成的不利后果，全志概不负责。

本档作为使用指导仅供参考。由于产品版本升级或其他原因，本档内容有可能修改，如有变更，恕不另行通知。全志尽全力在本档中提供准确的信息，但并不确保内容完全没有错误，因使用本档而发生损害（包括但不限于间接的、偶然的、特殊的损失）或发生侵犯第三方权利事件，全志概不负责。本档中的所有陈述、信息和建议并不构成任何明示或暗示的保证或承诺。

本档未以明示或暗示或其他方式授予全志的任何专利或知识产权。在您实施方案或使用产品的过程中，可能需要获得第三方的权利许可。请您自行向第三方权利人获取相关的许可。全志不承担也不代为支付任何关于获取第三方许可的许可费或版税（专利税）。全志不对您所使用的第三方许可技术做出任何保证、赔偿或承担其他义务。