



Unisoc Confidential For hiar

Kernel 4.14 LCD 客制化指导手册

文档版本
发布日期

V1.3
2020-09-21

版权所有 © 紫光展锐（上海）科技有限公司。保留一切权利。

本文件所含数据和信息都属于紫光展锐（上海）科技有限公司（以下简称紫光展锐）所有的机密信息，紫光展锐保留所有相关权利。本文件仅为信息参考之目的提供，不包含任何明示或默示的知识产权许可，也不表示有任何明示或默示的保证，包括但不限于满足任何特殊目的、不侵权或性能。当您接受这份文件时，即表示您同意本文件中内容和信息属于紫光展锐机密信息，且同意在未获得紫光展锐书面同意前，不使用或复制本文件的整体或部分，也不向任何其他方披露本文件内容。紫光展锐有权在未经事先通知的情况下，在任何时候对本文件做任何修改。紫光展锐对本文件所含数据和信息不做任何保证，在任何情况下，紫光展锐均不负任何与本文件相关的直接或间接的、任何伤害或损失。

请参照交付物中说明文档对紫光展锐交付物进行使用，任何人对紫光展锐交付物的修改、定制化或违反说明文档的指引对紫光展锐交付物进行使用造成的任何损失由其自行承担。紫光展锐交付物中的性能指标、测试结果和参数等，均为在紫光展锐内部研发和测试系统中获得的，仅供参考，若任何人需要对交付物进行商用或量产，需要结合自身的软硬件测试环境进行全面的测试和调试。

Unisoc Confidential For hiar

紫光展锐（上海）科技有限公司



前言

概述

本文档主要介绍了显示模组以及基于 Kernel 4.14 配置添加新的显示模组的方法。

读者对象


本文档适用于需要在紫光展锐最新的显示框架 DRM 上添加新的显示模组的客户。

缩略语

缩略语	英文全名	中文解释
DRM	Direct Rendering Manager	Linux 图形渲染框架
DPU	Display Processing Unit	显示控制器
LCD	Liquid Crystal Display	显示模组

符号约定

在本文中可能出现下列标志，它所代表的含义如下。

符号	说明
 说明	用于突出重要/关键信息、补充信息和小窍门等。 “说明”不是安全警示信息，不涉及人身、设备及环境伤害。

变更信息

文档版本	发布日期	修改说明
V1.0	2019-10-08	文档创建
V1.1	2019-12-26	适用产品信息增加 UIS8581E\SL8541E 平台
V1.2	2020-03-17	文档名修改、格式更新等

文档版本	发布日期	修改说明
V1.3	2020-09-21	更新模板

关键字

LCD

Unisoc Confidential For hiar

目 录

1 概览.....	1
2 Uboot 新屏配置.....	2
2.1 LCD Uboot 配置介绍	2
2.2 LCD driver 文件配置.....	2
2.2.1 添加驱动文件	2
2.2.2 修改驱动文件	2
2.3 LCD 添加编译规则	4
2.4 LCD 编译选项配置	4
3 Kernel 新屏配置.....	6
3.1 LCD Kernel 配置介绍	6
3.2 LCD DTS 文件配置.....	6
3.2.1 添加 dts 文件	6
3.2.2 修改 dts 文件	6
3.3 Kernel 背光配置	11
3.3.1 LCD 背光配置	11
3.3.2 OLED 背光配置	12
4 注意事项.....	13

图目录

图 1-1 Display 控制流程	1
图 2-1 uboot 配置	2
图 2-2 initial 数据格式	3
图 2-3 panel info 信息格式	3
图 2-4 panel_op 回调接口	3
图 2-5 注册 panel driver	4
图 2-6 supported panel 实例化	4
图 2-7 lcd 控制宏	5
图 3-1 kernel 配置	6
图 3-2 initial 数据格式	7
图 3-3 dts 配置	8

Unisoc Confidential For hiar

表目录

表 3-1 dtssi 文件属性说明.....	8
表 3-2 LCD 模组相关的属性说明	10
表 3-3 OLED 背光属性说明.....	12

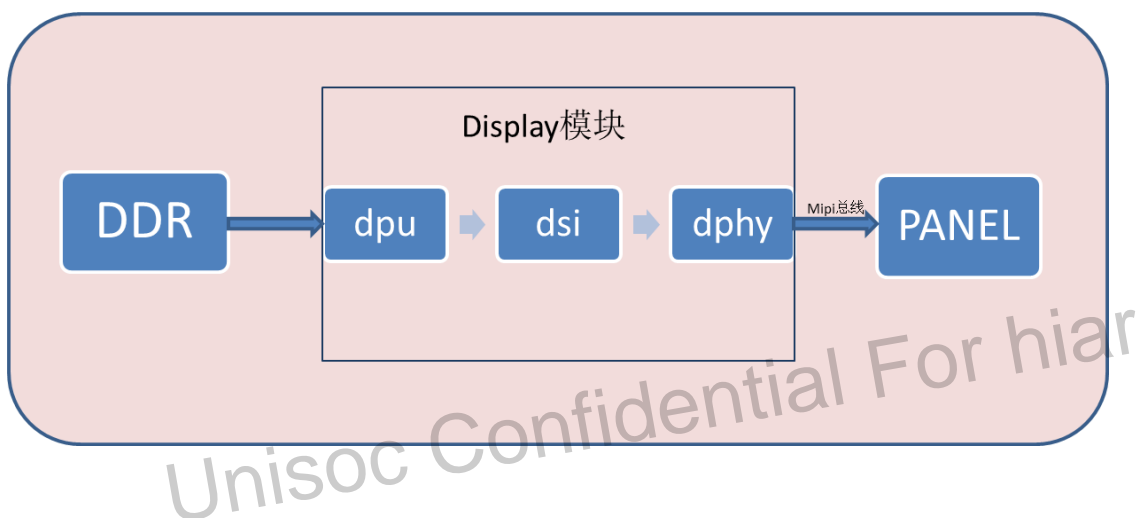
Unisoc Confidential For hiar

1 概览

以 UMS312 平台为例，display 模块包括 3 大块：dpu、dsi、dphy。核心模块 dpu 与显示相关，主要负责图层合成、图像变换、刷新屏幕等。dsi 和 dphy 与通讯有关，负责打包和传送数据。

上层程序将需要显示的图像绘制好并写到 DDR 的 buffer 中，然后 dpu 从 buffer 中取数据进行合成后送往 dsi 中进行打包，dphy 将打包好的数据通过 mipi 总线送往 panel 进行显示，如图 1-1。

图1-1 Display 控制流程



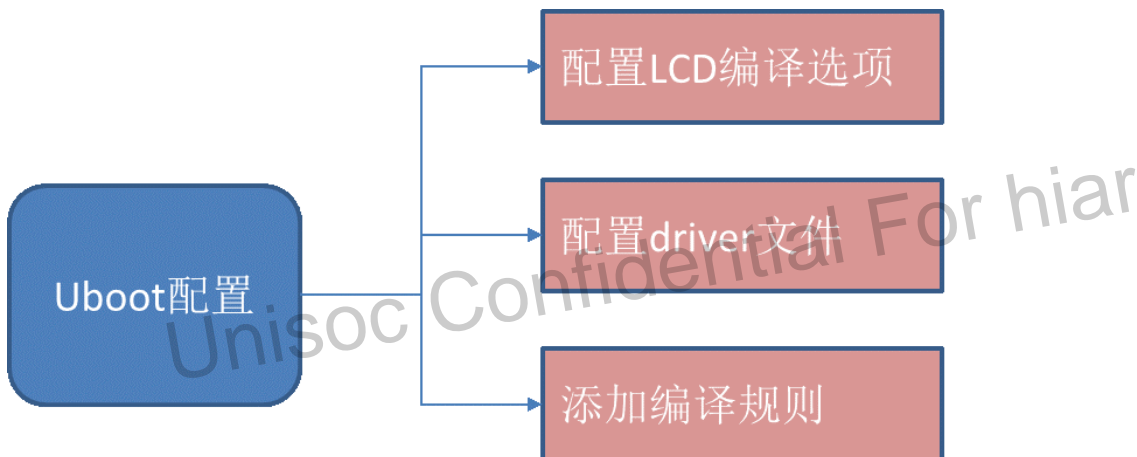
2 Uboot 新屏配置

Uboot 中关于 LCD 的配置和 Android 9.0 之前的版本基本一致，所以也可以参阅展锐在 SC9863A/SC7731E/SC9832E 等平台上 Android 9.0 之前的 LCD 客制化指导文档。

2.1 LCD Uboot 配置介绍

Uboot 部分的 LCD 控制，如图 2-1 所示，主要包括 3 个部分，配置 LCD 驱动文件、配置 LCD 编译选项、添加编译规则。下面以 ums312_1h10 这块 board 为例，添加 r61350 显示模组。

图2-1 uboot 配置



2.2 LCD driver 文件配置

2.2.1 添加驱动文件

Copy 已有驱动文件，并且按照以下命名规范重新命名 LCD 驱动文件：
`lcd_[DriverIC]_[ModuleVendor]_[BusType]_[Resolution]`

例如：u-boot15/drivers/video/sprd/lcd/lcd_r61350_truly_mipi_hd.c

2.2.2 修改驱动文件

步骤 1 将驱动文件中原 DDIC 名全部重新命名。原名：ili9881_xxx -> 新名：r61350_xxx。

步骤 2 配置初始化 code（由屏幕模组厂提供）为如图 2-2 格式。

图2-2 initial 数据格式

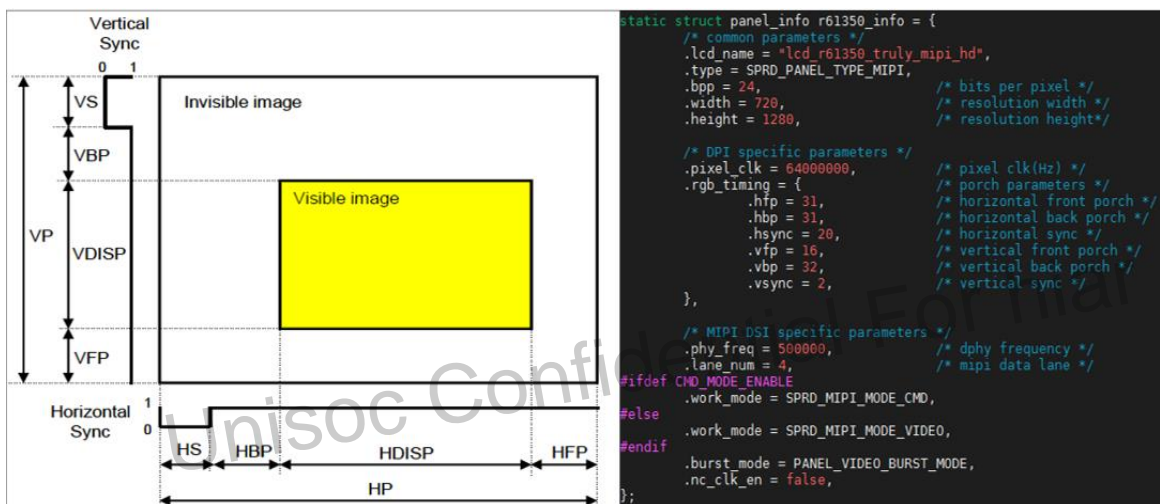
```
static uint8_t init_data[] = {  
    //Host I/F Setting  
    0x15, 0x00, 0x00, 0x02, 0x80, 0x00,  
    0x15, 0x00, 0x00, 0x02, 0xCC, 0x04,  
    0x15, 0x00, 0x00, 0x02, 0xE3, 0x01,  
  
    //DSI control  
    0x39, 0x00, 0x00, 0x03, 0x86, 0x61,  
  
    //display setting  
    0x39, 0x00, 0x00, 0x07, 0xC0, 0x23,  
  
    //display h-timing setting  
    0x39, 0x00, 0x00, 0x17, 0xC1, 0x08,  
};
```

初始化code在mipi_dsi_send_cmds函数中将强制转换为dsi_cmd_desc结构体。

```
struct dsi_cmd_desc {  
    uint8_t data_type; //mipi包的类型  
    uint8_t wait;      //延时时间(ms)  
    uint8_t wc_h;      //数据包长度高8位  
    uint8_t wc_l;      //数据包长度低8位  
    uint8_t payload[]; //data数据  
};
```

步骤3 配置 panel_info 结构体（参数由屏幕模组厂提供），如图 2-3。

图2-3 panel info 信息格式



说明

lcd_name: lcd 节点名，必须和 kernel 中的 lcd dts 节点名称保持一致，否则 kernel 无法匹配到正确的 lcd 驱动。

步骤4 实现 panel_ops 回调函数，如图 2-4。

图2-4 panel_op 回调接口

```
static struct panel_ops r61350_ops = {  
    .init = r61350_init, /* r61350初始化函数 */  
    .read_id = r61350_readid, /* r61350读取panel id函数 */  
    .power = r61350_power, /* r61350上电和复位函数 */  
};
```

步骤5 注册 panel driver，如图 2-5。

图2-5 注册 panel driver

```
struct panel_driver r61350_truly_driver = {
    .info = &r61350_info,
    .ops = &r61350_ops,
};
```

----结束

2.3 LCD 添加编译规则

步骤 1 找到并打开驱动文件所在目录的 Makefile: u-boot15/drivers/video/sprd/lcd/Makefile。

步骤 2 添加编译规则。

```
obj-$(CONFIG_LCD_R61350_TRULY_MIPI_HD) += lcd_r61350_truly_mipi_hd.o
```

其中 CONFIG_LCD_R61350_TRULY_MIPI_HD 在 board 头文件中定义。

步骤 3 找到并打开屏幕配置头文件: u-boot15/drivers/video/sprd/lcd/panel_cfg.h。

步骤 4 添加结构体声明。

```
extern struct panel_driver r61350_truly_driver;
```

其中 nt35532_truly_driver 结构体在 driver 文件中定义。

步骤 5 添加 panel 结构体, 如图 2-6。

图2-6 supported panel 实例化

<pre>static struct panel_cfg supported_panel[] = { #ifdef CONFIG_LCD_R61350_TRULY_MIPI_HD { .lcd_id = 0x61350, .drv = &r61350_truly_driver, }, #endif };</pre>	<p>CONFIG_LCD_R61350_TRULY_MIPI_HD —— 用于添加屏幕到uboot的遍历中, 在board头文件中定义</p> <p>.lcd_id —— 为自定义的屏幕id, 和DDIC的型号保持一致</p> <p>.drv —— 为panel驱动结构体指针, 指向driver结构体</p>
----------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

将驱动结构体关联到 supported_panel 中。在开机时, uboot 会枚举此结构体中的屏幕, 直到某个 driver 的 readid 函数返回成功。

----结束

2.4 LCD 编译选项配置

LCD Driver 文件添加和配置完成后, 就可以在 board 配置文件中打开添加的新的 LCD 模组了, 以 UMS312 平台为例:

- 编译选项通常放在 uboot 的 board 配置头文件中

u-boot15/include/configs/ums312_1h10.h

- 编辑 panel 配置信息

添加配置宏: #define CONFIG_LCD_R61350_TRULY_MIPI_HD

图2-7 lcd 控制宏

<pre>/* display config */ #define CONFIG_LCD #define CONFIG_CMD_BMP #define CONFIG_SPLASH_SCREEN #define CONFIG_SC2703_LCD_POWERON #define CONFIG_VIDEO_SPRD #define CONFIG_DPU_LITE_R2P0 #define CONFIG_DSIH_SPRD_CTRL_RXP0 #define CONFIG_DPHY_SPRD_SHARKLS #define CONFIG_LCD_R61350_TRULY_MIPI_HD #define CONFIG_LCD_ILI988IC_TRULY_MIPI_HD #define CONFIG_LCM_GPIO_RSTN 50 #define CONFIG_SYS_WHITE_ON_BLACK #define LCD_BPP LCD_COLOR32</pre>	<p>CONFIG_LCD_R61350_TRULY_MIPI_HD —— Panel, 添加新Panel时必须添加此项, 对应drivers/video/sprd/lcd/Makefile中obj-\$(CONFIG_LCD_R61350_TRULY_MIPI_HD) += lcd_r61350_truly_mipi_hd.o</p> <p>CONFIG_LCM_GPIO_AVDDEN 4 —— AVDD供电引脚 (按照需要配置)</p> <p>CONFIG_LCM_GPIO_AVEEEN 5 —— AVEE供电引脚 (按照需要配置)</p> <p>CONFIG_LCM_GPIO_RSTN 50 —— Reset引脚 (按照需要配置)</p> <p>LCD_BPP LCD_COLOR32 —— BPP为32, 为内存中的像素位宽, 即一个像素占32bit(非LCD总线上的位宽)</p>
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

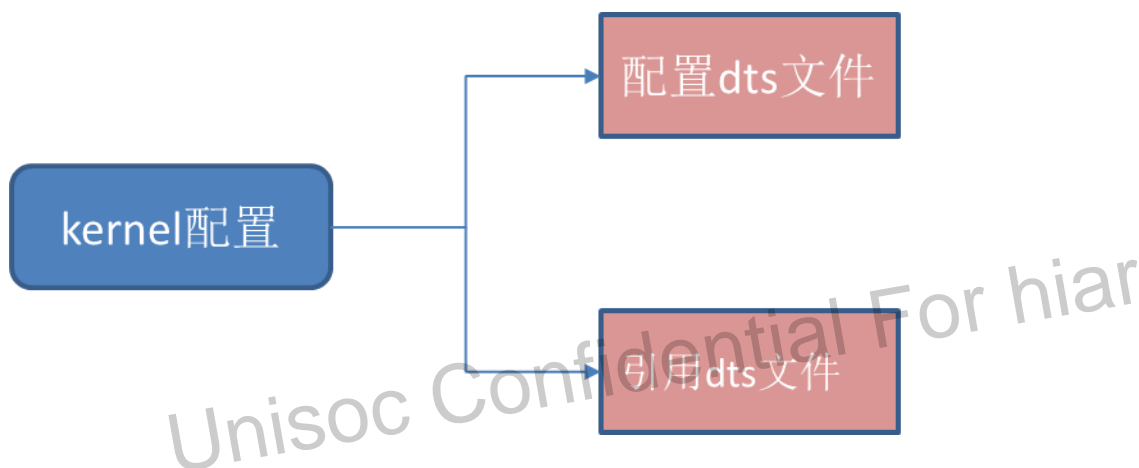
Unisoc Confidential For hiar

3 Kernel 新屏配置

3.1 LCD Kernel 配置介绍

LCD Kernel 的配置适用于紫光展锐最新的显示框架 DRM，Android 新版本逐渐从 ADF 框架迁移到 DRM 上，从内核 4.14 开始，UNISOC 的显示框架逐渐迁移到 DRM 上。目前 panel 通过 lcd dts 来实现客制化，使用统一的 sprd generic panel 驱动，暂不支持客制化 driver 的修改。

图3-1 kernel 配置



3.2 LCD DTS 文件配置

3.2.1 添加 dts 文件

Copy 已有驱动文件，并且按照以下命名规范重新命名 LCD 驱动文件：
lcd_[DriverIC]_[ModuleVendor]_[BusType]_[Resolution]

例如：kernel/arch/arm64/boot/dts/sprd/lcd/lcd_truly_r61350_mipi_hd.dtsi

3.2.2 修改 dts 文件

步骤 1 将 dtsi 文件中 DDIC 名全部重新命名。原名：xxx_ili9881c_xxx -> 新名：xxx_r61350_xxx。

步骤 2 配置初始化 code（由屏幕模组厂提供）为如图 3-2 格式。

图3-2 initial 数据格式

```
sprd,initial-command = [
    //Host I/F Setting
    15 00 00 02 B0 00
    15 00 00 02 CC 04
    15 00 00 02 E3 01
    //DSI control
    39 00 00 03 B6 61 2C
    //display setting
    39 00 00 07 C0 23 B2 22 10 C2 7F
    //display h-timing setting
    39 00 00 17 C1 0B 6F 01 80 00 00
    //stop recovery setting
    15 00 00 02 C3 75
];

struct dsi_cmd_desc {
    uint8_t data_type; //mipi包的类型
    uint8_t wait;      //延时时间(ms)
    uint8_t wc_h;      //数据包长度高8位
    uint8_t wc_l;      //数据包长度低8位
    uint8_t payload[]; //data数据
};
```

步骤3 配置 dts 属性。

参考设置如图 3-3 所示，参数保持跟 uboot 的配置一致。其中 panel node,sprd,dsi-work-mode,sprd,dsi-lane-number,sprd,phy-bit-clock,sprd,phy-escape-clock, sprd,reset-on-sequence,sprd,reset-off-sequence,sprd,initial-command,sprd,sleep-in-command,display-timings 为必配项，其他可根据实际情况酌情配置，也可以咨询展锐技术支持。

dtsi 文件属性说明见表 3-1。

Unisoc Confidential For hiar

图3-3 dts 配置

```

/ { lcds {
    lcd_r61350_truly_mipi_hd: lcd_r61350_truly_mipi_hd {

        sprd,dsi-work-mode = <1>; /* video burst mode*/
        sprd,dsi-lane-number = <4>;
        sprd,dsi-color-format = "rgb888";

        sprd,phy-bit-clock = <500000>; /* kbps */
        sprd,phy-escape-clock = <20000>; /* kHz */

        sprd,width-mm = <68>;
        sprd,height-mm = <121>;

        sprd,esd-check-enable = <0>;
        sprd,esd-check-mode = <0>;
        sprd,esd-check-period = <1000>;
        sprd,esd-check-register = <0x0A>;
        sprd,esd-check-value = <0x9C>;

        sprd,reset-on-sequence = <1 5>, <0 5>, <1 20>;
        sprd,reset-off-sequence = <1 5>;

        sprd,use-dcs-write;
        sprd,initial-command = [
            //Host I/F Setting
            15 00 00 02 B0 00
            15 00 00 02 CC 04
            15 00 00 02 E3 01
        ];

        sprd,sleep-in-command = [
            15 32 00 01 28
            15 96 00 01 10
            29 00 00 02 B0 00
            29 14 00 02 B1 01
        ];

        sprd,sleep-out-command = [
            15 78 00 01 11
            15 32 00 01 29
            15 01 00 02 B0 00
            15 01 00 02 B1 01
        ];

        display-timings {
            timing0 {
                clock-frequency = <64000000>;
                hactive = <720>;
                vactive = <1280>;
                hback-porch = <31>;
                hfront-porch = <31>;
                vback-porch = <32>;
                vfront-porch = <16>;
                hsync-len = <20>;
                vsync-len = <2>;
            };
        };
    };
};
    
```

表3-1 dtsi 文件属性说明

属性	说明
lcd_r61350_truly_mipi_hd	lcd 节点名，必须和 uboot 中的 lcd name 保持一致，否则 kernel 无法匹配到正确的 lcd 驱动
sprd,dsi-work-mode	该 LCD 采用何种 dsi mode 来传输图像数据，目前支持如下值： cmd mode(0)

属性	说明
	video burst mode (默认值 1) video non-burst mode with sync pulse(2) video non-burst mode with sync event(3)
sprd,dsi-lane-number	该 LCD 在接收 video 数据传输时的 datalane 总条数, 范围 1~4, 默认 4 条 lane
sprd,dsi-color-format	video stream 传输时的数据格式, 目前支持"rgb888"、"rgb666"、"rgb666_packed"、"rgb565"和"dsc", 默认使用 rgb888 格式
sprd,dsi-non-con_nuousclock	bool 属性, 表示使用非连续 clock, 即在 H/V proch 阶段, 或 DPU 停止送数阶段, clock lane 可以自动进入 LP11 状态, 以节省功耗
sprd,phy-bit-clock	High-Speed 模式下数据传输时, clock lane 的传输速率, 即 phy frequency, 单位为 Kbps, 默认值为 500Mbps
sprd,phy-escape-clock	Low-Power 模式下数据传输的速率, 单位 KHz, 最大不能超过 20Mhz, 默认值为 20MHz
sprd,width-mm	LCD 可视区域的物理尺寸, 宽度, 单位 mm, 默认值为 68mm
sprd,height-mm	LCD 可视区域的物理尺寸, 高度, 单位 mm, 默认值为 121mm
sprd,esd-check-enable	ESD Check 功能使能开关。1-开启, 0-关闭, 默认关闭
sprd,esd-check-mode	ESD 检测方式, 0 - register, 1 - TE, 默认采用 register 方式
sprd,esd-check-period	ESD 检测周期, 单位 ms, 默认值 1000ms
sprd,esd-check-register	如果采用 register 检测方式, 则该属性将被使用, 表明要检测的 register 的正常状态值, 默认为 0x9C, 即 Power Status 寄存器的正常状态值
sprd,esd-check-value	如果采用 register 检测方式, 则该属性将被使用, 表明要检测的 register 的正常状态值, 默认为 0x9C, 即 Power Status 寄存器的正常状态值
sprd,reset-on-sequence	LCD 上电时, reset 引脚的操作时序。<1 5>,<0 5>表示先保持高电平 5ms, 再保持低电平 5ms
sprd,reset-off-sequence	LCD 下电时, reset 引脚的操作时序。<1 5>,<0 5>表示先保持高电平 5ms, 再保持低电平 5ms
sprd,use-dcs-write	bool 属性, 表示是否使用 dcs 指令集发送 inital & sleep-in 指令, 默认使用 generic 指令集发送 cmd
sprd,ini_al-command	LCD 初始化指令
sprd,sleep-in-command	LCD 休眠指令
sprd,sleep-out-command	LCD 唤醒指令, 仅用于 debug, 正常休眠唤醒流程中不会使用

属性	说明
display-timings	video timing 信息节点
clock-frequency	pixel clock, 单位 Hz
hactive & vactive	LCD 显示分辨率, hactive-宽, vactive-高, 单位为 pixel
xxx-porch & xxx-len	VIDEO 屏的 Hporch/Vporch 参数。如果是 CMD 屏, 这些参数必须是<0>

步骤 4 引用配置好的 dts 文件。

找到并打开 board 配置头文件, 并且添加到 board 配置文件中:
kernel/arch/arm64/boot/dts/sprd/ums312-1h10.dts

```
#include "lcd/lcd_r61350_truly_mipi_hd.dtsi"
#include "lcd/lcd_ili9881c_truly_mipi_hd.dtsi"
```

通过#include 将需要适配的 lcd 外设与该 board 关联上, 这样 panel 驱动就会根据 uboot 传上来的 lcd_name 去"/lcds"节点下查找相应的 lcd 子节点, 从而完成 lcd 的匹配。

步骤 5 配置和 LCD 模组相关的 GPIO 和 Power Supply。LCD 模组相关的属性见表 3-2。

找到并打开 board 配置头文件, 并且添加到 board 配置文件中:
kernel/arch/arm64/boot/dts/sprd/ums312-1h10.dts

```
&dsi {
    status = "okay";
    #address-cells = <1>;
    #size-cells = <0>;

    panel: panel {
        compatible = "sprd,generic-mipi-panel";
        #address-cells = <1>;
        #size-cells = <0>;
        reg = <0>;

        reset-gpio = <&ap_gpio 50 GPIO_ACTIVE_HIGH>;
        port {
            reg = <1>;
            panel_in: endpoint {
                remote-endpoint = <&dphy_out>;
            };
        };
    };
};
```

表3-2 LCD 模组相关的属性说明

属性	说明
power-supply	可选属性, 给 LCD 模组供电的+/-5V 偏置电压, 指向 regulator 节点。
gpio	可选属性, 目前支持 3 个 GPIO, 分别为 reset-gpio、avdd-gpio 和

属性	说明
	avee-gpio。
sprd,backlight	与该 panel 相关联的 backlight 设备，用于确保在系统关机时，先关背光，后关显示。

----结束

3.3 Kernel 背光配置

目前 panel 已经可以支持 lcd 和 oled 两种形式的背光配置。

3.3.1 LCD 背光配置

- 配置 LCD 背光属性。

找到并打开 overlay board 配置头文件，添加 sprd_backlight 属性：
kernel/arch/arm64/boot/dts/sprd/ums312-1h10-overlay.dts

```
pwm_backlight: sprd_backlight {
    compatible = "pwm-backlight";
    pwms = <&pwms 2 40000>;
    pwm-names = "backlight";
    brightness-levels = <
        2 2 2 2 2 2 2 2 2 2
        2 2 2 3 4 5 6 7 8 9
        10 11 12 13 14 15 16 17 18 19
        20 21 22 23 24 25 26 27 28 29
        30 31 32 33 34 35 36 37 38 39
        200 201 202 203 204 205 206 207 208 209
        .....
        210 211 212 213 214 215 216 217 218 219
        220 221 222 223 224 225 226 227 228 229
        230 231 232 233 234 235 236 237 238 239
        240 241 242 243 244 245 246 247
    >;
    default-brightness-level = <33>;
};
```

说明

属性说明：Linux Kernel 标准属性，请查阅：Documentation/devicetree/bindings/leds/backlight/。

- 引用 LCD 背光属性。

```
&panel {  
    sprd,backlight = <&pwm_backlight>;  
};
```

3.3.2 OLED 背光配置

配置 OLED 背光属性，以 DDIC 型号为 sw43404 为例，添加 oled_backlight 属性：

kernel/arch/arm64/boot/dts/sprd/lcd/lcd_boe_sw43404_mipi_fhd.dtsi

```
oled-backlight {  
    default-brightness-level = <25>;  
    sprd,max-level = <255>;  
    brightness-levels = [  
        29 00 00 03 51 00 00  
    ];  
};
```

OLED 背光属性说明见表 3-3。

表3-3 OLED 背光属性说明

属性	说明
default-brightness-level	初始背光值索引，和 Linux 内核标准属性含义一致，默认为 25。
sprd,max-level	最大背光值索引，目前最大支持 255，后续可能会支持更多 level。
brightness-levels	背光占空比级别，级别数量和 sprd,max-level 索引值要一致，和 Linux 内核标准属性含义一致。

4

注意事项

展锐的 fb、overlay、write back、lcdon 等功能都是和显示模组的分辨率大小有关联的，所以在更换新的显示模组后，如果分辨率有改变的话，请务必告知展锐的技术支持人员。

Unisoc Confidential For hiar