密级状态：绝密（　　）　秘密（　　）　内部（ √ ）　公开（　）

RK3399_VR 分体机_显示屏_参数修改说明文档_V1.0_2016.9.3

（技术部，第二系统产品部门）

文件状态： [　] 正在修改 [√] 正式发布	当前版本：	V1.0
	作　　者：	王剑辉
	完成日期：	2016-09-03
	审　　核：	张文平
	完成日期：	2016-09-03

福州瑞芯微电子股份有限公司

Fuzhou　Rockchips　Semiconductor　Co . , Ltd

更新记录

版本	修改人	修改日期	修改说明	备注
V1.0	王剑辉	2016.9.3	初始版本	
V1.1	王剑辉	2017.1.4	更新宏说明和显示参数修改说明部分	

目 录

1 概述

本文档主要用于 RK3399VR 分体机项目，头盔端代码的开关配置说明、调试一款新屏的步骤说明。

2 DP 屏和 HDMI 屏配置开关介绍

Nanoc 屏的配置开关宏在 SDK 目录下面 SysConfig.h 文件里面，相关内容如下：

```
00163: *-------------------------------------------------------------------------
00164: */
00165: //configer Fusb302
00166: #define _FUSB302_
00167: #define _TOUCH_GSL1680_
00168: //configer sensor
00169: #define _MPU6500_
00170:
00171: //configer TC3588xx
00172: #define _TC358860XBG_
00173: //#define _TC358870XBG_
00174:
00175: //configer mipi lcd manufacturer
00176: //#define _AUO_2P1_60FPS_LCD_
00177: //#define _AUO_3P81_75FPS_LCD_
00178: //#define _SHARP_2P89_75FPS_LCD_
00179: #define _RAYKEN_5P46_60FPS_LCD_
00180:
00181: //-----------------------------------//
00182: //    AUO3.81 use _DP_MIPI_2LANE_    //
00183: //-----------------------------------//
00184: #ifdef _AUO_3P81_75FPS_LCD_
00185: #define _DP_MIPI_2LANE_
00186: #else
00187: #define _DP_MIPI_4LANE_
00188: #endif
00189:
00190: //configer EDID
00191: //#define _EEPROM_EDID_
00192: #define _GPIO_EDID_
00193:
00194: //configer music player
00195: #define _MUSIC_
00196: #ifdef _MUSIC_
00197:     #define _MUSIC_LRC_
00198:     #define AUDIOHOLDONPLAY          //hold on play switch.
00199: #endif
00200:
00201:
```

触摸板开关

sensor开关

显示输出方式开关，TC358860是dp，TC358870是HDMI

显示屏类型选择

EDID传输方式，一种是EEPROM存储通过I2C传输，另外一种是GPIO模拟I2C输出，省去了EEPROM

TC358860XBG 是 DP 的宏开关，当前使用 DP 屏时，需要把这个宏打开，把 HDMI 的宏关闭。

TC358870XBG 是 HDMI 的宏开关，当前使用 HDMI 屏时，把这个宏打开，把 DP 的宏关闭。

另外：

_TOUCH_GSL1680_ 是 TP 触摸板的宏开关

FUSB302 是 FUSB302 的宏开关

MPU6500 是 sensor 的宏开关

_DP_MIPI_2LANE_ 是表示用的 2 个 LANE（DP 接口的 MIPI 屏）的宏开关

_EEPROM_EDID_ 是把屏的信息写到 EEPROM 里面，当 nanoc 板子上面有贴 EEPROM 时，可以把这个宏打开

_GPIO_EDID_ 用 gpio 模拟 EEPROM 传输 EDID 信息，当 nanoc 板子没贴 EEROM 时，打开这个宏来实现 EDID 传输。

3 修改屏的参数

从 mipi 屏 Datasheet 得到屏的 Timing 参数（可以参考 RockChip_DSS Development Guide v1.2.pdf），然后把参数填到 excel 表格里面，下面举两个例子：

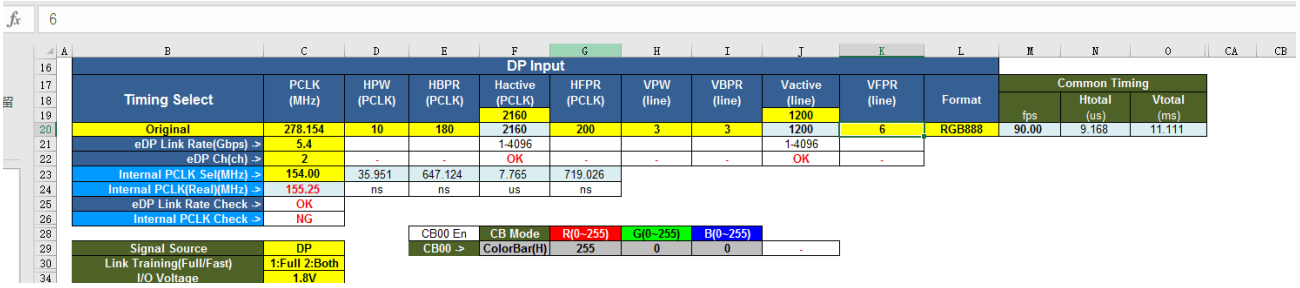
例一：

Timing 参数如下

```
disp_timings: display-timings {
        native-mode = <&timing0>;
        compatible = "rockchip,display-timings";
        timing0: timing0 {
                screen-type = <SCREEN_DUAL_MIPI>;
                lvds-format = <LVDS_8BIT_2>;
                out-face    = <OUT_P888>;
                clock-frequency = <278154000>;
                hactive = <2160>; //1080
                vactive = <1200>;
                hback-porch = <180>;
                hfront-porch = <200>;
                vback-porch = <3>;
                vfront-porch = <6>;
                hsync-len = <10>;
                vsync-len = <3>;
                hsync-active = <0>;
                vsync-active = <0>;
                de-active = <0>;
                pixelclk-active = <0>;
```

```
                swap-rb = <0>;
                swap-rg = <0>;
                swap-gb = <0>;
        };
};
```

修改后的 excel 表格如下：



					DP Input									
Timing Select	PCLK (MHz)	HPW (PCLK)	HBPR (PCLK)	Hactive (PCLK)	HFPR (PCLK)	VPW (line)	VBPR (line)	Vactive (line)	VFPR (line)	Format	Common Timing			
											fps	Htotal (us)	Vtotal (ms)	
				2160				1200						
Original	278.154	10	180	2160	200	3	3	1200	6	RGB888	90.00	9.168	11.111	
eDP Link Rate(Gbps) ->	5.4			1-4096				1-4096						
eDP Ch(ch) ->	2	-	-	OK	-	-	-	OK	-					
Internal PCLK Sel(MHz) ->	154.00	35.951	647.124	7.765	719.026									
Internal PCLK(Real)(MHz) ->	155.25	ns	ns	us	ns									
eDP Link Rate Check ->	OK													
Internal PCLK Check ->	NG													
		CB00 En	CB Mode	R(0~255)	G(0~255)	B(0~255)								
		CB00 ->	ColorBar(H)	255	0	0	-							
Signal Source	DP													
Link Training(Full/Fast)	1:Full 2:Both													
I/O Voltage	1.8V													

Timing 参数	----------	excel 表格参数
hsync-len	---------->	HPW
vsync-len	---------->	VPW
hback-porch	---------->	HBPR
hfront-porch	---------->	HFPR
vback-porch	---------->	VBPR
vfront-porch	---------->	VFPR
xres	---------->	Hactive
yres	---------->	Vactive

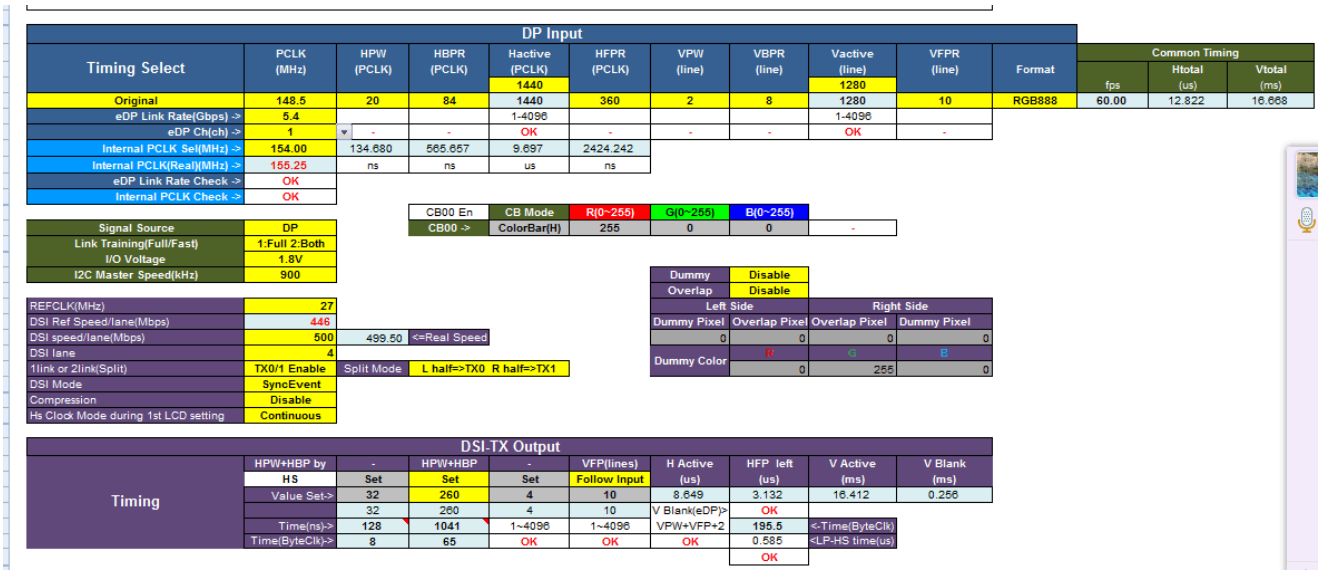
例二

Timing 参数如下：

```
 {
.mode = {
.name = "1440x1280@60Hz",
.refresh = 60,
.xres = 1440,
.yres = 1280,
.pixclock = 148500000,
.left_margin = 84,
```

```
.right_margin = 360,
.upper_margin = 8,
.lower_margin = 10,
.hsync_len = 20,
.vsync_len = 2,
.sync = 0,
.vmode = 0,
.flag = 0,
},
.vic = HDMI_VIDEO_DMT | 11,
.vic_2nd = 0,
.pixelrepeat = 1,
.interface = OUT_P888,
}
```

修改后的 excel 表格



DP Input

Timing Select	PCLK (MHz)	HPW (PCLK)	HBPR (PCLK)	Hactive (PCLK)	HFPR (PCLK)	VPW (line)	VBPR (line)	Vactive (line)	VFPR (line)	Format	Common Timing fps	Htotal (us)	Vtotal (ms)
				1440				1280					
Original	148.5	20	84	1440	360	2	8	1280	10	RGB888	60.00	12.822	16.668
eDP Link Rate(Gbps) ->	5.4			1-4096				1-4096					
eDP Ch(ch) ->	1	-	-	OK	-	-	-	OK	-				
Internal PCLK Sel(MHz) ->	154.00	134.680	565.657	9.697	2424.242								
Internal PCLK(Real)(MHz) ->	155.25	ns	ns	us	ns								
eDP Link Rate Check ->	OK												
Internal PCLK Check ->	OK												

CB00 En	CB Mode	R(0~255)	G(0~255)	B(0~255)	
CB00 ->	ColorBar(H)	255	0	0	-

Signal Source	DP
Link Training(Full/Fast)	1:Full 2:Both
I/O Voltage	1.8V
I2C Master Speed(kHz)	900

REFCLK(MHz)	27		
DSI Ref Speed/lane(Mbps)	446		
DSI speed/lane(Mbps)	500	499.50	<=Real Speed
DSI lane	4		
1link or 2link(Split)	TX0/1 Enable	Split Mode	L half=>TX0 R half=>TX1
DSI Mode	SyncEvent		
Compression	Disable		
Hs Clock Mode during 1st LCD setting	Continuous		

Dummy	Disable	
Overlap	Disable	
Left Side		Right Side
Dummy Pixel	Overlap Pixel	Overlap Pixel / Dummy Pixel
0	0	0 / 0
Dummy Color	R	G / B
	0	255 / 0

DSI-TX Output

Timing	HPW+HBP by	-	HPW+HBP	-	VFP(lines)	H Active (us)	HFP_left (us)	V Active (ms)	V Blank (ms)
	HS	Set	Set	Set	Follow Input				
	Value Set->	32	260	4	10	8.649	3.132	16.412	0.256
		32	260	4	10	V Blank(eDP)>	OK		
	Time(ns)->	128	1041	1~4096	1~4096	VPW+VFP+2	195.5	<-Time(ByteClk)	
	Time(ByteClk)->	8	65	OK	OK	OK	0.585	<LP-HS time(us)	
							OK		

Timing 参数 ---------- excel 表格参数

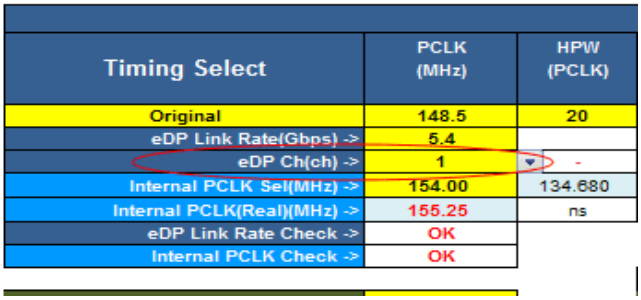
hsync_len ----------> HPW

vsync_len ----------> VPW

left_margin ----------> HBPR
right_margin ----------> HFPR
upper_margin ----------> VBPR
lower_margin ----------> VFPR
xres ----------> Hactive
yres ----------> Vactive

4 生成屏的初始化函数

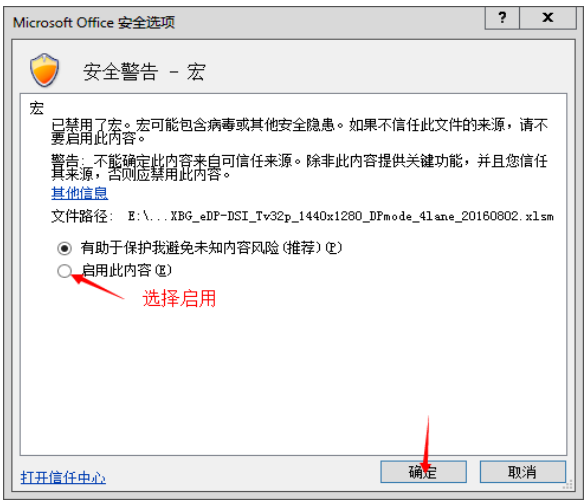
修改完 timing 后，还需要根据实际选择 DP mipi 屏（HDMI 屏没有）的几个 LANE，选择方法如下：



Timing Select	PCLK (MHz)	HPW (PCLK)
Original	148.5	20
eDP Link Rate(Gbps) ->	5.4	
eDP Ch(ch) ->	1	-
Internal PCLK Sel(MHz) ->	154.00	134.680
Internal PCLK(Real)(MHz) ->	155.25	ns
eDP Link Rate Check ->	OK	
Internal PCLK Check ->	OK	

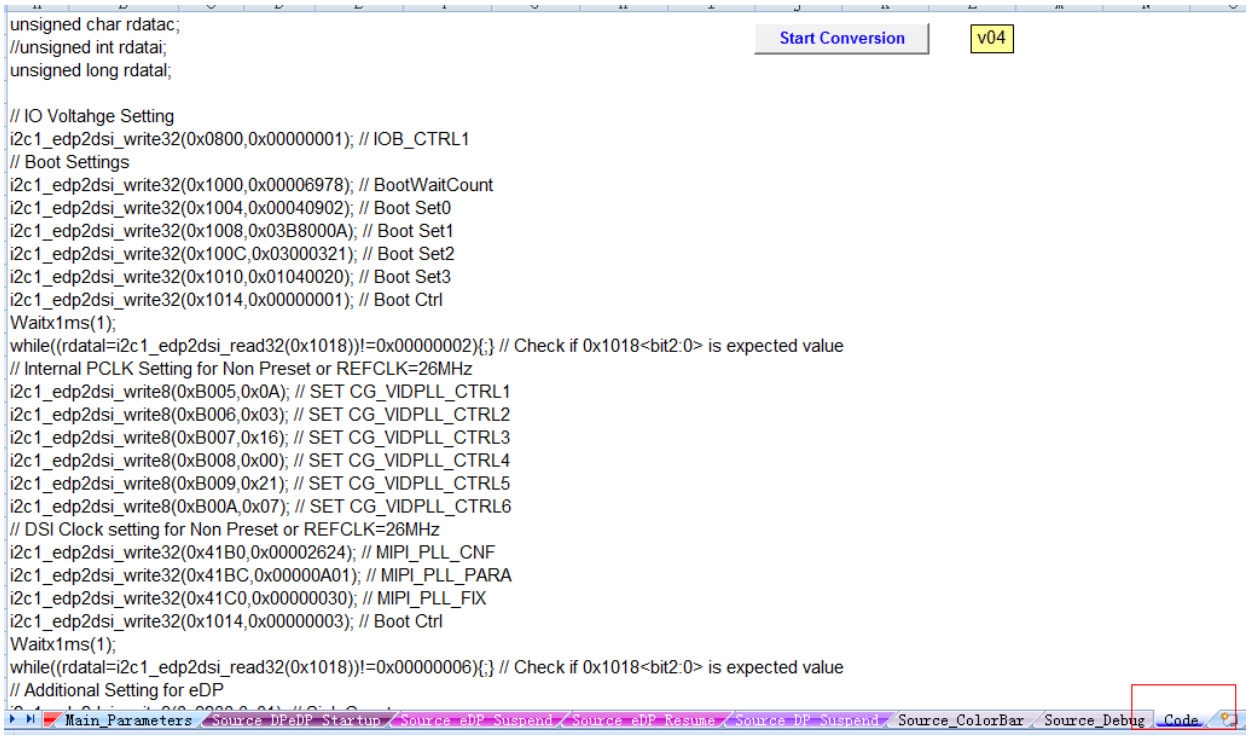
有三种 lane 供旋转分别为 1、2、4，图中选择的是 1 个 lane。

注意：如下图，有些 PC 电脑可能装的 Microsoft Excel 工具版本不一样，有可能宏已经被禁用，需要我们手动打开。



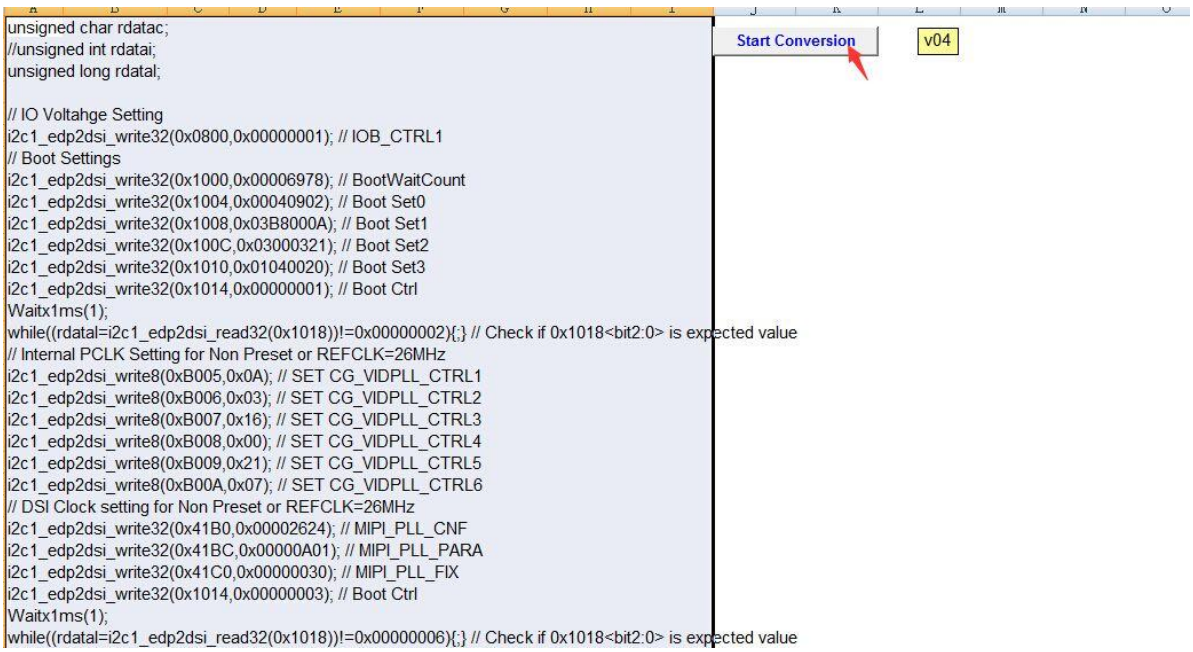
生成初始化函数步骤如下：

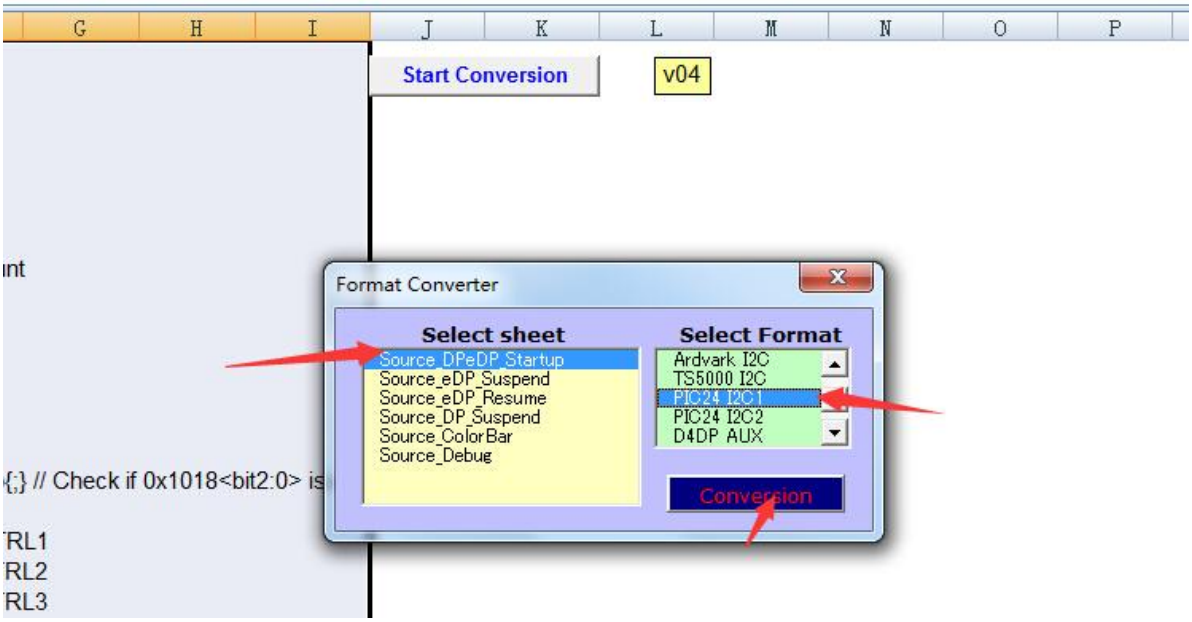
修改 timing 和 LANE 后，切换窗口到 Code 窗口，如下图



```
unsigned char rdatac;
//unsigned int rdatai;
unsigned long rdatal;

// IO Voltahge Setting
i2c1_edp2dsi_write32(0x0800,0x00000001); // IOB_CTRL1
// Boot Settings
i2c1_edp2dsi_write32(0x1000,0x00006978); // BootWaitCount
i2c1_edp2dsi_write32(0x1004,0x00040902); // Boot Set0
i2c1_edp2dsi_write32(0x1008,0x03B8000A); // Boot Set1
i2c1_edp2dsi_write32(0x100C,0x03000321); // Boot Set2
i2c1_edp2dsi_write32(0x1010,0x01040020); // Boot Set3
i2c1_edp2dsi_write32(0x1014,0x00000001); // Boot Ctrl
Waitx1ms(1);
while((rdatal=i2c1_edp2dsi_read32(0x1018))!=0x00000002){;} // Check if 0x1018<bit2:0> is expected value
// Internal PCLK Setting for Non Preset or REFCLK=26MHz
i2c1_edp2dsi_write8(0xB005,0x0A); // SET CG_VIDPLL_CTRL1
i2c1_edp2dsi_write8(0xB006,0x03); // SET CG_VIDPLL_CTRL2
i2c1_edp2dsi_write8(0xB007,0x16); // SET CG_VIDPLL_CTRL3
i2c1_edp2dsi_write8(0xB008,0x00); // SET CG_VIDPLL_CTRL4
i2c1_edp2dsi_write8(0xB009,0x21); // SET CG_VIDPLL_CTRL5
i2c1_edp2dsi_write8(0xB00A,0x07); // SET CG_VIDPLL_CTRL6
// DSI Clock setting for Non Preset or REFCLK=26MHz
i2c1_edp2dsi_write32(0x41B0,0x00002624); // MIPI_PLL_CNF
i2c1_edp2dsi_write32(0x41BC,0x00000A01); // MIPI_PLL_PARA
i2c1_edp2dsi_write32(0x41C0,0x00000030); // MIPI_PLL_FIX
i2c1_edp2dsi_write32(0x1014,0x00000003); // Boot Ctrl
Waitx1ms(1);
while((rdatal=i2c1_edp2dsi_read32(0x1018))!=0x00000006){;} // Check if 0x1018<bit2:0> is expected value
```





生成的初始化代码如下（选取部分截图）

```
unsigned char rdatac;
//unsigned int rdatai;
unsigned long rdatal;

// IO Voltahge Setting
i2c1_edp2dsi_write32(0x0800,0x00000001); // IOB_CTRL1
// Boot Settings
i2c1_edp2dsi_write32(0x1000,0x00006978); // BootWaitCount
i2c1_edp2dsi_write32(0x1004,0x00040902); // Boot Set0
i2c1_edp2dsi_write32(0x1008,0x03B8000A); // Boot Set1
i2c1_edp2dsi_write32(0x100C,0x03000321); // Boot Set2
i2c1_edp2dsi_write32(0x1010,0x01040020); // Boot Set3
i2c1_edp2dsi_write32(0x1014,0x00000001); // Boot Ctrl
Waitx1ms(1);
while((rdatal=i2c1_edp2dsi_read32(0x1018))!=0x00000002){;} // Check if 0x1018<bit2:0> is expected value
// Internal PCLK Setting for Non Preset or REFCLK=26MHz
i2c1_edp2dsi_write8(0xB005,0x0A); // SET CG_VIDPLL_CTRL1
i2c1_edp2dsi_write8(0xB006,0x03); // SET CG_VIDPLL_CTRL2
i2c1_edp2dsi_write8(0xB007,0x16); // SET CG_VIDPLL_CTRL3
i2c1_edp2dsi_write8(0xB008,0x00); // SET CG_VIDPLL_CTRL4
i2c1_edp2dsi_write8(0xB009,0x21); // SET CG_VIDPLL_CTRL5
i2c1_edp2dsi_write8(0xB00A,0x07); // SET CG_VIDPLL_CTRL6
// DSI Clock setting for Non Preset or REFCLK=26MHz
i2c1_edp2dsi_write32(0x41B0,0x00002624); // MIPI_PLL_CNF
i2c1_edp2dsi_write32(0x41BC,0x00000A01); // MIPI_PLL_PARA
i2c1_edp2dsi_write32(0x41C0,0x00000030); // MIPI_PLL_FIX
i2c1_edp2dsi_write32(0x1014,0x00000003); // Boot Ctrl
Waitx1ms(1);
while((rdatal=i2c1_edp2dsi_read32(0x1018))!=0x00000006){;} // Check if 0x1018<bit2:0> is expected value
// Additional Setting for eDP
```

Start Conversion

5 修改屏初始化代码填到 nanoc 代码里面

5.1 生成的初始化代码接口名称有些不一样，需要手动修改：

1.Waitx1ms(1);延时 1ms 的函数，需要改为 DelayMs(1);

2.Waitx1us(100);延时 100us 的函数，需要改为 DelayUs(100);

3.while((rdatac=i2c1_edp2dsi_read8(0x8202))!=0x07){;} // Check if 0x8203 is expected value.这种由于变量和读寄存器函数不一样，需要修改为

while((rdata8=i2c1_edp2dsi_read8(0x8202, &data8))!=0x77){;} // Check if 0x8203 is expected value.

4.while((rdatal=i2c1_edp2dsi_read32(0x4060))!=0x00000003){;} // Check if 0x2060/4060<bit1:0>=11b.

对应改为

while((rdata32=i2c1_edp2dsi_read32(0x4060,data32))!=0x00000003){;} // Check if

0x2060/4060<bit1:0>=11b.

下面截取部分修改前和修改后的代码对比：（上图是修改前，下图是修改后）

```
Waitx1ms(1);
while((rdatal=i2c1_edp2dsi_read32(0x1018))!=0x00000007){;} // Check if 0x1018<bit2:0> is expected value
// Video Size Related Setting for Non Preset
i2c1_edp2dsi_write32(0x0110,0x000002D0); // HTIM2_LEFT
i2c1_edp2dsi_write32(0x0124,0x000002D0); // HTIM2_RIGHT
i2c1_edp2dsi_write32(0x0148,0x000005A0); // DPRX_HTIM2
i2c1_edp2dsi_write32(0x2920,0x3E0B0870); // DSI0_PIC_SYN_PKT_A
i2c1_edp2dsi_write32(0x3920,0x3E0B0870); // DSI1_PIC_SYN_PKT_A
// eDP Settings for Link Training
while((rdatac=i2c1_edp2dsi_read8(0xB631))!=0x01){;} // Check if 0xB631<bit1:0>=01b.
i2c1_edp2dsi_write8(0x8001,0x14); // Max Link Rate
i2c1_edp2dsi_write8(0x8002,0x01); // Max Lane Count
```

```
DelayMs(1);
while((rdata32=i2c1_edp2dsi_read32(0x1018,data32))!=0x00000007){;} // Check if 0x1018<bit2:0> is expected value
// Video Size Related Setting for Non Preset
i2c1_edp2dsi_write32(0x0110,0x000002D0); // HTIM2_LEFT
i2c1_edp2dsi_write32(0x0124,0x000002D0); // HTIM2_RIGHT
i2c1_edp2dsi_write32(0x0148,0x000005A0); // DPRX_HTIM2
i2c1_edp2dsi_write32(0x2920,0x3E0B0870); // DSI0_PIC_SYN_PKT_A
i2c1_edp2dsi_write32(0x3920,0x3E0B0870); // DSI1_PIC_SYN_PKT_A
// eDP Settings for Link Training
while((rdata8=i2c1_edp2dsi_read8(0xB631,&data8))!=0x01){;} // Check if 0xB631<bit1:0>=01b.
printf ("init ------1\n");
i2c1_edp2dsi_write8(0x8001,0x14); // Max Link Rate 5.4Gb/s(0x14) 0x0A
i2c1_edp2dsi_write8(0x8002,0x01); // Max Lane Count
```

5.2 4LANE 的初始化比较特殊的注意点（1LANE 和 2LANE 可以忽略）：

1LANE 和 2LANE 的代码按照上面步骤就可以完成初始化。4LANE 的有两个需要注意的地方：

```
while((rdata8=i2c1_edp2dsi_read8(0xB631, &data8))!=0x01){;} // Check if 0xB631<bit1
//i2c1_edp2dsi_write8(0x8000,0x11); // DPCD Rev
i2c1_edp2dsi_write8(0x8001,0x14); // Max Link Rate //0X14
i2c1_edp2dsi_write8(0x8002,0x04); // Max Lane Count
i2c1_edp2dsi_write8(0xB608,0x0B); // Set AUXTXHSEN
i2c1_edp2dsi_write8(0xB800,0x1E); // Set CDR_PHASE_LP_EN
i2c1_edp2dsi_write8(0x8700,0x00); // DPCD 0700h
```

这个需要注释掉。

```
// Start Link Training
// By detecting VSYNC monitor output (INT rising edge)
// I2C direction control for DP connection
// Check if VSYNC is detected on monitor signal
i2c1_edp2dsi_write32(0x00B0,0x00000000); // I2CMD_SL
i2c1_edp2dsi_write32(0x00B0,0x00000000); // I2CMD_SL
i2c1_edp2dsi_write32(0x00B0,0x00000000); // I2CMD_SL
```

需要多次写才能成功。