# 一、RGBLCD显示原理简介

##  1、像素点

 于一个“小灯“，不管是液晶屏，还是手机，平板，RGBLCD屏幕他都是有由一个个的彩色小灯构成的。彩色点阵屏每个像素点有三个小灯，红色、绿色和蓝色，也叫做RGB。RGB就是光的三原色。通过调整RGB三种颜色的比例，就可以实现姹紫千红的世界。

##  2、分辨率

 要想显示文字，图片，视频等等就需要很多个像素点，分辨率说的就是像素点的个数，1080P、720P、2K、4K，8K。1080P=1920\*1080，表示一行有1920个像素点，一列有1080个。显示器有尺寸！24村，27村、55寸。尺寸不变的情况下，分辨率越高，显示效果越精细。4K=3840\*2160相当于4个1080P

 正点原子的RGB屏幕有：4.3寸480\*272，800\*480，7寸的800\*480和1024\*600，10.1寸的1280\*800。

 Iphone4屏幕尺寸是3.5寸，960\*640分辨率，PPI=327.

##  3、像素格式

 如何将RGB三种颜色进行量化，每种颜色用8bit表示，RGB就需要888共24bit。可以描述出2^24=中颜色16777216=1677万种颜色。现在流行10bit，HDR10，支持HDR效果的10bit面板，RGB10 10 10。

 在RGB888的基础上在加上8bit的ALPHA通道，也就是透明通道，ARGB8888=32位。

##  4、LCD屏幕接口

 RGB格式的屏幕，一般叫做RGB接口屏。

屏幕接口有：MIPI、LVDS、MCU、RGB接口。

正点原子屏幕ID：使用ID可以识别出不同的屏幕，在RGBLCD屏幕上对R7，G7，B7焊接上拉或下拉电阻实现不同的ID。，

正点原子的ALPHA地板RGB屏幕接口用了3个3157模拟开关。原因是防止LCD屏幕上的ID电阻影响到6ULL的启动。

##  5、LCD时间参数和LCD时序

 水平：

 HSYNC：水平同步信号，行同步信号，当出现HSYNC信号的时候表示新的一行开始显示

 1、产生HSYNC信号，表示新的一行开始显示，HSYNC信号得维持一段时间，这个时间叫做HSPW。

 2、HSYNC信号完成以后，需要一段时间延时，这段时间叫做HBP。

 3、显示1024个像素点的数据，1024个clk。

 3、一行像素显示完成以后，到HSYNC下一行信号的产生之间有个延时，叫做HFP。

 因此真正显示一行所需的时间计时HSPW+HBP+WIDTH(屏幕水平像素点个数，比如1024)+HFP=20+140+1024+160=1344CLK.

 垂直：

 VSYNC：垂直同步信号，帧同步信号，当出现VSYNC信号的时候表示新的一帧开始显示。

 1、VSYNC信号，持续一段时间，位VSPW。

 2、VSYNC信号完成以后，需要一段时间，叫做VBP

 3、VBP信号结束以后，就是要显示的行数，比如600行，

 4、所有的行显示完成以后，一段VFP延时，

 （VSPW+VBP+height(600)+VFP）\* （HSPW+HBP+WIDTH(1024)+HFP）=

 像素时钟：

##  6、显存

 显存：显示存储空间，采用ARGB8888=32bit=4B。这4个字节的数据表示一个像素点的信息，必须得存起来。1024\*600\*4=2.5MB。因此需要流出2.5MB的内存给LCD用，方法很简单，直接定义一个32位的数组，u32 lcdframe[1024\*600];

# 二、6ULL LCDIF控制器接口原理

 1、我们使用DOTCLK接口，也就是VSYNC、HSYNC、ENABLE(DE)和DOTCLK(PCLK)

 2、LCDIF\_CTRL寄存器，bit0必须置1，bit1设置数据格式24位全部有效，设置为0。Bit5设置LCDIF接口工作在主机模式下，要置1。Bit9:8设置输入像素格式位24bit，写3。Bit11：10，设置数据传输宽度位24bit，写3。Bit13：12设置数据交换，我们不需要交换设置位0。Bit15:14设置输入数据交换，不交换设置位0。Bit17置1，LCDIF工作在DOTCLK模式下。Bit19必须置1，因为工作在DOTCL模式。Bit31是复位功能必须置0.

 3、LCDIF\_CTRL1寄存器的bit19:16设置位0X7。24位的格式，

 4、LCDIF\_TRANSFER\_COUNT寄存器的bit15：0是LCD一行的像素数，1024。Bit31:16是LCD一共有多少行，600行。

 5、LCDIF\_VDCTRL0寄存器，bit17:0为vspw参数。Bit20设置vsync信号的宽度单位 ，设置为1。Bit21设置为1。Bit24设置ENABLE信号极性，为0的时候是低电平有效，为1是高电平，我们设置为1。Bit25设置CLK信号极性，设置为0.。bit26设置HSYNC信号极性，设置0，低电平有效，bit27设置VSYNC信号极性，设置为0，低电平有效。Bit28设置1，开始ENABLE信号。Bit29设置为0，VSYNC输出。

 6、LCDIF\_VDCTRL1寄存器为两个VSYNC信号之间的长度，那就是VSPW+ VBPD+HEIGHT+VFP。

 7、CDIF\_VDCTRL2寄存器bit17：0是两个HSYNC信号之间的长度，那就是hspw+hbp+width+hfp。Bit31:18为hspw。

 8、LCDIF\_VDCTRL3寄存器，bit15:0是vbp+vspw。Bit27:16是hbp+hspw。

 9、LCDIF\_VDCTRL4寄存器，bit17:0是一行有多少个像素点，1024

 10、LCDIF\_CUR\_BUF，LCD当前缓存，显存首地址。

 11、LCDIF\_NEXT\_BUF,LCD下一帧数据首地址。

 12、LCD IO初始化。

# 三、LCD像素时钟的设置。

 LCD需要一个CLK信号，这个时钟信号是6ULL的CLK引脚发送给RGB LCD的。比如7寸1024\*600的屏幕需要51.2MHz的CLK。

 LCDIF\_CLK\_ROOT就是6ULL的像素时钟。我们设置PLL5(video PLL)为LCD的时钟源，

 PLL5 CLK=Fref\*DIV\_SELECT=，DIV\_SELECT就是CCM\_ANALOG\_PLL\_VIDEO的bit6:0，也就是DIV\_SELEC位，可选范围27-54。设置PLL\_VIDEO寄存器的bit20:19为2，表示1分频。设置CCM\_ANALOG\_MISC2寄存器的bit31:30为0，也就是VIDOE\_DIV为0，1分频。

我们不是用小数分频器，因此CCM\_ANALOG\_PLL\_VIDEO\_NUM=0，在设置CCM\_ALALOG\_PLL\_VIDEO\_DENOM=0。

 CCM\_CSCDR2寄存器的bit17:15，设置LCDIF\_PRE\_CLK\_SEL，选择LCDIF\_CLK\_ROOT的时钟源，设置为0x2。表示LCDIF时钟源为PLL5。Bit14:12为LCDIF\_PRED位，设置前级分频，可以设置0~7，分别对应1~8分频。

 CCM\_CBCMR寄存器，bit25:23为LCDIF\_PODF，设置第二级分频，可以设置为0~7，分别对应1~8分频。

 继续设置CCM\_CSCDR2寄存器的bit11：9为LCDIF\_CLK\_SEL，选择LCD CLK的最终时钟源，设置为0，LCDIF的最终时钟源来源于pre-muxed LCDIF clock

# 四、LCD驱动程序编写

 1、如果使用的正点原子的开发板和RGB屏幕，那么在驱动LCD之前，要先读取屏幕ID。

# 无、LCD操作API函数编写