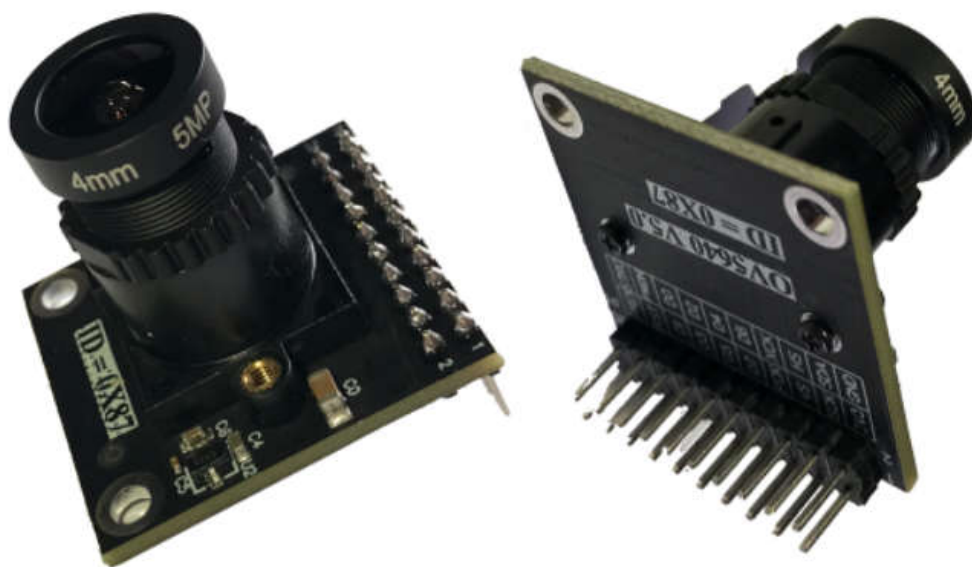


## OV5640 型 500W 高清摄像头模块用户手册

小梅哥编写整理，未经作者许可，严禁用于任何商业用途

OV5640 型 500W 像素高清摄像头模块是武汉芯路恒科技针对高校和企业项目应用，推出的一款高清 CMOS 摄像头模块。模块具有单电源 3~5V 供电输入，5 最高 500W 像素图像输出能力。带双 LED 补光灯，支持自动对焦功能。



### 模块介绍

OV5640\_V5 ( V5 是版本号，下面均以 OV5640 表示该产品 ) 是芯路恒科技推出的一款高性能 500W 像素高清摄像头模块。该模块采用 OmniVision 公司生产的一颗 1/4 英寸 CMOS QXGA ( 2592\*1944 ) 图像

# 武汉芯路恒科技 小梅哥 FPGA 团队

专注于培养您的 FPGA 独立开发能力 开发板 培训 项目研发三位一体

---

传感器：OV5640。OV5640 模块集成了 LDO，单电源 3~5V 供电即可工作，最高 500W 像素图像输出能力。带双 LED 补光灯，支持自动对焦功能，具有非常高的性价比。

OV5640 模块的特点如下：

- 1.4 $\mu$ m\*1.4 $\mu$ m 像素大小，并且使用 OmniBSI 技术以达到更高性能（高灵敏度、低串扰和低噪声）
- 自动图像控制功能：自动曝光（AEC）、自动白平衡（AWB）、自动消除灯光条纹、
- 自动黑电平校准（ABLC）和自动带通滤波器（ABF）等。
- 支持图像质量控制：色饱和度调节、色调调节、gamma 校准、锐度和镜头校准等
- 标准的 SCCB 接口，兼容 IIC 接口
- 支持 RawRGB、RGB(RGB565/RGB555/RGB444)、CCIR656、YUV(422/420)、YCbCr (422) 和压缩图像（JPEG）输出格式
- 支持 QSXGA（500W）图像尺寸输出，以及按比例缩小到其他任何尺寸
- 支持图像缩放、平移和窗口设置
- 支持图像压缩，即可输出 JPEG 图像数据

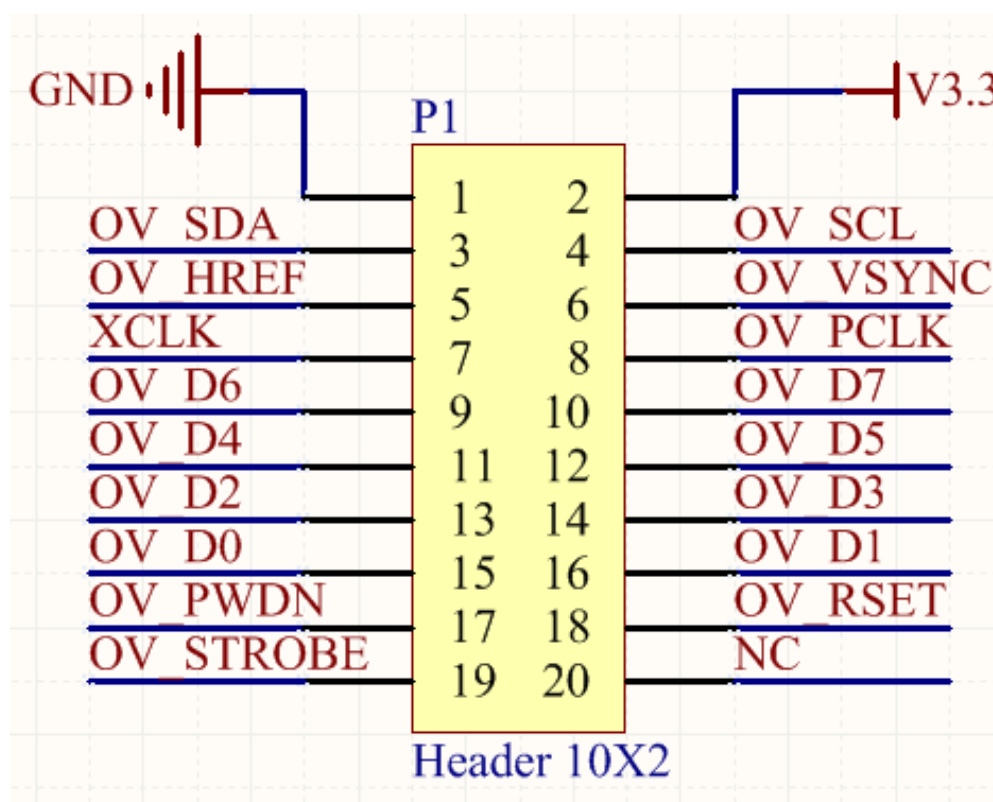
# 武汉芯路恒科技 小梅哥 FPGA 团队

专注于培养您的 FPGA 独立开发能力 开发板 培训 项目研发三位一体

- 支持数字视频接口 ( DVP )
- 自带嵌入式微处理器
- 集成 LDO , 仅提供 3.3V 电源即可正常工作

## 硬件电路说明

模块对用户提供一个 20 针的排针接口，可直接插接到芯路恒科技所有的 FPGA 开发板（包含 AC620、AC6102、AC601、Starter 开发板等）上，并提供有相应的例程，用户可以直接在这些开发板上，对模块进行测试。模块的 2\*10 接口信号图如下所示：



模块接口图

店铺: <https://xiaomeige.taobao.com>

技术博客: <http://www.cnblogs.com/xiaomeige/>

官方网站: [www.corecourse.cn](http://www.corecourse.cn)

技术群组: 615381411

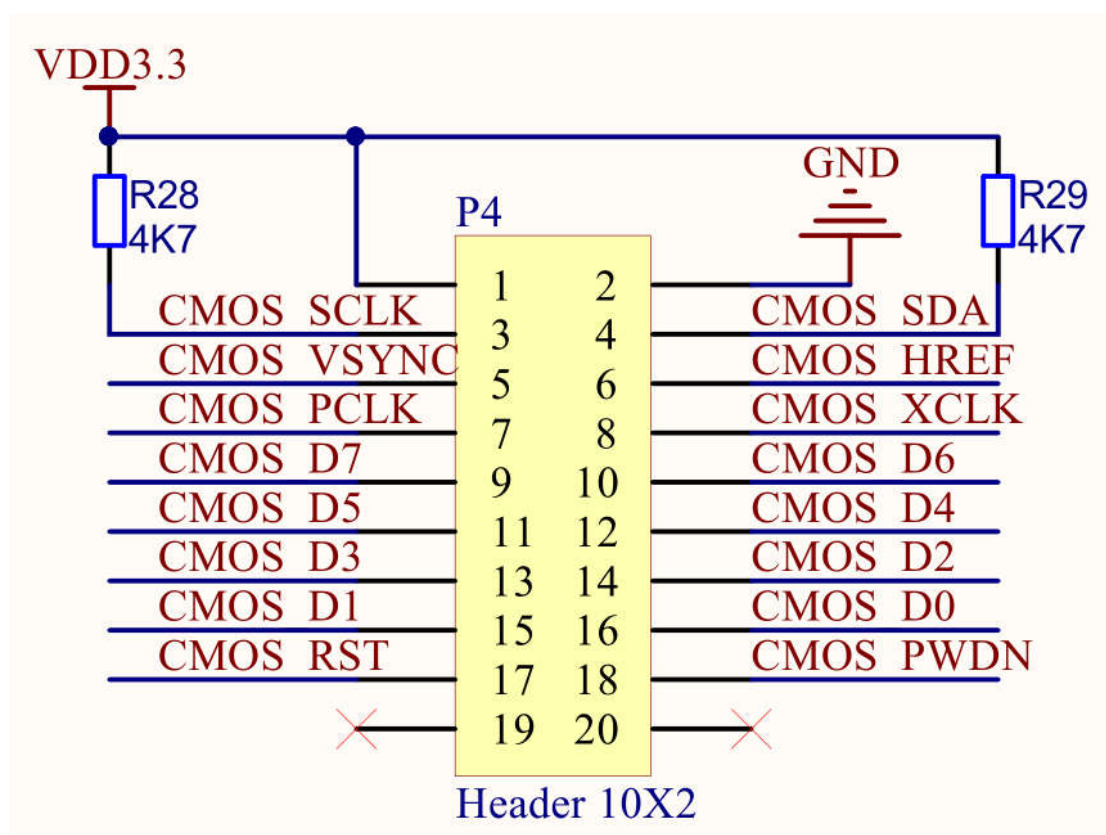
# 武汉芯路恒科技 小梅哥 FPGA 团队

专注于培养您的 FPGA 独立开发能力 开发板 培训 项目研发三位一体

信号	作用描述	信号	作用描述
V3.3	模块供电脚，接 3.3V 电源	OV_PCLK	像素时钟输出
GND	模块地线	OV_PWDN	掉电使能(高有效)
OV_SCL	SCCB 通信时钟信号	OV_VSYNC	帧同步信号输出
OV_SDA	SCCB 通信数据信号	OV_HREF	行同步信号输出
OV_D[7:0]	8 位数据输出	OV_RESET	复位信号(低有效)
XCLK	OV5640 输入参考时钟	OV_STROBE	预留

注意：OV5640 芯片 DVP 接口本身拥有 10 位的数据线，可以输出 10 位的 RAW 数据，但是在大多数情况下，使用高 8 位数据即可，因此模组在设计时，仅使用了 OV5640 芯片的 D9~D2 高 8 位，映射到模组上的 OV\_D7~OV\_D0。

虽然 OV5640 模块设计时主要针对芯路恒科技出产的 FPGA 开发板的 CMOS 摄像头接口，但是从理论上来说，任意一个开发板，只要是带有符合下图形式的接口，都能连接本模块并直接使用：

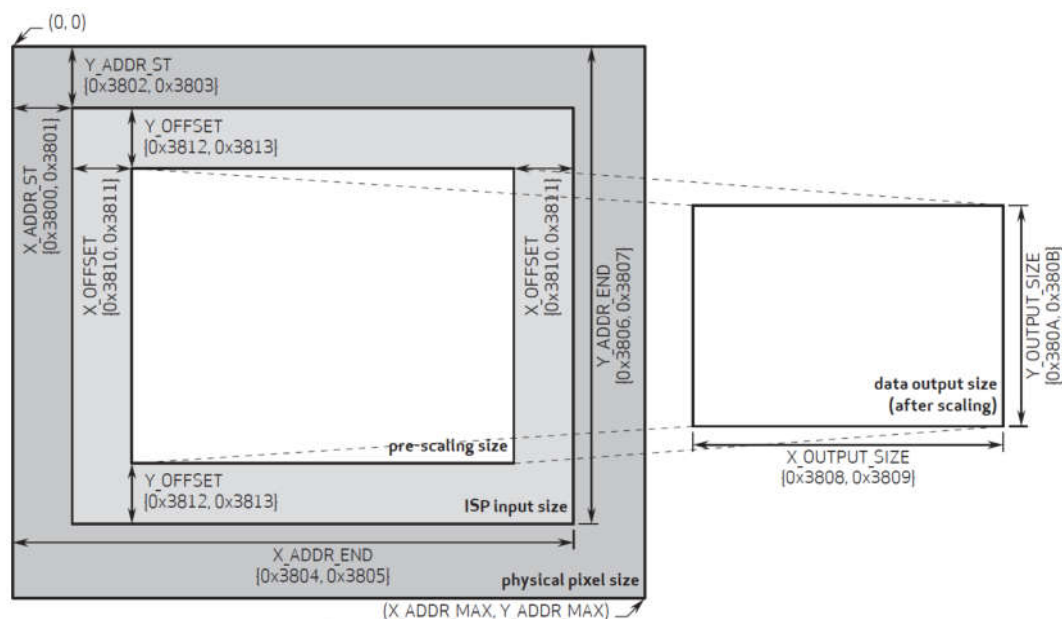


图上，CMOS\_SCLK 和 CMOS\_SDA 需要添加 2~10K 的上拉电阻。如果用户的板卡上没有该上拉电阻也没关系，可以使用 FPGA 管脚自带的片上上拉电阻。关于如何设置 FPGA 片上上拉电阻，可以参考小梅哥的下述博文：

Altera FPGA 开启引脚片上上拉电阻功能：  
<http://www.cnblogs.com/xiaomeige/p/8954743.html>

## 输出图像参数设置

接下来，我们介绍一下 OV5640 的：ISP ( Image Signal Processor ) 输入窗口设置、预缩放窗口设置和输出大小窗口设置，这几个设置与我们的正常使用密切相关，有必要了解一下。他们的设置关系，如下图所示：



## ISP 输入窗口设置 ( ISP input size)

该设置允许用户设置整个传感器区域( physical pixel size , 2632\*1951 ) 的感兴趣部分，也就是在传感器里面开窗 ( X\_ADDR\_ST、Y\_ADDR\_ST、

# 武汉芯路恒科技 小梅哥 FPGA 团队

专注于培养您的 FPGA 独立开发能力 开发板 培训 项目研发三位一体

---

X\_ADDR\_END 和 Y\_ADDR\_END), 开窗范围从 0\*0~2632\*1951 都可以设置, 该窗口所设置的范围, 将输入 ISP 进行处理。

ISP 输入窗口, 通过: 0X3800~0X3807 等 8 个寄存器进行设置, 这些寄存器的定义请看: OV5640\_CSP3\_DS\_2.01\_Ruisipusheng.pdf 这个文档 (下同)。

## 预缩放窗口设置 ( pre-scaling size)

该设置允许用户在 ISP 输入窗口的基础上, 再次设置将要用于缩放的窗口大小。该设置仅在 ISP 输入窗口内进行 x/y 方向的偏移 (X\_OFFSET/Y\_OFFSET)。通过: 0X3810~0X3813 等 4 个寄存器进行设置。

## 输出大小窗口设置 ( data output size)

该窗口是以预缩放窗口为原始大小, 经过内部 DSP 进行缩放处理后, 输出给外部的图像窗口大小。它控制最终的图像输出尺寸 (X\_OUTPUT\_SIZE/Y\_OUTPUT\_SIZE)。通过: 0X3808~0X380B 等 4 个寄存器进行设置。注意: 当输出大小窗口与预缩放窗口比例不一致时, 图像将进行缩放处理 (会变形), 仅当两者比例一致时, 输出比例才是 1:1 (正常)。

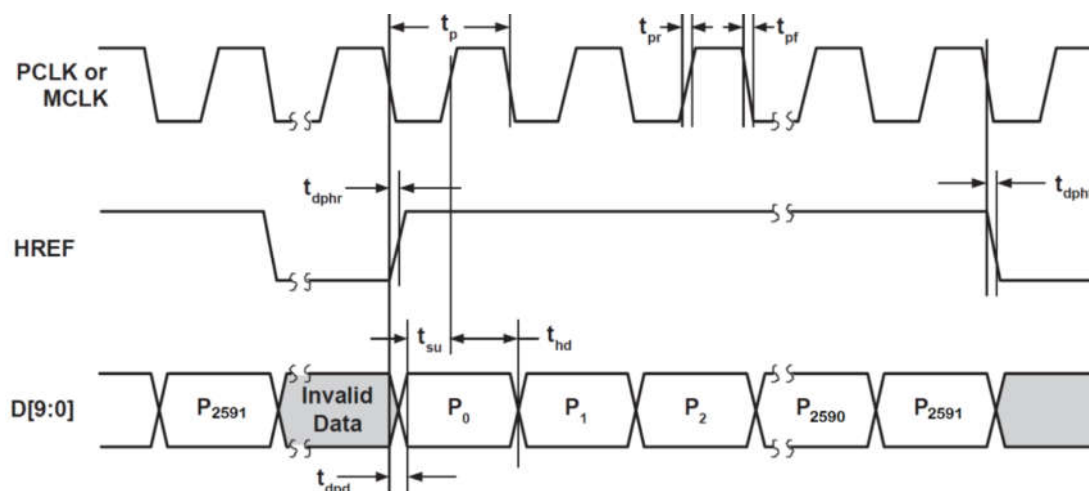
图中, 右侧 data output size 区域, 才是 OV5640 输出给外部的图像尺寸, 也就是显示在显示器上面的图像大小。输出大小窗口与预缩放窗口比例不一致时, 会进行缩放处理, 在显示器上面看到的图像将会变形。

## 图像输出时序

接下来，我们介绍一下 OV5640 的图像数据输出时序。下表为常见的分辨率定义和其对应的实际像素矩阵大小：

定义	分辨率	定义	分辨率
QSXGA	2592*1944	XGA	1024*768
QXGA	2048*1536	SVGA	800*600
UXGA	1600*1200	VGA	640*480
SXGA	1280*1024	QVGA	320*240
WXGA+	1440*900	QQVGA	160*120
WXGA	1280*800		

下图为 OV5640 在 DVP 接口模式下输出图像数据的时序图：





# 武汉芯路恒科技 小梅哥 FPGA 团队

专注于培养您的 FPGA 独立开发能力 开发板 培训 项目研发三位一体

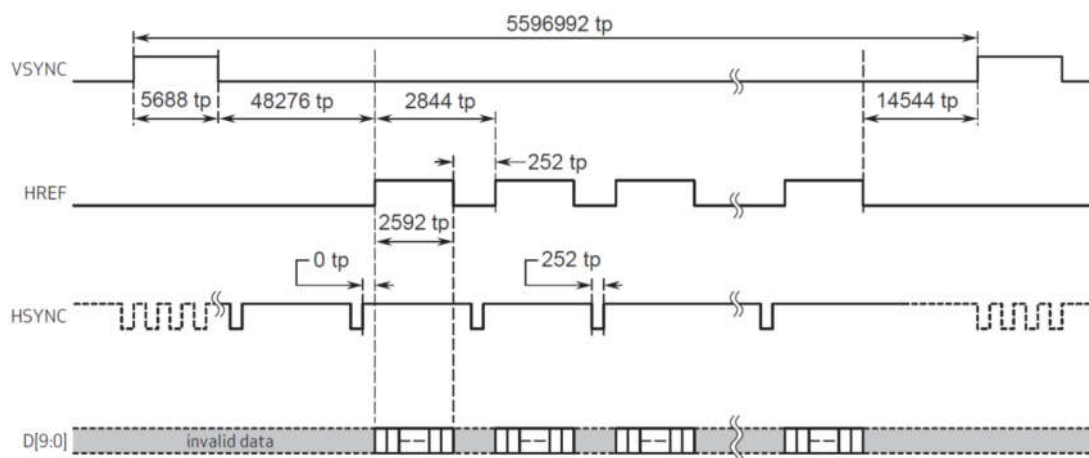
---

- PCLK，即像素时钟，一个 PCLK 时钟，输出一个像素（RAW 模式）或半个像素（RGB564、YUV422 模式）或 1/3 个像素（RGB888、YUV444 模式）
- VSYNC，即帧同步信号。当设置 VSYNC 信号极性为高有效时，VSYNC 信号的高电平期间输出一帧图像。
- HREF /HSYNC，即行同步信号，当设置 HSYNC 信号极性为高有效时，高电平期间输出一行图像。

从上图可以看出，图像数据在 HREF 为高的时候输出，当 HREF 变高后，每一个 PCLK 时钟，输出一个 8 位/10 位数据。我们采用 8 位接口，所以每个 PCLK 输出 1 个字节，且在 RGB565/YUV422 输出格式下，每个 tp=2 个 Tpclk，如果是 Raw 格式，则一个 tp=1 个 Tpclk。比如我们采用 QSXGA 时序，RGB565 格式输出，每 2 个字节组成一个像素的颜色（低字节在前，高字节在后），这样每行输出总共有 2592\*2 个 PCLK 周期，输出 2592\*2 个字节。

再看看帧时序（ QSXGA 模式），如下图所示：





上图清楚的表示了 OV5640 在 QSXGA 模式下的数据输出。我们按照这个时序去读取 OV5640 的数据，就可以得到图像数据。

## 输出图像格式

对于 OV5640 的图像输出数据格式，在不同的应用领域有所不同。在嵌入式系统如单片机、ARM 处理器系统中，一般常用 2 种输出方式：RGB565 和 JPEG。在图像处理系统中，一般常用 RGB565 模式和 YUV422 格式。而在图像采集系统中，为了充分保留图像传感器采集到的原始图像信息，会采用 RAW 格式输出。

当输出 RGB565、YUV422、RAW 格式数据的时候，时序完全就是上面两幅图介绍的关系。而当输出数据是 JPEG 数据的时候，同样也是这种方式输出（所以数据读取方法一模一样），不过 PCLK 数目大大减少了，且不连续，输出的数据是压缩后的 JPEG 数据，输出的 JPEG 数据以：0xFF,0xD8 开头，以 0xFF,0xD9 结尾，且在 0xFF,0xD8 之前，或者 0xFF,0xD9 之后，会有不定数

# 武汉芯路恒科技 小梅哥 FPGA 团队

专注于培养您的 FPGA 独立开发能力 开发板 培训 项目研发三位一体

---

量的其他数据存在（一般是 0），这些数据我们直接忽略即可，将得到的 0xFF,0xD8~0xFF,0xD9 之间的数据，保存为.jpg/.jpeg 文件，就可以直接在电脑上打开看到图像了。

在基于 FPGA 的系统中，一般使用 OV5640 进行图像处理，而进行图像处理时，一般使用 RGB565 模式和 YUV422 格式，JPEG 格式一般很少用到。在对颜色敏感的应用中，常用 RGB565 模式，如仓库物品分拣系统中通过不同的颜色值区分不同的产品。而其他更多的图像处理或图像识别系统中，一般是使用灰度图像进行处理，此时常用的图像模式为 YUV422 模式。RGB565 模式使用 16 位数据来表示一个像素点代表的红绿蓝分量，与我们最熟悉的颜色表示方式一致。而 YUV 模式则是采用色度和亮度分离的模式来表示一个像素点，当仅使用图像中的亮度值来表示一个像素的颜色时，就是传统的灰度图像，老式的黑白电视机就是仅能显示亮度值而无法显示色度值。

## 典型工作模式配置

接下来介绍 OV5640 应用于基于 FPGA 的图像采集系统中各种常见的典型参数配置。

## 基本初始化配置

基本初始化配置表以 800\*480@30FPS 的 RGB565 输出模式为例，具体内容详见文末附录。

## 修改信号极性

为得到正确的图像，OV5640 传感器的视频信号极性与基带芯片或 ISP 必须设置成一致。例如，默认状态下，OV5640 的 VSYNC 是低电平有效，即 VSYNC 信号在输出图像时为低电平。而我们常见的图像捕获系统中，往往以 VSYNC 信号为高电平代表图像数据有效，此时就需要设置 VSYNC 信号的极性为高电平有效。OV5640 可以通过设置地址为 0x4740 号寄存器的值来设置信号的极性，具体设置方法如下表所示：

VSYNC	0x4740[0]	1 – Vsync 为高时输出数据有效 0 – Vsync 为低时输出数据有效
HREF	0x4740[1]	0 – Href 为高时输出数据有效 1 – Href 为低时输出数据有效
PCLK	0x4740[5]	1 – 数据在下降沿输出 0 – 数据在上升沿输出

在我们的所有例程里，都有设置该寄存器的 bit0 为 1，即设置 VSYNC 信号为高电平有效。

## 修改帧率

OV5640 的图像输出帧率可以通过修改地址为 0x3035、0x3036、0x3037 的寄存器的值来修改，该寄存器实际上是设置了 OV5640 片上 PLL 的各种分频和倍频系数，例如在典型配置模式下，当输入时钟 XCLK 的信号频率为

店铺：<https://xiaomeige.taobao.com>

官方网站：[www.corecourse.cn](http://www.corecourse.cn)

技术博客：<http://www.cnblogs.com/xiaomeige/>

技术群组：615381411

# 武汉芯路恒科技 小梅哥 FPGA 团队

专注于培养您的 FPGA 独立开发能力 开发板 培训 项目研发三位一体

---

24MHz 时，设置 0x3035 寄存器的值为 0x21 可设置输出帧率为 30fps，设为 0x41 可设置输出帧率为 15fps、设为 0x81 可设置输出帧率为 7.5fps。

## 图像镜像翻转

因为 OV5640 是一款 BSI 图像传感器，成像光线是从芯片背面射入的，所以原始生成的图像看起来是左右相反的，故此需要对图像做镜像处理使其显示正常。设置镜像和翻转功能是通过设置寄存器 0x3820 和 0x3821 的值实现的。上电时，0x3820 的值默认为 0x40，0x3821 的值默认为 0x00。0x3820 寄存器的 bit2 和 bit1 分别设置 ISP 和传感器的翻转，0x3821 寄存器的 bit2 和 bit1 分别设置 ISP 和传感器的镜像。

一幅正放的图片，人眼看到的图像如下图所示，可以看到，文字都是正常的。

# 武汉芯路恒科技 小梅哥 FPGA 团队

专注于培养您的 FPGA 独立开发能力 开发板 培训 项目研发三位一体

---



当使用 OV5640 采集之后，如果不设置翻转和镜像功能，即寄存器的值为上电默认值， $0x3820 = 0x40$ ,  $0x3821 = 0x00$ 。其输出图像默认如下图所示：

店铺：<https://xiaomeige.taobao.com>

技术博客：<http://www.cnblogs.com/xiaomeige/>

官方网站：[www.corecourse.cn](http://www.corecourse.cn)

技术群组：615381411

# 武汉芯路恒科技 小梅哥 FPGA 团队

专注于培养您的 FPGA 独立开发能力 开发板 培训 项目研发三位一体



以图像下方的一排文字为参考，可以看到，图像相较于原始图像，在水平方向发生了镜像。

如果设置图像不镜像，仅翻转，即寄存器的值为， $0x3820 = 0x40|0x6;0x3821 = 0x00\&0xf9$ 。则是下图所示的样式。

# 武汉芯路恒科技 小梅哥 FPGA 团队

专注于培养您的 FPGA 独立开发能力 开发板 培训 项目研发三位一体



以原始图像下方的一排文字为参考，可以看到，图像相较于原始图像，在垂直方向发生了翻转，下方的文字，翻转到了上方。

如果设置图像既镜像，又翻转，即寄存器的值为， $0x3820 = 0x40|0x6;0x3821 = 0x00|0x06$ 。则是下图所示的样式。



# 武汉芯路恒科技 小梅哥 FPGA 团队

专注于培养您的 FPGA 独立开发能力 开发板 培训 项目研发三位一体



以原始图像下方的一排文字为参考，可以看到，图像左右镜像，且上下翻转了。

因此，当希望图像正常显示时，设置镜像和翻转都关闭即可。设置  $0x3820 = 0x40$  &  $0xf9$ ;  $0x3821 = 0x00$  &  $0xf9$ 。

## 调整图像尺寸

前面提到，设置输出图像尺寸可以通过设置 ISP 在传感器上的开窗，预缩放偏移和输出图像大小窗口。最常用的是设置输出图像大小窗口设置，地址为  $0x3808$  和  $0x3809$  的寄存器设置输出图像的高度，地址为  $0x380a$  和  $0x380b$  的寄存器设置输出图像的宽度。例如，需要设置输出图像大小为  $800 \times 480$  分辨

# 武汉芯路恒科技 小梅哥 FPGA 团队

专注于培养您的 FPGA 独立开发能力 开发板 培训 项目研发三位一体

率，则设置输出图像高度寄存器的值为 0x0320 ( 800d )，设置输出图像宽度寄存器的值为 0x01e0 ( 480d )。即设置：

0x3808 = 0x03; // DVPHO 800      0x3809 = 0x20; // DVPHO

0x380a = 0x01; // DVPVO480      0x380b = 0xe0; // DVPVO

## 调整图像输出模式

OV5640 输出图像制式支持多种制式，如 RGB、YUV、RAW。而 RGB 制式又包括 RGB888、RGB565、RGB555、RGB444 模式等，YUV 制式包括 YUV444、YUV422、YUV420 模式等，而每个模式下，又根据输出像素的各个字节代表的不同意义，又分为多种模式，例如对于 RGB565 模式，连续的两个字节代表一个像素的颜色值，在代表一个像素点的 2 个字节数据中，哪几位代表红色分量，哪几位代表绿色分量，哪几位代表蓝色分量，也是通过寄存器可以设置的。具体模式设置是通过 0x4300 这个寄存器设置的。这里仅介绍两种常用的模式设置，RGB565 和 YUV422。其他模式，用户可以通过查看 OV5640\_CSP3\_DS\_2.01\_Ruisipusheng.pdf 中相关描述。

0x4300 寄存器共 8 位，其中 bit[7:4]设置图像输出模式，bit[3:0]设置每个模式下输出像素内容的顺序。下表为 bit[7:4]的值和对应的输出模式的关系。

值	输出模式	值	输出模式	值	输出模式
0x0	RAW	0x4	YUV420	0x8	RGB555 format2
0x1	Y8	0x5	YUV420(MIPI)	0x9	RGB444 format1

店铺：<https://xiaomeige.taobao.com>

技术博客：<http://www.cnblogs.com/xiaomeige/>

官方网站：[www.corecourse.cn](http://www.corecourse.cn)

技术群组：615381411

# 武汉芯路恒科技 小梅哥 FPGA 团队

专注于培养您的 FPGA 独立开发能力 开发板 培训 项目研发三位一体

0x2	YUV444/RGB888	0x6	RGB565	0xa	RGB444 format2
0x3	YUV422	0x7	RGB555 format1	0xf	Bypass

说明：关于很多读者所关心的 JPEG 模式，并不是在这个寄存器中设置的，JPEG 输出是另外有单独的 JPEG 压缩引擎，对 YUV422 或 YUV420 格式的图像编码得到的。如果需要输出 JPEG 格式，则本寄存器应该设置高 4 位的值为 0x3 或 0x4，即选择 YUV422 或 YUV420 格式，然后再设置 JPEG 相关的寄存器。对于 JPEG 模式本手册不做讨论。

当设置了图像输出模式后，可以通过设置该寄存器的 bit[3:0]来设置输出图像的顺序。例如，对于 RGB565 模式(bit[7:4]=0x6)，设置 bit[3:0]为不同的值则可以实现不同的输出序列，如下表所示：

值	高字节内容定义	低字节内容定义
0x0	{b[4:0],g[5:3]}	{g[2:0],r[4:0]}
0x1	{r[4:0],g[5:3]}	{g[2:0],b[4:0]}
0x2	{g[4:0],r[5:3]}	{r[2:0],b[4:0]}
0x3	{b[4:0],r[5:3]}	{r[2:0],g[4:0]}
0x4	{g[4:0],b[5:3]}	{b[2:0],r[4:0]}
0x5	{r[4:0],b[5:3]}	{b[2:0],g[4:0]}
0x6~0xe	Not allowed	Not allowed
0xf	{g[2:0],b[4:0]}	{r[4:0],g[5:3]}

店铺：<https://xiaomeige.taobao.com>

技术博客：<http://www.cnblogs.com/xiaomeige/>

官方网站：[www.corecourse.cn](http://www.corecourse.cn)

技术群组：615381411

# 武汉芯路恒科技 小梅哥 FPGA 团队

专注于培养您的 FPGA 独立开发能力 开发板 培训 项目研发三位一体

对于我们常见的应用来说，一般是使用 RGB 或 BGR 序列，即设置寄存器的 bit[3:0] 的值为 0x0 或 0x1。

下表为 YUV422 模式 ( bit[7:4]=0x3 ) 时 bit[3:0] 的值与对应的输出序列的关系：

0x0	YUYV	0x3	VYUY
0x1	YVYU	0x4~0xe	Not Allowed
0x2	UYVY	0xf	UYVY

在提供的 OV5640 的实例中，基于 AC6102 开发板的 UVC 摄像头实验设置的是 1280\*720@30fps 的输出速率，且图像模式为 UVC 要求的 YUV 模式。而在其他的基于 AC620 或 AC6102 开发板上直接使用 VGA 显示器或 TFT 屏显示的时候，设置的是 RGB565 模式。

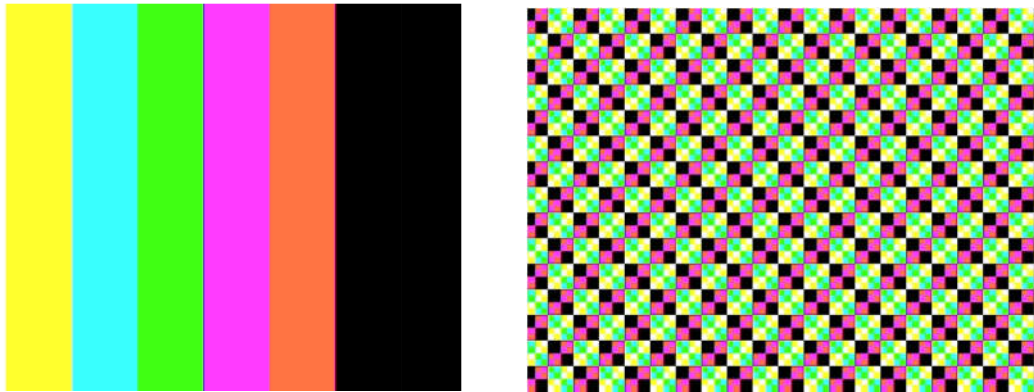
## 彩条测试模式

通过设置地址为 0x503d 寄存器中相应位的值，可以设置 OV5640 输出的图像内容是传感器采集到的图像还是测试信号发生器模块产生的测试信号。

0x503d 寄存器的最高位 bit7 为测试信号发生器的开关控制位，设置为 1 即可开启测试信号发生器。bit[1:0] 为测试信号类型选择寄存器不同的值可以选择不同的测试模式，如彩条 ( Color Bar )、棋盘格 ( Color square )。

# 武汉芯路恒科技 小梅哥 FPGA 团队

专注于培养您的 FPGA 独立开发能力 开发板 培训 项目研发三位一体



例如要设置彩条测试模式，设置 0x503d 寄存器的值为 0x80 即可。

## 附录：基于 800\*480@30FPS 的 RGB565 输出模式的基本配置表

详细内容可以参见 AC620\_OV5640V2\_SDRAM\_TFT800\_30FPS 例程代码

中 I2C\_OV5640\_RGB565\_Config.v 文件内容。

```
//15fps VGA YUV output
// 24MHz input clock, 24MHz PCLK
SET_OV5640 + 0 : LUT_DATA = 24'h3103_11; // system clock from
pad, bit[1]
SET_OV5640 + 1 : LUT_DATA = 24'h3008_82; // software reset,
bit[7]
SET_OV5640 + 2 : LUT_DATA = 24'h3008_42; // software power down,
bit[6]
SET_OV5640 + 3 : LUT_DATA = 24'h3103_03; // system clock from
PLL, bit[1]
SET_OV5640 + 4 : LUT_DATA = 24'h3017_ff; // FREX, Vsync, HREF,
PCLK, D[9:6] output enable
SET_OV5640 + 5 : LUT_DATA = 24'h3018_ff; // D[5:0], GPIO[1:0]
output enable
SET_OV5640 + 6 : LUT_DATA = 24'h3034_1a; // MIPI 10-bit
```

店铺: <https://xiaomeige.taobao.com>

技术博客: <http://www.cnblogs.com/xiaomeige/>

官方网站: [www.corecourse.cn](http://www.corecourse.cn)

技术群组: 615381411

# 武汉芯路恒科技 小梅哥 FPGA 团队

专注于培养您的 FPGA 独立开发能力 开发板 培训 项目研发三位一体

```
SET_OV5640 + 7 : LUT_DATA = 24'h3037_13; // PLL root divider,
bit[4], PLL pre-divider, bit[3:0]
SET_OV5640 + 8 : LUT_DATA = 24'h3108_01; // PCLK root divider,
bit[5:4], SCLK2x root divider, bit[3:2]
SET_OV5640 + 9 : LUT_DATA = 24'h3630_36; // SCLK root divider,
bit[1:0]
SET_OV5640 + 10 : LUT_DATA = 24'h3631_0e;
SET_OV5640 + 11 : LUT_DATA = 24'h3632_e2;
SET_OV5640 + 12 : LUT_DATA = 24'h3633_12;
SET_OV5640 + 13 : LUT_DATA = 24'h3621_e0;
SET_OV5640 + 14 : LUT_DATA = 24'h3704_a0;
SET_OV5640 + 15 : LUT_DATA = 24'h3703_5a;
SET_OV5640 + 16 : LUT_DATA = 24'h3715_78;
SET_OV5640 + 17 : LUT_DATA = 24'h3717_01;
SET_OV5640 + 18 : LUT_DATA = 24'h370b_60;
SET_OV5640 + 19 : LUT_DATA = 24'h3705_1a;
SET_OV5640 + 20 : LUT_DATA = 24'h3905_02;
SET_OV5640 + 21 : LUT_DATA = 24'h3906_10;
SET_OV5640 + 22 : LUT_DATA = 24'h3901_0a;
SET_OV5640 + 23 : LUT_DATA = 24'h3731_12;
SET_OV5640 + 24 : LUT_DATA = 24'h3600_08; // VCM control
SET_OV5640 + 25 : LUT_DATA = 24'h3601_33; // VCM control
SET_OV5640 + 26 : LUT_DATA = 24'h302d_60; // system control
SET_OV5640 + 27 : LUT_DATA = 24'h3620_52;
SET_OV5640 + 28 : LUT_DATA = 24'h371b_20;
SET_OV5640 + 29 : LUT_DATA = 24'h471c_50;
SET_OV5640 + 30 : LUT_DATA = 24'h3a13_43; // pre-gain = 1.047x
SET_OV5640 + 31 : LUT_DATA = 24'h3a18_00; // gain ceiling
SET_OV5640 + 32 : LUT_DATA = 24'h3a19_f8; // gain ceiling = 15.5x
SET_OV5640 + 33 : LUT_DATA = 24'h3635_13;
SET_OV5640 + 34 : LUT_DATA = 24'h3636_03;
SET_OV5640 + 35 : LUT_DATA = 24'h3634_40;
SET_OV5640 + 36 : LUT_DATA = 24'h3622_01;
// 50/60Hz detection 50/60Hz 灯光条纹过滤
SET_OV5640 + 37 : LUT_DATA = 24'h3c01_34; // Band auto, bit[7]
SET_OV5640 + 38 : LUT_DATA = 24'h3c04_28; // threshold low sum
SET_OV5640 + 39 : LUT_DATA = 24'h3c05_98; // threshold high sum
SET_OV5640 + 40 : LUT_DATA = 24'h3c06_00; // light meter 1
threshold[15:8]
SET_OV5640 + 41 : LUT_DATA = 24'h3c07_08; // light meter 1
threshold[7:0]
```

店铺: <https://xiaomeige.taobao.com>

技术博客: <http://www.cnblogs.com/xiaomeige/>

官方网站: [www.corecourse.cn](http://www.corecourse.cn)

技术群组: 615381411

# 武汉芯路恒科技 小梅哥 FPGA 团队

专注于培养您的 FPGA 独立开发能力 开发板 培训 项目研发三位一体

```
SET_OV5640 + 42 : LUT_DATA = 24'h3c08_00; // light meter 2
threshold[15:8]
SET_OV5640 + 43 : LUT_DATA = 24'h3c09_1c; // light meter 2
threshold[7:0]
SET_OV5640 + 44 : LUT_DATA = 24'h3c0a_9c; // sample number[15:8]
SET_OV5640 + 45 : LUT_DATA = 24'h3c0b_40; // sample number[7:0]
SET_OV5640 + 46 : LUT_DATA = 24'h3810_00; // Timing Hoffset[11:8]
SET_OV5640 + 47 : LUT_DATA = 24'h3811_10; // Timing Hoffset[7:0]
SET_OV5640 + 48 : LUT_DATA = 24'h3812_00; // Timing Voffset[10:8]
SET_OV5640 + 49 : LUT_DATA = 24'h3708_64;
SET_OV5640 + 50 : LUT_DATA = 24'h4001_02; // BLC start from line
2
SET_OV5640 + 51 : LUT_DATA = 24'h4005_1a; // BLC always update
SET_OV5640 + 52 : LUT_DATA = 24'h3000_00; // enable blocks
SET_OV5640 + 53 : LUT_DATA = 24'h3004_ff; // enable clocks
SET_OV5640 + 54 : LUT_DATA = 24'h300e_58; // MIPI power down, DVP
enable
SET_OV5640 + 55 : LUT_DATA = 24'h302e_00;
SET_OV5640 + 56 : LUT_DATA = 24'h4300_61; // RGB565
SET_OV5640 + 57 : LUT_DATA = 24'h501f_01; // RGB565
SET_OV5640 + 58 : LUT_DATA = 24'h440e_00;
SET_OV5640 + 59 : LUT_DATA = 24'h5000_a7; // Lenc on, raw gamma
on, BPC on, WPC on, CIP on
// AEC target 自动曝光控制
SET_OV5640 + 60 : LUT_DATA = 24'h3a0f_30; // stable range in high
SET_OV5640 + 61 : LUT_DATA = 24'h3a10_28; // stable range in low
SET_OV5640 + 62 : LUT_DATA = 24'h3a1b_30; // stable range out
high
SET_OV5640 + 63 : LUT_DATA = 24'h3a1e_26; // stable range out low
SET_OV5640 + 64 : LUT_DATA = 24'h3a11_60; // fast zone high
SET_OV5640 + 65 : LUT_DATA = 24'h3a1f_14; // fast zone low
// Lens correction for ? 镜头补偿
SET_OV5640 + 66 : LUT_DATA = 24'h5800_23;
SET_OV5640 + 67 : LUT_DATA = 24'h5801_14;
SET_OV5640 + 68 : LUT_DATA = 24'h5802_0f;
SET_OV5640 + 69 : LUT_DATA = 24'h5803_0f;
SET_OV5640 + 70 : LUT_DATA = 24'h5804_12;
SET_OV5640 + 71 : LUT_DATA = 24'h5805_26;
SET_OV5640 + 72 : LUT_DATA = 24'h5806_0c;
SET_OV5640 + 73 : LUT_DATA = 24'h5807_08;
SET_OV5640 + 74 : LUT_DATA = 24'h5808_05;
```

店铺: <https://xiaomeige.taobao.com>

技术博客: <http://www.cnblogs.com/xiaomeige/>

官方网站: [www.corecourse.cn](http://www.corecourse.cn)

技术群组: 615381411



# 武汉芯路恒科技 小梅哥 FPGA 团队

专注于培养您的 FPGA 独立开发能力 开发板 培训 项目研发三位一体

```
SET_OV5640 + 75 : LUT_DATA = 24'h5809_05;
SET_OV5640 + 76 : LUT_DATA = 24'h580a_08;
SET_OV5640 + 77 : LUT_DATA = 24'h580b_0d;
SET_OV5640 + 78 : LUT_DATA = 24'h580c_08;
SET_OV5640 + 79 : LUT_DATA = 24'h580d_03;
SET_OV5640 + 80 : LUT_DATA = 24'h580e_00;
SET_OV5640 + 81 : LUT_DATA = 24'h580f_00;
SET_OV5640 + 82 : LUT_DATA = 24'h5810_03;
SET_OV5640 + 83 : LUT_DATA = 24'h5811_09;
SET_OV5640 + 84 : LUT_DATA = 24'h5812_07;
SET_OV5640 + 85 : LUT_DATA = 24'h5813_03;
SET_OV5640 + 86 : LUT_DATA = 24'h5814_00;
SET_OV5640 + 87 : LUT_DATA = 24'h5815_01;
SET_OV5640 + 88 : LUT_DATA = 24'h5816_03;
SET_OV5640 + 89 : LUT_DATA = 24'h5817_08;
SET_OV5640 + 90 : LUT_DATA = 24'h5818_0d;
SET_OV5640 + 91 : LUT_DATA = 24'h5819_08;
SET_OV5640 + 92 : LUT_DATA = 24'h581a_05;
SET_OV5640 + 93 : LUT_DATA = 24'h581b_06;
SET_OV5640 + 94 : LUT_DATA = 24'h581c_08;
SET_OV5640 + 95 : LUT_DATA = 24'h581d_0e;
SET_OV5640 + 96 : LUT_DATA = 24'h581e_29;
SET_OV5640 + 97 : LUT_DATA = 24'h581f_17;
SET_OV5640 + 98 : LUT_DATA = 24'h5820_11;
SET_OV5640 + 99 : LUT_DATA = 24'h5821_11;
SET_OV5640 + 100 : LUT_DATA = 24'h5822_15;
SET_OV5640 + 101 : LUT_DATA = 24'h5823_28;
SET_OV5640 + 102 : LUT_DATA = 24'h5824_46;
SET_OV5640 + 103 : LUT_DATA = 24'h5825_26;
SET_OV5640 + 104 : LUT_DATA = 24'h5826_08;
SET_OV5640 + 105 : LUT_DATA = 24'h5827_26;
SET_OV5640 + 106 : LUT_DATA = 24'h5828_64;
SET_OV5640 + 107 : LUT_DATA = 24'h5829_26;
SET_OV5640 + 108 : LUT_DATA = 24'h582a_24;
SET_OV5640 + 109 : LUT_DATA = 24'h582b_22;
SET_OV5640 + 110 : LUT_DATA = 24'h582c_24;
SET_OV5640 + 111 : LUT_DATA = 24'h582d_24;
SET_OV5640 + 112 : LUT_DATA = 24'h582e_06;
SET_OV5640 + 113 : LUT_DATA = 24'h582f_22;
SET_OV5640 + 114 : LUT_DATA = 24'h5830_40;
SET_OV5640 + 115 : LUT_DATA = 24'h5831_42;
```

店铺: <https://xiaomeige.taobao.com>

技术博客: <http://www.cnblogs.com/xiaomeige/>

官方网站: [www.corecourse.cn](http://www.corecourse.cn)

技术群组: 615381411

# 武汉芯路恒科技 小梅哥 FPGA 团队

专注于培养您的 FPGA 独立开发能力 开发板 培训 项目研发三位一体

```
SET_OV5640 + 116: LUT_DATA = 24'h5832_24;
SET_OV5640 + 117: LUT_DATA = 24'h5833_26;
SET_OV5640 + 118: LUT_DATA = 24'h5834_24;
SET_OV5640 + 119: LUT_DATA = 24'h5835_22;
SET_OV5640 + 120: LUT_DATA = 24'h5836_22;
SET_OV5640 + 121: LUT_DATA = 24'h5837_26;
SET_OV5640 + 122: LUT_DATA = 24'h5838_44;
SET_OV5640 + 123: LUT_DATA = 24'h5839_24;
SET_OV5640 + 124: LUT_DATA = 24'h583a_26;
SET_OV5640 + 125: LUT_DATA = 24'h583b_28;
SET_OV5640 + 126: LUT_DATA = 24'h583c_42;
SET_OV5640 + 127: LUT_DATA = 24'h583d_ce; // lenc BR offset
// AWB 自动白平衡
SET_OV5640 + 128: LUT_DATA = 24'h5180_ff; // AWB B block
SET_OV5640 + 129: LUT_DATA = 24'h5181_f2; // AWB control
SET_OV5640 + 130: LUT_DATA = 24'h5182_00; // [7:4] max local
counter, [3:0] max fast counter
SET_OV5640 + 131: LUT_DATA = 24'h5183_14; // AWB advanced
SET_OV5640 + 132: LUT_DATA = 24'h5184_25;
SET_OV5640 + 133: LUT_DATA = 24'h5185_24;
SET_OV5640 + 134: LUT_DATA = 24'h5186_09;
SET_OV5640 + 135: LUT_DATA = 24'h5187_09;
SET_OV5640 + 136: LUT_DATA = 24'h5188_09;
SET_OV5640 + 137: LUT_DATA = 24'h5189_75;
SET_OV5640 + 138: LUT_DATA = 24'h518a_54;
SET_OV5640 + 139: LUT_DATA = 24'h518b_e0;
SET_OV5640 + 140: LUT_DATA = 24'h518c_b2;
SET_OV5640 + 141: LUT_DATA = 24'h518d_42;
SET_OV5640 + 142: LUT_DATA = 24'h518e_3d;
SET_OV5640 + 143: LUT_DATA = 24'h518f_56;
SET_OV5640 + 144: LUT_DATA = 24'h5190_46;
SET_OV5640 + 145: LUT_DATA = 24'h5191_f8; // AWB top limit
SET_OV5640 + 146: LUT_DATA = 24'h5192_04; // AWB bottom limit
SET_OV5640 + 147: LUT_DATA = 24'h5193_70; // red limit
SET_OV5640 + 148: LUT_DATA = 24'h5194_f0; // green limit
SET_OV5640 + 149: LUT_DATA = 24'h5195_f0; // blue limit
SET_OV5640 + 150: LUT_DATA = 24'h5196_03; // AWB control
SET_OV5640 + 151: LUT_DATA = 24'h5197_01; // local limit
SET_OV5640 + 152: LUT_DATA = 24'h5198_04;
SET_OV5640 + 153: LUT_DATA = 24'h5199_12;
SET_OV5640 + 154: LUT_DATA = 24'h519a_04;
```

店铺: <https://xiaomeige.taobao.com>

技术博客: <http://www.cnblogs.com/xiaomeige/>

官方网站: [www.corecourse.cn](http://www.corecourse.cn)

技术群组: 615381411

# 武汉芯路恒科技 小梅哥 FPGA 团队

专注于培养您的 FPGA 独立开发能力 开发板 培训 项目研发三位一体

```
SET_OV5640 + 155: LUT_DATA = 24'h519b_00;
SET_OV5640 + 156: LUT_DATA = 24'h519c_06;
SET_OV5640 + 157: LUT_DATA = 24'h519d_82;
SET_OV5640 + 158: LUT_DATA = 24'h519e_38; // AWB control
// Gamma 伽玛曲线
SET_OV5640 + 159: LUT_DATA = 24'h5480_01; // Gamma bias plus on,
bit[0]
SET_OV5640 + 160: LUT_DATA = 24'h5481_08;
SET_OV5640 + 161: LUT_DATA = 24'h5482_14;
SET_OV5640 + 162: LUT_DATA = 24'h5483_28;
SET_OV5640 + 163: LUT_DATA = 24'h5484_51;
SET_OV5640 + 164: LUT_DATA = 24'h5485_65;
SET_OV5640 + 165: LUT_DATA = 24'h5486_71;
SET_OV5640 + 166: LUT_DATA = 24'h5487_7d;
SET_OV5640 + 167: LUT_DATA = 24'h5488_87;
SET_OV5640 + 168: LUT_DATA = 24'h5489_91;
SET_OV5640 + 169: LUT_DATA = 24'h548a_9a;
SET_OV5640 + 170: LUT_DATA = 24'h548b_aa;
SET_OV5640 + 171: LUT_DATA = 24'h548c_b8;
SET_OV5640 + 172: LUT_DATA = 24'h548d_cd;
SET_OV5640 + 173: LUT_DATA = 24'h548e_dd;
SET_OV5640 + 174: LUT_DATA = 24'h548f_ea;
SET_OV5640 + 175: LUT_DATA = 24'h5490_1d;
// color matrix 色彩矩阵
SET_OV5640 + 176: LUT_DATA = 24'h5381_1e; // CMX1 for Y
SET_OV5640 + 177: LUT_DATA = 24'h5382_5b; // CMX2 for Y
SET_OV5640 + 178: LUT_DATA = 24'h5383_08; // CMX3 for Y
SET_OV5640 + 179: LUT_DATA = 24'h5384_0a; // CMX4 for U
SET_OV5640 + 180: LUT_DATA = 24'h5385_7e; // CMX5 for U
SET_OV5640 + 181: LUT_DATA = 24'h5386_88; // CMX6 for U
SET_OV5640 + 182: LUT_DATA = 24'h5387_7c; // CMX7 for V
SET_OV5640 + 183: LUT_DATA = 24'h5388_6c; // CMX8 for V
SET_OV5640 + 184: LUT_DATA = 24'h5389_10; // CMX9 for V
SET_OV5640 + 185: LUT_DATA = 24'h538a_01; // sign[9]
SET_OV5640 + 186: LUT_DATA = 24'h538b_98; // sign[8:1]
// UV adjust UV 色彩饱和度调整
SET_OV5640 + 187: LUT_DATA = 24'h5580_06; // saturation on,
bit[1]
SET_OV5640 + 188: LUT_DATA = 24'h5583_40;
SET_OV5640 + 189: LUT_DATA = 24'h5584_10;
SET_OV5640 + 190: LUT_DATA = 24'h5589_10;
```

店铺: <https://xiaomeige.taobao.com>

官方网站: [www.corecourse.cn](http://www.corecourse.cn)

技术博客: <http://www.cnblogs.com/xiaomeige/>

技术群组: 615381411

# 武汉芯路恒科技 小梅哥 FPGA 团队

专注于培养您的 FPGA 独立开发能力 开发板 培训 项目研发三位一体

```
SET_OV5640 + 191: LUT_DATA = 24'h558a_00;
SET_OV5640 + 192: LUT_DATA = 24'h558b_f8;
SET_OV5640 + 193: LUT_DATA = 24'h501d_40; // enable manual offset
of contrast
// CIP 锐化和降噪
SET_OV5640 + 194: LUT_DATA = 24'h5300_08; // CIP sharpen MT
threshold 1
SET_OV5640 + 195: LUT_DATA = 24'h5301_30; // CIP sharpen MT
threshold 2
SET_OV5640 + 196: LUT_DATA = 24'h5302_10; // CIP sharpen MT
offset 1
SET_OV5640 + 197: LUT_DATA = 24'h5303_00; // CIP sharpen MT
offset 2
SET_OV5640 + 198: LUT_DATA = 24'h5304_08; // CIP DNS threshold 1
SET_OV5640 + 199: LUT_DATA = 24'h5305_30; // CIP DNS threshold 2
SET_OV5640 + 200: LUT_DATA = 24'h5306_08; // CIP DNS offset 1
SET_OV5640 + 201: LUT_DATA = 24'h5307_16; // CIP DNS offset 2
SET_OV5640 + 202: LUT_DATA = 24'h5309_08; // CIP sharpen TH
threshold 1
SET_OV5640 + 203: LUT_DATA = 24'h530a_30; // CIP sharpen TH
threshold 2
SET_OV5640 + 204: LUT_DATA = 24'h530b_04; // CIP sharpen TH
offset 1
SET_OV5640 + 205: LUT_DATA = 24'h530c_06; // CIP sharpen TH
offset 2
SET_OV5640 + 206: LUT_DATA = 24'h5025_00;
SET_OV5640 + 207: LUT_DATA = 24'h3008_02; // wake up from
standby, bit[6]

// 800*480, 30fps
// input clock 24Mhz,
SET_OV5640 + 208: LUT_DATA = 24'h3035_21; // PLL 21:30fps
41:15fps 81:7.5fps
SET_OV5640 + 209: LUT_DATA = 24'h3036_69; // PLL
SET_OV5640 + 210: LUT_DATA = 24'h3c07_07; // lightmeter 1
threshold[7:0]
SET_OV5640 + 211: LUT_DATA = 24'h3820_47; // flip
SET_OV5640 + 212: LUT_DATA = 24'h3821_01; // no mirror
SET_OV5640 + 213: LUT_DATA = 24'h3814_31; // timing X inc
SET_OV5640 + 214: LUT_DATA = 24'h3815_31; // timing Y inc
```

店铺: <https://xiaomeige.taobao.com>

技术博客: <http://www.cnblogs.com/xiaomeige/>

官方网站: [www.corecourse.cn](http://www.corecourse.cn)

技术群组: 615381411

# 武汉芯路恒科技 小梅哥 FPGA 团队

专注于培养您的 FPGA 独立开发能力 开发板 培训 项目研发三位一体

```
SET_OV5640 + 215: LUT_DATA = 24'h3800_00; // HS
SET_OV5640 + 216: LUT_DATA = 24'h3801_00; // HS
SET_OV5640 + 217: LUT_DATA = 24'h3802_00; // VS
SET_OV5640 + 218: LUT_DATA = 24'h3803_fa; // VS
SET_OV5640 + 219: LUT_DATA = 24'h3804_0a; // HW SET_OV5640 + :
LUT_DATA = HE}
SET_OV5640 + 220: LUT_DATA = 24'h3805_3f; // HW SET_OV5640 + :
LUT_DATA = HE}
SET_OV5640 + 221: LUT_DATA = 24'h3806_06; // VH SET_OV5640 + :
LUT_DATA = VE}
SET_OV5640 + 222: LUT_DATA = 24'h3807_a9; // VH SET_OV5640 + :
LUT_DATA = VE}
SET_OV5640 + 223: LUT_DATA = 24'h3808_03; // DVPHO 800
SET_OV5640 + 224: LUT_DATA = 24'h3809_20; // DVPHO
SET_OV5640 + 225: LUT_DATA = 24'h380a_01; // DVPVO 480
SET_OV5640 + 226: LUT_DATA = 24'h380b_e0; // DVPVO
SET_OV5640 + 227: LUT_DATA = 24'h380c_07; // HTS
SET_OV5640 + 228: LUT_DATA = 24'h380d_64; // HTS
SET_OV5640 + 229: LUT_DATA = 24'h380e_02; // VTS
SET_OV5640 + 230: LUT_DATA = 24'h380f_e4; // VTS
SET_OV5640 + 231: LUT_DATA = 24'h3813_04; // timing V offset
SET_OV5640 + 232: LUT_DATA = 24'h3618_00;
SET_OV5640 + 233: LUT_DATA = 24'h3612_29;
SET_OV5640 + 234: LUT_DATA = 24'h3709_52;
SET_OV5640 + 235: LUT_DATA = 24'h370c_03;
SET_OV5640 + 236: LUT_DATA = 24'h3a02_02; // 60Hz max exposure
SET_OV5640 + 237: LUT_DATA = 24'h3a03_e0; // 60Hz max exposure
SET_OV5640 + 238: LUT_DATA = 24'h3a14_02; // 50Hz max exposure
SET_OV5640 + 239: LUT_DATA = 24'h3a15_e0; // 50Hz max exposure
SET_OV5640 + 240: LUT_DATA = 24'h4004_02; // BLC line number
SET_OV5640 + 241: LUT_DATA = 24'h3002_1c; // reset JFIFO, SFIFO,
JPG
SET_OV5640 + 242: LUT_DATA = 24'h3006_c3; // disable clock of
JPEG2x, JPEG
SET_OV5640 + 243: LUT_DATA = 24'h4713_03; // JPEG mode 3
SET_OV5640 + 244: LUT_DATA = 24'h4407_04; // Quantization scale
SET_OV5640 + 245: LUT_DATA = 24'h460b_37;
SET_OV5640 + 246: LUT_DATA = 24'h460c_20;
SET_OV5640 + 247: LUT_DATA = 24'h4837_16; // MIPI global timing
SET_OV5640 + 248: LUT_DATA = 24'h3824_04; // PCLK manual divider
```

店铺: <https://xiaomeige.taobao.com>

技术博客: <http://www.cnblogs.com/xiaomeige/>

官方网站: [www.corecourse.cn](http://www.corecourse.cn)

技术群组: 615381411

# 武汉芯路恒科技 小梅哥 FPGA 团队

专注于培养您的 FPGA 独立开发能力 开发板 培训 项目研发三位一体

```
SET_OV5640 + 249: LUT_DATA = 24'h5001_83; // SDE on, CMX on, AWB  
on  
SET_OV5640 + 250: LUT_DATA = 24'h3503_00; // AEC/AGC on  
  
SET_OV5640 + 251: LUT_DATA = 24'h4740_01; // VS 1  
  
//      SET_OV5640 + 252: LUT_DATA = 24'h503d80; // color bar  
//      SET_OV5640 + 253: LUT_DATA = 24'h474100; //
```

店铺: <https://xiaomeige.taobao.com>

技术博客: <http://www.cnblogs.com/xiaomeige/>

官方网站: [www.corecourse.cn](http://www.corecourse.cn)

技术群组: 615381411