

OV5640 摄像头显示例程

黑金动力社区 2023-02-24

1 实验简介

本实验将采用 500 万像素的 OV5640 摄像头模组(模块型号: AN5642)为大家显示更高分辨率的视频画面。OV5640 摄像头模组最大支持 QSXGA (2592x1944)的拍照功能, 支持 1080P、720P、VGA、QVGA 视频图像输出。本实验将 OV5640 配置为 RGB565 输出, 先将视频数据写入外部存储器, 再从外部存储器读取送到 VGA、HDMI、LCD 等显示模块。

2 实验原理

2.1 OV5640 传感器简介

OV5640 摄像头模组采用美国 OmniVision(豪威)CMOS 芯片图像传感器 OV5640, 支持自动对焦的功能。OV5640 芯片支持 DVP 和 MIPI 接口,本实验所用 OV5640 摄像头模组通过 DVP 接口和 FPGA 连接实现图像的传输。

2.2 OV5640 的参数说明

像素: 硬件像素 500W;

感光芯片: OV5640;

感光尺寸: 1/4;

功能支持: 自动对焦, 自动曝光控制(AEC),自动白平衡(AWB);

图像格式: RAW RGB, RGB565/555/444, YUV422/420 和 JPEG 压缩;

捕获画面: QSXGA(2592x1944), 1080p, 1280x960, VGA(640x480), QVGA(320x240);

工作温度: -30~70°C, 稳定工作温度为 0~50°C

2.3 OV5640 的寄存器配置

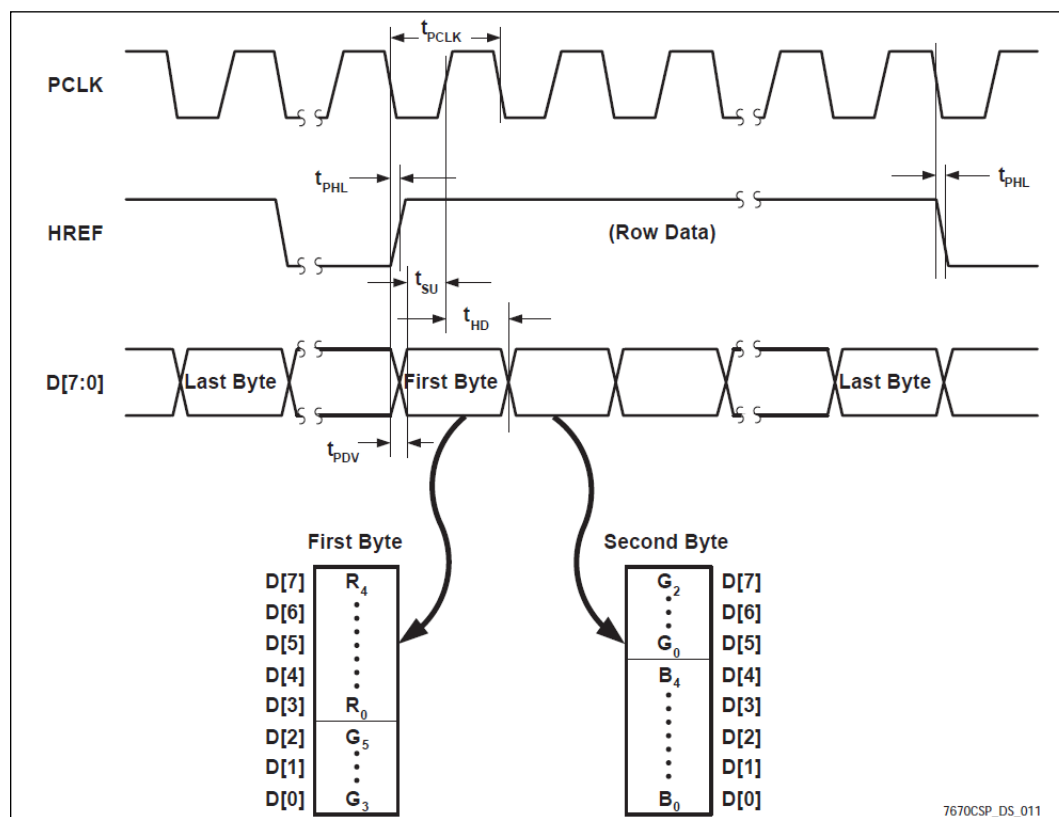
OV5640 的寄存器配置是通过 FPGA 的 I2C（也称为 SCCB 接口）接口来配置。用户需要配置正确的寄存器值让 OV5640 输出我们需要的图像格式，实验中我们把摄像头输出分辨率和显示设备分辨率配置成一样的，OV5640 的摄像头输出的数据格式在以下的 0x4300 的寄存器里配置，在我们的例程中 OV5640 配置成 RGB565 的输出格式。

Format Control 00			
Bit[7:4]: Output format of formatter module			
0x0: RAW			
Bit[3:0]: Output sequence			
0x0: BGBG... / GRGR...			
0x1: GBGB... / RGRG...			
0x2: GRGR... / BGBG...			
0x3: RGRG... / GBGB...			
0x4~0xF: Not allowed			
0x1: Y8			
Bit[3:0]: Does not matter			
0x2: YUV444/RGB888 (not available for full resolution)			
Bit[3:0]: Output sequence			
0x0: YUYVUY..., or BGRBGR...			
0x1: YVUYVU..., or GRBGRB...			
0x2: UYVUYV..., or BGRBGR...			
0x3: VYVUYU..., or RGBRGB...			
0x4: UYVUYV..., or BRGBRG...			
0x5: VUYVUY..., or RBGRBG...			
0x6~0xE: Not allowed			
0xF: UYVUYV..., or BGRBGR...			
0x3: YUV422			
Bit[3:0]: Output sequence			
0x0: YUYV...			
0x1: YVYU...			
0x2: UYVY...			
0x3: VYUY...			
0x4~0xE: Not allowed			
0xF: UYVY...			
0x4: YUV420			
Bit[3:0]: Output sequence			
0x0: YUYV...			
0x1: YVYU...			
0x2: UYVY...			
0x3: VYUY...			
0x4~0xE: Not allowed			
0xF: UYVY...			

关于 OV5640 的寄存器还有很多很多，但很多寄存器用户无需去了解，寄存器的配置用户可以按照 OV5640 的应用指南来配置就可以了。如果您想了解更多的寄存器的信息，可以参考 OV5640 的 datasheet 中的寄存器说明。

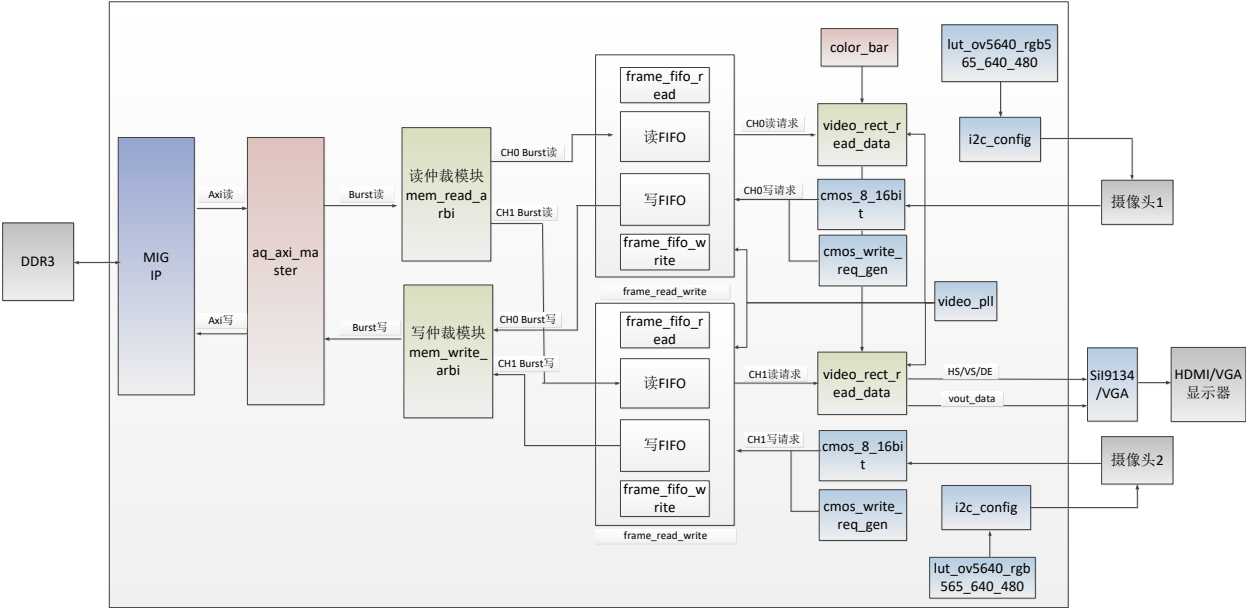
2.4 OV5640 的 RGB565 输出格式

OV5640 在 HREF 信号为高时输出一行的图像数据，输出数据在 PCLK 的上升沿的时候有效。因为 RGB565 显示每个像数为 16bit，但 OV5640 每个 PCLK 输出的是 8bit，所以每个图像的像数分两次输出，第一个 Byte 输出为 R4~R0 和 G5~G3，第二个 Byte 输出为 G2~G0 和 B4~B0，将前后 2 个字节拼接起来就是 16Bit RGB565 数据。

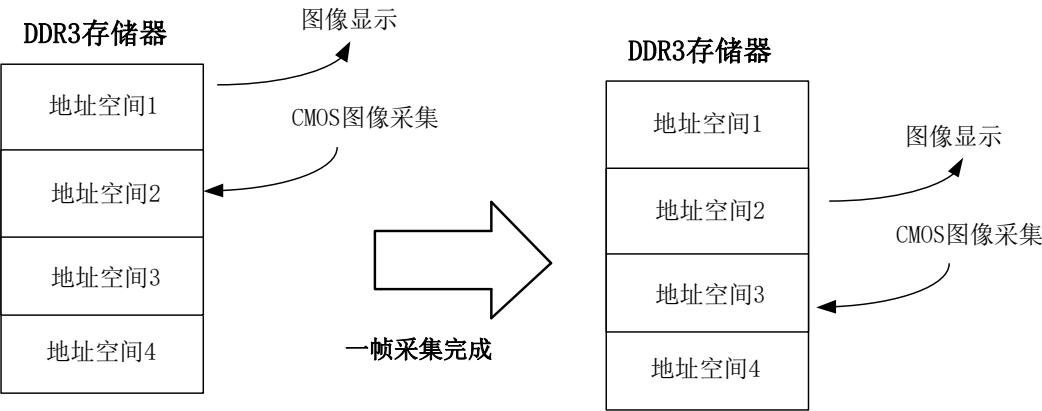


3 程序设计

程序中把两路摄像头的视频图像采集进来后转换成 16 位的数据，分别写入 FIFO 中，因为涉及到 2 路视频数据的存储，所以需要有一个写总裁模块把 2 路视频数据仲裁后写入到 DDR3 存储器里。视频显示也是同样需要读总裁模块从 DDR3 里读取视频数据，通过 video_rect_read_data 模块把彩条，第一路视频和第二路视频组合成一个画面，在 VGA\HDMI 显示器里同时显示出来。这里配置的摄像头的输出视频图像大小为 640*480，通过 I2C 总线配置 OV5640 的寄存器实现。



前面的实验已经为本实验做了大量的铺垫，包括 I2C 寄存器的配置、外部存储器的读写，本程序一个比较关键的地方在于视频同时读写，如果做到读写不冲突？在设计帧读写模块时就已经考虑到这点，所以有帧基地址选择，最大 4 帧选择，每次读视频帧地址和正在写的帧地址是不同的，而是上次写入的帧地址，这样就可以避免读写冲突，避免视频画面裂开错位。



cmos_8_16bit 模块完成输入 8bit 数据到 16bit 数据，数据位宽变成 2 倍，时钟频率不变，所以 16bit 数据是隔一个时钟周期有效，并不是一直有效。

信号名称	方向	说明
rst	in	异步复位输入，高复位
pclk	in	传感器像素时钟输入

pdata_i	in	传感器 8bit 数据输入
de_i	in	数据有效 (HREF)
pdata_o	out	16bit 数据输出
hblank	out	de_i 延时一个时钟周期
de_o	out	数据输出有效

cmos_8_16bit 模块端口

cmos_write_req_gen 模块完成 ov5640 数据写入请求生成，请生成 write_addr_index 写地址选择和 read_addr_index 读地址选择。

信号名称	方向	说明
rst	in	异步复位输入，高复位
pclk	in	传感器像素时钟输入
cmos_vsync	in	场同步输入，没一帧视频都会变化一次，可以用于一帧的开始或结束
write_req	out	写数据请求
write_addr_index	out	写帧地址选择
read_addr_index	out	读帧地址选择
write_req_ack	in	写请求应答

cmos_write_req_gen 模块端口

frame_read_write 模块我们已经在前面的例程中使用过，在这里就是把摄像头采集的数据存入写 FIFO 里，然后产生 DDR3 的 Burst 写请求，写入到 DDR3 中，另外读也是一样，当读 FIFO 里的数据小于一定值时，产生 DDR3 的 Burst 读请求。在这里我们实例化了 2 个 frame_read_write 模块，分别对应 2 路视频的数据存储和读取。这里每路视频的 DDR3 的存储地址是不一样的。

第一路视频的存储地址如下：

```
.read_finish      (
.read_addr_0      (25' d0                ), //The first frame address is 0
.read_addr_1      (25' d2073600          ), //The second frame address is 25' d207
.read_addr_2      (25' d4147200          ),
.read_addr_3      (25' d8220800          ),
.read_addr_index  (ch0_read_addr_index   ),
.read_len         (25' d196608           ), //Frame size 1024 * 768 * 16 / 64
```

第二路视频的存储地址如下：

```
.read_finish      (                                     ),
.read_addr_0      (25'd0294400                        ), //The first frame address is 0
.read_addr_1      (25'd10368000                       ), //The second frame address is 25'd20
.read_addr_2      (25'd12441600                       ),
.read_addr_3      (25'd14515200                       ),
.read_addr_index   (ch1_read_addr_index                 ),
.read_len         (25'd196608                         ), //Frame size 1024 * 768 * 16 / 64
```

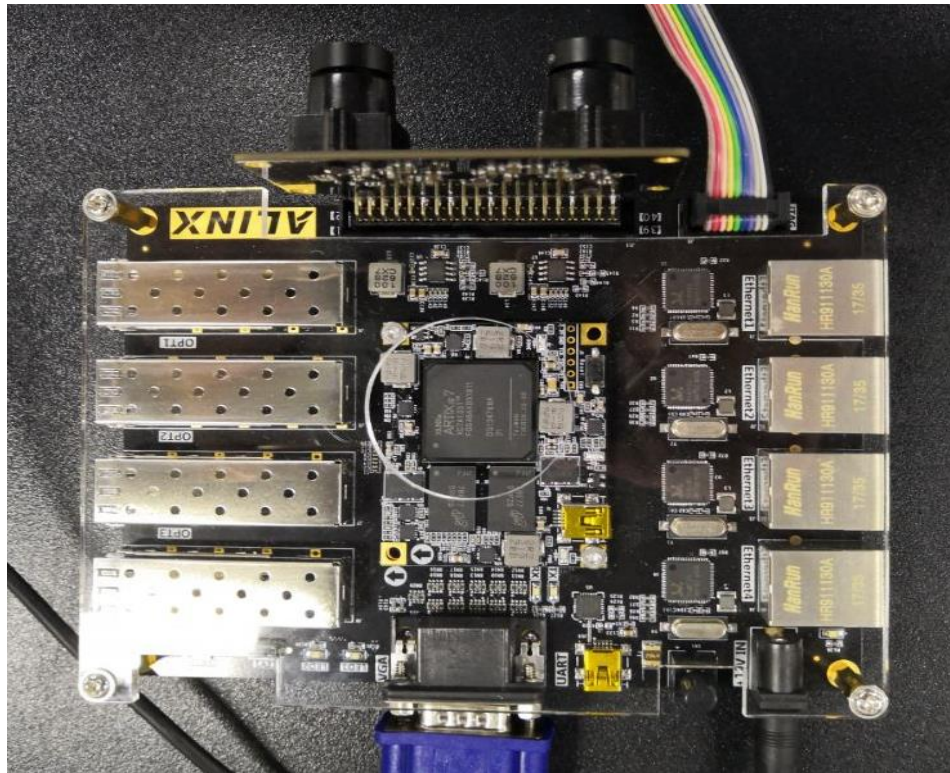
video_rect_read_data 模块功能跟 HDMI 字符显示 osd_display 模块的功能类似，本例程中是从 DDR3 里读取的视频图像跟彩条图像叠加，因为 ax7035 使用的是双目摄像头，所以需要调用 2 个 video_rect_read_data 模块。

信号名称	方向	说明
video_clk	in	视频的像素时钟
rst	in	复位信号
video_left_offset	in	视频显示的水平偏移地址
video_top_offset	in	视频显示的垂直偏移地址
video_width	in	视频的宽度
video_height	in	视频的高度
read_req	out	读一帧图像数据请求
read_req_ack	in	读请求应答
read_en	out	读数据使能
read_data	in	读到的数据
timing_hs	in	输入的行同步信号
timing_vs	in	输入的列同步信号
timing_de	in	输入的数据有效信号
timing_data	in	输入的数据信号
hs	out	输出的行同步信号
vs	out	输出的列同步信号
de	out	输出的数据有效信号
vout_data	out	输出的数据信号

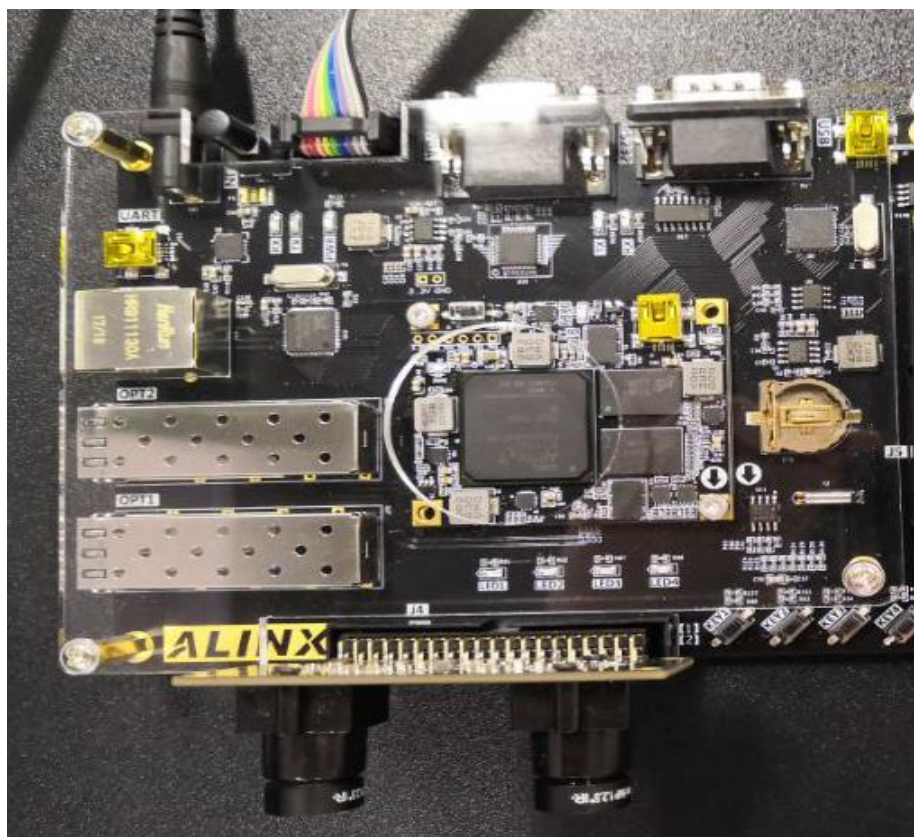
4 实验现象

(1) 将摄像头模块插入开发板，模块依次接入

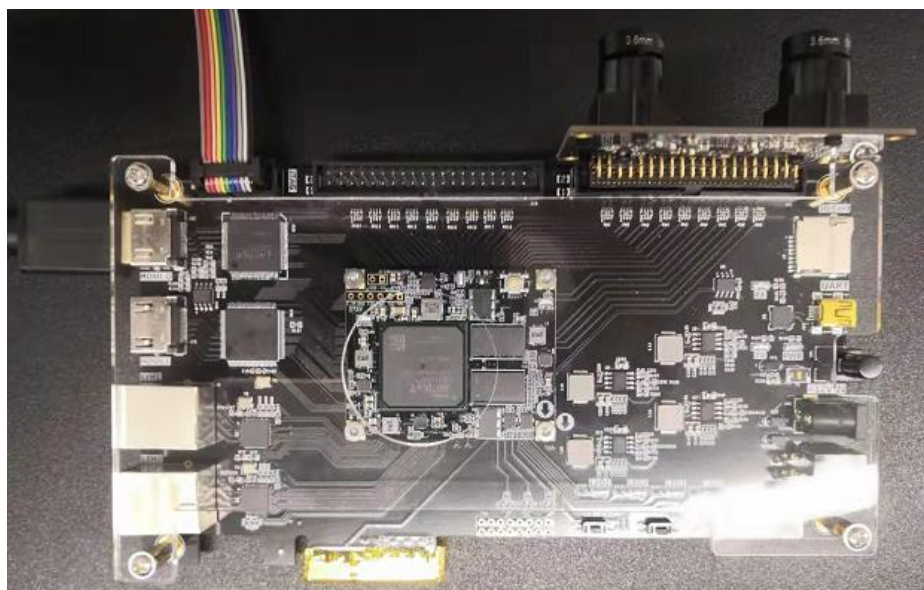
AX7101(AX7201)/AX7102(AX7202)/AX7103(AX7203)的扩展口 J11、J4、J11。保证 1 脚对齐，1 脚在焊盘形状和其他引脚是有明显区别的，是方形的。



AN5642 摄像头模块连接 AX7101(AX7201)连接图



AN5642 摄像头模块连接 AX7102(AX7202)连接图



AN5642 摄像头模块连接 AX7103(AX7203)连接图

- (2) 如果使用 HDMI/VGA 来显示，连接好 HDMI/VGA 显示器，如果使用液晶屏显示，液晶屏模块（AN430/AN070）依次接入 AX7102(AX7202)/AX7103(AX7203)的扩展口 J5、J13。

- (3) 下载实验程序，可以看到摄像头模块输出的视频。注意：*ov5640 模块焦距是可调的，如果焦距不合适，图像会模糊，旋转镜头，可以调节焦距。摄像头模块要轻拿轻放，不要用手触摸元器件。*