

# 基于 RISC-V 的多功能游戏机

何连杰；董铠阳

## 目录

<b>1</b>	<b>设计概述</b>	<b>2</b>
1.1	设计目的	2
1.2	应用领域	2
1.3	主要技术特点	2
1.4	关键性能指标	2
1.5	主要创新点	2
<b>2</b>	<b>系统组成及功能说明</b>	<b>3</b>
2.1	整体介绍	3
2.2	各模块介绍	4
2.2.1	蜂鸟 e203 RISC-V 内核	4
2.2.2	ps2 手柄和红外遥控解析模块	4
2.2.3	iCB 总线至 AXI 总线转换模块	4
2.2.4	DDR3 AXI0 接口读写控制模块	4
2.2.5	以太网模块	5
2.2.6	LCD 模块	6
2.2.7	HDMI 模块	6
2.2.8	SD 卡模块	7
<b>3</b>	<b>完成情况及性能参数</b>	<b>8</b>
3.1	基础要求	8
3.2	加分项 1：支持游戏切换	8
3.3	加分项 2：支持更高的图像刷新率	8
3.4	加分项 3：支持标准游戏手柄	8
3.5	加分项 4：驱动多种传感器	8
3.6	加分项 5：能通过以太网进行游戏转播	9
<b>4</b>	<b>总结</b>	<b>9</b>
4.1	可拓展之处	9
4.2	心得体会	9

## 5 附录

10

## 1 设计概述

### 1.1 设计目的

基于紫光同创 FPGA 平台，移植 RISC-V 并实现多款趣味游戏的多功能游戏机，能够运行 NES 模拟器，驱动多种外设输出图像，接收控制手柄信号。

### 1.2 应用领域

嵌入式微处理器；娱乐；FPGA

### 1.3 主要技术特点

基于紫光同创 FPGA 平台，移植 RISC-V 并实现多款趣味游戏的多功能游戏机。RISC-V 核系统总线上挂载 DDR3 和 FLASH，有充足的内存空间。多个外设均是由 FPGA 直接驱动，CPU 只负责调度。多款游戏分开设计存放在 FLASH 中，切换游戏时才将其载入读写速度最快的片上 RAM 中。音频存放在 SD 卡中，可以放下足够多的歌曲。RISC-V 可以运行 NES 模拟器，畅玩 NES 游戏。

### 1.4 关键性能指标

- (1) CPU 主频 50M，Dhrystone 跑分 1.28DMIPS/MHz
- (2) 支持 HDMI 输出音频和视频：视频参数为 640\*480@(60Hz)，音频采样率为 48000
- (3) 支持 RGB-LCD 输出：640\*480@(60Hz)
- (4) 支持千兆以太网传输视频
- (5) 支持 PS2 手柄
- (6) 支持红外遥控
- (7) 支持多游戏切换
- (8) 支持歌曲播放

### 1.5 主要创新点

- (1) 精简芯来科技的开源蜂鸟 e203 RISC-V 内核，使其 CPU 主频从原来的 16M 提升到 50M。
- (2) 图形帧存放在 DDR3 指定地址处，FPGA 的 hdmi 控制模块可以直接访问 DDR3，以 60Hz 的频率刷新屏幕。hdmi 控制模块读取的数据同时通过 fifo 缓冲送给 LCD 模块和以太网模块。CPU 在需要时可以精确到像素更新图形帧。
- (3) PS2 手柄和红外模块不采用传统的 CPU 软驱动，而是采用 FPGA 硬驱动。
- (4) hdmi 驱动不外接 PHY，而是从 PHY 层开始实现，且同时传输音频和视频，符合 hdmi 标准规范。

## 2 系统组成及功能说明

### 2.1 整体介绍

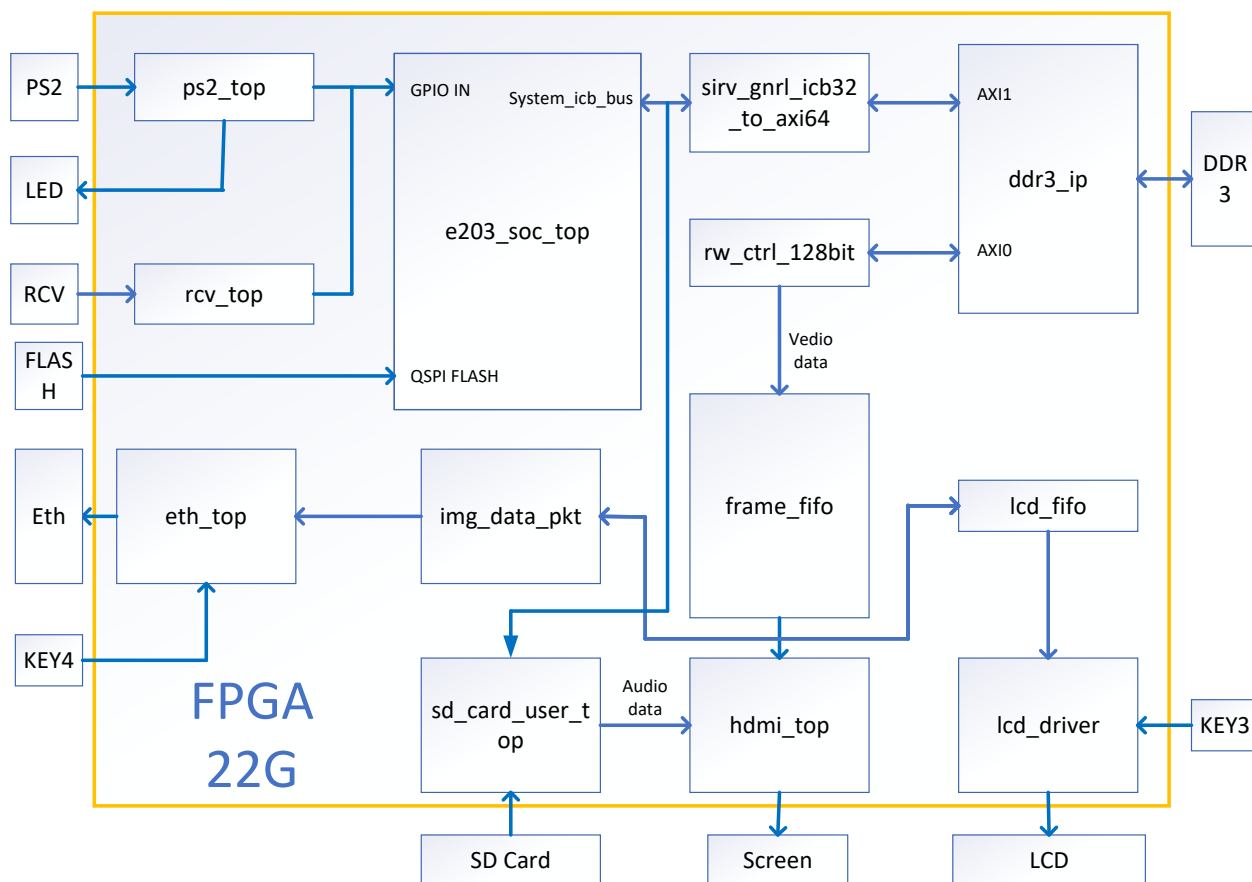


figure 1: 系统框架图

整体框架图见 figure1。其中'e203\_soc\_top'为蜂鸟 e203 RISC-V 内核，主频为 50M，负责全局的控制。其左边的'ps2\_top'和'rcv\_top'分别为 PS2 手柄和红外遥控解析模块，其解析出的按键值通过 GPIO 传给 e203 内核。PS2 解析模块解析出的按键值同时通过 LED 展示。'serv\_gnrl\_icb32\_to\_axi64'负责将 e203 的 32 位的 icb 系统总线转换位 64 位的 axi 总线，从而可以通过 'ddr3\_ip' 模块的 AXI1 接口访问 ddr3。e203 也有专用的接口用于访问 FLASH，还可以通过系统总线控制 SD 卡读写模块。

'rw\_ctrl\_128bit' 模块从 'ddr3\_ip' 模块的 AXI0 接口访问 ddr3，从中读取出视频数据送入 'frame\_fifo' 帧缓冲模块。该模块的输出又接给三个模块，分别是 'hdm\_top' 模块，'img\_data\_pkt' 模块和 'lcd\_fifo' 模块。其中 'lcd\_fifo' 模块用于再次缓冲视频数据，其输出接给 'lcd\_driver' 模块，该模块为用于驱动 LCD 屏幕。'img\_data\_pkt' 模块将读取的视频数据从 16 位拼接为 32 位，并在每帧的第一个像素数据前插入控制信息，之后存入内部的 fifo 中。'eth\_top' 模块从 'img\_data\_pkt' 模块内部的 fifo 中读取数据作为 UDP 协议中的数据段数据，最后驱动以太网传输数据。

'sd\_card\_user\_top' 模块从 SD 卡中读取音频数据传输给 'hdm\_top' 模块。'hdm\_top' 模块将获取的音频数据和视频数据通过 hdmi 接口输出。

## 2.2 各模块介绍

### 2.2.1 蜂鸟 e203 RISC-V 内核

该模块位芯来科技的开源 RISC-V 内核，但是对其进行了精简，去掉了 jtag 模块，去掉了大部分外设，去掉了内部的时钟生成模块，从而使其可以运行在 50M 的主频。由于没有了 jtag 模块，无法通过 riscv 调试器进行烧录程序，因此需要手动将程序烧录进 flash 中，烧录起始地址为 0x400000(内核的复位起始 PC 值设置为 0x20400000，从 0x20000000 至 0x3ffffff 属于外部 FLASH 的访问区间，故复位后会先访问 FLASH)。在 FLASH 力运行一段 boot 程序将主程序搬移如 Itcm(片上 RAM) 和 ddr3 中，优先搬入 itcm，如 figure2所示。由于开发板上只有一块 FLASH，故需要和 FPGA 比特流共享一块 FLASH。

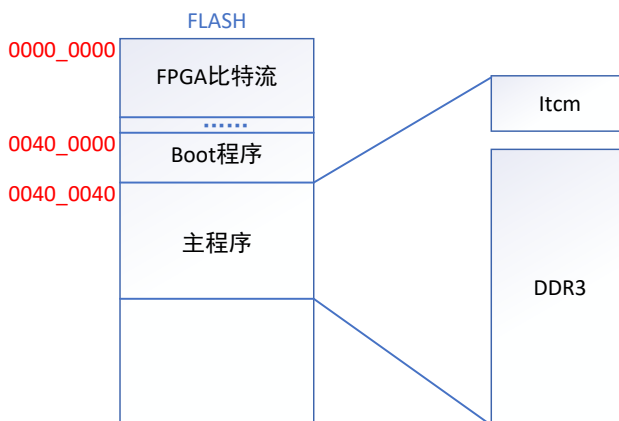


figure 2: 程序烧入 FLASH 后在 FLASH 内运行 boot 程序将主程序搬移入 itcm 和 ddr3

### 2.2.2 ps2 手柄和红外遥控解析模块

这两个模块分别负责解析 ps2 手柄和红外遥控器发送来的信号，从中获取出按键值。解析出的按键值直接连接至 e203 内核的 GPIO 输入，从而程序设计时只需要读取 GPIO 的输入值即可判断按键值，示意图见 figure3。

### 2.2.3 iCB 总线至 AXI 总线转换模块

由于蜂鸟 e203 内核使用的总线是 32 位的 ICB(Internal Chip Bus)，而 DDR3 IP 模块提供的用户接口是 AXI 总线，且最小宽度为 64 位。因此 e203 访问 DDR3 时需要先将 32 位 ICB 总线转换为 64 位 AXI 总线。蜂鸟 e203 源代码中提供了将 32 位的 ICB 总线转换为 32 位的 AXI 总线的模块，因此只需要稍加修改即可。

### 2.2.4 DDR3 AXI0 接口读写控制模块

该模块会不断的从 DDR3 的地址 0x00200000 处突发读出图片数据送入图形帧缓冲模块，其他模块从图形帧缓冲模块中读图形数据送往各自的驱动模块显示出图片。每秒读取 60 次，每次读取 640\*480

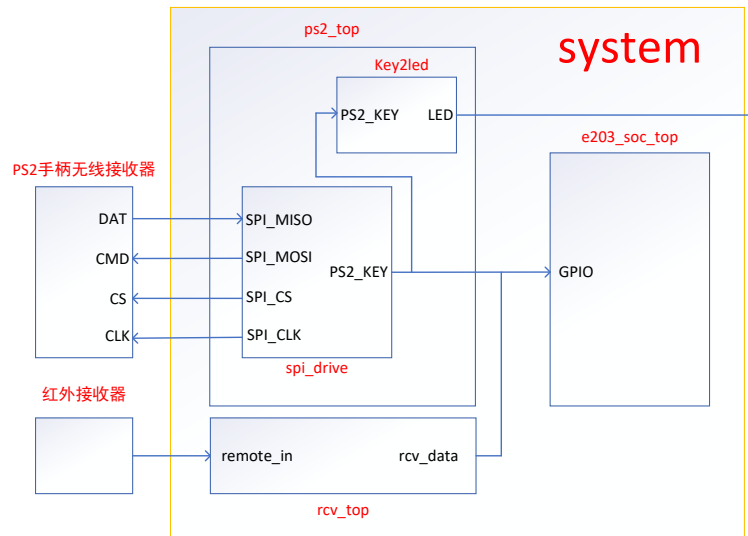


figure 3: PS2 手柄和红外接收模块

的 16bit 数据，吞吐量为  $60 \times 640 \times 480 \times 16 = 295\text{Mbps}$ ，而 ddr3 ip 模块提供的最大传输速率为 1000Mbps，因此能够满足需求。

CPU 总线的 0x80000000 至 0xffffffff 为系统内存访问区间，其中从 0x80000000 开始的 32KB 属于 itcm，从 0x90000000 开始的 32KB 属于 dtcm，其余属于外部 DDR3。因此往屏幕上的 x,y 坐标写入指定像素的程序如下：

```
1  uint16_t*fig_p=(uint16_t*)0x80200000;
2  uint16_t width=640;    //屏幕宽度
3  uint16_t height=480;   //屏幕高度
4  uint16_t x=10;
5  uint16_t y=10;
6  uint16_t RED=0xf800;    //红色RGB565格式
7  *(fig_p+width*y+x)=RED; //往屏幕x,y坐标处写入红色
```

## 2.2.5 以太网模块

该模块是基于正点原子的 PGL22G 开发板例程修改的，整体框图见 figure4。'img\_data\_pkt' 模块将读取的 16 位像素拼接成 32 位，并在每帧的开始插入额外的控制信息，然后存入内部的 fifo 中供 udp 模块读取。同时'img\_data\_pkt' 会给出每一包传输的数据长度以及开始传输信号。udp 模块按照 UDP 协议将数据封装成包，通过 GMII 接口 (Gigabit Medium Independent Interface) 传输给以太网物理层。由于外接的以太网 PHY 芯片使用的是 RGMII 接口 (Reduced GMII)，因此还需要由'gmii\_to\_rgmii' 模块进行接口转换。因为不需要通过以太网接收数据，因此 udp 模块只实现了发送功能，没有接收功能。接收方的 IP 地址固定且采用广播 MAC 地址，因此也不需要利用 ARP 协议 (Address Resolution Protocol) 进行地址解析。

上位机利用控制信息定位每帧的第一个数据包，然后接收 480 个数据包后开始解析数据，恢复出图片。上位机是我们团队自己编写的。

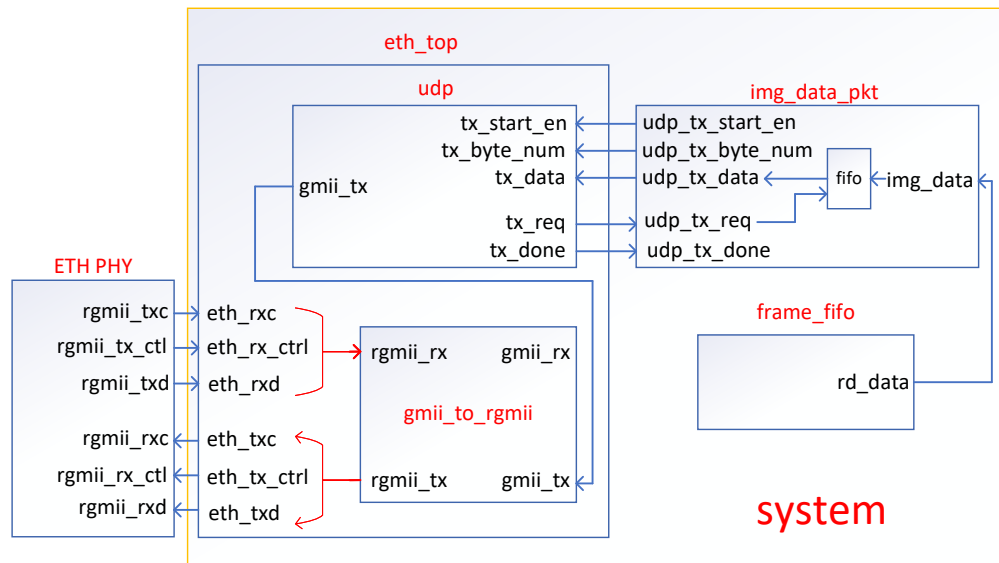


figure 4: 以太网模块整体框图

### 2.2.6 LCD 模块

帧缓冲模块的读信号是由 hdmi 模块给出的，而 hdmi 模块和 LCD 模块的时钟域并不一致，因此需要将像素数据再次通过 fifo 缓存。hdmi 输出的分辨率为 640\*480，而我们使用 LCD 屏幕分辨率为 1024\*600，因此驱动 LCD 屏幕时只驱动了左上角的 640\*480 区域，其余部分像素值给 0。但是若仍然按照 1024\*600@(60Hz) 的参考时序参数驱动 LCD 屏幕，则需要的缓冲 fifo 过大，因此我们魔改了 LCD 屏幕的时序参数。整体框图见5。

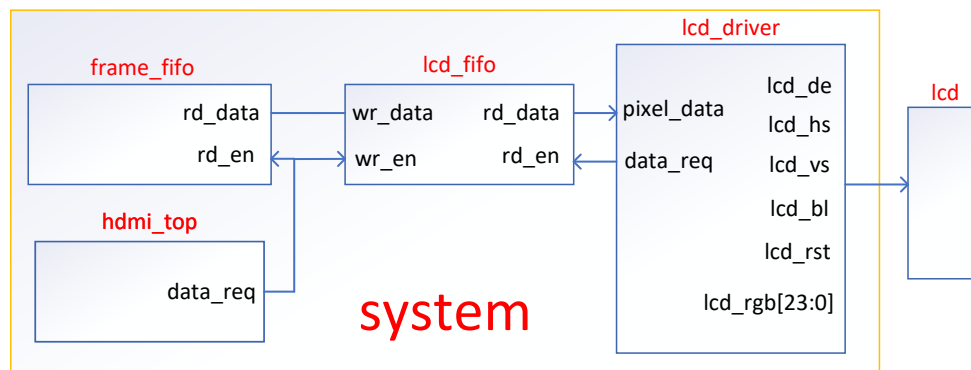


figure 5: LCD 模块整体框图

### 2.2.7 HDMI 模块

整体框图见 figure6。该模块是基于 github 上的开源模块修而来的，源代码位于：<https://github.com/hdl-util/hdmi>

首先将系统的 50M 时钟作为参考时钟，用锁相环生成三个时钟，频率分别为 25M, 125M, 24M。其

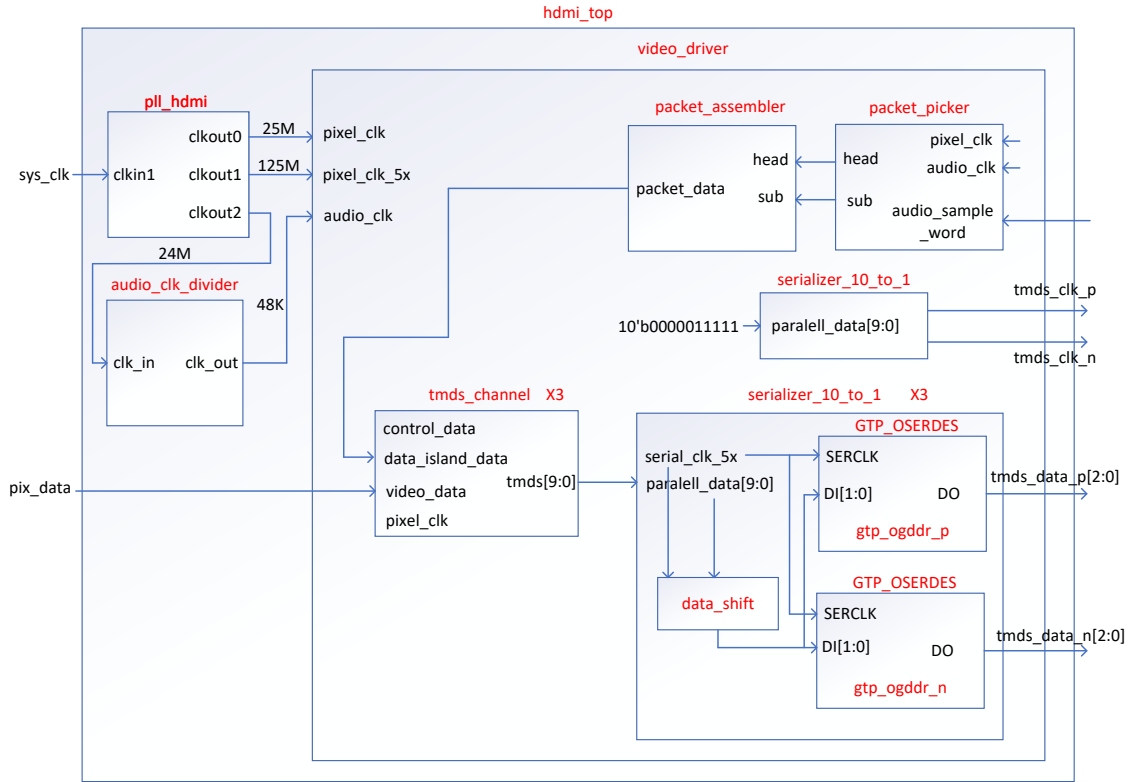


figure 6: hdmi 模块整体框图

中 25M 时钟作为像素时钟，125M 时钟作为并串转换的参考时钟。24M 时钟在进行 500 分频生成 48K 的时钟作为音频信号采样时钟。

'tmds\_channel' 模块完成视频数据，数据岛数据和控制数据的编码，生成长度为 10 比特的并行数据。该并行数据送入'serializer\_10\_to\_1' 模块进行并串转换，由于该模块的输入时钟为像素时钟的 5 倍，因此只需要进行 2:1 的并串转换，一共进行 5 次，从而将 10 比特的并行数据转换为串行数据，且输出的是串行信号是差分信号对。

'packet\_picker' 模块将输入的音频数据封装成包，包括 3 个字节的包头 head 和 4\*7 个字节的包体 sub。'packet\_assembler' 模块完成音频数据包的校验位计算，最终生成完整的音频数据包。具体结构见附录 figure12[1]。

### 2.2.8 SD 卡模块

该模块通过 SPI 协议不断读取 SD 里的音频数据，读取的起始地址和终止地址由 CPU 设置，同时 CPU 也可以控制其是否进行读取，具体程序如下：

```

1  uint32_t*Music_P=(uint32_t*)0x80400000; //SD卡模块控制寄存器基地址
2  *(Music_P+1)=21312; //设置读取SD卡的起始扇区地址
3  *(Music_P+2)=28714; //设置读取SD卡的终止扇区地址
4  *(Music_P)=1; //SD卡模块读使能
    
```



## 3 完成情况 & 性能参数

### 3.1 基础要求

在基于紫光同创 22G 的开发板上成功移植芯来科技蜂鸟 e203 RISC-V 内核，主频 50M，Dhrystone 跑分 1.28DMIPS/MHz，片上内存 64KB，片外拓展了 DDR3。实现了两款基础游戏和 NES 模拟器，可运行大部分 NES 游戏。基础游戏能保持在 60 帧，而 NES 游戏基本只有 1 帧，只能验证可以运行。

NES 游戏：超级玛丽，见 figure7；曼罗拉蛇，见 figure8

基础游戏：fly bird，见 figure9；2048，见 figure10



figure 7: 超级玛丽

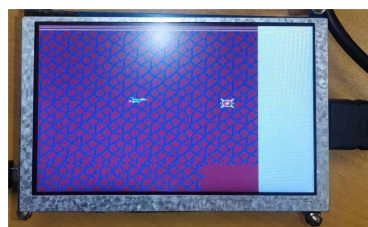


figure 8: 曼罗拉蛇

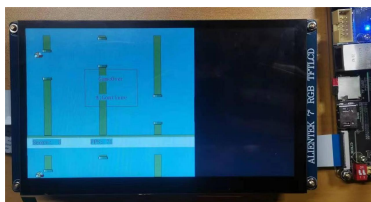


figure 9: fly bird

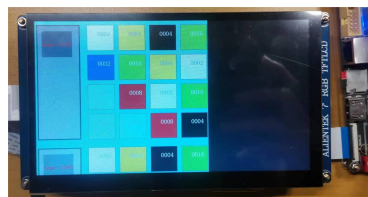


figure 10: 2048

### 3.2 加分项 1：支持游戏切换

这几款游戏是分开设计的，并存放在 FLASH 里的不同地址区间，boot 程序可以选择将哪款游戏 boot 进系统内存，及片上 itcm 和片外 ddr3，具体见 figure2。主菜单见 figure11

### 3.3 加分项 2：支持更高的图像刷新率

图像刷新率固定为 60Hz，图像分辨率为 640\*480。需要注意的是图像刷新率并不等于游戏帧率。可以同时支持 HDMI，LCD 屏输出图像。HDMI 输出见 figure8，LCD 屏输出见 figure9。

### 3.4 加分项 3：支持标准游戏手柄

支持 PS2 手柄控制，也支持红外遥控控制。

### 3.5 加分项 4：驱动多种传感器

支持氛围灯，即 LED 灯会实时显示出 PS2 的按键值 (姑且算是氛围灯吧)。





figure 11: 主菜单

能够进行游戏的背景音乐播放，音频使用 HDMI 输出，符合 HDMI 的标准协议。还可以进行音乐的切换，选择自己喜欢的 BGM。主菜单见 figure11。

### 3.6 加分项 5: 能通过以太网进行游戏转播

支持用 UDP 协议传输图像，并通过自制上位机接收图像。上位机接收效果见图7或图11。

## 4 总结

### 4.1 可拓展之处

- (1) 可以通过写汇编优化 NES 模拟器，是其游戏性能提升。
- (2) 只使用了 PGL22G 大概一半的逻辑资源，可以利用剩下的资源进行图像的插值处理，如最邻近插值等，使输出图像适配不同的分辨率。

### 4.2 心得体会

学会了充分利用 FPGA，进行软硬件协同设计，程序需要的外设尽可能通过 FPGA 去驱动，而不是使用 CPU 的 GPIO 或者外接芯片解决。

## 参考文献

- [1] Philips Consumer Electronics International B.V. Silicon Image Inc. Sony Corporation Thomson Inc. Toshiba Corporation Hitachi Ltd., Matsushita Electric Industrial Co Ltd. High-definition multimedia interface, 2006.

## 5 附录

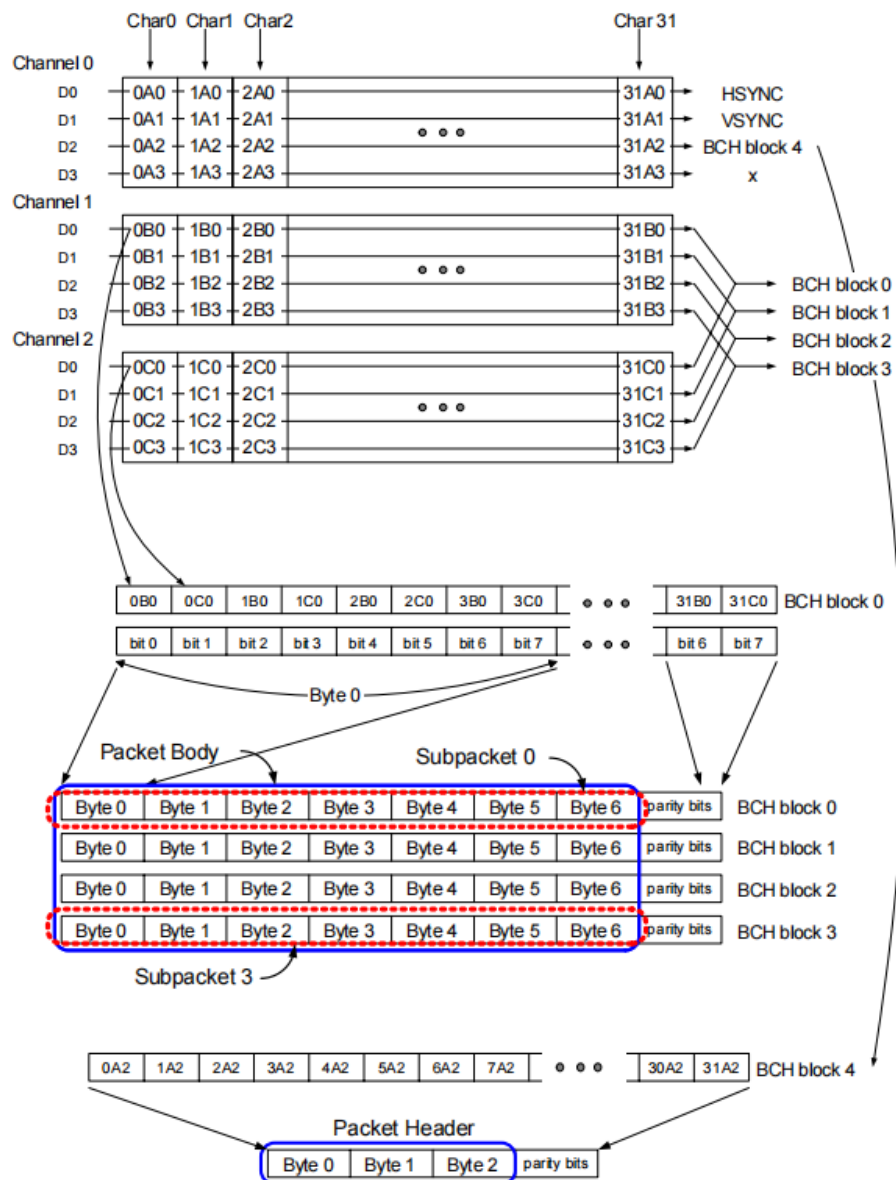


figure 12: hdmi 协议数据岛包和 ECC 结构