

2020 年新工科联盟-Xilinx 暑期学校团队项目设计文档

作品名称	Catmario
板卡型号	xc7s15ftgb196-1
所在班级	东南大学电子学院 A 班
成员姓名、学号、学校	雷弈（06017334） 东南大学电子科学与工程学院 倪雨晴（06017211） 东南大学电子科学与工程学院
Github 链接	https://github.com/06a17301/sea_catmario

第一部分

设计概述 /Design Introduction

1、我们借鉴了一款名叫猫里奥的游戏，想通过这次项目回顾经典游戏，希望能探索其中的代码设计、算法原理，提升自我水平。

考虑到游戏的实时性，以及游戏中对状态机的大量应用，利用 FPGA 并行模式的优势实现游戏 Catmario 在 SEA 上的复现。Catmario 游戏中的主角是猫里奥，可以左右移动以及跳跃，在路途中越过障碍，摔入空隙就会死亡，到达目的地即成功。我们通过 Verilog 语言编写 Catmario 的代码，并下载到 SEA 上，结合适当的外部连接设备实现游戏的正常显示和运行。

2、分工

分工表	
雷弈 06017334	与视频显示有关的部分、键盘检测及控制部分的部分代码、小怪兽的代码以及全部代码的整合与调试
倪雨晴 06017211	键盘检测及控制部分的部分代码，顶层状态机的设计及代码，资料搜索与整理，报告书的编写，视频的剪辑制作

3、作品展示



注：详细内容可以看 catmario_video.mp4

第二部分

系统组成及功能说明 /System Construction & Function Description

2.1 系统功能说明

计划实现——1、跳跃：能够实现向上跳高、向上跳的同时也能向左向右前进；

2、隐藏方块：在跳跃时，会由于跳跃触发，带有问号的方块会变成新方块，特定位置的空中会出现实物格子挡住前进的路线；

3、隐藏攻击（小鱼或者其他事物）：猫里奥前进时，会突然触发攻击，即会有小鱼从下方/空中/前方出现攻击猫里奥。

已实现——游戏画面在 HDMI 屏上显示，游戏主角猫里奥可以左右移动以及跳跃，可在跳跃的触发下将问号方块变成新方块，在路途中越过障碍（包括隐藏怪兽）到达终点即获得胜利，掉入空隙就会死亡。可通过手柄按键进行控制。

2.2 各模块介绍

2.2.1 主状态机模块

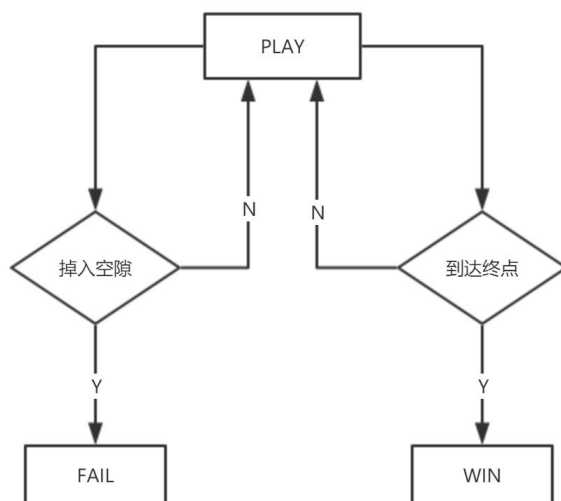


图 1 主状态机流程图

游戏共分为三个状态：游戏进行状态，胜利状态，失败状态。游戏一启动就能通过按键操作进行游戏；在游戏进行状态中，若猫里奥到达终点，则进入胜利状态；若猫里奥掉入空隙，则进入失败状态。之后可重新开始游戏，重复上述过程。

2.2.2 键控猫模块

猫里奥的控制是由外接按键控制，按键分别给上、左、右三个方向赋值，来控制猫里奥的移动。我们通过 `cat_x`、`cat_y` 数组储存猫的身体位置信息，用 `key_right`、`key_left` 以及 `key_jump` 来判断猫里奥的运动状态，最终实现猫里奥的运动；

其中跳跃 `key_jump` 代码的编写较为关键——我们分为了左跳、上跳和右跳以便控制。

001 表示跳跃，代码如下

```

3'b001:begin
    if (down==0)
        begin
            if(((cat_y0-cat_y)>=32*5)|(hit_wall_up==1)) begin
                cat_y<=cat_y+16;
            end
        end
    end
  
```

```
        down<=1;
        end
    else
        begin
            cat_y<=cat_y-16;
        end
    end
else if((down==1)&&(hit_wall_down==0))
    cat_y<=cat_y+16;
else if(hit_wall_down==1) begin
    cat_y<=cat_y;
    down<=0;
end
else begin
    cat_y<=cat_y;
    down<=down;
end
end
```

011 表示向右跳，代码如下

```
3'b011:begin
    if (down==0)
        begin
            if(((cat_y0-cat_y)>=32*5)|(hit_wall_up==1)|(hit_wall_right==1))begin
                down<=1;
                cat_y<=cat_y+16;
            end
        else
            begin
                cat_x<=cat_x+8;
                cat_y<=cat_y-16;
            end
        end
    else if((down==1)&&(hit_wall_down==0)&&(hit_wall_right==0))begin
        cat_x<=cat_x+8;
        cat_y<=cat_y+16;
    end
    else if((hit_wall_down==1)|| (hit_wall_right==0)) begin
        cat_y<=cat_y;
        cat_x<=cat_x+8;
        down<=0;
    end
    else if((hit_wall_down==0)|| (hit_wall_right==1)) begin
        cat_y<=cat_y-16;
        cat_x<=cat_x;
    end
    else begin
        cat_x<=cat_x;
        cat_y<=cat_y;
    end
end
```

由于游戏画面中砖块分布复杂，我们对地图进行分块标记来分别进行行为判断。以猫里奥头部左上坐标为基础，设置了6个状态：

000 表示空白处，猫里奥可随意行动；

001 表示砖块处和下方边界，猫里奥不可向下行动；

010 表示左方边界，猫里奥不可向左行动；

011 表示右方边界，猫里奥不可向右行动；
100 表示上方边界，猫里奥不可向上行动；
101 表示问号下方坐标，当猫里奥的头部坐标与其重叠时，问号消失。
边界分块如图所示：

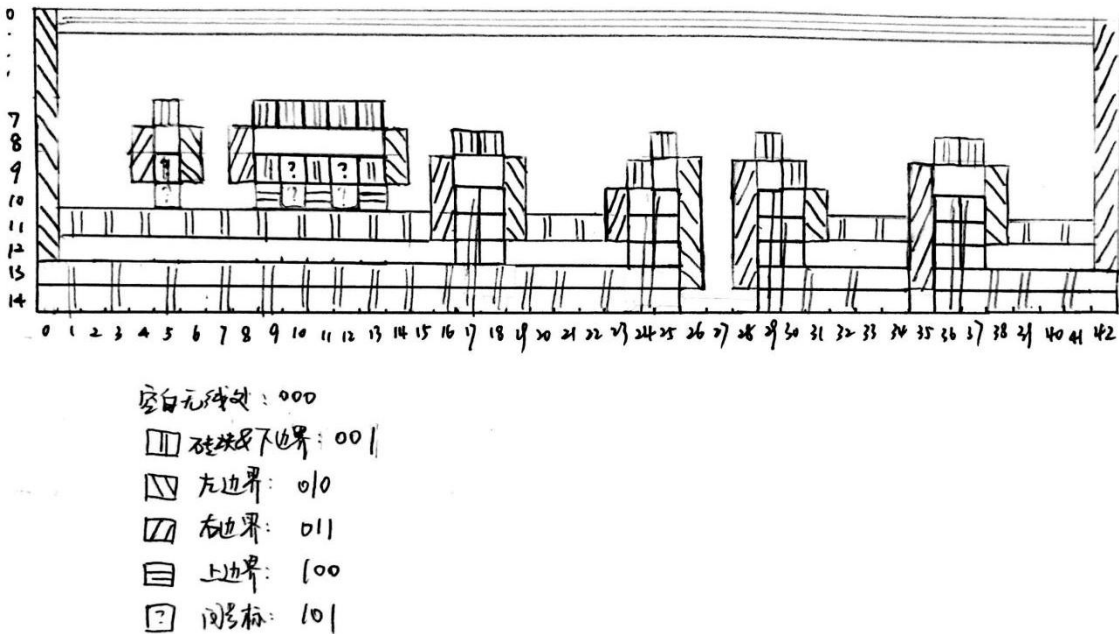


图 2 地图边界分块图

后来在调试代码的过程中发现了一些位置需要特殊考虑，例如拐角处、边缘部分等，因此在对猫里奥是否撞击到墙的判断里需要加上特例的情况。

2.2.3 小怪兽控制模块

为了增加游戏的趣味性，我们添加了小怪兽这个隐藏攻击，即当猫里奥碰撞第三个问号砖块时小怪兽会突然跳出，猫里奥需要进行躲避，若碰到小怪兽则游戏失败；

我们通过 `monster_x`、`monster_y` 数组储存小怪兽的身体位置信息，是的小怪兽从问号砖块上面跳出后，向画面左边移动直到消失。

2.2.3 HDMI 显示模块

这部分包含了已有的 HDMI 驱动和视频显示模块，利用 SEA 的 miniHDMI 接口，将图像信号传至显示器，采用 1920*1080 像素的分辨率。HDMI 时序如下：

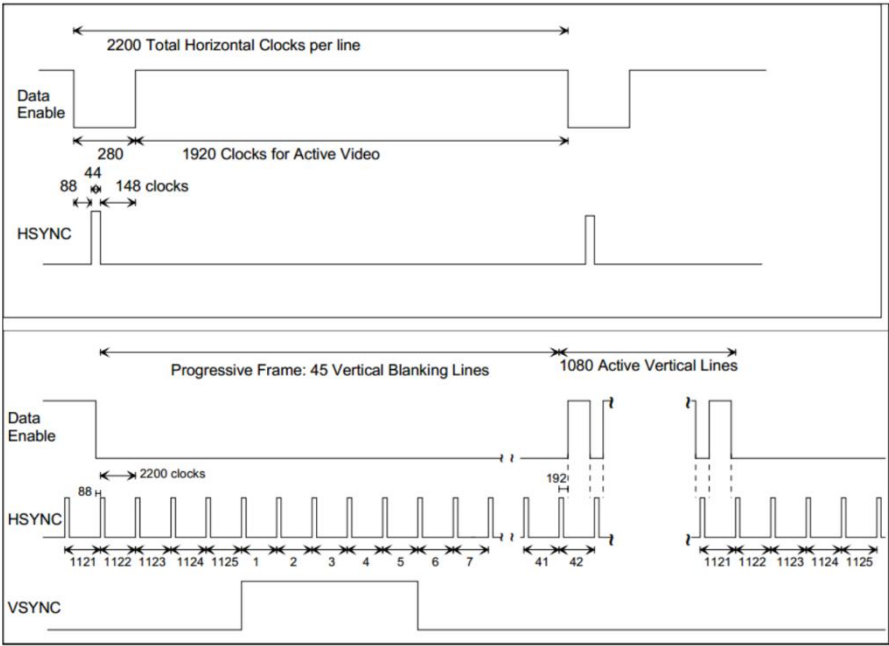


图 3 HDMI 时序图

1920*1080 分辨率的时序参数如下表：

显示模式	时钟 (MHz)	行时序（像素数）					帧时序（行数）				
		a	b	c	d	e	o	p	q	f	s
1920*1080@60	148.5	12	40	1920	28	2000	4	18	1080	3	1105

由于猫里奥的位置一直在改变，因此将猫里奥图像左上角的坐标作为变量及参考坐标，根据当前的猫里奥和砖块信息，通过对扫描坐标的判断，输出相应的显示使能信号，在相应的位置上显示相应的元素。我们设定了数组[11:0]RGB_Data，其中4位二进制为一组，共三组，来显示RGB三色的占比，进而达到可以显示不同颜色的效果。

通过控制part变量来划分区域，将地图分为砖块、问号砖块、猫里奥、小怪兽和其他区域，再分别对其进行RGB赋值显色。当WIN或者FAIL时，令RGB_Data=24'h000000全屏显示黑色，中间显示单词。

由于现有的RGB信号为并行的，而需要传送给miniHDMI接口的信号是一对串行的差分信号，因此利用rgb2dvi ip核进行信号的转化：

（其中vid_pVDE为数据有效信号，vid_pHSync和vid_pVSync为行、场同步信号。

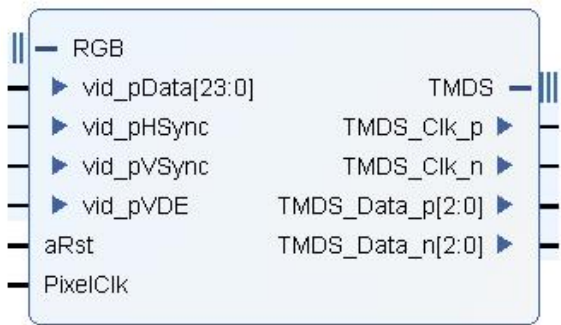


图 4 rgb2dvi ip 核 引脚图

第三部分

完成情况及性能参数 /Final Design & Performance Parameters

（作品已实现的功能及性能指标）

总体来说，我们已经实现了大部分的预期功能。

- 1) 游戏的整个流程以及各个状态的转换已经完全实现，基本还原了猫里奥游戏；
- 2) 基本实现了在复杂的地图上进行跳跃的预期功能；
- 3) 实现了未知方块被向上顶到后变成隐藏方块的预期功能；
- 4) 实现了隐藏怪物的攻击的预期功能；
- 5) 由于时间、资源等问题，我们未能实现之前设想的陷阱功能。

第四部分

总结 /Conclusions

（谈一谈完成暑期学校课程后的收获与感想。请每位组员分开写。）

06017334 雷弈

通过这次的暑期学校课程，我对 FPGA 有了更深刻的理解，对模块化设计有了更直观的体会，对 Verilog 硬件描述语言的运用也更加灵活。在实现我们 catmario 项目的过程中，考虑到板卡的资源限制，需要让同一图像通过 HDMI 在屏幕上分块显示出来，这也是我遇到的第一个难题，在查阅了相关资料并在调试代码的过程中不断思考，我最终成功实现了这一设想。然而，这也只是第一步，往后的人物动态显示、跳跃部分算法如何处理、模块之间相互的关系如何、如何优化代码使得时序得以收敛等等，这些问题随着模块的增多、代码的增长陆续出现，因此控制变量调试代码、分模块排除问题占据了我大部分的时间。

这其中我觉得收获最大的地方就是经历了从一个模块的功能实现到几个模块分别实现功能，再到将模块搭建在一起，让他们实现全部正确功能的过程，这是一个很奇妙的过程，因为以往我在竞赛或者课程中都是负责部分模块，并没有实现过最终的模块整合。我相信这次的暑期学校课程对我以后的硬件设计学习将产生很大的影响。

当然，我们的项目还有很多不足之处，比如一些边界判断不够清楚，有时会造成猫里奥掉进本应属于砖块的区域之中，通过部分实践和思考，我认为这是因为我们在最初用猫里奥图像的左上角作为它的参考坐标并将其一直保留到键控部分的原因，如果在对猫里奥行为控制部分用猫里奥中心处的坐标作为参考坐标，一方面边界判断会变简单，另一方面也更便于理解。我最后添加了 monster 模块之后，在 key2move 模块里有关猫里奥遇到怪物就死亡的部分就采用了这种方式，使得整个代码简化了不少。但是由于时间原因，不能再将整个代码都进行优化了。

06017211 倪雨晴

这次暑期学校从 7 月 20 日开始课程学习，到 8 月 1 日完成项目，期间我们逐步收获了许多关于 FPGA 的知识。通过综合设计课程，我们对 FPGA 的开发以及 Verilog 代码的编写有了深刻的实践以及体会。

在实现 Catmario 的过程中,之前计划的是可以通过手机 APP 蓝牙进行按键控制人物行动,我也在前期完成了手机 APP 的设计制作,但是在进行实际操作时,发现手机 APP 蓝牙控制难以实现长按来控制行动,游戏体验感不高,所以我们只能改变方向,放弃使用蓝牙键控,也幸好学校为我们提供了游戏手柄,让我们能够便捷的实现按键控制。此外,最令我印象深刻的还是地图边界分块代码的编写,首先要给画面进行分区并分配状态,为了能够准确的限制人物行为,应当充分考虑其头部坐标的行为路径,然后需要编写一共 645 个位置的状态。而且一开始的设想在实践之后发现了问题,就要全部重新编写,非常考验我们的细致严谨耐心的工作能力。

由于这次项目的完成时间有限,我们仍有一些想法未能实现,希望后续有机会我们能对这个游戏进行完善,比如:

- 1) 游戏界面以及关卡更加丰富多样;
- 2) 方向控制更加舒适顺畅;
- 3) 能够增加特殊的隐藏陷阱、猫里奥能够进行攻击;
- 4) 可以增加背景音乐和特别的音效;等等

虽然一开始的学习经历令我忙碌手足无措,但是后面的学习实践让我收获了很多。相信通过这次暑期学校的学习与实践,我之后的硬件学习会更上一层楼,也希望学校能多举办一些类似的学习课程。