硬件课程设计总结报告

--基于Xilinx的FPGA的打地鼠游戏机设计

院　　系：电子与信息工程系

专　　业：通信工程

班　　级：通信0901

小组成员：蔡光明 U200913804

小组成员：陈仁亮 U200913790

小组成员：汪晓轩 U200913806

指导教师：杨小献

目　录

一、摘要.....3

二、项目概述.....3

三、项目设计目标（功能性能指标）描述.....3

四、团队组成与任务分工.....4

五、系统总体设计.....4

六、系统软件设计与实现.....5

1.各模块输入输出端口说明.....5

2.各模块框图.....7

3.测试平台设计.....11

4.引脚分配及下载过程.....12

七、系统硬件设计与实现.....14

八、系统的测试与结果分析.....17

九、心得体会与项目总结.....17

十、致谢.....18

十一、参考文献.....18

附录：1.实验平台 Xilinx 公司的 SPARTAN3E 实验板介绍.....19

2.源程序.....19

一．摘要

本次硬件课程设计我们主要是应用 Xilinx 大学计划 FPGA 实验板 Xilinx Spartan™-3E XC3S500E FPGA 实验开发板,并结合自己焊接的 PCB 板外围电路来共同实现一个打地鼠游戏机，游戏参与者根据系统中的亮灯提示控制相应按键，模拟打地鼠的实际情况。对按键控制越准确，说明参与者的身体越灵活，反应越快。我们使用 FPGA 芯片作为中央控制器控制整个系统，将整个系统划分为：编码模块，分频模块，时钟选择模块（即速度控制模块），随机数产生模块，比较匹配模块（即得分模块），液晶显示模块，背景音乐模块等模块，所有算法由软件结合硬件电路来实现，利用 ISE 工作平台和硬件描述语言 VHDL，并通过 Xilinx FPGA 芯片实现。

The hardware design course we are mainly used Xilinx FPGA experiment board university program Xilinx Spartan ™-3 E XC3S500E FPGA experiment development board, and with his welding PCB peripheral circuit to achieve a common ground mouse playing video games, games participants in the system according to the light hint control corresponding key, a simulation to the actual situation of the rat. The more accurate control of buttons, the more flexible body that the participants, the faster response. We use the FPGA chip as the central controller control the entire system, the whole system is divided into: code modules, points frequency modules, the clock choose module (namely speed control module), random number produce module, more matching module (namely scored modules), liquid crystal display module, background music module module, all by combining software algorithm hardware circuit to realize, using the ISE work platform and hardware description language VHDL, and Xilinx FPGA chip realize through.

关键词： 打地鼠　FPGA　VHDL　ISE

二、项目概述

应用 Xilinx 大学计划 FPGA 实验板 Xilinx Spartan™-3E XC3S500E FPGA 实验开发板,并结合自己焊接的 PCB 板外围电路来共同实现一个打地鼠游戏机，游戏参与者根据系统中的亮灯提示控制相应按键，模拟打地鼠的实际情况。对按键控制越准确，说明参与者的身体越灵活，反应越快。

三、项目设计目标（功能性能指标）描述

基本功能：用 8 个灯作目标，与之对应有 8 个按键进行控制。每一次 8 个灯中随机出现一个灯处于“亮”的状态，在灯亮的时间内要求按到对应的按键，若按到则加 1 分，且灯熄灭；否则命数扣一条。

设定初始命数为 9，得分每达到 10 分便加 1 条命，失误一次扣一条命，命数为 0 时游戏结束。

游戏分四个难度级别，每个级别灯闪亮的速度不同，级别越高，速度越快，灯亮的时间越短。得分每达 10 分，难度自动升一级。

设有暂停/继续和开始/停止功能，能记录和更新历史最高分数。控制 LCD 进行相应的显示。

每个难度级别对应产生不同的音乐，与灯闪的频率节奏相当。

扩展功能：用 VGA 显示图案的随机闪动

四、团队组成与任务分工

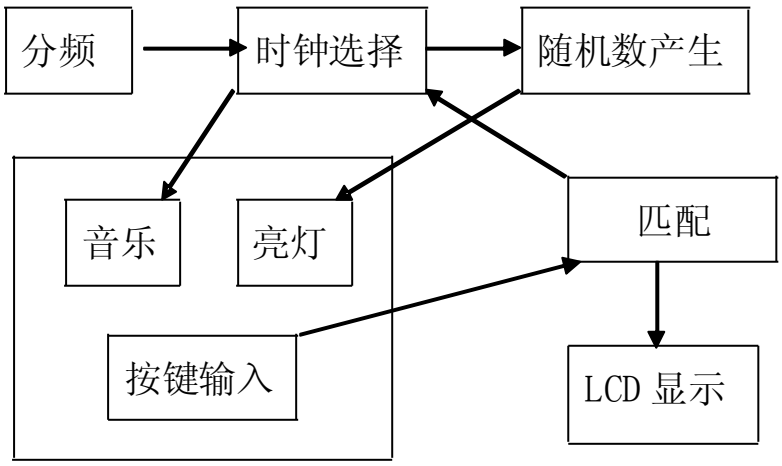
蔡光明（U200913804 通信 0901）：负责软件部分 music 音乐模块，lcd 液晶显示模块，以及硬件部分 PCB 板的焊接，调试及封装。

陈仁亮（U200913790 通信 0901）：负责软件部分 code 编码模块，random 随机数产生模块，compare 匹配模块及软件调试

汪晓轩（U200913806 通信 0901）：负责 divider 分频模块，clk-sel 时钟选择模块，vga 视频显示模块及软件调试

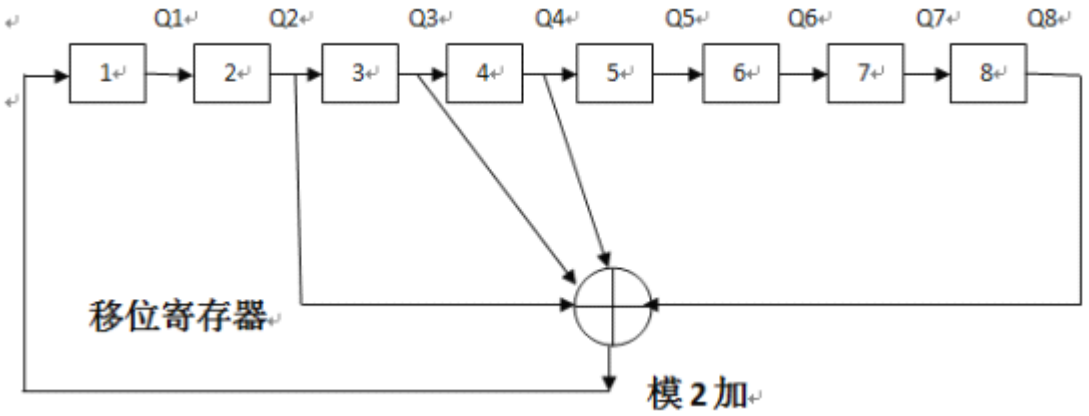
五、系统总体设计

系统框图与说明



外围扩展电路

各框图说明：
①分频模块是将输入的 FPGA 的 50MHz 的时钟分频为各种需要的时钟。
②时钟选择模块根据按键匹配的结果，选择对应难度的时钟，即选择 A,B,C,D 其中的一个，然后控制其他的模块。
③随机数产生模块使用的是 m 序列，是基于线性反馈移位寄存器的原理实现的。本实验中使用的是 8 级 m 序列发生器，产生一个长度为 255 个时钟脉冲周期的二进制伪随机序列。其逻辑框图如下：



取其中的三位(如后三位)通过 3-8 译码器来点亮 LED 灯。
④采样编码模块，将 8 个输入按键进行采样，编码成 3 位数。
⑤匹配模块，是把用户的按键输入值与对应于亮灯的随机数进行比较，如果相同则增加分数值和生命值，不同则减少分数值和生命值。
⑥LCD 显示模块，是把变化的分数值、生命值和历史最高分数在 LCD 屏幕上显示出来，让用户知道自己最新的分数、生命值及历史最高分数状况。
⑦音乐模块是将不同的难度用不同的音乐驱动蜂鸣器来加以区别。

六、系统软件设计与实现

1. 各模块输入输出端口说明：

1)顶层模块 game
输入信号 IN：

```
clk50m:in std_logic;              --50MHz 的输入时钟
k1,k2,k3,k4,k5,k6,k7,k8:in std_logic;    --8 个输入按键
start:in std_logic;           --开始信号，高电平有效
clear:in std_logic;           --最高分清零信号，clear=1 时清除最高分纪录
en:in STD_LOGIC;                  --暂停信号，en=0 时暂停
```

输出信号 OUT：

```
win,lose:out std_logic;       --win=1 表明游戏胜利，lose=1 表明游戏失败
led:out std_logic_vector(2 downto 0);    --产生的三位随机数，送到 74LS138
                                          进行译码，从而点亮一个 LED
```

```
leda,ledb,ledc,ledd:out std_logic;       --不同的难度分别用不同的LED灯
                                          来指示
sound:out std_logic;                  --为输出的喇叭信号
LCD_RS : out  STD_LOGIC;
LCD_RW : out  STD_LOGIC;
LCD_EN : out  STD_LOGIC;
hs : out std_logic;                              --屏幕行扫描
vs : out std_logic;                              --屏幕场扫描
red : out std_logic;                             --显示红色输出
green : out std_logic;                           --显示绿色输出
blue : out std_logic;                             --显示蓝色输出
data : out  STD_LOGIC_VECTOR (3 downto 0)); --数据信号输出
```

2)采样编码模块code

输入信号IN：

```
clk50m:in std_logic;
clk:in std_logic;
start:in std_logic;
k1,k2,k3,k4,k5,k6,k7,k8:in std_logic;
```

输出信号OUT：

```
key:out std_logic_vector(2 downto 0)
```

3)分频模块divider

输入信号IN：

```
clk50m : in  STD_LOGIC;
start:in STD_LOGIC;
```

输出信号OUT：

```
A : out  STD_LOGIC;
B : out  STD_LOGIC;
C: out  STD_LOGIC;
D : out  STD_LOGIC;
clk500:out STD_LOGIC;
clk4:out std_logic;
clk6m:out std_logic
```

4)时钟选择模块clk_sel

输入信号IN：

```
clk50m:in std_logic;
start:in std_logic;
A,B,C,D:in std_logic;
en_A,en_B,en_C,en_D:in std_logic;
```

输出信号OUT：

```
leda,ledb,ledc,ledd:out std_logic;
clkout:out std_logic
```

5)随机数产生模块 random
输入信号 IN：
```
clk : in  STD_LOGIC;
start:in STD_LOGIC;
en:in STD_LOGIC;
```
输出信号 OUT：
```
led : out  STD_LOGIC_VECTOR (2 downto 0)
```

6)匹配模块 compare
输入信号 IN：
```
start:in std_logic;
clear:in std_logic;
en:in std_logic;
clk : in  STD_LOGIC;
key : in  STD_LOGIC_VECTOR (2 downto 0);
led : in  STD_LOGIC_VECTOR (2 downto 0);
```
输出信号 OUT：
```
win,lose:out STD_LOGIC;
en_A : out  STD_LOGIC;
en_B : out  STD_LOGIC;
en_C : out  STD_LOGIC;
en_D : out  STD_LOGIC;
highscore_out:out STD_LOGIC_VECTOR(11 downto 0);
score_out : out  STD_LOGIC_VECTOR (11 downto 0);
life_out : out  STD_LOGIC_VECTOR (7 downto 0));
```

7)音乐产生模块 music
输入信号 IN：
```
clk6m,clk:in std_logic;
en_A,en_B,en_C,en_D:in std_logic;
```
输出信号 OUT：
```
sound:out std_logic
```

8)LCD 显示模块 lcd
输入信号 IN：
```
clk500:in std_logic;
start : in  STD_LOGIC;
win,lose:in std_logic;
highscore:in STD_LOGIC_VECTOR(11 downto 0);
score: in  STD_LOGIC_VECTOR (11 downto 0);
life: in  STD_LOGIC_VECTOR (7 downto 0);
```
输出信号 OUT：
```
LCD_RS : out  STD_LOGIC;
```

```
LCD_RW : out  STD_LOGIC;
LCD_EN : out  STD_LOGIC;
data : out  STD_LOGIC_VECTOR (3 downto 0)
```

9）VGA 显示模块

输入信号 IN：

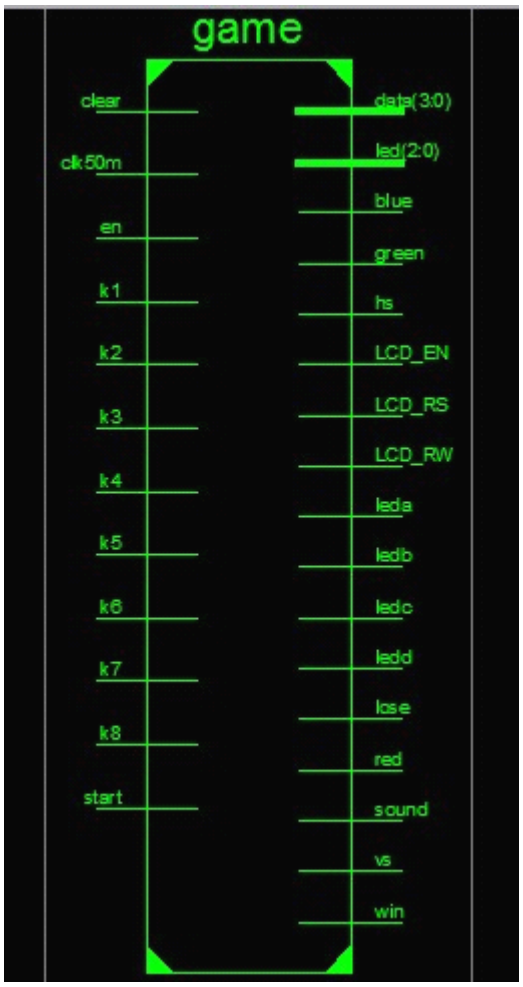
```
start:in std_logic;          --开始信号，高电平有
en:in STD_LOGIC;                  --暂停信号，en=0 时暂停
led : in  STD_LOGIC_VECTOR (2 downto 0);
key : in  STD_LOGIC_VECTOR (2 downto 0);
clk50m:in std_logic;
win,lose: in std_logic
```

输出信号 OUT：

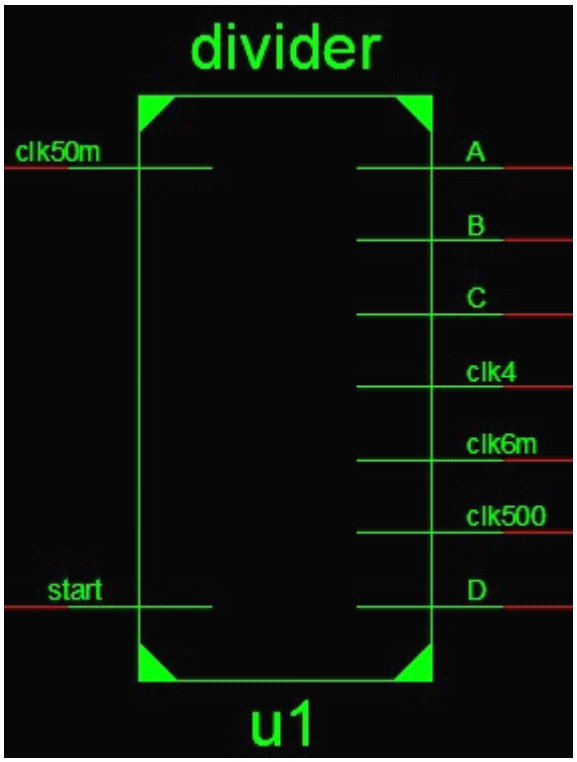
```
hs : out std_logic;                                --屏幕行扫描
vs : out std_logic;                                --屏幕场扫描
red : out std_logic;                               --显示红色输出
green : out std_logic;                             --显示绿色输出
blue : out std_logic;                               --显示蓝色输出
```

2.各模块框图

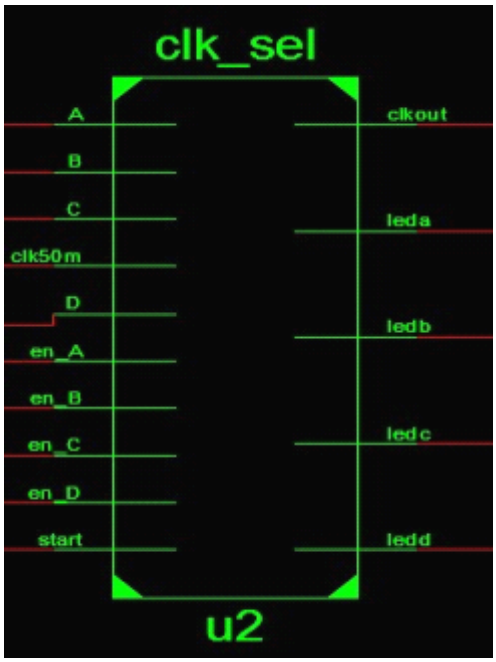
全系统的框图如下：


game
clear
clk50m
en
k1
k2
k3
k4
k5
k6
k7
k8
start
data(3:0)
led(2:0)
blue
green
hs
LCD_EN
LCD_RS
LCD_RW
leda
ledb
ledc
ledd
lose
red
sound
vs
win

a)分频模块框图：

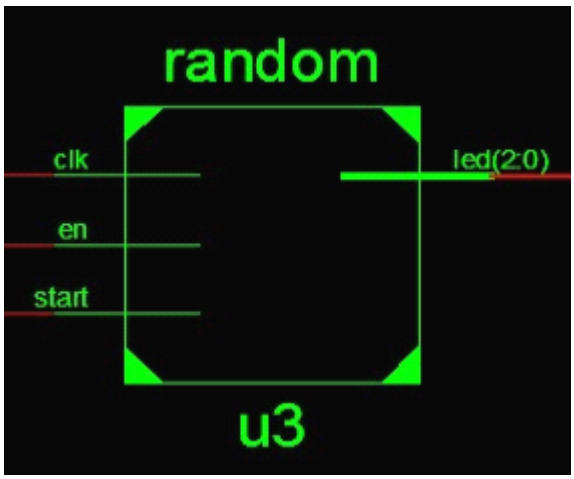

divider
clk50m
start
A
B
C
clk4
clk6m
clk500
D
u1

输入为 50MHz 的时钟信号，开始信号 start；
输出为 A,B,C,D 四种难度的时钟，500Hz 的时钟驱动 LCD，以及 4Hz 和 6.25MHz 的时钟供音乐模块使用。
b)时钟选择模块的框图为：



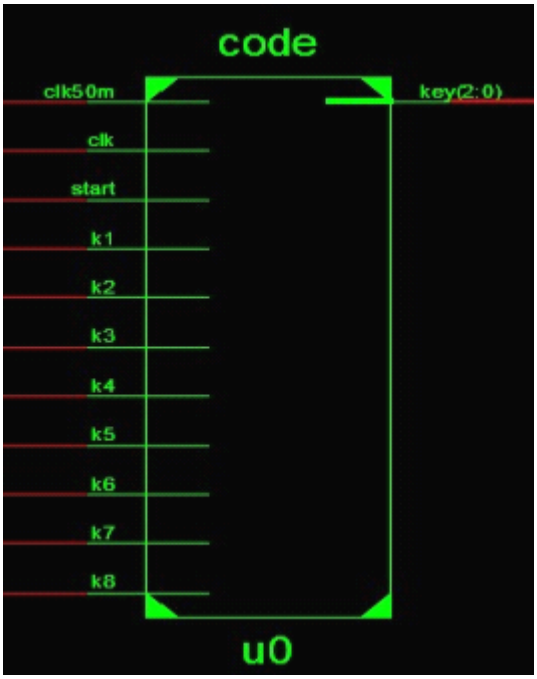
输入为 A,B,C,D 四种难度的时钟，50MHz 的时钟 clk50m，开始信号 start，
en_A,en_B,en_C,en_D 四种时钟的选择信号；
输出为 leda，ledb，ledc，ledd 四个对应不同难度的 LED，选择的 A,B,C,D 难度中的一个时钟。

c)随机数产生模块的框图为：



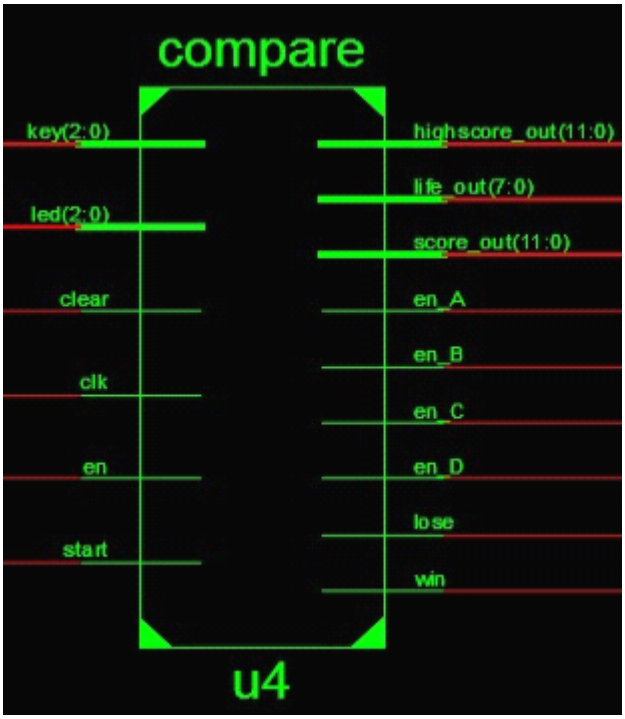
输入为选择的难度时钟 clk，暂停信号 en，开始信号 start；
输出为产生的三位随机数。

d) 采样编码模块的框图为:



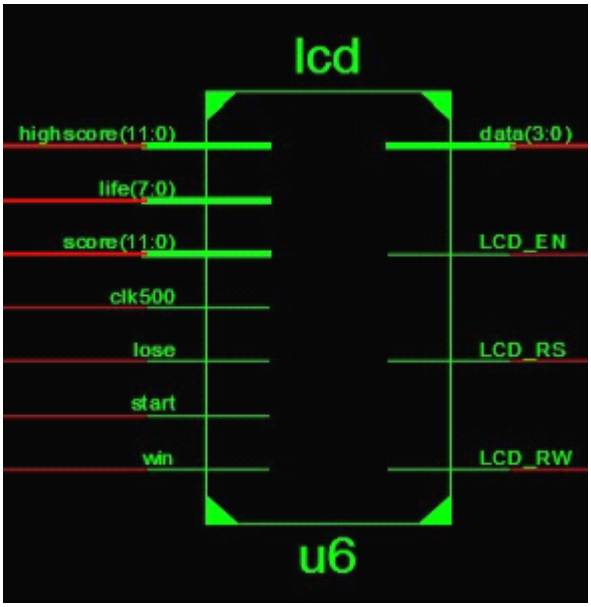
输入为 50MHz 的时钟信号 clk50m，选择的难度信号 clk，开始信号 start，以及八个按键；
输出为编码值 key

e）匹配模块的框图为:



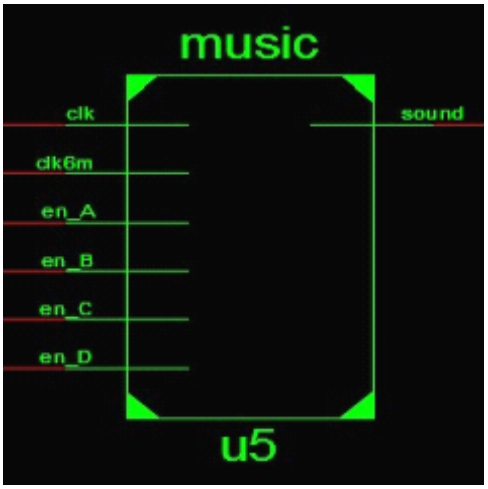
输入选择的难度时钟 clk，最高分清零信号 clear，暂停信号 en，开始信号 start，编码 key，随机数 led；
输出为最高分 highscore_out，当前分数 score_out，当前命数 life_out，不同难度的选择信号 en_A，en_B，en_C，en_D，输信号 lose，赢信号 win。

f) LCD 显示模块的框图为：



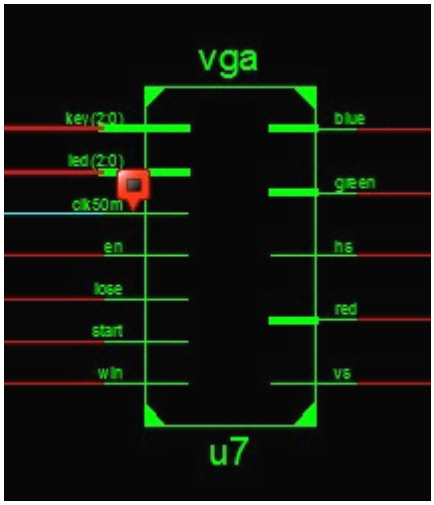
输入为最高分 highscore，当前分数 score，当前命数 life，500Hz 的时钟 clk500，输信号 lose，赢信号 win，开始信号 start；
输出为数据输出信号 data，LCD 使能信号 LCD_EN，LCD 读写信号 LCD_RW，LCD 写指令/数据信号 LCD_RS。

g) 音乐模块的框图为：



输入为不同难度的选择信号 en_A,en_B,en_C,en_D,6.25MHz 的时钟信号 clk6m，4Hz 的时钟信号 clk4；
输出为蜂鸣器的信号 sound。

h)vga 显示模块的框图：



输入为按键输入编码后的值 key（2:0），产生的随机数 led（2:0），时钟信号 clk50m，暂停信号 en，开始信号 start，输信号 lose，赢信号 win；

输出为屏幕行扫描 hs，屏幕场扫描 vs；红色输出 red ，绿色输出 green ，蓝色输出 blue

3. 测试平台设计

a)分频模块:仿真文件为 test_divider.tbw。

b)时钟选择模块:仿真文件为 test_clk_sel.tbw。

c)随机数产生模块:仿真文件为 test_random.tbw。

d)采样编码模块:仿真文件为 test_code.tbw。

e) 匹配模块:仿真文件为 test_compare.tbw。

f)LCD 显示模块:仿真文件为 test_lcd.tbw。

g) 音乐模块:由于音乐模块中涉及到分频，且分频较大，仿真得不到结果，故没有对音乐模块进行仿真。

h)vga 显示模块:仿真文件为 test_vga.tbw。

4.引脚分配及下载过程：

引脚分频图如下：

```
NET "clear" LOC = H18;
NET "clk50m" LOC = C9;
NET "data[0]" LOC = R15;
NET "data[1]" LOC = R16;
NET "data[2]" LOC = P17;
NET "data[3]" LOC = M15;
NET "en" LOC = L14;
NET "k1" LOC = A6;
NET "k2" LOC = B6;
NET "k3" LOC = E7;
NET "k4" LOC = F7;
NET "k5" LOC = D7;
NET "k6" LOC = C7;
NET "k7" LOC = F8;
NET "k8" LOC = E8;
NET "LCD_EN" LOC = M18;
NET "LCD_RS" LOC = L18;
NET "LCD_RW" LOC = L17;
NET "led[0]" LOC = B4;
NET "led[1]" LOC = A4;
NET "led[2]" LOC = D5;
NET "leda" LOC = F11;
NET "ledb" LOC = E11;
NET "ledc" LOC = E12;
NET "ledd" LOC = F12;
NET "lose" LOC = E9;
NET "sound" LOC = C5;
NET "start" LOC = L13;
NET "win" LOC = F9;
```

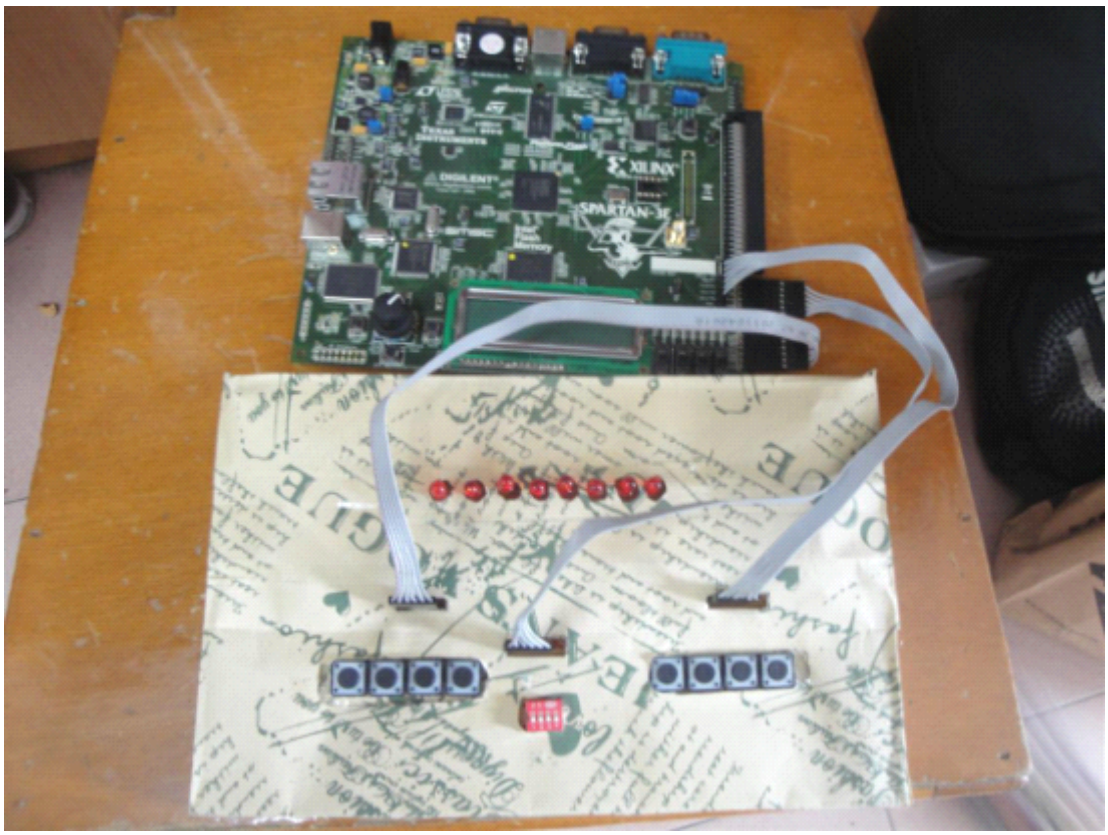
```
NET "green" LOC = H15;
NET "hs" LOC = F15;
NET "red" LOC = H14;
NET "vs" LOC = F14;

# PlanAhead Generated physical constraints

NET "blue" LOC = G15;
```

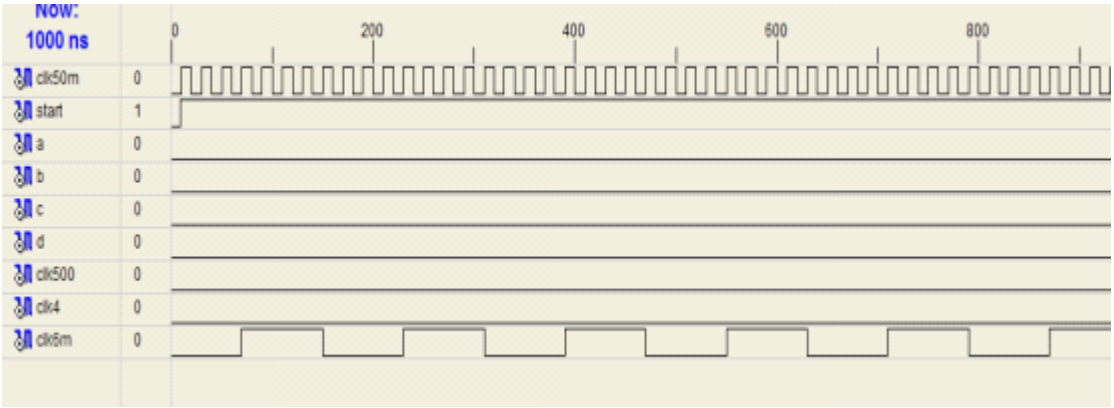
七、系统硬件设计与实现：

在打地鼠游戏机设计中我们采用了 Xilinx Spartan™-3E XC3S500E FPGA 实验开发板（其拥有多个 I/O 接口及强大功能，具体介绍见附录）以及焊接在 PCB 板上的外围电路（板上包括自己设计的发光二极管、按键输入和蜂鸣器，3—8 译码器 74LS138 等模块）



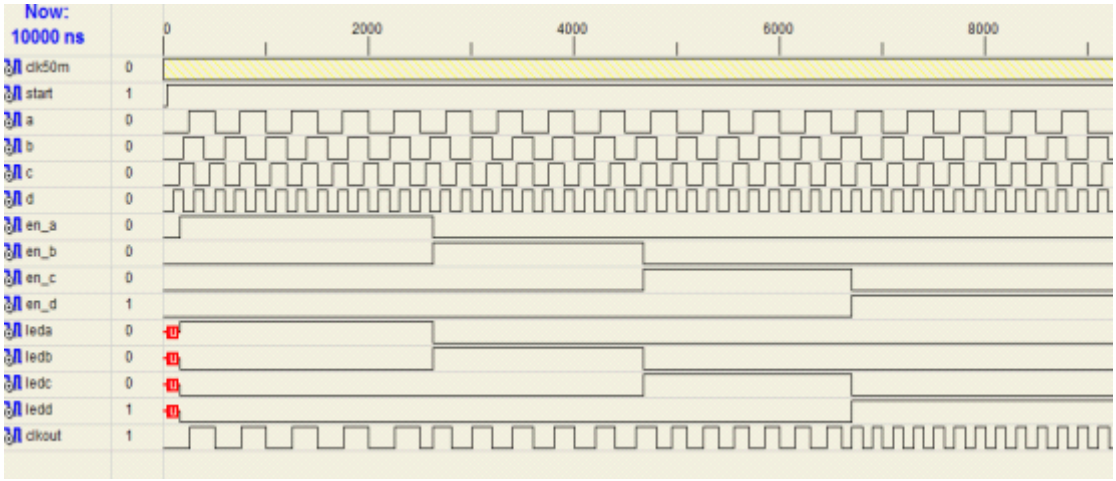
八、系统测试与结果分析：

①分频模块仿真结果：

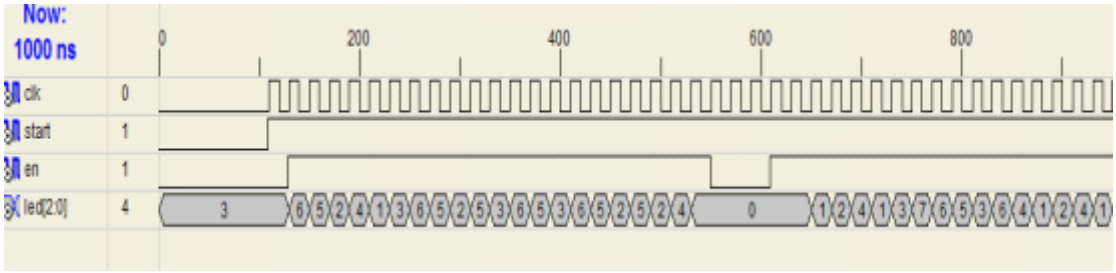


由于 A,B,C,D 的频率较小，得不到仿真结果，所以将代码烧录到板子上通过 LED 灯的亮灭来观察 a,b,c,d 的频率，观察结果与预期相符。

②时钟选择模块仿真结果：

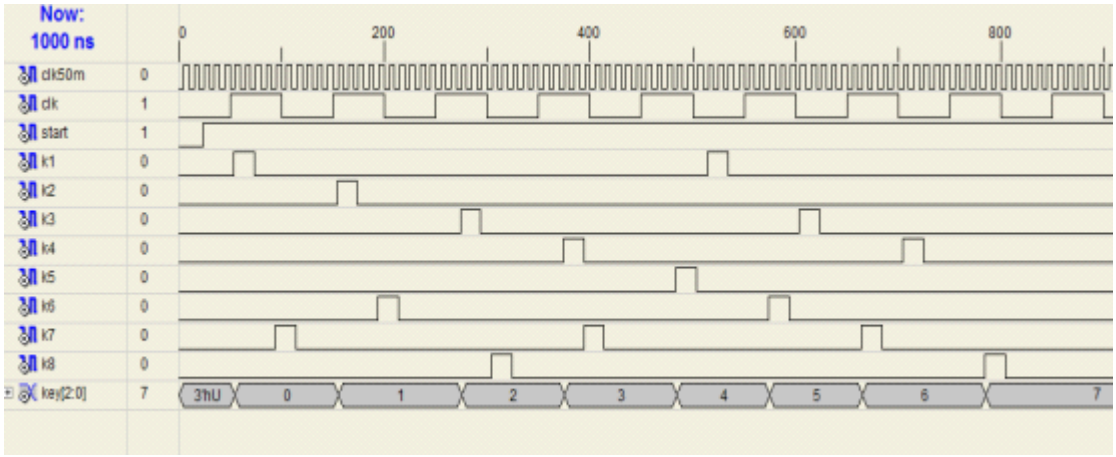


③随机数产生模块仿真结果：



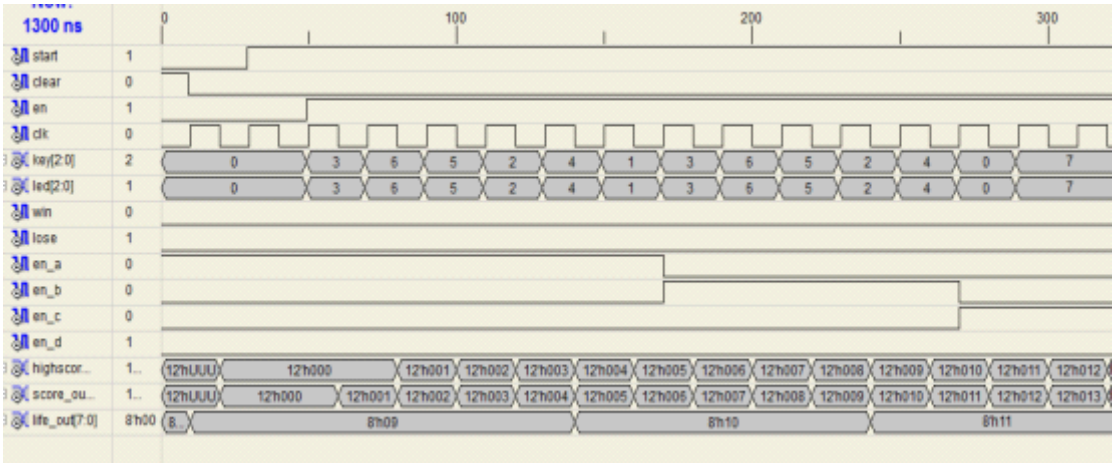
由仿真图可知，产生的 led 可认为是随机数。

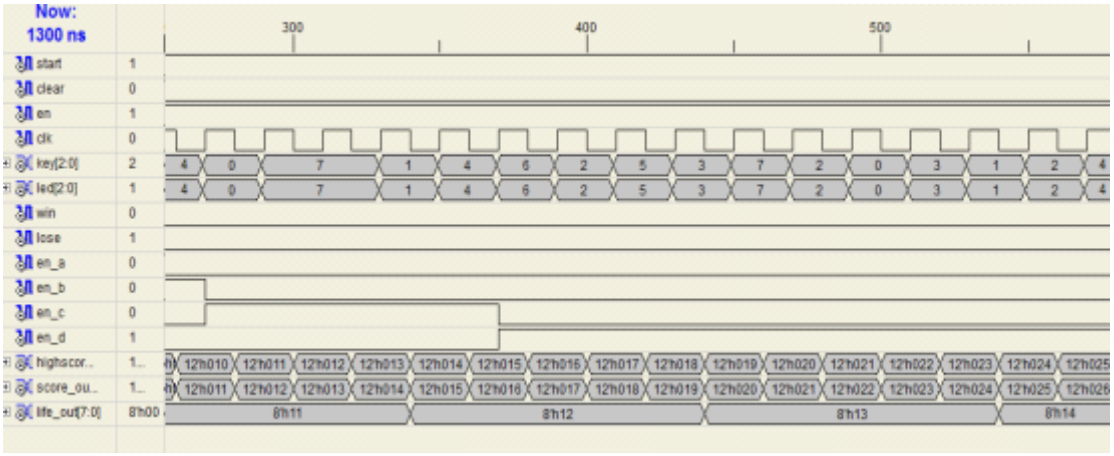
④采样编码模块仿真结果：

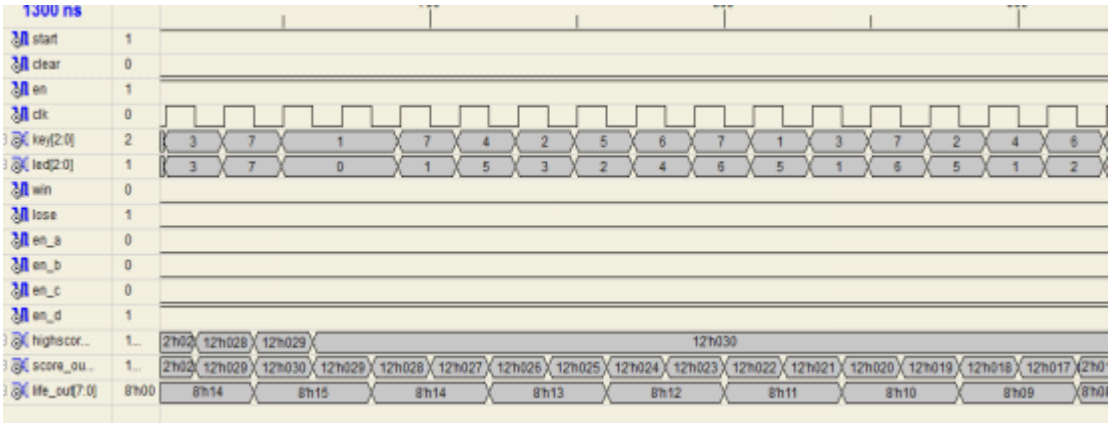


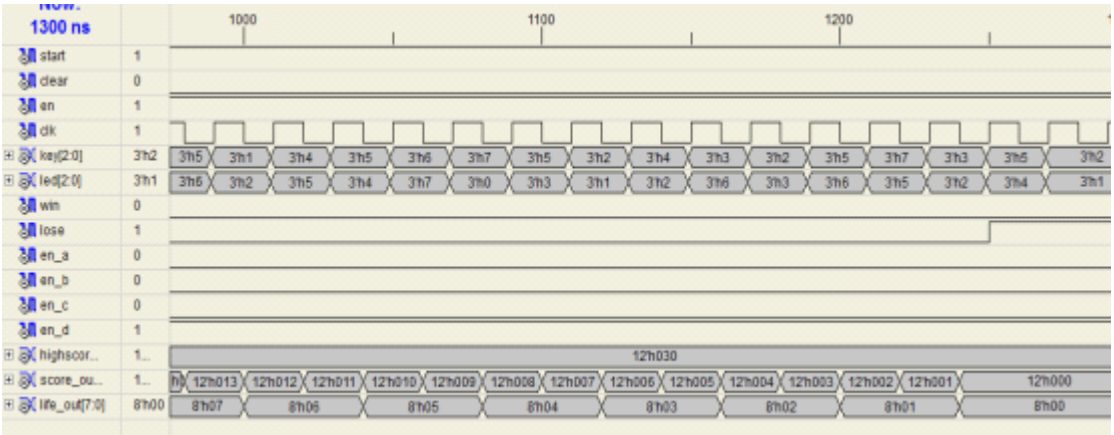
由仿真图可知，在一个 clk 周期内虽然有多次按键输入，但只有第一次的按键有效，即在一个 clk 周期内只采样编码一次。

⑤匹配模块仿真结果：


1300 ns
start
clear
en
clk
key[2:0]
led[2:0]
win
lose
en_a
en_b
en_c
en_d
highscor...
score_ou...
life_out[7:0]

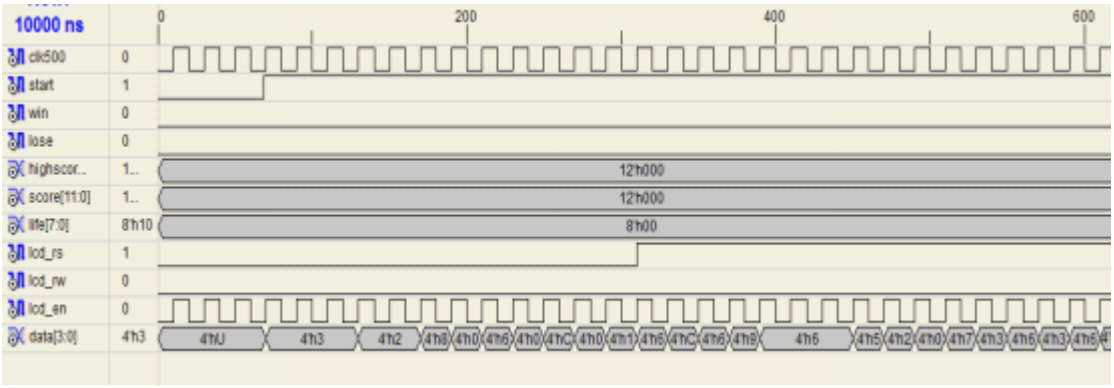

Now:
1300 ns
start
clear
en
clk
key[2:0]
led[2:0]
win
lose
en_a
en_b
en_c
en_d
highscor...
score_ou...
life_out[7:0]

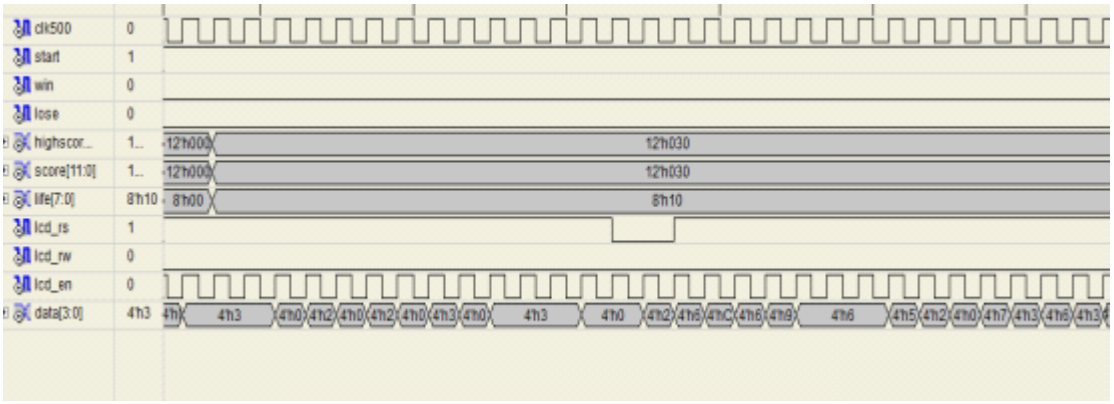

1300 ns
start
clear
en
clk
key[2:0]
led[2:0]
win
lose
en_a
en_b
en_c
en_d
highscor...
score_ou...
life_out[7:0]

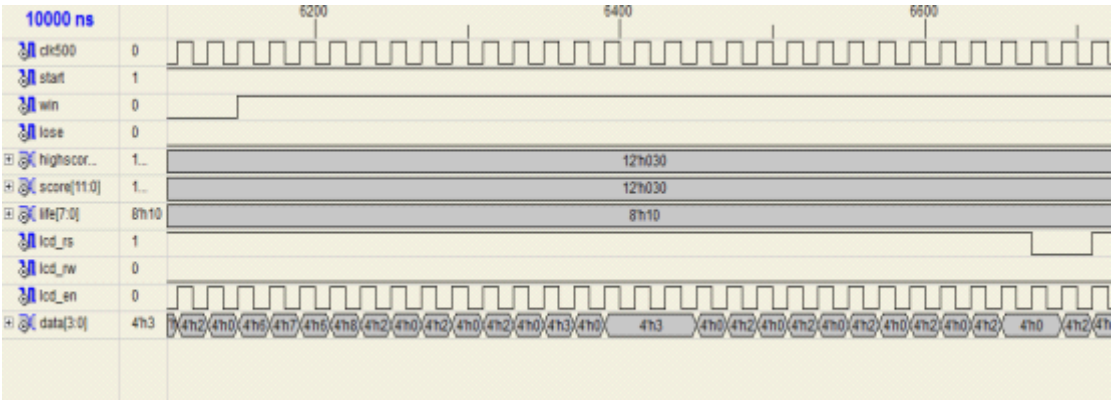

1300 ns
start
clear
en
clk
key[2:0]
led[2:0]
win
lose
en_a
en_b
en_c
en_d
highscor...
score_ou...
life_out[7:0]

由以上波形可看出，匹配时，分数增加一分，分数每增加 5 分，命数增加一条，分数在 5 以下时难度为 A，在 5-10 之间时难度为 B,在 10-15 之间时难度为 C,在 15 以上时难度为 D,且难度只升不降，不匹配时，分数减一分，分数每减两分，命数扣一条，最高分只升不降。最高分比当前分数滞后一个周期。当命数为 0 时游戏结束。

⑥LCD 驱动模块仿真结果：







九、心得体会与项目总结：

1.心得体会：

最初选题时，之所以选择打地鼠游戏机的设计一是觉得比较有趣味性，让我想起了之前陪小侄女去儿童游乐园玩时，有见过类似游戏，有塑料锤子敲下随机冒出来的小动物，规定时间内打到即得分，该游戏反应的就是人的反应速度和身体的灵活性，挺有意思，再者，也选修了 FPGA 的课程，正好能学以致用。

虽然从很早以前就有一些准备，查阅各种资料，学习硬件编程语言，使用软件等等，但

真正开始还是有些晚的，我们小组成员分工合作先把一些相对简单的模块写了，比如分频、时钟选择、随机数产生、采样编码等。等到最后两周我们也完成了另外的三个模块。

在这里，我想要指出的是，我们在写音乐模块时，因为根据老师所给课设题目要求，需要用不同的频率驱动喇叭，使得游戏难度越大，相应的音乐频率越快。可是当我们真的直接用不同频率驱动不同难度时，时间一长，耳朵就受不了，所以我们改用不同难度对应不同的乐曲，通过从网上查找资料，了解了音乐产生的原理，其实是每一个音阶对应不同的频率，于是从网上找了四个谱子，照着谱子把音阶填上去了，但是结果和预期有很大差别，好在能区分出四个曲子不同，基本满足要求。

另外，在写 LCD 驱动模块时，因为需要显示三种不同的结果，所以要用 if else 语句，而且每种情况显示的长度不相同，于是就会出现上一次显示的字符在下一个不同情况时还在，我们首先想到的方法是用清屏指令，即每一种情况完后用清屏指令清屏，但是显示屏不稳定，会出现有些字符看不到的情况，最后采用的办法是用空白字符填充所有不需要的地方，这样就满足了要求。

再就是匹配模块，最先是用难度时钟进行匹配加减分，但是仿真得不到正确结果，于是就改用 50MHz 的频率进行匹配加减分，需要设置标志位，保证在一个难度时钟内只匹配一次，另外需要将编码延时一个难度周期，只有在前后两次按键不一样时才进行匹配，这样的好处就是分数更新很快，几乎没有延时，但缺点就是如果前后两次按键一样时就不进行匹配，既不加分也不减分，不满足要求。最后我们采用的办法是用难度时钟的上升沿进行匹配，在难度时钟的下降沿进行加减分，这样能满足要求，不过加减分仍有一定的延迟。

最麻烦的当属 VGA 模块了，VGA 模块是我们在最后几天才开始接触的，所以时间也比较赶。在两天的努力下，我们终于大致做出了，虽然不那么美观，但可以在显示屏上准确显示，但我们发现这时却不可以准确地打地鼠了，调试修改，反反复复，却仍有一定问题。

以上是软件部分，事实上我们的硬件部分基本上是最后两周才开始行动。除了用到 Xilinx 大学计划给定的 Xilinx Spartan™-3E XC3S500E FPGA 实验开发板，我们还需自己设计制作外围扩展板，老师也为我们提供了焊接用的 PCB 板。我们首先设计出了电路图，用到了 8 个发光二极管，3-8 译码器 74LS138，以及 8 个按键开关等，由于担心直接开始在 PCB 板上焊接不妥，出现错误也难改，于是我们先用面包板插好电路图，进行测试，刚开始一直出问题，要么灯不亮，要么地鼠打中也不加分，后来通过反复的检错排错，终于试验成功！

在焊接 PCB 板的过程中，我们也花费了不少时间，由于我们外围板的设计较简单，所以也没专门用 protel99se 软件画 PCB 图，只是先在板上设计好各器件方位才开始焊接。由于好久没接触这焊接工艺，早已生疏，所以一开始焊得很费劲，且十分难看，后来才慢慢顺手些，焊完第一次测试也问题重重，通过反复修改，第二天才得以成功，可以很好地模拟打地鼠的情况了。实属不易啊。

最后为了使作品美观化产品化，我们还特意对 PCB 板进行了包装和封装。做出来效果还不错，恩，还挺有成就感。

这次硬件课设很好的锻炼了我的编写代码的能力，以及出现问题时通过询问老师、和其他同学讨论、上网查找资料、图书馆查阅相关书籍等各种途径找到问题并很好地解决，很好地锻炼了收集信息的能力。

2.项目总结：

1.刚开始运行时会出现不稳定的情况，有时候一开始就提示游戏结束，不过重新开始就可以了。

2.如果有更多的时间，我们相信我们会把 VGA 部分做的更好。

十、致谢：

本次硬件课设历时时间之久，这一路走来幸得老师和同学的指导与帮助，在此我们十分感激。首先要感谢的便是我们的指导老师杨小献老师，从最初的课题讲解到我们实验设计过程中的疑难解答，对我们进度的关心，到后期器件的提供与指导，杨老师为我们提供了舒适的实验环境，实验室全天开放也让整个项目的进度能够快很多。另外，我们还要感谢身边同院系同学的帮助，感谢他们对于那些疑难问题的不厌其烦的讲解。还要感谢其他组的成员，和他们的交流让我们认识到自己的不足，给自己以鞭策，激励自己不断提高。

最后，感谢电信系和赛灵思 FPGA 大学计划。另外，特别要提的是赛灵思公司的免费申请样片服务，这让我们增加了接近和了解赛灵思芯片和赛灵思公司的机会，也给我们提供了一个可以发挥我们想象力的平台。

十一、参考文献

[1]李云松 宋锐 雷杰等 Xilinx FPGA 设计基础 西安电子科技大学出版社
[2]孟宪元，钱伟康，FPGA 嵌入式系统设计，电子工业出版社，2007
[3]罗杰，电子线路设计.实验.测试，电子工业出版社，2008

附录：

1.实验平台 Xilinx 公司的 SPARTAN3E 实验板介绍

- XILINX XC3S500E Spartan-3E FPGA：提供了最多 232 个 I/O 引脚和 10000 个逻辑单元。
- XILINX 4Mbit Flash 配置 PROM。
- XILINX XC2C64A CoolRunner 系列 CPLD：提供用户使用或辅助 FPGA 配置。
- 64MByte、16 位数据宽度、100MHz 的 DDR SDRAM 接口。
- 16MByte 并行 INTEL 公司的 NOR FLASH:可存储 FPGA 配置信息或 MicroBlaze 指令序列。
- 16Mbits ST 半导体的 SPI 串行 FLASH：可存储 FPGA 配置信息或 MicroBlaze 指令序列。
- 2 行，每行可显示 16 个字符的 LCD：用来显示 FPGA 输出信息。
- PS/2 接口：用来外联键盘或鼠标，扩展输入设备
- VGA 接口：可显示 64 种颜色。
- 10/100M 以太网接口：提供了以太网物理层接口，便于 MAC 层 IP 的验证。
- 两个标准 RS232 接口：可方便连接 PC 和其他工业设备进行数据传输。
- USB 的下载接口配置接口。
- 板载 50MHz 晶体振荡器。
- 4 输出基于 SPI 接口的数模转换器。
- 2 输入基于 SPI 接口、带可编程增益放大的模数转换器。
- ChipScope 在线调试接口：可提供在线调试信号功能。
- 4 个拨码开关、1 个旋钮、4 个按键：可作为通用逻辑输入。
- 8 盏 LED：可用来显示 FPGA 的输出信息。
- 提供 8 脚 DIP 封装的辅助时钟输入
- 提供了标准外扩接口，供用户灵活添加使用。

2.源程序

①采样编码模块：

```
entity code is
port(clk50m:in std_logic;          --50MHz 的输入时钟
     clk:in std_logic;             --时钟选择后得到的时钟，即不同难度对应的时钟
     start:in std_logic;           --开始信号，高电平有效
     k1,k2,k3,k4,k5,k6,k7,k8:in std_logic;     --8 个输入按键
     key:out std_logic_vector(2 downto 0)); --按键编码后得到的编码
end code;
architecture Behavioral of code is
signal flag:std_logic;          --标志位
begin
process(clk50m,clk,start)
begin
if start='0' then                    --初始时令 flag=0
   flag<='0';
elsif clk50m='1' then--用 50MHz 的时钟进行采样编码，且在一个难度时钟内只采样编码一次
   if flag='0' then
      if k8='1' then
         key<="111";
         flag<='1';                  --flag=1 时表明已经采样编码完毕
      elsif k7='1' then
         key<="110";
         flag<='1';
      elsif k6='1' then
         key<="101";
         flag<='1';
      elsif k5='1' then
         key<="100";
         flag<='1';
      elsif k4='1' then
         key<="011";
         flag<='1';
      elsif k3='1' then
         key<="010";
         flag<='1';
      elsif k2='1' then
         key<="001";
         flag<='1';
      elsif k1='1' then
         key<="000";
         flag<='1';
      end if;
   end if;
```

```
elsif rising_edge(clk) then          --在难度时钟的上升沿将 flag 置 0，表明可进行采样编码
    if flag='1' then
        flag<='0';
    end if;
end if;
end process;
end Behavioral;
```

②分频模块：

```
entity divider is
    Port ( clk50m : in    STD_LOGIC;        --50MHz 的输入时钟
           start:in STD_LOGIC;              --开始信号，高电平有效
           A : out    STD_LOGIC;            --难度 A 的周期为 2.5 秒
           B : out    STD_LOGIC;            --难度 B 的周期为 2.0 秒
           C : out    STD_LOGIC;            --难度 C 的周期为 1.5 秒
           D : out    STD_LOGIC;            --难度 D 的周期为 1.0 秒
           clk500:out STD_LOGIC;            --clk500=500Hz
           clk4:out std_logic;              --clk=4Hz
           clk6m:out std_logic);            --clk6m=6.25MHz
end divider;
architecture Behavioral of divider is
signal ma,mb,mc,md,m4,m500,m6m:std_logic:='0';
signal cnt1:integer range 1 to 50000;
signal cnt2:integer range 1 to 6250000;
signal cnt3:integer range 1 to 2;
signal cnt4:integer range 1 to 5;
signal cnt5:integer range 1 to 3;
begin
A<=ma;
B<=mb;
C<=mc;
D<=md;
clk500<=m500;
clk6m<=m6m;
clk4<=m4;
process(clk50m)
    variable n:integer range 1 to 5:=1;
begin
    if rising_edge(clk50m) then      --50MHz8 分频后得到 6.25MHz 的信号 clk6m
        n:=n+1;
        if n=5 then
            m6m<=not m6m;
            n:=1;
        end if;
        if cnt1=50000 then                  --50MHz100000 分频后得到 500Hz 的信号
```

```
clk500
            m500<=not m500;
            cnt1<=1;
        else
            cnt1<=cnt1+1;
        end if;
        if cnt2=6250000 then          ----50MHz12500000 分频后得到 4Hz 的信号 clk4
            m4<=not m4;
            cnt2<=1;
        else
            cnt2<=cnt2+1;
        end if;
    end if;
end process;
process(start,m4,md)
begin
    if start='0' then                                   --start=0 时初始化
        ma<='0';
        mb<='0';
        mc<='0';
        md<='0';
        cnt3<=1;
        cnt5<=1;
    else
        if rising_edge(m4) then                         --4Hz 时钟 4 分频后得到 1Hz
时钟 D
            if cnt3=2 then
                cnt3<=1;
                md<=not md;
            else
                cnt3<=cnt3+1;
            end if;
            if cnt4=5 then                              --4Hz 时钟 10 分频后得到 0.4Hz 时
钟 A
                ma<=not ma;
                cnt4<=1;
            else
                cnt4<=cnt4+1;
            end if;

            if cnt5=3 then                              --4Hz 时钟 6 分频后得到 0.67Hz 时
钟 C
                mc<=not mc;
                cnt5<=1;
```

```
                else
                    cnt5<=cnt5+1;
                end if;
            end if;
            if rising_edge(md) then                    --1Hz 时钟 2 分频后得到 0.5Hz 的时
钟 B
                mb<=not mb;
            end if;
        end if;
end process;
end Behavioral;
```

③时钟选择模块：

```
entity clk_sel is
port(clk50m:in std_logic;          --50MHz 的输入时钟
        start:in std_logic;        --开始信号，高电平有效
        A,B,C,D:in std_logic;      --A,B,C,D 分别对应难度 A，B，C，D 的时钟
 en_A,en_B,en_C,en_D:in std_logic;--en_A,en_B,en_C,en_D 分别为 A，B，C，D 的使能信号
        leda,ledb,ledc,ledd:out std_logic;--不同的难度用不同的 LED 来指示
        clkout:out std_logic);     --不同的难度输出对应的难度时钟
end clk_sel;
architecture Behavioral of clk_sel is
signal m:std_logic_vector(3 downto 0);
signal clk:std_logic:='0';
begin
m<=en_A&en_B&en_C&en_D;
clkout<=clk;
process(clk50m)
begin
if start='0' then
    clk<='0';
elsif rising_edge(clk50m) then
    if m=8 then        --当 en_A='1'时，时钟对应为难度 A 的时钟，且指示难度 A 的 LED
亮
        clk<=A;
        leda<='1';
        ledb<='0';
        ledc<='0';
        ledd<='0';
    elsif m=4 then          --当 en_B='1'时，时钟对应为难度 B 的时钟，且指示难度 B 的
LED 亮
        clk<=B;
        leda<='0';
        ledb<='1';
        ledc<='0';
```

```
            ledd<='0';
        elsif m=2 then          --当 en_C='1'时，时钟对应为难度 C 的时钟，且指示难度 C 的
LED 亮
            clk<=C;
            leda<='0';
            ledb<='0';
            ledc<='1';
            ledd<='0';
        elsif m=1 then          --当 en_D='1'时，时钟对应为难度 D 的时钟，且指示难度 D 的
LED 亮
            clk<=D;
            leda<='0';
            ledb<='0';
            ledc<='0';
            ledd<='1';
        else
            clk<='0';
        end if;
end if;
end process;
end Behavioral;
```

④随机数产生模块：

```
entity random is
    Port ( clk : in   STD_LOGIC;            --难度时钟
           start:in STD_LOGIC;              --开始信号，高电平有效
           en:in STD_LOGIC;                 --暂停信号，en=0 时暂停
           led : out   STD_LOGIC_VECTOR (2 downto 0));  --产生的三位随机数
end random;
architecture Behavioral of random is
signal m:std_logic_vector(7 downto 0):="10000000";
begin
led<=m(2 downto 0);
process(clk,en,start)
begin
    if start='0' then
        m<="11000011";
    elsif en='0' then
        m<=m;
    elsif rising_edge(clk) then
        m(7 downto 1)<=m(6 downto 0);
        m(0)<=m(7) xor m(3) xor m(2) xor m(1);
    end if;
end process;
end Behavioral;
```

⑤匹配模块：

```
entity compare is
port(      start:in std_logic;        --开始信号，高电平有效
           clear:in std_logic;        --最高分清零信号，clear=1 时清除最高分纪录
           en:in std_logic;                --暂停信号，en=0 时暂停
           clk : in    STD_LOGIC;          --clk50m 为 50MHz 的输入时钟，clk 为难度时钟
         key : in    STD_LOGIC_VECTOR (2 downto 0);       --key 为编码后得到的编码值
         led : in    STD_LOGIC_VECTOR (2 downto 0);--led 为产生的三位随机数
           win,lose:out STD_LOGIC;     --win=1 表明游戏胜利，lose=1 表明游戏失败
         en_A : out    STD_LOGIC; --难度 A 的使能信号
         en_B : out    STD_LOGIC; --难度 B 的使能信号
         en_C : out    STD_LOGIC; --难度 C 的使能信号
         en_D : out    STD_LOGIC;   --难度 D 的使能信号
           highscore_out:out STD_LOGIC_VECTOR(11 downto 0); --最高分
         score_out : out    STD_LOGIC_VECTOR (11 downto 0);       --当前分数
         life_out : out    STD_LOGIC_VECTOR (7 downto 0));          --当前生命数
end compare;
architecture Behavioral of compare is
signal mscore,mhighscore:integer range 0 to 999;
signal mlife:integer range 0 to 99;
signal result,mwin,mlose:std_logic;
signal cnt1:integer range 0 to 4;
signal cnt2:integer range 0 to 1;
type state is(na,nb,nc,nd);
signal nandu:state;
begin
result<=mwin or mlose;            --当 result=1 时游戏结束，LCD 上会显示输或赢
win<=mwin;
lose<=mlose;
--得 5 分加一条命，失误一次扣一分，失误达到 2 次扣一条命，当分数为 0 时，失误一次不减分，但扣一条命
process(clk,start,en,clear)
begin
     if mscore>mhighscore then
          mhighscore<=mscore;
     else
          mhighscore<=mhighscore;
     end if;
     if start='0' then
          mscore<=0;         --初始分数为 0
          mlife<=9;          --初始命数为 9
          cnt1<=0;
          mhighscore<=mhighscore;
          cnt2<=0;
```

```
        mwin<='0';
        mlose<='0';
        nandu<=na;
        en_A<='1';
        en_B<='0';
        en_C<='0';
        en_D<='0';
    elsif clear='1' then
        mhighscore<=0;
    elsif en='0' then
        mscore<=mscore;
        mhighscore<=mhighscore;
        mlife<=mlife;
    elsif rising_edge(clk) then
        if result='0' then
        if led=key then
            if mlife=99 then
                mwin<='1';
            elsif mlife<99 then
                if mscore=999 then
                    mwin<='1';
                elsif mscore<999 then
                    mscore<=mscore+1;
                    if cnt1=4 then
                        mlife<=mlife+1;
                        cnt1<=0;
                    else
                        cnt1<=cnt1+1;
                    end if;
                end if;
            elsif mlife=0 then
                mlose<='1';
            end if;
        else
            if mscore=0 and mlife=0 then
                mscore<=0;
                mlife<=0;
                mlose<='1';
            elsif mlife=0 then
                mlife<=0;
                mscore<=mscore;
                mlose<='1';
            elsif mscore=0 and mlife>0 then
                mscore<=0;
```

```
            mlife<=mlife-1;
        else
            mscore<=mscore-1;
            if cnt2=1 then
                mlife<=mlife-1;
                cnt2<=0;
            else
                cnt2<=cnt2+1;
            end if;
        end if;
end if;
end if;
case nandu is
when na=>          --当分数小于 5 分时，难度为 A
        en_A<='1';
        en_B<='0';
        en_C<='0';
        en_D<='0';
        if mscore>=5 and mscore<10 then
            nandu<=nb;
        else
            nandu<=na;
        end if;
when nb=>   --当分数在 5-10 之间时，难度为 B
        en_A<='0';
        en_B<='1';
        en_C<='0';
        en_D<='0';
        if mscore>=10 and mscore<15 then
            nandu<=nc;
        else
            nandu<=nb;
        end if;
when nc=>   --当分数在 10-15 之间时，难度为 C
        en_A<='0';
        en_B<='0';
        en_C<='1';
        en_D<='0';
        if mscore>=15 and mscore<20 then
            nandu<=nd;
        else
            nandu<=nc;
        end if;
when nd=>   --当分数在 15 以上时，难度为 D
```

```
                en_A<='0';
                en_B<='0';
                en_C<='0';
                en_D<='1';
                nandu<=nd;
            when others=>
                en_A<='1';
                en_B<='0';
                en_C<='0';
                en_D<='0';
                nandu<=na;
            end case;
        end if;
end process;
process(clk)
begin
        if falling_edge(clk) then
            if mscore>=900 then
                score_out(11 downto 8)<=x"9";
                if (mscore-900)>=90 then
                    score_out(7 downto 4)<=x"9";
                    score_out(3 downto 0)<=conv_std_logic_vector((mscore-990),4);
                elsif (mscore-900)>=80 then
                    score_out(7 downto 4)<=x"8";
                    score_out(3 downto 0)<=conv_std_logic_vector((mscore-980),4);
                elsif (mscore-900)>=70 then
                    score_out(7 downto 4)<=x"7";
                    score_out(3 downto 0)<=conv_std_logic_vector((mscore-970),4);
                elsif (mscore-900)>=60 then
                    score_out(7 downto 4)<=x"6";
                    score_out(3 downto 0)<=conv_std_logic_vector((mscore-960),4);
                elsif (mscore-900)>=50 then
                    score_out(7 downto 4)<=x"5";
                    score_out(3 downto 0)<=conv_std_logic_vector((mscore-950),4);
                elsif (mscore-900)>=40 then
                    score_out(7 downto 4)<=x"4";
                    score_out(3 downto 0)<=conv_std_logic_vector((mscore-940),4);
                elsif (mscore-900)>=30 then
                    score_out(7 downto 4)<=x"3";
                    score_out(3 downto 0)<=conv_std_logic_vector((mscore-930),4);
                elsif (mscore-900)>=20 then
                    score_out(7 downto 4)<=x"2";
                    score_out(3 downto 0)<=conv_std_logic_vector((mscore-920),4);
                elsif (mscore-900)>=10 then
```

```
                score_out(7 downto 4)<=x"1";
                score_out(3 downto 0)<=conv_std_logic_vector((mscore-910),4);
            else
                score_out(7 downto 4)<=x"0";
                score_out(3 downto 0)<=conv_std_logic_vector((mscore-900),4);
            end if;
            ......
        if mhighscore>=900 then
            highscore_out(11 downto 8)<=x"9";
            if (mhighscore-900)>=90 then
                highscore_out(7 downto 4)<=x"9";
                highscore_out(3 downto
0)<=conv_std_logic_vector((mhighscore-990),4);
            elsif (mhighscore-900)>=80 then
                highscore_out(7 downto 4)<=x"8";
                highscore_out(3 downto
0)<=conv_std_logic_vector((mhighscore-980),4);
            elsif (mhighscore-900)>=70 then
                highscore_out(7 downto 4)<=x"7";
                highscore_out(3 downto
0)<=conv_std_logic_vector((mhighscore-970),4);
            elsif (mhighscore-900)>=60 then
                highscore_out(7 downto 4)<=x"6";
                highscore_out(3 downto
0)<=conv_std_logic_vector((mhighscore-960),4);
            elsif (mhighscore-900)>=50 then
                highscore_out(7 downto 4)<=x"5";
                highscore_out(3 downto
0)<=conv_std_logic_vector((mhighscore-950),4);
            elsif (mhighscore-900)>=40 then
                highscore_out(7 downto 4)<=x"4";
                highscore_out(3 downto
0)<=conv_std_logic_vector((mhighscore-940),4);
            elsif (mhighscore-900)>=30 then
                highscore_out(7 downto 4)<=x"3";
                highscore_out(3 downto
0)<=conv_std_logic_vector((mhighscore-930),4);
            elsif (mhighscore-900)>=20 then
                highscore_out(7 downto 4)<=x"2";
                highscore_out(3 downto
0)<=conv_std_logic_vector((mhighscore-920),4);
            elsif (mhighscore-900)>=10 then
                highscore_out(7 downto 4)<=x"1";
                highscore_out(3 downto
```

```
0)<=conv_std_logic_vector((mhighscore-910),4);
                else
                    highscore_out(7 downto 4)<=x"0";
                    highscore_out(3 downto
0)<=conv_std_logic_vector((mhighscore-900),4);
                end if;
                ......
            if falling_edge(clk) then
            if (mlife)>=90 then
                life_out(7 downto 4)<=x"9";
                life_out(3 downto 0)<=conv_std_logic_vector((mlife-90),4);
            elsif (mlife)>=80 then
                life_out(7 downto 4)<=x"8";
                life_out(3 downto 0)<=conv_std_logic_vector((mlife-80),4);
            elsif (mlife)>=70 then
                life_out(7 downto 4)<=x"7";
                life_out(3 downto 0)<=conv_std_logic_vector((mlife-70),4);
            elsif (mlife)>=60 then
                life_out(7 downto 4)<=x"6";
                life_out(3 downto 0)<=conv_std_logic_vector((mlife-60),4);
            elsif (mlife)>=50 then
                life_out(7 downto 4)<=x"5";
                life_out(3 downto 0)<=conv_std_logic_vector((mlife-50),4);
            elsif (mlife)>=40 then
                life_out(7 downto 4)<=x"4";
                life_out(3 downto 0)<=conv_std_logic_vector((mlife-40),4);
            elsif (mlife)>=30 then
                life_out(7 downto 4)<=x"3";
                life_out(3 downto 0)<=conv_std_logic_vector((mlife-30),4);
            elsif (mlife)>=20 then
                life_out(7 downto 4)<=x"2";
                life_out(3 downto 0)<=conv_std_logic_vector((mlife-20),4);
            elsif (mlife)>=10 then
                life_out(7 downto 4)<=x"1";
                life_out(3 downto 0)<=conv_std_logic_vector((mlife-10),4);
            else
                life_out(7 downto 4)<=x"0";
                life_out(3 downto 0)<=conv_std_logic_vector((mlife),4);
            end if;
            .....
end process;
end Behavioral;
```

⑥音乐产生模块：

```
entity music is
```

```
port(clk6m,clk:in std_logic;              --clk6m 为输入的 6.25MHz 的时钟，clk 为 4Hz 的时钟
     en_A,en_B,en_C,en_D:in std_logic; --分别为 A，B，C，D 的选择信号
     sound:out std_logic);                --为输出的喇叭信号
end music;

architecture Behavioral of music is
signal hml:std_logic_vector(11 downto 0);
signal di,ori:integer range 0 to 16383;
signal msound:std_logic:='0';
signal cnt:integer range 0 to 63;
begin
sound<=msound;
process(clk6m)
begin
    if rising_edge(clk6m) then
        if di=16383 then
            msound<=not msound;
            di<=ori;
        else
            di<=di+1;
        end if;
    end if;
end process;
process(clk)
begin
if rising_edge(clk) then
    case hml is
        when x"003"=>        --当 hml=x"003"时，sound 为 6.25MHz 的信号经过 2*
（16383-7280)=18206 次分频后得到的时钟信号，即 343.29Hz
            ori<=7281;
        when x"005"=>        --当 hml=x"005"时，sound 为 6.25MHz 的信号经过 2*
（16383-8729)=15308 次分频后得到的时钟信号，即 408.28Hz
            ori<=8730;
        when x"006"=>        --当 hml=x"006"时，sound 为 6.25MHz 的信号经过 2*
（16383-9564)=13638 次分频后得到的时钟信号，即 458.28Hz
            ori<=9565;
        when x"007"=>        --当 hml=x"007"时，sound 为 6.25MHz 的信号经过 2*
（16383-10309)=12148 次分频后得到的时钟信号，即 514.49Hz
            ori<=10310;
        when x"010"=>        --当 hml=x"010"时，sound 为 6.25MHz 的信号经过 2*
（16383-10646)=11474 次分频后得到的时钟信号，即 544.71Hz
            ori<=10647;
        when x"020"=>        --当 hml=x"020"时，sound 为 6.25MHz 的信号经过 2*
（16383-11271)=10224 次分频后得到的时钟信号，即 611.31Hz
```

```
                ori<=11272;
            when x"030"=>          --当 hml=x"030"时，sound 为 6.25MHz 的信号经过 2*
（16383-11830)=9106??分频后得到的时钟信号，即 686.36Hz
                ori<=11831;
            when x"050"=>          --当 hml=x"050"时，sound 为 6.25MHz 的信号经过 2*
（16383-12555)=7656 次分频后得到的时钟信号，即 816.35Hz
                ori<=12556;
            when x"060"=>          --当 hml=x"060"时，sound 为 6.25MHz 的信号经过 2*
（16383-12973)=6820 次分频后得到的时钟信号，即 916.42Hz
                ori<=12974;
            when x"100"=>          --当 hml=x"100"时，sound 为 6.25MHz 的信号经过 2*
（16383-13516)=5734 次分频后得到的时钟信号，即 1089.99Hz
                ori<=13516;
            when others=>
                ori<=16383;
            end case;
        end if;
end process;
process(clk)
begin
        if rising_edge(clk) then
            if cnt=63 then
                cnt<=0;
            else
                cnt<=cnt+1;
            end if;
                --不同难度下会产生不同的音乐
                if en_A='1' then
                    case cnt is
                        when 0=>hml<=x"005";
                        when 1=>hml<=x"006";
                        when 2=>hml<=x"010";
                        when 3=>hml<=x"010";
                        when 4=>hml<=x"010";
                        when 5=>hml<=x"020";
                        when 6=>hml<=x"010";
                        when 7=>hml<=x"020";
                        when 8=>hml<=x"030";
                        when 9=>hml<=x"050";
                        when 10=>hml<=x"050";
                        when 11=>hml<=x"050";
                        when 12=>hml<=x"060";
                        when 13=>hml<=x"050";
                        when 14=>hml<=x"060";
```

```
when 15=>hml<=x"060";
when 16=>hml<=x"060";
when 17=>hml<=x"100";
when 18=>hml<=x"060";
when 19=>hml<=x"030";
when 20=>hml<=x"050";
when 21=>hml<=x"030";
when 22=>hml<=x"050";
when 23=>hml<=x"060";
when 24=>hml<=x"060";
when 25=>hml<=x"100";
when 26=>hml<=x"060";
when 27=>hml<=x"050";
when 28=>hml<=x"030";
when 29=>hml<=x"050";
when 30=>hml<=x"050";
when 31=>hml<=x"060";
when 32=>hml<=x"030";
when 33=>hml<=x"030";
when 34=>hml<=x"020";
when 35=>hml<=x"030";
when 36=>hml<=x"020";
when 37=>hml<=x"020";
when 38=>hml<=x"020";
when 39=>hml<=x"010";
when 40=>hml<=x"005";
when 41=>hml<=x"006";
when 42=>hml<=x"010";
when 43=>hml<=x"010";
when 44=>hml<=x"010";
when 45=>hml<=x"020";
when 46=>hml<=x"010";
when 47=>hml<=x"020";
when 48=>hml<=x"030";
when 49=>hml<=x"050";
when 50=>hml<=x"050";
when 51=>hml<=x"050";
when 52=>hml<=x"060";
when 53=>hml<=x"050";
when 54=>hml<=x"060";
when 55=>hml<=x"060";
when 56=>hml<=x"060";
when 57=>hml<=x"100";
when 58=>hml<=x"060";
```

```
                    when 59=>hml<=x"030";
                    when 60=>hml<=x"050";
                    when 61=>hml<=x"003";
                    when 62=>hml<=x"005";
                    when 63=>hml<=x"006";
                end case;
            ......
            end if;
    end if;
end process;
end Behavioral;
```

⑦LCD 显示模块：

```
entity lcd is
port(    clk500:in std_logic;                    --输入的 500Hz 的时钟
         start : in   STD_LOGIC;                 --开始信号，高电平有效
         win,lose:in std_logic;                  --win=1 表明游戏胜利，lose=1 表明游戏失败
         highscore:in STD_LOGIC_VECTOR(11 downto 0);   --最高分
         score: in   STD_LOGIC_VECTOR (11 downto 0);   --当前分数
       life: in   STD_LOGIC_VECTOR (7 downto 0);       --当前生命
       LCD_RS : out   STD_LOGIC;
       LCD_RW : out   STD_LOGIC;
       LCD_EN : out   STD_LOGIC;
       data : out   STD_LOGIC_VECTOR (3 downto 0)); --数据信号输出
end lcd;
architecture Behavioral of lcd is
--译码函数完成字符译码
function putc(data:character) return std_logic_vector is
    variable result:std_logic_vector(7 downto 0);
begin
    case data is
        when 'E'=>   result:=x"45";
        when 'I'=>   result:=x"49";
        when 'L'=>   result:=x"4C";
        when 'N'=>   result:=x"4E";
        when 'O'=>   result:=x"4F";
        when 'S'=>   result:=x"53";
        when 'U'=>   result:=x"55";
        when 'W'=>   result:=x"57";
        when 'Y'=>   result:=x"59";
        when 'c'=>   result:=x"63";
        when 'e'=>   result:=x"65";
        when 'f'=>   result:=x"66";
        when 'g'=>   result:=x"67";
        when 'h'=>   result:=x"68";
```

```
            when 'i'=>    result:=x"69";
            when 'l'=>    result:=x"6C";
            when 'o'=>    result:=x"6F";
            when 'r'=>    result:=x"72";
            when 's'=>    result:=x"73";
            when '!'=>    result:=x"21";
            when others => result:=x"20";
        end case;
        return result;
end putc;
--译码函数完成 0~9 的 ASCII 译码
function putn(num:in std_logic_vector(3 downto 0)) return std_logic_vector is
        variable fig:std_logic_vector(7 downto 0);
begin
        case num is
            when "0000"=>   fig:="00110000";
            when "0001"=>   fig:="00110001";
            when "0010"=>   fig:="00110010";
            when "0011"=>   fig:="00110011";
            when "0100"=>   fig:="00110100";
            when "0101"=>   fig:="00110101";
            when "0110"=>   fig:="00110110";
            when "0111"=>   fig:="00110111";
            when "1000"=>   fig:="00111000";
            when "1001"=>   fig:="00111001";
            when others=>   fig:="00100000";
        end case;
        return fig;
end putn;
type istate is(
                        write_instr,        --写命令字
                        write_dataup,       --写 LCD 高四位
                        write_datadown,     --写 LCD 低四位
                        set_addrup,             --设定位置高四位
                        set_addrdown,           --设定位置低四位
                        ret_homeup,             --光标归位指令高四位
                        ret_homedown);          --光标归位指令低四位
signal state:istate;
signal cnt:integer range 0 to 29:=0;
begin
LCD_RW<='0';
LCD_EN<=clk500;
process(clk500,start)               --LCD 显示控制
begin
```

```
if start='0' then
        state<=write_instr;
        LCD_RS<='0';
        cnt<=0;
elsif rising_edge(clk500) then
            case state is
            when write_instr=>--写命令字到 LCD 控制器，初始化液晶，并设置显示格式
                LCD_RS<='0';
                case cnt is
                    --写命令字 0x33
                    when 0=> data<="0011";
                    when 1=> data<="0011";
                    --写命令字 0x32
                    when 2=> data<="0011";
                    when 3=> data<="0010";
                    --0x28：功能设置，4 位接口模式，2 行显示
                    when 4=> data<="0010";
                    when 5=> data<="1000";
                    ---0x06：模式设定，输入后光标右移
                    when 6=> data<="0000";
                    when 7=> data<="0110";
                    --0x0c：显示设定，显示功能开
                    --不显示光标，光标不闪烁
                    when 8=> data<="0000";
                    when 9=> data<="1100";
                    --0x01：清屏
                    when 10=> data<="0000";
                    when 11=> data<="0001";
                    when others=> data<="0000";
                end case;
                if cnt>=11 then
                    cnt<=0;
                    state<=write_dataup;
                else
                    cnt<=cnt+1;
                    state<=write_instr;
                end if;
            when write_dataup=>
                LCD_RS<='1';
                --当 win=‘1’时第一行显示 YOU WIN!
                --第二行显示 high   xxx
                if win='1' then
                    case cnt is
                    when 0=> data<=putc('Y')(7 downto 4);
```

```
        when 1=> data<=putc('O')(7 downto 4);
        when 2=> data<=putc('U')(7 downto 4);
        when 3=> data<=putc(' ')(7 downto 4);
        when 4=> data<=putc('W')(7 downto 4);
        when 5=> data<=putc('I')(7 downto 4);
        when 6=> data<=putc('N')(7 downto 4);
        when 7=> data<=putc('!')(7 downto 4);
        when 8=> data<=putc(' ')(7 downto 4);
        when 9=> data<=putc(' ')(7 downto 4);
        when 10=> data<=putc(' ')(7 downto 4);
        when 11=> data<=putc(' ')(7 downto 4);
        when 12=> data<=putc(' ')(7 downto 4);
        when 13=> data<=putc(' ')(7 downto 4);
        when 14=> data<=putc(' ')(7 downto 4);
        --转到第二行显示
        when 15=> data<=putc('h')(7 downto 4);
        when 16=> data<=putc('i')(7 downto 4);
        when 17=> data<=putc('g')(7 downto 4);
        when 18=> data<=putc('h')(7 downto 4);
        when 19=> data<=putc(' ')(7 downto 4);
        when 20=> data<=putc(' ')(7 downto 4);
        when 21=> data<=putc(' ')(7 downto 4);
        when 22=> data<=putn(highscore(11 downto 8))(7 downto 4);
        when 23=> data<=putn(highscore(7 downto 4))(7 downto 4);
        when 24=> data<=putn(highscore(3 downto 0))(7 downto 4);
        when 25=> data<=putc(' ')(7 downto 4);
        when 26=> data<=putc(' ')(7 downto 4);
        when 27=> data<=putc(' ')(7 downto 4);
        when 28=> data<=putc(' ')(7 downto 4);
        when 29=> data<=putc(' ')(7 downto 4);
        when others=> data<="0000";
        end case;
        state<=write_datadown;
    --当 lose=‘1’时第一行显示 YOU LOSE!
    --第二行显示 high   xxx
    elsif lose='1' then
        case cnt is
        when 0=> data<=putc('Y')(7 downto 4);
        when 1=> data<=putc('O')(7 downto 4);
        when 2=> data<=putc('U')(7 downto 4);
        when 3=> data<=putc(' ')(7 downto 4);
        when 4=> data<=putc('L')(7 downto 4);
        when 5=> data<=putc('O')(7 downto 4);
        when 6=> data<=putc('S')(7 downto 4);
```

```
        when 7=> data<=putc('E')(7 downto 4);
        when 8=> data<=putc('!')(7 downto 4);
        when 9=> data<=putc(' ')(7 downto 4);
        when 10=> data<=putc(' ')(7 downto 4);
        when 11=> data<=putc(' ')(7 downto 4);
        when 12=> data<=putc(' ')(7 downto 4);
        when 13=> data<=putc(' ')(7 downto 4);
        when 14=> data<=putc(' ')(7 downto 4);
        --转到第二行显示
        when 15=> data<=putc('h')(7 downto 4);
        when 16=> data<=putc('i')(7 downto 4);
        when 17=> data<=putc('g')(7 downto 4);
        when 18=> data<=putc('h')(7 downto 4);
        when 19=> data<=putc(' ')(7 downto 4);
        when 20=> data<=putc(' ')(7 downto 4);
        when 21=> data<=putc(' ')(7 downto 4);
        when 22=> data<=putn(highscore(11 downto 8))(7 downto 4);
        when 23=> data<=putn(highscore(7 downto 4))(7 downto 4);
        when 24=> data<=putn(highscore(3 downto 0))(7 downto 4);
        when 25=> data<=putc(' ')(7 downto 4);
        when 26=> data<=putc(' ')(7 downto 4);
        when 27=> data<=putc(' ')(7 downto 4);
        when 28=> data<=putc(' ')(7 downto 4);
        when 29=> data<=putc(' ')(7 downto 4);
        when others=> data<="0000";
        end case;
        state<=write_datadown;
    --其他情况下第一行显示 life score high
    --第二行显示  xx    xxx   xxx 分别代表命数、分数和历史最高分
    else
        case cnt is
            when 0=> data<=putc('l')(7 downto 4);
            when 1=> data<=putc('i')(7 downto 4);
            when 2=> data<=putc('f')(7 downto 4);
            when 3=> data<=putc('e')(7 downto 4);
            when 4=> data<=putc(' ')(7 downto 4);
            when 5=> data<=putc('s')(7 downto 4);
            when 6=> data<=putc('c')(7 downto 4);
            when 7=> data<=putc('o')(7 downto 4);
            when 8=> data<=putc('r')(7 downto 4);
            when 9=> data<=putc('e')(7 downto 4);
            when 10=> data<=putc(' ')(7 downto 4);
            when 11=> data<=putc('h')(7 downto 4);
            when 12=> data<=putc('i')(7 downto 4);
```

```
                when 13=> data<=putc('g')(7 downto 4);
                when 14=> data<=putc('h')(7 downto 4);
                --转到第二行显示
                when 15=> data<=putc(' ')(7 downto 4);
                when 16=> data<=putc(' ')(7 downto 4);
                when 17=> data<=putn(life(7 downto 4))(7 downto 4);
                when 18=> data<=putn(life(3 downto 0))(7 downto 4);
                when 19=> data<=putc(' ')(7 downto 4);
                when 20=> data<=putc(' ')(7 downto 4);
                when 21=> data<=putc(' ')(7 downto 4);
                when 22=> data<=putn(score(11 downto 8))(7 downto 4);
                when 23=> data<=putn(score(7 downto 4))(7 downto 4);
                when 24=> data<=putn(score(3 downto 0))(7 downto 4);
                when 25=> data<=putc(' ')(7 downto 4);
                when 26=> data<=putc(' ')(7 downto 4);
                when 27=> data<=putn(highscore(11 downto 8))(7 downto 4);
                when 28=> data<=putn(highscore(7 downto 4))(7 downto 4);
                when 29=> data<=putn(highscore(3 downto 0))(7 downto 4);
                when others=>data<="0000";
            end case;
            state<=write_datadown;
        end if;
    when write_datadown=>
        --当 win='1'时第一行显示 YOU WIN!
        --第二行显示 high   xxx
        if win='1' then
            case cnt is
            when 0=> data<=putc('Y')(3 downto 0);
            when 1=> data<=putc('O')(3 downto 0);
            when 2=> data<=putc('U')(3 downto 0);
            when 3=> data<=putc(' ')(3 downto 0);
            when 4=> data<=putc('W')(3 downto 0);
            when 5=> data<=putc('I')(3 downto 0);
            when 6=> data<=putc('N')(3 downto 0);
            when 7=> data<=putc('!')(3 downto 0);
            when 8=> data<=putc(' ')(3 downto 0);
            when 9=> data<=putc(' ')(3 downto 0);
            when 10=> data<=putc(' ')(3 downto 0);
            when 11=> data<=putc(' ')(3 downto 0);
            when 12=> data<=putc(' ')(3 downto 0);
            when 13=> data<=putc(' ')(3 downto 0);
            when 14=> data<=putc(' ')(3 downto 0);
            --转到第二行显示
            when 15=> data<=putc('h')(3 downto 0);
```

```
        when 16=> data<=putc('i')(3 downto 0);
        when 17=> data<=putc('g')(3 downto 0);
        when 18=> data<=putc('h')(3 downto 0);
        when 19=> data<=putc(' ')(3 downto 0);
        when 20=> data<=putc(' ')(3 downto 0);
        when 21=> data<=putc(' ')(3 downto 0);
        when 22=> data<=putn(highscore(11 downto 8))(3 downto 0);
        when 23=> data<=putn(highscore(7 downto 4))(3 downto 0);
        when 24=> data<=putn(highscore(3 downto 0))(3 downto 0);
        when 25=> data<=putc(' ')(3 downto 0);
        when 26=> data<=putc(' ')(3 downto 0);
        when 27=> data<=putc(' ')(3 downto 0);
        when 28=> data<=putc(' ')(3 downto 0);
        when 29=> data<=putc(' ')(3 downto 0);
        when others=> data<="0000";
        end case;
        if cnt=14 then
            cnt<=15;
            state<=set_addrup;
        elsif cnt=29 then
            cnt<=0;
            state<=ret_homeup;
        else
            cnt<=cnt+1;
            state<=write_dataup;
        end if;
    --当 lose=‘1’时第一行显示 YOU LOSE!
    --第二行显示 high   xxx
    elsif lose='1' then
        case cnt is
        when 0=> data<=putc('Y')(3 downto 0);
        when 1=> data<=putc('O')(3 downto 0);
        when 2=> data<=putc('U')(3 downto 0);
        when 3=> data<=putc(' ')(3 downto 0);
        when 4=> data<=putc('L')(3 downto 0);
        when 5=> data<=putc('O')(3 downto 0);
        when 6=> data<=putc('S')(3 downto 0);
        when 7=> data<=putc('E')(3 downto 0);
        when 8=> data<=putc('!')(3 downto 0);
        when 9=> data<=putc(' ')(3 downto 0);
        when 10=> data<=putc(' ')(3 downto 0);
        when 11=> data<=putc(' ')(3 downto 0);
        when 12=> data<=putc(' ')(3 downto 0);
        when 13=> data<=putc(' ')(3 downto 0);
```

```
        when 14=> data<=putc(' ')(3 downto 0);
        --转到第二行显示
        when 15=> data<=putc('h')(3 downto 0);
        when 16=> data<=putc('i')(3 downto 0);
        when 17=> data<=putc('g')(3 downto 0);
        when 18=> data<=putc('h')(3 downto 0);
        when 19=> data<=putc(' ')(3 downto 0);
        when 20=> data<=putc(' ')(3 downto 0);
        when 21=> data<=putc(' ')(3 downto 0);
        when 22=> data<=putn(highscore(11 downto 8))(3 downto 0);
        when 23=> data<=putn(highscore(7 downto 4))(3 downto 0);
        when 24=> data<=putn(highscore(3 downto 0))(3 downto 0);
        when 25=> data<=putc(' ')(3 downto 0);
        when 26=> data<=putc(' ')(3 downto 0);
        when 27=> data<=putc(' ')(3 downto 0);
        when 28=> data<=putc(' ')(3 downto 0);
        when 29=> data<=putc(' ')(3 downto 0);
        when others=> data<="0000";
        end case;
        if cnt=14 then
            cnt<=15;
            state<=set_addrup;
        elsif cnt=29 then
            cnt<=0;
            state<=ret_homeup;
        else
            cnt<=cnt+1;
            state<=write_dataup;
        end if;
    --其他情况下第一行显示 life score high
    --第二行显示   xx    xxx   xxx 分别代表命数、分数和历史最高分
    else
        case cnt is
            when 0=> data<=putc('l')(3 downto 0);
            when 1=> data<=putc('i')(3 downto 0);
            when 2=> data<=putc('f')(3 downto 0);
            when 3=> data<=putc('e')(3 downto 0);
            when 4=> data<=putc(' ')(3 downto 0);
            when 5=> data<=putc('s')(3 downto 0);
            when 6=> data<=putc('c')(3 downto 0);
            when 7=> data<=putc('o')(3 downto 0);
            when 8=> data<=putc('r')(3 downto 0);
            when 9=> data<=putc('e')(3 downto 0);
            when 10=> data<=putc(' ')(3 downto 0);
```

```
            when 11=> data<=putc('h')(3 downto 0);
            when 12=> data<=putc('i')(3 downto 0);
            when 13=> data<=putc('g')(3 downto 0);
            when 14=> data<=putc('h')(3 downto 0);
            --转到第二行显示
            when 15=> data<=putc(' ')(3 downto 0);
            when 16=> data<=putc(' ')(3 downto 0);
            when 17=> data<=putn(life(7 downto 4))(3 downto 0);
            when 18=> data<=putn(life(3 downto 0))(3 downto 0);
            when 19=> data<=putc(' ')(3 downto 0);
            when 20=> data<=putc(' ')(3 downto 0);
            when 21=> data<=putc(' ')(3 downto 0);
            when 22=> data<=putn(score(11 downto 8))(3 downto 0);
            when 23=> data<=putn(score(7 downto 4))(3 downto 0);
            when 24=> data<=putn(score(3 downto 0))(3 downto 0);
            when 25=> data<=putc(' ')(3 downto 0);
            when 26=> data<=putc(' ')(3 downto 0);
            when 27=> data<=putn(highscore(11 downto 8))(3 downto 0);
            when 28=> data<=putn(highscore(7 downto 4))(3 downto 0);
            when 29=> data<=putn(highscore(3 downto 0))(3 downto 0);
            when others=>data<="0000";
        end case;
        if cnt=14 then
            cnt<=15;
            state<=set_addrup;
        elsif cnt=29 then
            cnt<=0;
            state<=ret_homeup;
        else
            cnt<=cnt+1;
            state<=write_dataup;
        end if;
    end if;
    --设定显示地址
    when set_addrup=>                    --设定地址为 40H（第二行第 1 列）
        LCD_RS<='0';
        data<="1100";
        state<=set_addrdown;
    when set_addrdown=>
        data<="0000";
        state<=write_dataup;
    --光标归位
    when ret_homeup=>
        LCD_RS<='0';
```

```
                    data<="0000";
                    state<=ret_homedown;
                when ret_homedown=>
                    data<="0010";
                    state<=write_dataup;

                when others=>
                    state<=write_instr;
                end case;
            end if;
    end process;
end Behavioral;
```

⑧VGA 模块

```
entity vga is
port(
      start:in std_logic;            --开始信号，高电平有效
       en:in STD_LOGIC;                    --暂停信号，en=0 时暂停
      hs : out std_logic;                                --屏幕行扫描
      vs : out std_logic;                                --屏幕场扫描
      red : out std_logic;                               --显示红色输出
      green : out std_logic;                             --显示绿色输出
      blue : out std_logic;                               --显示蓝色输出
       led : in   STD_LOGIC_VECTOR (2 downto 0);--led 为产生的三位随机数
       key : in   STD_LOGIC_VECTOR (2 downto 0);      --key 为编码后得到的编码值

       clk50m : in   STD_LOGIC;
       win,lose: in std_logic

);
end vga;

architecture Behavioral of vga is
signal hloc : std_logic_vector(9 downto 0);                    --行位置
signal vloc : std_logic_vector(9 downto 0);                    --场位置
signal rgbx,rgby: std_logic_vector(2 downto 0);                 --用于暂存屏幕对应的颜
色
signal sysclk : std_logic;                                      --屏幕扫描时钟：20MHZ
    signal hsyncb : std_logic;                                      --行同步信号
    signal vsyncb : std_logic;                                      --场同步信号
  signal enable : std_logic;                                       --屏幕显示使能信号，
高电平有效
    signal rgbp,rgb: std_logic_vector(2 downto 0);                  --用于暂存屏幕对应的
颜色
    signal hcnt: std_logic_vector(9 downto 0);                      --用于行计数
```

```
    signal vcnt: std_logic_vector(9 downto 0);                    --用于场计数

-----------------------------------------------------constant-------------------------------
-----------------
   constant H_PIXELS: INTEGER:=255;                                           --行显示
点数
   constant H_FRONT:    INTEGER:=165;                                          --行前
消隐点数
   constant H_BACK:     INTEGER:=180;                                          --行后
消隐点数
   constant H_SYNCTIME:INTEGER:=100;                                           --行同
步点数
   constant H_PERIOD: INTEGER:= H_SYNCTIME + H_PIXELS + H_FRONT + H_BACK;--行周
期计数值
   constant V_LINES: INTEGER:=255;                                            --场显示
点数
   constant V_FRONT: INTEGER:=115;                                             --场前消
隐点数
   constant V_BACK: INTEGER:=130;                                              --场后消
隐点数
   constant V_SYNCTIME: INTEGER:=2;                                            --场同步
点数
   constant V_PERIOD: INTEGER:= V_SYNCTIME + V_LINES + V_FRONT + V_BACK; --场周
期计数值

-----------------------------------------------------------------------------------------------

begin

process(start,en,hloc,vloc,led,key,lose,win)
begin
---------------------------------------------------开机画面-------------
if((en='0') and (start='1'))then
   if((vloc(7 downto 1)>=0 and vloc(7 downto 1)<=9) or
      (vloc(7 downto 1)>=118 and    vloc(7 downto 1)<=127)) then
        if(hloc(7 downto 1)>=0 and hloc(7 downto 1)<=127)then
              rgbx<="100";rgby<="010";end if; --yellow
   elsif((vloc(7 downto 1)>=10 and vloc(7 downto 1)<=14) or
         (vloc(7 downto 1)>=80 and vloc(7 downto 1)<=91) or
         (vloc(7 downto 1)>=108 and    vloc(7 downto 1)<=117))then
        if(hloc(7 downto 1)>=10 and hloc(7 downto 1)<=117)then
```

```
            rgbx<="100";rgby<="000";
        else rgbx<="100";rgby<="010";
        end if;
---the mouse's face
    elsif(vloc(7 downto 1)=15)then---line 15
        if((hloc(7 downto 1)>=33 and hloc(7 downto 1)<=41)
          or(hloc(7 downto 1)>=86 and hloc(7 downto 1)<=93))then
            rgbx<="000";rgby<="000";---black,mouse
        elsif((hloc(7 downto 1)>=10 and hloc(7 downto 1)<=32) or
            (hloc(7 downto 1)>=42 and hloc(7 downto 1)<=85) or
            (hloc(7 downto 1)>=94 and hloc(7 downto 1)<=117))then
            rgbx<="100";rgby<="000";---red,back scene
        else
            rgbx<="100";rgby<="010";---yellow,edge
        end if;
    elsif(vloc(7 downto 1)=16)then---line 16
        if((hloc(7 downto 1)>=31 and hloc(7 downto 1)<=43)
          or(hloc(7 downto 1)>=84 and hloc(7 downto 1)<=95))then
            rgbx<="000";rgby<="000";---black,mouse
        elsif((hloc(7 downto 1)>=10 and hloc(7 downto 1)<=30)
            or(hloc(7 downto 1)>=44 and hloc(7 downto 1)<=83)
            or(hloc(7 downto 1)>=96 and hloc(7 downto 1)<=117))then
            rgbx<="100";rgby<="000";---red
        else
            rgbx<="100";rgby<="010";---yellow,edge
        end if;
    elsif(vloc(7 downto 1)=17)then---line 17
        if((hloc(7 downto 1)>=30 and hloc(7 downto 1)<=44)
          or(hloc(7 downto 1)>=83 and hloc(7 downto 1)<=96))then
            rgbx<="000";rgby<="000";---black
        elsif((hloc(7 downto 1)>=10 and hloc(7 downto 1)<=29)
            or(hloc(7 downto 1)>=45 and hloc(7 downto 1)<=82)
            or(hloc(7 downto 1)>=97 and hloc(7 downto 1)<=117))then
            rgbx<="100";rgby<="000";---red
        else
            rgbx<="100";rgby<="010";---yellow,edge
        end if;
    elsif(vloc(7 downto 1)=18)then---line 18
        if((hloc(7 downto 1)>=29 and hloc(7 downto 1)<=45)
          or(hloc(7 downto 1)>=82 and hloc(7 downto 1)<=97))then
            rgbx<="000";rgby<="000";
        elsif((hloc(7 downto 1)>=0 and hloc(7 downto 1)<=9)
            or(hloc(7 downto 1)>=118 and hloc(7 downto 1)<=127))then
            rgbx<="100";rgby<="010";
```

```
    else
        rgbx<="100";rgby<="000";
    end if;
elsif(vloc(7 downto 1)=19)then---19
    if((hloc(7 downto 1)>=28 and hloc(7 downto 1)<=46)
      or(hloc(7 downto 1)>=81 and hloc(7 downto 1)<=98))then
        rgbx<="000";rgby<="000";
    elsif((hloc(7 downto 1)>=0 and hloc(7 downto 1)<=9)
        or(hloc(7 downto 1)>=118 and hloc(7 downto 1)<=127))then
        rgbx<="100";rgby<="010";
    else
        rgbx<="100";rgby<="000";
    end if;
elsif(vloc(7 downto 1)=20)then---20
    if((hloc(7 downto 1)>=27 and hloc(7 downto 1)<=47)
      or(hloc(7 downto 1)>=80 and hloc(7 downto 1)<=99))then
      rgbx<="000";rgby<="000";
    elsif((hloc(7 downto 1)>=0 and hloc(7 downto 1)<=9)
      or(hloc(7 downto 1)>=118 and hloc(7 downto 1)<=127))then
      rgbx<="100";rgby<="010";
    else
      rgbx<="100";rgby<="000";
    end if;
elsif(vloc(7 downto 1)=21)then---21
    if((hloc(7 downto 1)>=26 and hloc(7 downto 1)<=48)
      or(hloc(7 downto 1)>=79 and hloc(7 downto 1)<=100))then
       rgbx<="000";rgby<="000";
    elsif((hloc(7 downto 1)>=0 and hloc(7 downto 1)<=9)
      or(hloc(7 downto 1)>=118 and hloc(7 downto 1)<=127))then
       rgbx<="100";rgby<="010";
    else
      rgbx<="100";rgby<="000";
    end if;
elsif(vloc(7 downto 1)=22)then---22
    if((hloc(7 downto 1)>=25 and hloc(7 downto 1)<=49)
      or(hloc(7 downto 1)>=78 and hloc(7 downto 1)<=101))then
        rgbx<="000";rgby<="000";
    elsif((hloc(7 downto 1)>=0 and hloc(7 downto 1)<=9)
        or(hloc(7 downto 1)>=118 and hloc(7 downto 1)<=127))then
        rgbx<="100";rgby<="010";
    else
        rgbx<="100";rgby<="000";
    end if;
elsif(vloc(7 downto 1)=23 or vloc(7 downto 1)=24)then---24,23
```

```
  if((hloc(7 downto 1)>=24 and hloc(7 downto 1)<=50)
  or(hloc(7 downto 1)>=77 and hloc(7 downto 1)<=102))then
       rgbx<="000";rgby<="000";
  elsif((hloc(7 downto 1)>=0 and hloc(7 downto 1)<=9)
       or(hloc(7 downto 1)>=118 and hloc(7 downto 1)<=127))then
       rgbx<="100";rgby<="010";
  else
       rgbx<="100";rgby<="000";
  end if;
 elsif(vloc(7 downto 1)=25 or vloc(7 downto 1)=26)then--25,26
  if((hloc(7 downto 1)>=23 and hloc(7 downto 1)<=51)
    or(hloc(7 downto 1)>=76 and hloc(7 downto 1)<=103))then
       rgbx<="000";rgby<="000";
    elsif((hloc(7 downto 1)>=0 and hloc(7 downto 1)<=9)
       or(hloc(7 downto 1)>=118 and hloc(7 downto 1)<=127))then
       rgbx<="100";rgby<="010";
    else
       rgbx<="100";rgby<="000";
    end if;
 elsif(vloc(7 downto 1)=27 or vloc(7 downto 1)=28 or vloc(7 downto
1)=29)then---27,28,29
  if((hloc(7 downto 1)>=22 and hloc(7 downto 1)<=52)
    or(hloc(7 downto 1)>=75 and hloc(7 downto 1)<=104))then
    rgbx<="000";rgby<="000";
       elsif((hloc(7 downto 1)>=0 and hloc(7 downto 1)<=9)
       or(hloc(7 downto 1)>=118 and hloc(7 downto 1)<=127))then
    rgbx<="100";rgby<="010";
    else
    rgbx<="100";rgby<="000";
    end if;
 elsif(vloc(7 downto 1)=30)then---30
  if((hloc(7 downto 1)>=22 and hloc(7 downto 1)<=51)
    or(hloc(7 downto 1)>59 and hloc(7 downto 1)<=68)
    or(hloc(7 downto 1)>=76 and hloc(7 downto 1)<=104))then
    rgbx<="000";rgby<="000";
       elsif((hloc(7 downto 1)>=0 and hloc(7 downto 1)<=9)
       or(hloc(7 downto 1)>=118 and hloc(7 downto 1)<=127))then
    rgbx<="100";rgby<="010";
    else
    rgbx<="100";rgby<="000";
    end if;
 elsif(vloc(7 downto 1)=31)then---31
  if((hloc(7 downto 1)>=22 and hloc(7 downto 1)<=51)
    or(hloc(7 downto 1)>=57 and hloc(7 downto 1)<=70)
```

```
      or(hloc(7 downto 1)>=76 and hloc(7 downto 1)<=104))then
      rgbx<="000";rgby<="000";
          elsif((hloc(7 downto 1)>=0 and hloc(7 downto 1)<=9)
          or(hloc(7 downto 1)>=118 and hloc(7 downto 1)<=127))then
      rgbx<="100";rgby<="010";
      else
      rgbx<="100";rgby<="000";
      end if;

elsif(vloc(7 downto 1)=32)then---32
    if((hloc(7 downto 1)>=22 and hloc(7 downto 1)<=50)
      or(hloc(7 downto 1)>=56 and hloc(7 downto 1)<=71)
      or(hloc(7 downto 1)>=77 and hloc(7 downto 1)<=104))then
      rgbx<="000";rgby<="000";
          elsif((hloc(7 downto 1)>=0 and hloc(7 downto 1)<=9)
          or(hloc(7 downto 1)>=118 and hloc(7 downto 1)<=127))then
      rgbx<="100";rgby<="010";
      else
      rgbx<="100";rgby<="000";
      end if;
elsif(vloc(7 downto 1)=33)then---33
    if((hloc(7 downto 1)>=23 and hloc(7 downto 1)<=49)
      or(hloc(7 downto 1)>=55 and hloc(7 downto 1)<=72)
      or(hloc(7 downto 1)>=78 and hloc(7 downto 1)<=103))then
      rgbx<="000";rgby<="000";
          elsif((hloc(7 downto 1)>=0 and hloc(7 downto 1)<=9)
          or(hloc(7 downto 1)>=118 and hloc(7 downto 1)<=127))then
      rgbx<="100";rgby<="010";
      else
      rgbx<="100";rgby<="000";
      end if;

elsif(vloc(7 downto 1)=34)then---34
    if(hloc(7 downto 1)>=23 and hloc(7 downto 1)<=103)then
    rgbx<="000";rgby<="000";
          elsif((hloc(7 downto 1)>=0 and hloc(7 downto 1)<=9)
          or(hloc(7 downto 1)>=118 and hloc(7 downto 1)<=127))then
      rgbx<="100";rgby<="010";
      else
      rgbx<="100";rgby<="000";
      end if;

elsif(vloc(7 downto 1)=35 or vloc(7 downto 1)=36)then---35,36
    if(hloc(7 downto 1)>=24 and hloc(7 downto 1)<=102)then
```

```
    rgbx<="000";rgby<="000";
         elsif((hloc(7 downto 1)>=0 and hloc(7 downto 1)<=9)
         or(hloc(7 downto 1)>=118 and hloc(7 downto 1)<=127))then
      rgbx<="100";rgby<="010";
      else
      rgbx<="100";rgby<="000";
      end if;
elsif(vloc(7 downto 1)=37)then---37
    if(hloc(7 downto 1)>=25 and hloc(7 downto 1)<=101)then
    rgbx<="000";rgby<="000";
         elsif((hloc(7 downto 1)>=0 and hloc(7 downto 1)<=9)
         or(hloc(7 downto 1)>=118 and hloc(7 downto 1)<=127))then
      rgbx<="100";rgby<="010";
      else
      rgbx<="100";rgby<="000";
      end if;

elsif(vloc(7 downto 1)=38)then---38
    if(hloc(7 downto 1)>=26 and hloc(7 downto 1)<=100)then
    rgbx<="000";rgby<="000";
         elsif((hloc(7 downto 1)>=0 and hloc(7 downto 1)<=9)
         or(hloc(7 downto 1)>=118 and hloc(7 downto 1)<=127))then
      rgbx<="100";rgby<="010";
      else
      rgbx<="100";rgby<="000";
      end if;
elsif(vloc(7 downto 1)=39)then--39
    if(hloc(7 downto 1)>=27 and hloc(7 downto 1)<=99)then
    rgbx<="000";rgby<="000";
         elsif((hloc(7 downto 1)>=0 and hloc(7 downto 1)<=9)
         or(hloc(7 downto 1)>=118 and hloc(7 downto 1)<=127))then
      rgbx<="100";rgby<="010";
      else
      rgbx<="100";rgby<="000";
      end if;

elsif(vloc(7 downto 1)=40)then---40
    if(hloc(7 downto 1)>=28 and hloc(7 downto 1)<=98)then
      rgbx<="000";rgby<="000";
         elsif((hloc(7 downto 1)>=0 and hloc(7 downto 1)<=9)
         or(hloc(7 downto 1)>=118 and hloc(7 downto 1)<=127))then
      rgbx<="100";rgby<="010";
      else
      rgbx<="100";rgby<="000";
```

```
    end if;

elsif(vloc(7 downto 1)=41)then
   if(hloc(7 downto 1)>=29 and hloc(7 downto 1)<=97)then
   rgbx<="000";rgby<="000";
        elsif((hloc(7 downto 1)>=0 and hloc(7 downto 1)<=9)
        or(hloc(7 downto 1)>=118 and hloc(7 downto 1)<=127))then
    rgbx<="100";rgby<="010";
    else
    rgbx<="100";rgby<="000";
    end if;

elsif(vloc(7 downto 1)=42)then
   if(hloc(7 downto 1)>=30 and hloc(7 downto 1)<=96)then
   rgbx<="000";rgby<="000";
        elsif((hloc(7 downto 1)>=0 and hloc(7 downto 1)<=9)
        or(hloc(7 downto 1)>=118 and hloc(7 downto 1)<=127))then
    rgbx<="100";rgby<="010";
    else
    rgbx<="100";rgby<="000";
    end if;

elsif(vloc(7 downto 1)=43)then
   if(hloc(7 downto 1)>=31 and hloc(7 downto 1)<=95)then
   rgbx<="000";rgby<="000";
        elsif((hloc(7 downto 1)>=0 and hloc(7 downto 1)<=9)
        or(hloc(7 downto 1)>=118 and hloc(7 downto 1)<=127))then
    rgbx<="100";rgby<="010";
    else
    rgbx<="100";rgby<="000";
    end if;

elsif(vloc(7 downto 1)=44)then
   if((hloc(7 downto 1)>=33 and hloc(7 downto 1)<=39)
    or (hloc(7 downto 1)>=43 and hloc(7 downto 1)<=84)
    or (hloc(7 downto 1)>=88 and hloc(7 downto 1)<=94))then
    rgbx<="000";rgby<="000";
        elsif((hloc(7 downto 1)>=0 and hloc(7 downto 1)<=9)
        or(hloc(7 downto 1)>=118 and hloc(7 downto 1)<=127))then
    rgbx<="100";rgby<="010";
    else
    rgbx<="100";rgby<="000";
    end if;
```

```
elsif(vloc(7 downto 1)=45)then
   if(hloc(7 downto 1)>=42 and hloc(7 downto 1)<=85)then
      rgbx<="000";rgby<="000";
         elsif((hloc(7 downto 1)>=0 and hloc(7 downto 1)<=9)
         or(hloc(7 downto 1)>=118 and hloc(7 downto 1)<=127))then
     rgbx<="100";rgby<="010";
     else
     rgbx<="100";rgby<="000";
     end if;

elsif(vloc(7 downto 1)=46 or vloc(7 downto 1)=47)then
   if(hloc(7 downto 1)>=41 and hloc(7 downto 1)<=86)then
   rgbx<="000";rgby<="000";
         elsif((hloc(7 downto 1)>=0 and hloc(7 downto 1)<=9)
         or(hloc(7 downto 1)>=118 and hloc(7 downto 1)<=127))then
     rgbx<="100";rgby<="010";
     else
     rgbx<="100";rgby<="000";
     end if;

elsif(vloc(7 downto 1)=48 or vloc(7 downto 1)=49 or vloc(7 downto 1)=50)then
   if(hloc(7 downto 1)>=40 and hloc(7 downto 1)<=87)then
   rgbx<="000";rgby<="000";
         elsif((hloc(7 downto 1)>=0 and hloc(7 downto 1)<=9)
         or(hloc(7 downto 1)>=118 and hloc(7 downto 1)<=127))then
     rgbx<="100";rgby<="010";
     else
     rgbx<="100";rgby<="000";
     end if;

elsif(vloc(7 downto 1)>=51 and vloc(7 downto 1)<=60)then
   if(hloc(7 downto 1)>=39 and hloc(7 downto 1)<=88)then
   rgbx<="000";rgby<="000";
         elsif((hloc(7 downto 1)>=0 and hloc(7 downto 1)<=9)
         or(hloc(7 downto 1)>=118 and hloc(7 downto 1)<=127))then
     rgbx<="100";rgby<="010";
     else
     rgbx<="100";rgby<="000";
     end if;

  ELSIF(Vloc(7 downto 1)>=61 AND Vloc(7 downto 1)<=63)THEN
      IF((HLOC(7 DOWNTO 1)>=0 AND HLOC(7 DOWNTO 1)<=9)OR(HLOC(7 DOWNTO
1)>=118    -----RED EDGE
         AND HLOC(7 DOWNTO 1)<=127) )
```

```
      THEN
          RGBX<="100";RGBY<="010";
      ELSIF(HLOC(7 DOWNTO 1)>=40 AND HLOC(7 DOWNTO 1)<=87) then
          RGBX<="000";RGBY<="000";
       ELSE
          RGBX<="100";RGBY<="000";
       END IF;
ELSIF(Vloc(7 downto 1)>=64 AND Vloc(7 downto 1)<=65)THEN
      IF((HLOC(7 DOWNTO 1)>=0 AND HLOC(7 DOWNTO 1)<=9)OR(HLOC(7 DOWNTO
1)>=118   -----RED EDGE
         AND HLOC(7 DOWNTO 1)<=127)
       )THEN
          RGBX<="100";RGBY<="010";
      ELSIF(HLOC(7 DOWNTO 1)>=41 AND HLOC(7 DOWNTO 1)<=86)then
          RGBX<="000";RGBY<="000";
       ELSE
          RGBX<="100";RGBY<="000";
       END IF;
ELSIF(Vloc(7 downto 1)>=66 AND Vloc(7 downto 1)<=67)THEN
      IF((HLOC(7 DOWNTO 1)>=0 AND HLOC(7 DOWNTO 1)<=9)OR(HLOC(7 DOWNTO
1)>=118   -----RED EDGE
         AND HLOC(7 DOWNTO 1)<=127)
       )THEN
          RGBX<="100";RGBY<="010";
      ELSIF(HLOC(7 DOWNTO 1)>=42 AND HLOC(7 DOWNTO 1)<=85)then
          RGBX<="000";RGBY<="000";
       ELSE
          RGBX<="100";RGBY<="000";
       END IF;
ELSIF(Vloc(7 downto 1)>=68 AND Vloc(7 downto 1)<=69)THEN
      IF((HLOC(7 DOWNTO 1)>=0 AND HLOC(7 DOWNTO 1)<=9)OR(HLOC(7 DOWNTO
1)>=118   -----RED EDGE
         AND HLOC(7 DOWNTO 1)<=127)
       )THEN
          RGBX<="100";RGBY<="010";
      ELSIF(HLOC(7 DOWNTO 1)>=43 AND HLOC(7 DOWNTO 1)<=84)then
          RGBX<="000";RGBY<="000";
       ELSE
          RGBX<="100";RGBY<="000";
       END IF;
ELSIF(Vloc(7 downto 1)=70 )THEN
      IF((HLOC(7 DOWNTO 1)>=0 AND HLOC(7 DOWNTO 1)<=9)OR(HLOC(7 DOWNTO
1)>=118   -----RED EDGE
         AND HLOC(7 DOWNTO 1)<=127)
```

```
        )THEN
           RGBX<="100";RGBY<="010";
      ELSIF(HLOC(7 DOWNTO 1)>=44 AND HLOC(7 DOWNTO 1)<=83)then
           RGBX<="000";RGBY<="000";
        ELSE
           RGBX<="100";RGBY<="000";
        END IF;
ELSIF(Vloc(7 downto 1)=71 )THEN
     IF((HLOC(7 DOWNTO 1)>=0 AND HLOC(7 DOWNTO 1)<=9)OR(HLOC(7 DOWNTO
1)>=118   -----RED EDGE
         AND HLOC(7 DOWNTO 1)<=127)
        )THEN
           RGBX<="100";RGBY<="010";
      ELSIF(HLOC(7 DOWNTO 1)>=45 AND HLOC(7 DOWNTO 1)<=82)then
           RGBX<="000";RGBY<="000";
        ELSE
           RGBX<="100";RGBY<="000";
        END IF;
ELSIF(Vloc(7 downto 1)=72 )THEN
     IF((HLOC(7 DOWNTO 1)>=0 AND HLOC(7 DOWNTO 1)<=9)OR(HLOC(7 DOWNTO
1)>=118   -----RED EDGE
         AND HLOC(7 DOWNTO 1)<=127)
        )THEN
           RGBX<="100";RGBY<="010";
      ELSIF(HLOC(7 DOWNTO 1)>=46 AND HLOC(7 DOWNTO 1)<=81)then
           RGBX<="000";RGBY<="000";
        ELSE
           RGBX<="100";RGBY<="000";
        END IF;
ELSIF(Vloc(7 downto 1)=73 )THEN
     IF((HLOC(7 DOWNTO 1)>=0 AND HLOC(7 DOWNTO 1)<=9)OR(HLOC(7 DOWNTO
1)>=118   -----RED EDGE
         AND HLOC(7 DOWNTO 1)<=127)
        )THEN
           RGBX<="100";RGBY<="010";
      ELSIF(HLOC(7 DOWNTO 1)>=47 AND HLOC(7 DOWNTO 1)<=80)then
           RGBX<="000";RGBY<="000";
        ELSE
           RGBX<="100";RGBY<="000";
        END IF;
ELSIF(Vloc(7 downto 1)=74 )THEN
     IF((HLOC(7 DOWNTO 1)>=0 AND HLOC(7 DOWNTO 1)<=9)OR(HLOC(7 DOWNTO
1)>=118   -----RED EDGE
         AND HLOC(7 DOWNTO 1)<=127)
```

```
        )THEN
           RGBX<="100";RGBY<="010";
      ELSIF(HLOC(7 DOWNTO 1)>=48 AND HLOC(7 DOWNTO 1)<=79)then
           RGBX<="000";RGBY<="000";
        ELSE
           RGBX<="100";RGBY<="000";
        END IF;
ELSIF(Vloc(7 downto 1)=75 )THEN
     IF((HLOC(7 DOWNTO 1)>=0 AND HLOC(7 DOWNTO 1)<=9)OR(HLOC(7 DOWNTO
1)>=118   -----RED EDGE
         AND HLOC(7 DOWNTO 1)<=127)
        )THEN
           RGBX<="100";RGBY<="010";
      ELSIF(HLOC(7 DOWNTO 1)>=49 AND HLOC(7 DOWNTO 1)<=78)then
           RGBX<="000";RGBY<="000";
        ELSE
           RGBX<="100";RGBY<="000";
        END IF;
ELSIF(Vloc(7 downto 1)=76 )THEN
     IF((HLOC(7 DOWNTO 1)>=0 AND HLOC(7 DOWNTO 1)<=9)OR(HLOC(7 DOWNTO
1)>=118   -----RED EDGE
         AND HLOC(7 DOWNTO 1)<=127)
        )THEN
           RGBX<="100";RGBY<="010";
      ELSIF(HLOC(7 DOWNTO 1)>=50 AND HLOC(7 DOWNTO 1)<=77)then
           RGBX<="000";RGBY<="000";
        ELSE
           RGBX<="100";RGBY<="000";
        END IF;

ELSIF(Vloc(7 downto 1)=77 )THEN
     IF((HLOC(7 DOWNTO 1)>=0 AND HLOC(7 DOWNTO 1)<=9)OR(HLOC(7 DOWNTO
1)>=118   -----RED EDGE
         AND HLOC(7 DOWNTO 1)<=127)
        )THEN
           RGBX<="100";RGBY<="010";
      ELSIF(HLOC(7 DOWNTO 1)>=52 AND HLOC(7 DOWNTO 1)<=75)then
           RGBX<="000";RGBY<="000";
        ELSE
           RGBX<="100";RGBY<="000";
        END IF;
ELSIF(Vloc(7 downto 1)=78 )THEN
     IF((HLOC(7 DOWNTO 1)>=0 AND HLOC(7 DOWNTO 1)<=9)OR(HLOC(7 DOWNTO
1)>=118   -----RED EDGE
```

```
          AND HLOC(7 DOWNTO 1)<=127)
        )THEN
           RGBX<="100";RGBY<="010";
      ELSIF(HLOC(7 DOWNTO 1)>=55 AND HLOC(7 DOWNTO 1)<=72)then
           RGBX<="000";RGBY<="000";
        ELSE
           RGBX<="100";RGBY<="000";
        END IF;
ELSIF(Vloc(7 downto 1)=79 )THEN
      IF((HLOC(7 DOWNTO 1)>=0 AND HLOC(7 DOWNTO 1)<=9)OR(HLOC(7 DOWNTO
1)>=118   -----RED EDGE
          AND HLOC(7 DOWNTO 1)<=127)
        )THEN
           RGBX<="100";RGBY<="010";
      ELSIF(HLOC(7 DOWNTO 1)>=59 AND HLOC(7 DOWNTO 1)<=68)then
           RGBX<="000";RGBY<="000";
        ELSE
           RGBX<="100";RGBY<="000";
        END IF;
--"PLAY"字样
ELSIF(Vloc(7 downto 1)>=92 AND Vloc(7 downto 1)<=93)THEN
      IF((HLOC(7 DOWNTO 1)>=0 AND HLOC(7 DOWNTO 1)<=9)OR(HLOC(7 DOWNTO
1)>=118   -----RED EDGE
          AND HLOC(7 DOWNTO 1)<=127)OR(HLOC(7 DOWNTO 1)>=40 AND HLOC(7
DOWNTO 1)<=49)
          OR(HLOC(7 DOWNTO 1)>=53 AND HLOC(7 DOWNTO 1)<=54)OR(HLOC(7
DOWNTO 1)>=66 AND HLOC(7 DOWNTO 1)<=75)
          OR(HLOC(7 DOWNTO 1)>=79 AND HLOC(7 DOWNTO 1)<=80)OR(HLOC(7
DOWNTO 1)>=87 AND HLOC(7 DOWNTO 1)<=88)
        )THEN
           RGBX<="100";RGBY<="010";
        ELSE
           RGBX<="100";RGBY<="000";
        END IF;
ELSIF(Vloc(7 downto 1)>=94 AND Vloc(7 downto 1)<=96)THEN
      IF((HLOC(7 DOWNTO 1)>=0 AND HLOC(7 DOWNTO 1)<=9)OR(HLOC(7 DOWNTO
1)>=118   -----RED EDGE
          AND HLOC(7 DOWNTO 1)<=127)OR HLOC(7 DOWNTO 1)=40 OR HLOC(7
DOWNTO 1)=41 OR HLOC(7 DOWNTO 1)=48
           OR HLOC(7 DOWNTO 1)=49 OR HLOC(7 DOWNTO 1)=53 OR HLOC(7 DOWNTO
1)=54 OR HLOC(7 DOWNTO 1)=66
            OR HLOC(7 DOWNTO 1)=67 OR HLOC(7 DOWNTO 1)=74 OR HLOC(7 DOWNTO
1)=75 OR HLOC(7 DOWNTO 1)=79
             OR HLOC(7 DOWNTO 1)=80 OR HLOC(7 DOWNTO 1)=87 OR HLOC(7 DOWNTO
```

```
1)=88
        )THEN
           RGBX<="100";RGBY<="010";
        ELSE
           RGBX<="100";RGBY<="000";
        END IF;
ELSIF(Vloc(7 downto 1)=97 )THEN
      IF((HLOC(7 DOWNTO 1)>=0 AND HLOC(7 DOWNTO 1)<=9)OR(HLOC(7 DOWNTO
1)>=118    -----RED EDGE
         AND HLOC(7 DOWNTO 1)<=127)OR HLOC(7 DOWNTO 1)=40 OR HLOC(7
DOWNTO 1)=41 OR HLOC(7 DOWNTO 1)=48
          OR HLOC(7 DOWNTO 1)=49 OR HLOC(7 DOWNTO 1)=53 OR HLOC(7 DOWNTO
1)=54 OR HLOC(7 DOWNTO 1)=66
           OR HLOC(7 DOWNTO 1)=67 OR HLOC(7 DOWNTO 1)=74 OR HLOC(7 DOWNTO
1)=75 OR HLOC(7 DOWNTO 1)=81
            OR HLOC(7 DOWNTO 1)=80 OR HLOC(7 DOWNTO 1)=87 OR HLOC(7 DOWNTO
1)=86
        )THEN
           RGBX<="100";RGBY<="010";
        ELSE
           RGBX<="100";RGBY<="000";
        END IF;
ELSIF(Vloc(7 downto 1)=98 )THEN
      IF((HLOC(7 DOWNTO 1)>=0 AND HLOC(7 DOWNTO 1)<=9)OR(HLOC(7 DOWNTO
1)>=118    -----RED EDGE
         AND HLOC(7 DOWNTO 1)<=127)OR HLOC(7 DOWNTO 1)=40 OR HLOC(7
DOWNTO 1)=41 OR HLOC(7 DOWNTO 1)=48
          OR HLOC(7 DOWNTO 1)=49 OR HLOC(7 DOWNTO 1)=53 OR HLOC(7 DOWNTO
1)=54 OR HLOC(7 DOWNTO 1)=66
           OR HLOC(7 DOWNTO 1)=67 OR HLOC(7 DOWNTO 1)=74 OR HLOC(7 DOWNTO
1)=75 OR HLOC(7 DOWNTO 1)=81
            OR HLOC(7 DOWNTO 1)=82 OR HLOC(7 DOWNTO 1)=85 OR HLOC(7 DOWNTO
1)=86
        )THEN
           RGBX<="100";RGBY<="010";
        ELSE
           RGBX<="100";RGBY<="000";
        END IF;
ELSIF(Vloc(7 downto 1)=99 )THEN
      IF((HLOC(7 DOWNTO 1)>=0 AND HLOC(7 DOWNTO 1)<=9)OR(HLOC(7 DOWNTO
1)>=118    -----RED EDGE
         AND HLOC(7 DOWNTO 1)<=127)OR HLOC(7 DOWNTO 1)=40 OR HLOC(7
DOWNTO 1)=41 OR HLOC(7 DOWNTO 1)=48
          OR HLOC(7 DOWNTO 1)=49 OR HLOC(7 DOWNTO 1)=53 OR HLOC(7 DOWNTO
```

```
1)=54 OR (HLOC(7 DOWNTO 1)>=66
            AND HLOC(7 DOWNTO 1)<=75) OR HLOC(7 DOWNTO 1)=83
             OR HLOC(7 DOWNTO 1)=82 OR HLOC(7 DOWNTO 1)=85 OR HLOC(7 DOWNTO
1)=84
        )THEN
           RGBX<="100";RGBY<="010";
        ELSE
           RGBX<="100";RGBY<="000";
        END IF;
ELSIF(Vloc(7 downto 1)=100 )THEN
      IF((HLOC(7 DOWNTO 1)>=0 AND HLOC(7 DOWNTO 1)<=9)OR(HLOC(7 DOWNTO
1)>=118    -----RED EDGE
         AND HLOC(7 DOWNTO 1)<=127)OR HLOC(7 DOWNTO 1)=40 OR HLOC(7
DOWNTO 1)=41 OR HLOC(7 DOWNTO 1)=48
          OR HLOC(7 DOWNTO 1)=49 OR HLOC(7 DOWNTO 1)=53 OR HLOC(7 DOWNTO
1)=54 OR (HLOC(7 DOWNTO 1)>=66
           AND HLOC(7 DOWNTO 1)<=75) OR HLOC(7 DOWNTO 1)=83 OR HLOC(7
DOWNTO 1)=84
        )THEN
           RGBX<="100";RGBY<="010";
        ELSE
           RGBX<="100";RGBY<="000";
        END IF;
ELSIF(Vloc(7 downto 1)=101 OR Vloc(7 downto 1)=102 )THEN
      IF((HLOC(7 DOWNTO 1)>=0 AND HLOC(7 DOWNTO 1)<=9)OR(HLOC(7 DOWNTO
1)>=118    -----RED EDGE
         AND HLOC(7 DOWNTO 1)<=127)OR(HLOC(7 DOWNTO 1)>=40 AND HLOC(7
DOWNTO 1)<=49)OR HLOC(7 DOWNTO 1)=53
          OR HLOC(7 DOWNTO 1)=54 OR HLOC(7 DOWNTO 1)=66
        OR HLOC(7 DOWNTO 1)=67 OR HLOC(7 DOWNTO 1)=74 OR HLOC(7 DOWNTO
1)=75 OR HLOC(7 DOWNTO 1)=83 OR HLOC(7 DOWNTO 1)=84
        )THEN
           RGBX<="100";RGBY<="010";
        ELSE
           RGBX<="100";RGBY<="000";
        END IF;
ELSIF(Vloc(7 downto 1)>=103 AND Vloc(7 downto 1)<=105)THEN
      IF((HLOC(7 DOWNTO 1)>=0 AND HLOC(7 DOWNTO 1)<=9)OR(HLOC(7 DOWNTO
1)>=118    -----RED EDGE
         AND HLOC(7 DOWNTO 1)<=127)OR HLOC(7 DOWNTO 1)=40 OR HLOC(7
DOWNTO 1)=41
          OR HLOC(7 DOWNTO 1)=53 OR HLOC(7 DOWNTO 1)=54 OR HLOC(7 DOWNTO
1)=66
           OR HLOC(7 DOWNTO 1)=67 OR HLOC(7 DOWNTO 1)=74 OR HLOC(7 DOWNTO
```

```
1)=75 OR HLOC(7 DOWNTO 1)=83
              OR HLOC(7 DOWNTO 1)=84
        )THEN
           RGBX<="100";RGBY<="010";
        ELSE
           RGBX<="100";RGBY<="000";
        END IF;
ELSIF(Vloc(7 downto 1)>=106 AND Vloc(7 downto 1)<=107)THEN
      IF((HLOC(7 DOWNTO 1)>=0 AND HLOC(7 DOWNTO 1)<=9)OR(HLOC(7 DOWNTO
1)>=118   -----RED EDGE
         AND HLOC(7 DOWNTO 1)<=127)OR HLOC(7 DOWNTO 1)=40 OR HLOC(7
DOWNTO 1)=41
         OR (HLOC(7 DOWNTO 1)>=53 AND HLOC(7 DOWNTO 1)<=62) OR HLOC(7
DOWNTO 1)=66
            OR HLOC(7 DOWNTO 1)=67 OR HLOC(7 DOWNTO 1)=74 OR HLOC(7 DOWNTO
1)=75 OR HLOC(7 DOWNTO 1)=83
              OR HLOC(7 DOWNTO 1)=84
        )THEN
           RGBX<="100";RGBY<="010";
        ELSE
           RGBX<="100";RGBY<="000";
        END IF;
   end if;




----------打地鼠画面
ELSIF(en='1' AND start='1' and lose='0' and win='0' )THEN
    if ((hloc(7 downto 2)>=4 and hloc(7 downto 2)<=14) and
         (vloc(7 downto 2)>=9 and vloc(7 downto 2)<=29))then
    IF(led="000" and key="000")then
    rgbx<="001";rgby<="000";
    elsif led="000"    then
    rgbx<="100";rgby<="000";
    else rgbx<="000";rgby<="000";end if;

    elsif((hloc(7 downto 2)>=19 and hloc(7 downto 2)<=29) and
         (vloc(7 downto 2)>=9 and vloc(7 downto 2)<=29))then
    if( led="001"and key="001")then
    rgbx<="001";rgby<="000";
    elsif led="001" then
```

```
rgbx<="100";rgby<="000";
else rgbx<="000";rgby<="000";end if;

elsif((hloc(7 downto 2)>=34 and hloc(7 downto 2)<=44) and
      (vloc(7 downto 2)>=9 and vloc(7 downto 2)<=29))then
if(led="010" and key="010")THEN
rgbx <=   "001";rgby<="000";
elsif led="010" then
rgbx<="100";rgby<="000";
else rgbx<="000";rgby<="000";end if;

elsif((hloc(7 downto 2)>=49 and hloc(7 downto 2)<=59) and
      (vloc(7 downto 2)>=9 and vloc(7 downto 2)<=29))then
if(led="011" and key="011")then
rgbx <=   "001";rgby<="000";
elsif led="011" then
rgbx<="100";rgby<="000";
else rgbx<="000";rgby<="000";end if;

elsif((hloc(7 downto 2)>=4 and hloc(7 downto 2)<=14) and
      (vloc(7 downto 2)>=39 and vloc(7 downto 2)<=59))then
if(led="100" and key="100")then
rgbx <=   "001";rgby<="000";
elsif led="100" then
rgbx<="100";rgby<="000";
else rgbx<="000";rgby<="000";end if;

elsif((hloc(7 downto 2)>=19 and hloc(7 downto 2)<=29) and
      (vloc(7 downto 2)>=39 and vloc(7 downto 2)<=59))then
if(led="101" and key="101")then
rgbx <=   "001";rgby<="000";
elsif led="101" then
rgbx<="100";rgby<="000";
else rgbx<="000";rgby<="000";end if;

elsif((hloc(7 downto 2)>=34 and hloc(7 downto 2)<=44) and
      (vloc(7 downto 2)>=39 and vloc(7 downto 2)<=59))then
if(led="110" and key="110")then
rgbx <=   "001";rgby<="000";
elsif led="110" then
rgbx<="100";rgby<="000";
else rgbx<="000";rgby<="000";end if;

elsif((hloc(7 downto 2)>=49 and hloc(7 downto 2)<=59) and
```

```
        (vloc(7 downto 2)>=39 and vloc(7 downto 2)<=59))then
    if(led="111" and key="111")then
    rgbx <=    "001";rgby<="000";
    elsif led="111" then
    rgbx<="100";rgby<="000";
    else rgbx<="000";rgby<="000";end if;




    elsif(((hloc(7 downto 2)>0 and hloc(7 downto 2)<3) or (hloc(7 downto 2)>15 and
hloc(7 downto 2)<18)
        or(hloc(7 downto 2)>30 and hloc(7 downto 2)<33) or (hloc(7 downto 2)>45 and
hloc(7 downto 2)<48)
        or (hloc(7 downto 2)>60 and hloc(7 downto 2)<63)) or
        ((vloc(7 downto 2)>0 and vloc(7 downto 2)<8) or (vloc(7 downto 2)>30 and vloc(7
downto 2)<38)
        or (vloc(7 downto 2)>60 and vloc(7 downto 2)<63))) then
      RGBX<="010"; RGBY<="000";




    end if;



----------------------------------------------------游戏失败画面显示
-----------------------------------------
ELSIF(en='1' and start='1' and lose='1')THEN
  case vloc(7 downto 2) is
  WHEN "000101"|"111010" =>
  IF(HLOC(7 DOWNTO 2)>=25 AND HLOC(7 DOWNTO 2)<=38)THEN
  RGBY<="111";
  ELSE
  RGBY<="000";
  END IF;
  WHEN "000110"|"111001"=>
  IF(HLOC(7 DOWNTO 2)>=22 AND HLOC(7 DOWNTO 2)<=41)THEN
  RGBY<="111";
  ELSE
  RGBY<="000";
```

```
END IF;
WHEN "000111"|"111000"=>
IF(HLOC(7 DOWNTO 2)>=20 AND HLOC(7 DOWNTO 2)<=43)THEN
RGBY<="111";
ELSE
RGBY<="000";
END IF;
WHEN "001000"|"110111"=>
IF(HLOC(7 DOWNTO 2)>=18 AND HLOC(7 DOWNTO 2)<=45)THEN
RGBY<="111";
ELSE
RGBY<="000";
END IF;
WHEN "001001"|"110110"=>
IF(HLOC(7 DOWNTO 2)>=16 AND HLOC(7 DOWNTO 2)<=47)THEN
RGBY<="111";
ELSE
RGBY<="000";
END IF;
WHEN "001010"|"110101"=>
IF(HLOC(7 DOWNTO 2)>=15 AND HLOC(7 DOWNTO 2)<=48)THEN
RGBY<="111";
ELSE
RGBY<="000";
END IF;
WHEN "001011"|"110100"=>
IF(HLOC(7 DOWNTO 2)>=14 AND HLOC(7 DOWNTO 2)<=49)THEN
RGBY<="111";
ELSE
RGBY<="000";
END IF;
WHEN "001100"|"110011"=>
IF(HLOC(7 DOWNTO 2)>=13 AND HLOC(7 DOWNTO 2)<=50)THEN
RGBY<="111";
ELSE
RGBY<="000";
END IF;
WHEN "001101"|"110010"=> --13+50
IF(HLOC(7 DOWNTO 2)>=12 AND HLOC(7 DOWNTO 2)<=51)THEN
RGBY<="111";
ELSE
RGBY<="000";
END IF;
WHEN "001110"|"110001"=> --14 49
```

```
IF(HLOC(7 DOWNTO 2)>=11 AND HLOC(7 DOWNTO 2)<=52)THEN
RGBY<="111";
ELSE
RGBY<="000";
END IF;
WHEN "001111"|"110000"=> --15 48
IF(HLOC(7 DOWNTO 2)>=10 AND HLOC(7 DOWNTO 2)<=53)THEN
RGBY<="111";
ELSE
RGBY<="000";
END IF;
WHEN "010000"|"010001"|"101110"|"101111" => --16,17,46,47
IF(HLOC(7 DOWNTO 2)>=9 AND HLOC(7 DOWNTO 2)<=54)THEN
RGBY<="111";
ELSE
RGBY<="000";
END IF;
WHEN "010010"|"010011"|"101101" =>--18,19,45
IF(HLOC(7 DOWNTO 2)>=8 AND HLOC(7 DOWNTO 2)<=55)THEN
RGBY<="111";
ELSE
RGBY<="000";
END IF;

WHEN "101100" =>--44
IF(HLOC(7 DOWNTO 2)>=8 AND HLOC(7 DOWNTO 2)<=55)THEN
   IF(HLOC(7 DOWNTO 2)=23 OR HLOC(7 DOWNTO 2)=40)THEN
   RGBY<="100";
   ELSE RGBY<="111" ; END IF;
ELSE
RGBY<="000";
END IF;

WHEN "010100"=>--20
IF(HLOC(7 DOWNTO 2)>=7 AND HLOC(7 DOWNTO 2)<=56)THEN
RGBY<="111";
ELSE
RGBY<="000";
END IF;

WHEN "101010"=>--42
IF(HLOC(7 DOWNTO 2)>=7 AND HLOC(7 DOWNTO 2)<=56)THEN
   IF(HLOC(7 DOWNTO 2)=25 OR HLOC(7 DOWNTO 2)=38)THEN
   RGBY<="100";
```

```
   ELSE RGBY<="111"; END IF;
ELSE
RGBY<="000";
END IF;

WHEN "101011"=>--43
IF(HLOC(7 DOWNTO 2)>=7 AND HLOC(7 DOWNTO 2)<=56)THEN
   IF(HLOC(7 DOWNTO 2)=24 OR HLOC(7 DOWNTO 2)=39)THEN
   RGBY<="100";
   ELSE RGBY<="111"; END IF;
ELSE
RGBY<="000";
END IF;



WHEN "010101" => --21
IF(HLOC(7 DOWNTO 2)>=7 AND HLOC(7 DOWNTO 2)<=56)THEN
  if (hloc(7 DOWNTO 2)=14 or hloc(7 DOWNTO 2) = 23 or hloc(7 DOWNTO 2)=49
     or hloc(7 DOWNTO 2) = 40 ) then
     rgby <= "100";
else    RGBY<="111"; END IF;
ELSE
RGBY<="000";
END IF;

WHEN "100111"=>--39
IF(HLOC(7 DOWNTO 2)>=6 AND HLOC(7 DOWNTO 2)<=57)THEN
RGBY<="111";
ELSE
RGBY<="000";
END IF;

WHEN "101000" =>--40
IF(HLOC(7 DOWNTO 2)>=6 AND HLOC(7 DOWNTO 2)<=57)THEN
  IF(HLOC(7 DOWNTO 2)>=27 AND HLOC(7 DOWNTO 2)<=36)THEN
  RGBY<="100";
  ELSE    RGBY<="111"; END IF;
ELSE
RGBY<="000";
END IF;

WHEN "101001" =>--41
IF(HLOC(7 DOWNTO 2)>=6 AND HLOC(7 DOWNTO 2)<=57)THEN
```

```
    IF(HLOC(7 DOWNTO 2)=26 OR HLOC(7 DOWNTO 2)=37)THEN
    RGBY<="100";
    ELSE RGBY<="111"; END IF;
   ELSE
   RGBY<="000";
   END IF;

   WHEN "010110" =>--22
   IF(HLOC(7 DOWNTO 2)>=6 AND HLOC(7 DOWNTO 2)<=57)THEN
    if (hloc(7 DOWNTO 2)=15 or hloc(7 DOWNTO 2)= 22 or hloc(7 DOWNTO 2)=48
       or hloc(7 DOWNTO 2) = 41 ) then
       rgby <= "100";
    else    RGBY<="111";END IF;
   ELSE
   RGBY<="000";
   END IF;

   WHEN "010111" =>--23
   IF(HLOC(7 DOWNTO 2)>=6 AND HLOC(7 DOWNTO 2)<=57)THEN
      if (hloc(7 DOWNTO 2)=16 or hloc(7 DOWNTO 2) = 21 or hloc(7 DOWNTO 2)=47
       or hloc(7 DOWNTO 2) = 42 ) then
       rgby <= "100";
      else    RGBY<="111";END IF;
   ELSE
   RGBY<="000";
   END IF;

WHEN "011000" =>--24
   IF(HLOC(7 DOWNTO 2)>=6 AND HLOC(7 DOWNTO 2)<=57)THEN
    if (hloc(7 DOWNTO 2)=17 or hloc(7 DOWNTO 2) = 20 or hloc(7 DOWNTO 2)=46
       or hloc(7 DOWNTO 2) = 43 ) then
       rgby <= "100";
    else    RGBY<="111";END IF;
   ELSE
   RGBY<="000";
   END IF;

   WHEN
"011111"|"100000"|"100001"|"100010"|"100011"|"100100"|"100101"|"100110"=>--31,32,3
3,34,35,36,37,38
   IF(HLOC(7 DOWNTO 2)>=5 AND HLOC(7 DOWNTO 2)<=58)THEN
   RGBY<="111";
   ELSE
   RGBY<="000";
```

```
 END IF;

WHEN "011001" =>                                --25
 IF(HLOC(7 DOWNTO 2)>=5 AND HLOC(7 DOWNTO 2)<=58)THEN
  if (hloc(7 DOWNTO 2)=18 or hloc(7 DOWNTO 2) = 19 or hloc(7 DOWNTO 2)=45
       or hloc(7 DOWNTO 2)= 44 ) then
     rgby <= "100";
 else    RGBY<="111"; END IF;
 ELSE
 RGBY<="000";
 END IF;


 WHEN "011010"=>                                --26
 IF(HLOC(7 DOWNTO 2)>=5 AND HLOC(7 DOWNTO 2)<=58)THEN
    if (hloc(7 DOWNTO 2)=19 or hloc(7 DOWNTO 2) = 18 or hloc(7 DOWNTO 2)=44
     or hloc(7 DOWNTO 2) = 45 ) then
     rgby <= "100";
 else RGBY<="111"; END IF;
 ELSE
 RGBY<="000";
 END IF;


 WHEN "011011" =>                        --27
 IF(HLOC(7 DOWNTO 2)>=5 AND HLOC(7 DOWNTO 2)<=58)THEN
    if (hloc(7 DOWNTO 2)=20 or hloc(7 DOWNTO 2) = 17 or hloc(7 DOWNTO 2)=43
     or hloc(7 DOWNTO 2) = 46 ) then
     rgby <= "100";
 else RGBY<="111"; END IF;
 ELSE
 RGBY<="000";
 END IF;


 WHEN "011100" =>                    --28
 IF(HLOC(7 DOWNTO 2)>=5 AND HLOC(7 DOWNTO 2)<=58)THEN
    if (hloc(7 DOWNTO 2)=21 or hloc(7 DOWNTO 2) = 16 or hloc(7 DOWNTO 2)=42
     or hloc(7 DOWNTO 2) = 47 ) then
     rgby <= "100";
 else     RGBY<="111"; END IF;
 ELSE
 RGBY<="000";
 END IF;
```

```
   WHEN "011101" =>                  --29
   IF(HLOC(7 DOWNTO 2)>=5 AND HLOC(7 DOWNTO 2)<=58)THEN
    if (hloc(7 DOWNTO 2)=22 or hloc(7 DOWNTO 2) = 15 or hloc(7 DOWNTO 2)=41
       or hloc(7 DOWNTO 2) = 48 ) then
       rgby <= "100";
   else RGBY<="111";    END IF;
   ELSE
   RGBY<="000";
   END IF;

    WHEN "011110" =>                  --30
   IF(HLOC(7 DOWNTO 2)>=5 AND HLOC(7 DOWNTO 2)<=58)THEN
      if (hloc(7 DOWNTO 2)=23 or hloc(7 DOWNTO 2) = 14 or hloc(7 DOWNTO 2)=40
       or hloc(7 DOWNTO 2) = 49 ) then
       rgby <= "100";
   else RGBY<="111";     END IF;
   ELSE
   RGBY<="000";
   END IF;



   WHEN OTHERS=>
   RGBY<="000";
   END CASE;

-------------------------------------------------------------------------------------------------
----------
  ------------------------------------------------------游戏胜利画面显示
-----------------------------------------
ELSIF(en='1' and start='1' and win='1')THEN
   case Vloc(7 downto 2) is
   WHEN "000101"|"111010" =>
   IF(HLOC(7 DOWNTO 2)>=25 AND HLOC(7 DOWNTO 2)<=38)THEN
   RGBY<="111";
   ELSE
   RGBY<="000";
   END IF;
   WHEN "000110"|"111001"=>
   IF(HLOC(7 DOWNTO 2)>=22 AND HLOC(7 DOWNTO 2)<=41)THEN
   RGBY<="111";
   ELSE
   RGBY<="000";
```

```
END IF;
WHEN "000111"|"111000"=>
IF(HLOC(7 DOWNTO 2)>=20 AND HLOC(7 DOWNTO 2)<=43)THEN
RGBY<="111";
ELSE
RGBY<="000";
END IF;
WHEN "001000"|"110111"=>
IF(HLOC(7 DOWNTO 2)>=18 AND HLOC(7 DOWNTO 2)<=45)THEN
RGBY<="111";
ELSE
RGBY<="000";
END IF;
WHEN "001001"|"110110"=>
IF(HLOC(7 DOWNTO 2)>=16 AND HLOC(7 DOWNTO 2)<=47)THEN
RGBY<="111";
ELSE
RGBY<="000";
END IF;
WHEN "001010"|"110101"=>
IF(HLOC(7 DOWNTO 2)>=15 AND HLOC(7 DOWNTO 2)<=48)THEN
RGBY<="111";
ELSE
RGBY<="000";
END IF;
WHEN "001011"|"110100"=>
IF(HLOC(7 DOWNTO 2)>=14 AND HLOC(7 DOWNTO 2)<=49)THEN
RGBY<="111";
ELSE
RGBY<="000";
END IF;
WHEN "001100"|"110011"=>       ---12,51
IF(HLOC(7 DOWNTO 2)>=13 AND HLOC(7 DOWNTO 2)<=50)THEN
RGBY<="111";
ELSE
RGBY<="000";
END IF;
WHEN "001101"=>--|50
IF(HLOC(7 DOWNTO 2)>=12 AND HLOC(7 DOWNTO 2)<=51)THEN
RGBY<="111";
ELSE
RGBY<="000";
END IF;
WHEN "001110"=>--|49
```

```
IF(HLOC(7 DOWNTO 2)>=11 AND HLOC(7 DOWNTO 2)<=52)THEN
RGBY<="111";
ELSE
RGBY<="000";
END IF;
WHEN "001111"=>--|48
IF(HLOC(7 DOWNTO 2)>=10 AND HLOC(7 DOWNTO 2)<=53)THEN
RGBY<="111";
ELSE
RGBY<="000";
END IF;
WHEN "010000"|"010001" =>--|46|47
IF(HLOC(7 DOWNTO 2)>=9 AND HLOC(7 DOWNTO 2)<=54)THEN
RGBY<="111";
ELSE
RGBY<="000";
END IF;
WHEN "010010"|"010011"=>--|44|45
IF(HLOC(7 DOWNTO 2)>=8 AND HLOC(7 DOWNTO 2)<=55)THEN
RGBY<="111";
ELSE
RGBY<="000";
END IF;
WHEN "010100"|"010101"=>--|42|43
IF(HLOC(7 DOWNTO 2)>=7 AND HLOC(7 DOWNTO 2)<=56)THEN
RGBY<="111";
ELSE
RGBY<="000";
END IF;
WHEN "010110" =>---haicha    (39)|40|41
IF(HLOC(7 DOWNTO 2)>=6 AND HLOC(7 DOWNTO 2)<=57)THEN
   if((hloc(7 downto 2)>=15 and hloc(7 downto 2)<=17)or
  (hloc(7 downto 2)>=20 and hloc(7 downto 2)<=22)or
  (hloc(7 downto 2)>=41 and hloc(7 downto 2)<=43)or
  (hloc(7 downto 2)>=46 and hloc(7 downto 2)<=48))then
     rgby<="100";
   else rgby<="111";end if;      ------FACE,HOW TO YELLOW????
ELSE
RGBY<="000";
END IF;
WHEN "010111"|"011000"=>
IF(HLOC(7 DOWNTO 2)>=6 AND HLOC(7 DOWNTO 2)<=57)THEN
   if((hloc(7 downto 2)>=14 and hloc(7 downto 2)<=23)or
  (hloc(7 downto 2)>=40 and hloc(7 downto 2)<=49))then
```

```
     rgby<="100";
   else rgby<="111";end if;
ELSE
RGBY<="000";
END IF;
WHEN "011001" =>--||||(29)|
IF(HLOC(7 DOWNTO 2)>=5 AND HLOC(7 DOWNTO 2)<=58)THEN
   if((hloc(7 downto 2)>=14 and hloc(7 downto 2)<=23)or
  (hloc(7 downto 2)>=40 and hloc(7 downto 2)<=49))then
     rgby<="100";
   else rgby<="111";end if;
ELSE
RGBY<="000";
END IF;
WHEN"011010"=>
IF(HLOC(7 DOWNTO 2)>=5 AND HLOC(7 DOWNTO 2)<=58)THEN
   if((hloc(7 downto 2)>=15 and hloc(7 downto 2)<=22) or
  (hloc(7 downto 2)>=41 and hloc(7 downto 2)<=48))then
   rgby<="100";
   else rgby<="111";end if;
ELSE
RGBY<="000";
END IF;
WHEN"011011"=>
IF(HLOC(7 DOWNTO 2)>=5 AND HLOC(7 DOWNTO 2)<=58)THEN
   if((hloc(7 downto 2)>=16 and hloc(7 downto 2)<=21) or
  (hloc(7 downto 2)>=42 and hloc(7 downto 2)<=47))then
   rgby<="100";
   else rgby<="111";end if;
ELSE
RGBY<="000";
END IF;
WHEN"011100"=>
IF(HLOC(7 DOWNTO 2)>=5 AND HLOC(7 DOWNTO 2)<=58)THEN
   if((hloc(7 downto 2)>=17 and hloc(7 downto 2)<=20)or
  (hloc(7 downto 2)>=43 and hloc(7 downto 2)<=46))then
   rgby<="100";
   else rgby<="111";end if;
ELSE
RGBY<="000";
END IF;
WHEN"011101"=>
IF(HLOC(7 DOWNTO 2)>=5 AND HLOC(7 DOWNTO 2)<=58)THEN
   if((hloc(7 downto 2)>=18 and hloc(7 downto 2)<=19)or
```

```
  (hloc(7 downto 2)>=44 and hloc(7 downto 2)<=45))then
    rgby<="100";
    else rgby<="111";end if;
  ELSE
  RGBY<="000";
  END IF;
  WHEN"011110"|"011111"|"100000"|"100001"|"100010"|"100011"|"100100"
  |"100101"|"100110"=>
  IF(HLOC(7 DOWNTO 2)>=5 AND HLOC(7 DOWNTO 2)<=58)THEN
  rgby<="111";
  ELSE
  RGBY<="000";
  END IF;
  WHEN"100111"=>
  IF(HLOC(7 DOWNTO 2)>=6 AND HLOC(7 DOWNTO 2)<=57)THEN
  rgby<="111";
  ELSE
  RGBY<="000";
  END IF;
  WHEN"101000"=>
  IF(HLOC(7 DOWNTO 2)>=6 AND HLOC(7 DOWNTO 2)<=57)THEN
    if(hloc(7 DOWNTO 2)=18 or hloc(7 DOWNTO 2)=19 or hloc(7 DOWNTO 2)=20
    or hloc(7 DOWNTO 2)= 43 or hloc(7 DOWNTO 2) = 44 or hloc(7 DOWNTO 2) =
45)then
    rgby<="100";
    else rgby<="111";end if;
  ELSE
  RGBY<="000";
  END IF;
  WHEN "101001"=>
  IF(HLOC(7 DOWNTO 2)>=6 AND HLOC(7 DOWNTO 2)<=57)THEN
    if(hloc(7 DOWNTO 2)=19 or hloc(7 DOWNTO 2)=20 or hloc(7 DOWNTO 2) = 43
    or hloc(7 DOWNTO 2) = 44)then
    rgby<="100";
    else rgby<="111"; end if;
  ELSE
  RGBY<="000";
  END IF;
  WHEN"101010"=>
  IF(HLOC(7 DOWNTO 2)>=7 AND HLOC(7 DOWNTO 2)<=56)THEN
    if(hloc(7 DOWNTO 2)=19 or hloc(7 DOWNTO 2)=20 or hloc(7 DOWNTO 2) = 43
    or hloc(7 DOWNTO 2) = 44)then
    rgby<="100";
    else rgby<="111";end if;
```

```
ELSE
RGBY<="000";
END IF;
WHEN "101011"=>
IF(HLOC(7 DOWNTO 2)>=7 AND HLOC(7 DOWNTO 2)<=56)THEN
   if(hloc(7 DOWNTO 2)=20 or hloc(7 DOWNTO 2)=21 or hloc(7 DOWNTO 2) = 42
   or hloc(7 DOWNTO 2) = 43)then
   rgby<="100";
   else rgby<="111";end if;
ELSE
RGBY<="000";
END IF;
WHEN"101100"=>
IF(HLOC(7 DOWNTO 2)>=8 AND HLOC(7 DOWNTO 2)<=55)THEN
   if(hloc(7 DOWNTO 2)=21 or hloc(7 DOWNTO 2)=22 or hloc(7 DOWNTO 2) = 41
   or hloc(7 DOWNTO 2) = 42)then
   rgby<="100";
   else rgby<="111";end if;
ELSE
RGBY<="000";
END IF;
WHEN "101101"=>
 IF(HLOC(7 DOWNTO 2)>=8 AND HLOC(7 DOWNTO 2)<=55)THEN
   if(hloc(7 DOWNTO 2)=22 or hloc(7 DOWNTO 2)=23 or hloc(7 DOWNTO 2) = 40
   or hloc(7 DOWNTO 2) = 41)then
   rgby<="100";
   else rgby<="111";end if;
ELSE
RGBY<="000";
END IF;
WHEN"101110"=>
IF(HLOC(7 DOWNTO 2)>=9 AND HLOC(7 DOWNTO 2)<=54)THEN
   if(hloc(7 DOWNTO 2)=23 or hloc(7 DOWNTO 2)=24 or hloc(7 DOWNTO 2) = 39
   or hloc(7 DOWNTO 2) = 40)then
   rgby<="100";
   else rgby<="111";end if;
ELSE
RGBY<="000";
END IF;
WHEN"101111"=>
IF(HLOC(7 DOWNTO 2)>=9 AND HLOC(7 DOWNTO 2)<=54)THEN
   if(hloc(7 DOWNTO 2)=24 or hloc(7 DOWNTO 2)=25 or hloc(7 DOWNTO 2) = 38
   or hloc(7 DOWNTO 2) = 39)then
   rgby<="100";
```

```
    else rgby<="111";end if;
  ELSE
  RGBY<="000";
  END IF;
  WHEN"110000"=>
  IF(HLOC(7 DOWNTO 2)>=10 AND HLOC(7 DOWNTO 2)<=53)THEN
    if(hloc(7 DOWNTO 2)=26 or hloc(7 DOWNTO 2)=27 or hloc(7 DOWNTO 2) = 36
    or hloc(7 DOWNTO 2) = 37)then
    rgby<="100";
    else rgby<="111";end if;
  ELSE
  RGBY<="000";
  END IF;
  WHEN"110001"=>
  IF(HLOC(7 DOWNTO 2)>=11 AND HLOC(7 DOWNTO 2)<=52)THEN
    if(hloc(7 DOWNTO 2)=28 or hloc(7 DOWNTO 2)=29 or hloc(7 DOWNTO 2) = 34
    or hloc(7 DOWNTO 2) = 35)then
    rgby<="100";
    else rgby<="111";end if;
  ELSE
  RGBY<="000";
  END IF;
  WHEN"110010"=>
  IF(HLOC(7 DOWNTO 2)>=12 AND HLOC(7 DOWNTO 2)<=51)THEN
    if(hloc(7 DOWNTO 2)=31 or hloc(7 DOWNTO 2) = 32)then
    rgby<="100";
    else rgby<="111";end if;
  ELSE
  RGBY<="000";
  END IF;
  WHEN OTHERS=>
  RGBY<="000";
  END CASE;
  END IF;


-------------------------------------------------------------------------------------------------
-------
```

```
end process;
-------------------------------------------
 process(clk50m,start)
 begin
    if start='0' then sysclk<='0';
      elsif clk50m'event and clk50m='1' then sysclk<=not sysclk;
    end if;
 end process;
-------------------------------------------------屏幕扫描显示
-----------------------------------------------



   rgb(2) <= rgbp(2) and enable;
   rgb(1) <= rgbp(1) and enable;
   rgb(0) <= rgbp(0) and enable;

   process(rgbx,rgby)
   begin
   rgbp <= rgbx xor rgby;
   end process;
--产生行计数（记录每行的点数），H_PERIOD 为行周期计数值
process(sysclk)
begin
   if (sysclk'event and sysclk = '1') then
   if hcnt < H_PERIOD then
     hcnt <= hcnt + 1;
   else--当行计数到达计数周期时将重置
     hcnt <= (others => '0');
   end if;
 end if;
end process;
--产生场记数（记录每帧中的行数），V_PERIOD 为场周期计数值
process(hsyncb)
begin
    if (hsyncb'event and hsyncb = '1') then
    if vcnt < V_PERIOD then
    vcnt <= vcnt + 1;
   else--当场计数到达计数周期时将重置
    vcnt <= (others => '0');
   end if;
 end if;
end process;
--产生行同步信号，H_PIXELS 为行显示点数，H_FRONT 为前消隐点数，H_SYNCTIME 为行
同步点数
```

```
process(sysclk)
begin
   if (sysclk'event and sysclk = '1') then
   if (hcnt >= (H_PIXELS + H_FRONT) and hcnt < (H_PIXELS + H_SYNCTIME + H_FRONT))
then
   hsyncb <= '0';
   else
     hsyncb <= '1';
   end if;
 end if;
end process;
--产生场同步信号，V_LINES 为场显示点数，V_FRONT 为前消隐点数，V_SYNCTIME 场同步
点数
process(hsyncb)
begin
   if (hsyncb'event and hsyncb = '1') then
   if (vcnt >= (V_LINES + V_FRONT) and vcnt < (V_LINES + V_SYNCTIME + V_FRONT)) then
     vsyncb <= '0';
   else
     vsyncb <= '1';
       end if;
 end if;
end process;
--屏幕显示使能信号赋值
process (sysclk)
begin
 if sysclk'EVENT and sysclk = '1' then
   if hcnt >= H_PIXELS or vcnt >= V_LINES then
     enable <= '0';
     else
       enable <= '1';
     end if;
 end if;
end process;

   hloc <= hcnt;
   vloc <= vcnt;
   hs <= hsyncb;
   vs <= vsyncb;
   red <= rgb(2);
   green <= rgb(1);
   blue <= rgb(0);
--------------------------------------------------------------------------------------------------
--------------------------
```

```
end Behavioral;
```

⑧顶层模块：

```
entity game is
port(clk50m:in std_logic;                --50MHz 的输入时钟
     k1,k2,k3,k4,k5,k6,k7,k8:in std_logic;      --8 个输入按键
     start:in std_logic;                 --开始信号，高电平有效
     clear:in std_logic;                 --最高分清零信号，clear=1 时清除最高分纪录
     en:in STD_LOGIC;                        --暂停信号，en=0 时暂停
     win,lose:out std_logic;             --win=1 表明游戏胜利，lose=1 表明游戏失败
     led:out std_logic_vector(2 downto 0);  --产生的三位随机数，送到 74LS138 进行译
码，从而点亮一个 LED
     leda,ledb,ledc,ledd:out std_logic;  --不同的难度分别用不同的 LED 灯来指示
     sound:out std_logic;                    --为输出的喇叭信号
     LCD_RS : out    STD_LOGIC;
    LCD_RW : out    STD_LOGIC;
    LCD_EN : out    STD_LOGIC;
    data : out    STD_LOGIC_VECTOR (3 downto 0));   --数据信号输出
end game;
architecture Behavioral of game is
signal A,B,C,D,clk,clkout,clk4,clk500,clk6m:std_logic;
signal key:STD_LOGIC_VECTOR (2 downto 0);
signal mwin,mlose:std_logic;
signal en_A,en_B,en_C,en_D: std_logic;
signal mled:std_logic_vector(2 downto 0);
signal score,highscore:STD_LOGIC_VECTOR(11 downto 0);
signal life:STD_LOGIC_VECTOR (7 downto 0);
--采样编码模块声明
component code
port(clk50m:in std_logic;
     clk:in std_logic;
     start:in std_logic;
     k1,k2,k3,k4,k5,k6,k7,k8:in std_logic;
     key:out std_logic_vector(2 downto 0));
end component;
--分频模块声明
component divider
   Port ( clk50m : in    STD_LOGIC;
                start:in STD_LOGIC;
          A : out    STD_LOGIC;
          B : out    STD_LOGIC;
```

```
                C: out    STD_LOGIC;
                D : out    STD_LOGIC;
                clk500:out STD_LOGIC;
                clk4:out std_logic;
                clk6m:out std_logic);
end component;
--时钟选择模块声明
component clk_sel
port(clk50m:in std_logic;
        start:in std_logic;
        A,B,C,D:in std_logic;
        en_A,en_B,en_C,en_D:in std_logic;
        leda,ledb,ledc,ledd:out std_logic;
        clkout:out std_logic);
end component;
--随机数产生模块声明
component random
      Port ( clk : in    STD_LOGIC;
                    start:in STD_LOGIC;
                    en:in STD_LOGIC;
                led : out    STD_LOGIC_VECTOR (2 downto 0));
end component;
--匹配模块声明
component compare
port(      start:in std_logic;
           clear:in std_logic;
           en:in std_logic;
           clk : in    STD_LOGIC;
        key : in    STD_LOGIC_VECTOR (2 downto 0);
        led : in    STD_LOGIC_VECTOR (2 downto 0);
           win,lose:out STD_LOGIC;
        en_A : out    STD_LOGIC;
        en_B : out    STD_LOGIC;
        en_C : out    STD_LOGIC;
        en_D : out    STD_LOGIC;
           highscore_out:out STD_LOGIC_VECTOR(11 downto 0);
        score_out : out    STD_LOGIC_VECTOR (11 downto 0);
        life_out : out    STD_LOGIC_VECTOR (7 downto 0));
end component;
--音乐产生模块声明
component music
port(clk6m,clk:in std_logic;
        en_A,en_B,en_C,en_D:in std_logic;
        sound:out std_logic);
```

```
end component;
--ＬＣＤ显示模块声明
component lcd
port(       clk500:in std_logic;
            start : in    STD_LOGIC;
            win,lose:in std_logic;
            highscore:in STD_LOGIC_VECTOR(11 downto 0);
            score: in    STD_LOGIC_VECTOR (11 downto 0);
          life: in    STD_LOGIC_VECTOR (7 downto 0);
          LCD_RS : out    STD_LOGIC;
          LCD_RW : out    STD_LOGIC;
          LCD_EN : out    STD_LOGIC;
          data : out    STD_LOGIC_VECTOR (3 downto 0));  --数据信号输出
end component;
-VGA 显示模块
component vga
port(
        start:in std_logic;              --开始信号，高电平有效
          en:in STD_LOGIC;                        --暂停信号，en=0 时暂停
        hs : out std_logic;                                    --屏幕行扫描
        vs : out std_logic;                                   --屏幕场扫描
        red : out std_logic;                                   --显示红色输出
        green : out std_logic;                                 --显示绿色输出
        blue : out std_logic;                                   --显示蓝色输出
          led : in    STD_LOGIC_VECTOR (2 downto 0);
          key : in    STD_LOGIC_VECTOR (2 downto 0);
          clk50m:in std_logic;
          win,lose: in std_logic

);
end component;
begin
win<=mwin;
lose<=mlose;
led<=mled;
u0:code port map(clk50m,clkout,start,k1,k2,k3,k4,k5,k6,k7,k8,key);
u1:divider port map(clk50m,start,A,B,C,D,clk500,clk4,clk6m);
u2:clk_sel port map(clk50m,start,A,B,C,D,en_A,en_B,en_C,en_D,leda,ledb,ledc,ledd,clkout);
u3:random port map(clkout,start,en,mled);
u4:compare port
map(start,clear,en,clkout,key,mled,mwin,mlose,en_A,en_B,en_C,en_D,highscore,score,life);
u5:music port map(clk6m,clk4,en_A,en_B,en_C,en_D,sound);
u6:lcd port
map(clk500,start,mwin,mlose,highscore,score,life,LCD_RS,LCD_RW,LCD_EN,data);
```

```
end Behavioral;
```