## 一、项目介绍

本项目参考了经典格斗游戏拳皇，在FPGA中实现了拳皇中的八神庵和草稚京两个角色，在游戏性上也实现了拳皇的简化版的操作。

在项目根目录下包括以下一些文件和文件夹：

buildtxt文件夹下是一个visual studio2012的c++代码工程，需要在win7 64位系统下运行，负责处理素材生成二进制文件data，由于需要读取bmp文件，所以需要使用opencv库，版本2.4.10，但opencv太大没有放进来；另外在该工程中有27张图片素材，通过该工程转成二进制文件写入sram。

project文件夹下是拳皇的FPGA工程代码，其中的output\_files下有编译好的sof和pof文件。

data和pre.pdf分别是写入sram的二进制文件和展示文件，还有一个就是本文档是报告文件。

## 二、总体设计思路

## 三、关键技术分析

#### 1. sram读取

sram读取必须满足时序要求，先设置模式为读取，每次给定一个21位的地址，其中最高两位为0，以25MHz的频率在一个周期之后读取上一个周期给定的地址的值，这里需要延迟计算，并不像软件代码直接获取数据。

#### 2. 颜色存储

sram一个地址对应了32bit的数据，采用8bit保存一个像素颜色的存储方式，稍微牺牲准确度，大幅提高了2M内存的利用率，完整保存下了所有素材（1.92M），另外还降低了组合逻辑的延迟，因为四个像素代码中是乘除4。

具体来说，rgb颜色本来需要各3bit，加上透明需要1bit，舍弃蓝色最低位并用特殊颜色代替透明就可以用8bit保存一个像素颜色。

#### 3. 一次处理40个像素颜色

## 四、下载验证方法

图片的下载：使用RLab把二进制数据文件data下载到SRAM。

使用Quartus 13.0打开king.qpf文件，直接编译然后下载，由于时序还存在些问题不保证每次下载都正常，可能要多次下载。

演示说明

输入输出使用PS/2键盘和VGA显示器。

进入开始界面后按回车进入游戏界面

P1的方向控制为WASD，拳为H，踢为J，防御为K

P2的方向控制为小键盘上下左右，拳为小键盘1，踢为小键盘2，防御为小键盘3

## 五、实验中遇到的问题及解决方法