

专用集成电路设计方法 课程报告

姓名:林山青 学号:16307130142

设计目标

使用FPGA设计游戏俄罗斯方块,实现最基本的功能,包括:

- 方块的旋转和移动
- 游戏逻辑与进程
- 分数的记录
- VGA显示信号
- 游戏声音

为了方便起见,硬件描述语言使用SystemVerilog.

快速测试

本项目支持使用Verilator进行软件层面的仿真,前提是你的系统中有Verilator仿真器.

```
$ make verilator
```

如果需要对所有的模块进行VCS仿真,则可以:

```
$ make vcs
```

板子的测试环境为ZedBoard,如果需要生成Vivado位流文件,则可以:

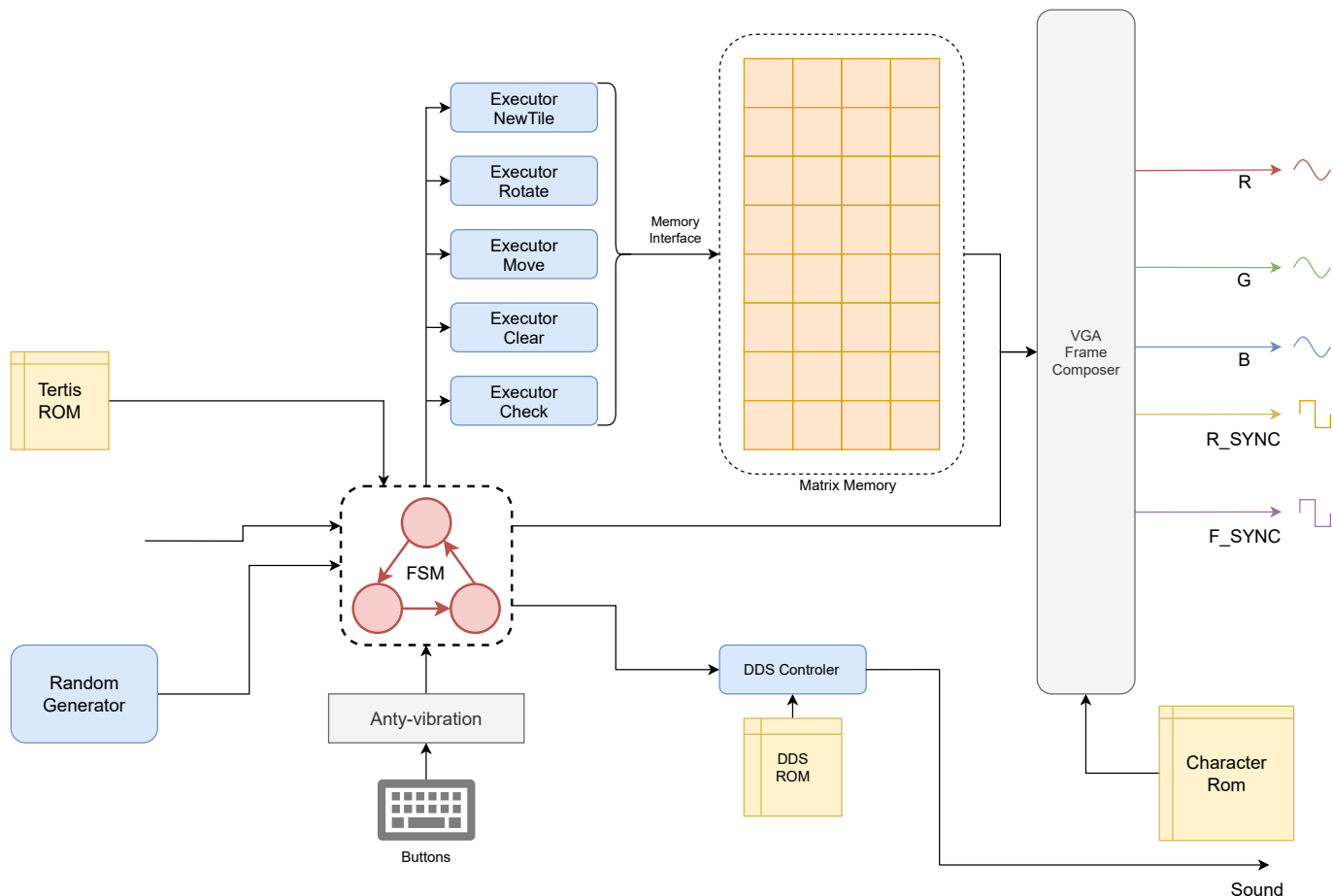
```
$ make vivado
```

系统框图

系统本身的显示部分的大类是由两部分组成:

1. 后端,包括游戏的逻辑控制,游戏状态的更新.主要包括的模块有状态机FSM和更新游戏状态的执行单元Executor.
2. 前端,主要是处理输入输出功能,包括键盘的防抖动处理,VGA控制器以及用于产生声音的DDS模块.

整个系统的框架图如下,后面会对每一个模块的设计方式单独讨论:

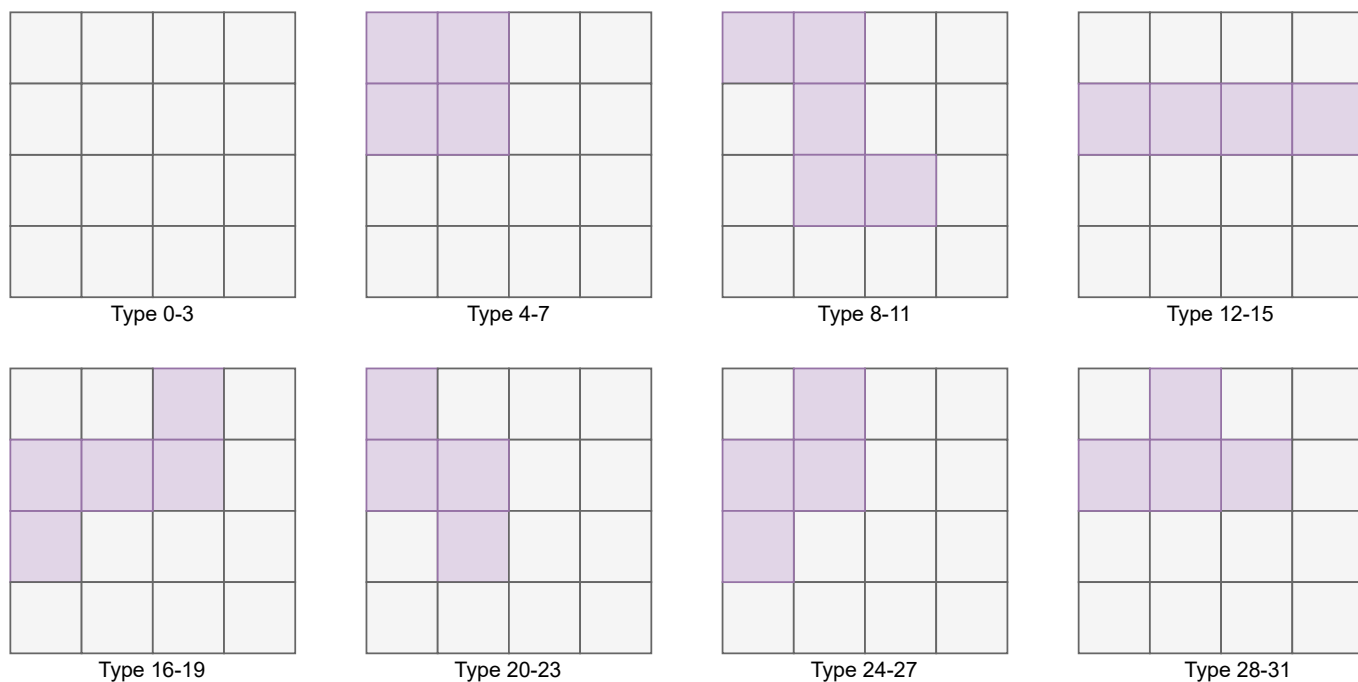


ROM模块说明

本小节对两个ROM进行说明,包括定义方式以及内存中结果的数据格式。

Tetris ROM

这个ROM的作用是存放了俄罗斯方块所有的方块信息,包括起始位置和终止位置,以方便进行后续的操作。首先需要对块进行编码。俄罗斯方块一共有七种,其中最大的块长度为4,经过旋转之后的块长度也为4。所以统一使用 4×4 的内存区域进行编码,如下图所示:



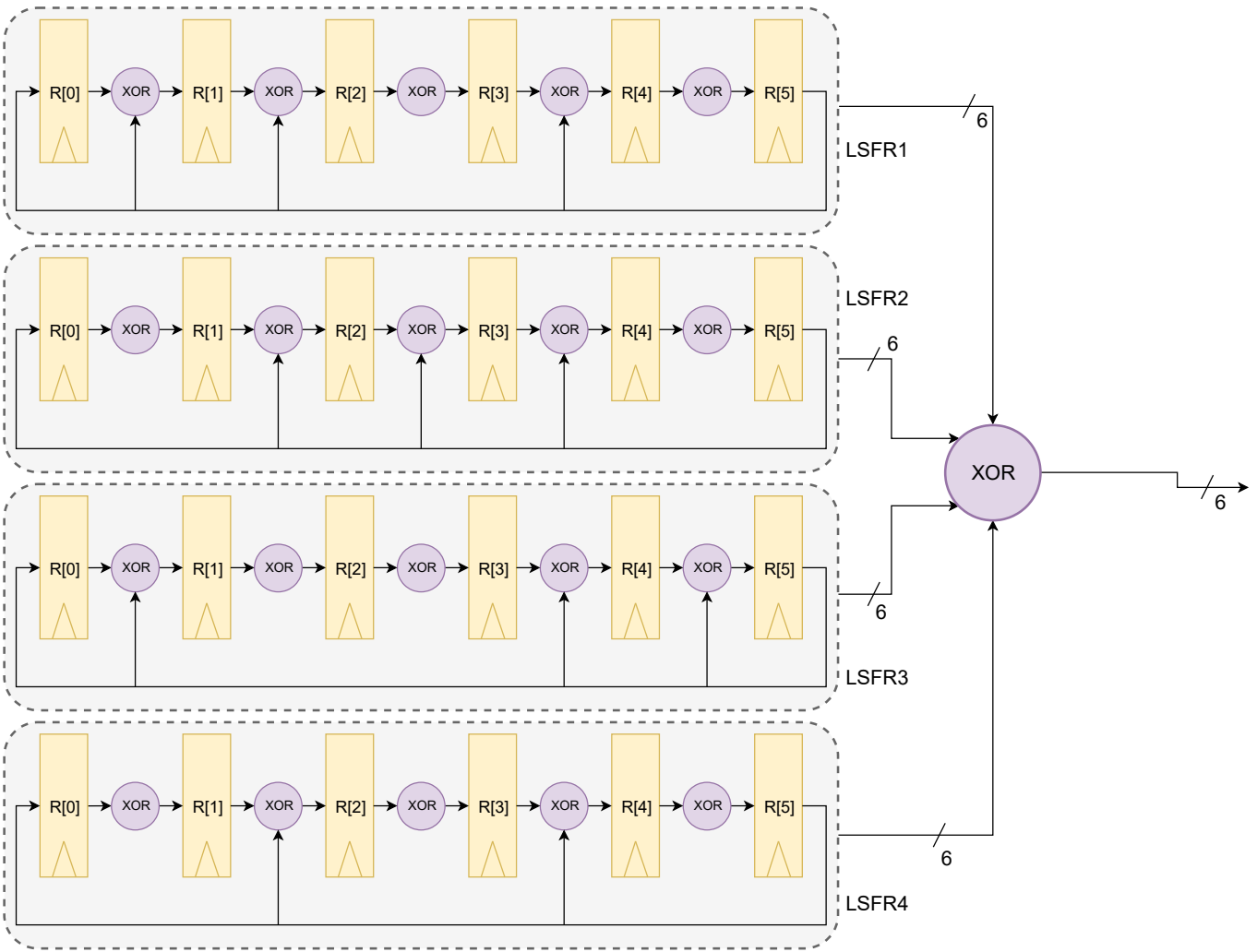
其中0号块为空块,用于初始化当前状态内存,其余的块则按照两处的位置

行为模块说明

接下来分类对不同的逻辑模块的行为进行讨论.本部分只给出了每个模块的框图和行为,暂时不给出对应实现的SystemVerilog代码.如需要参见代码的设计,请参考后续的内容.

Random Generator

随机数生成器采用最简单的线性反馈移位寄存器(Linear Feedback Shift Register),由于单组LFSR产生的随机程度不够大,这里使用四个不同的LSFR的输出结果的xor结果.下图给出了一种可能的配置结构:



随机数的生成实际上是冗余的.比如如果需要8位二进制数,最后的结果可能要求生成32位二进制数,然后再进行折叠xor,最终确保比较好的随机性.

Matrix Memory

矩阵内存本质上就是有两个Read口和一个Write口的SRAM阵列.为了方便对齐,矩阵的宽度有16个,矩阵的高度有32个.

Anti-Vibration

消除抖动的解决方法无非有两种:

1. 加长采样时间,让按钮在下一个上升沿来临是达到稳定.
2. 取多个采样周期的与(如果是低电平有效则取或),如果出现抖动则不会出现结果.

其实两种操作的本质十分接近,无非都是增加采样的时间.这里使用的是第一种方法,即使用一个25Hz(40ms)的时钟去采集输入电平.同时由于内部电路全部使用高电平有效的信号,这里还需要对输入信号进行一个转换.

Current Tile Memory

这个模块则用于记录当前要操作的块的属性,包括:

1. 块的左上角在Matrix Memory坐标位置,横坐标取值可能取为 $[-1,13]$,纵坐标取值可能为 $[-3,30]$.
2. 当前块的信息,包括块的形状,块在内存中的id,块的最小xy坐标值的点和块最大xy坐标的点.主要用于移动时边界的判断.
3. 当前块的下一个旋转块的信息,主要用于能否旋转的判断.

模块代码说明

本部分介绍了模块对应的实现代码的重要思想.对于具体的代码,请参考源文件.

Style

为了增强可读性,所有的代码具有这样的格式:

Declaration

为了方便代码的实现,在include目录下的tetris.sv中声明了以下数据类型:

Random Generator

Matrix Memory

Anti-Vibration

Testbench