

Lab2 实验说明

本实验分为两个部分，第一部分为序列检测电路代码示例以及新功能的实现，第二部分为移位取余的设计。下面开始分部分介绍。

一 序列检测示例代码

1.1 实现功能

随机序列中 xxx101101xxx 检测序列“101101”（左先），每检测到一个序列输出一个周期的脉冲，输入:clk,rstn,data_in, 输出:detector, 仅检测不重叠的 101101 序列。

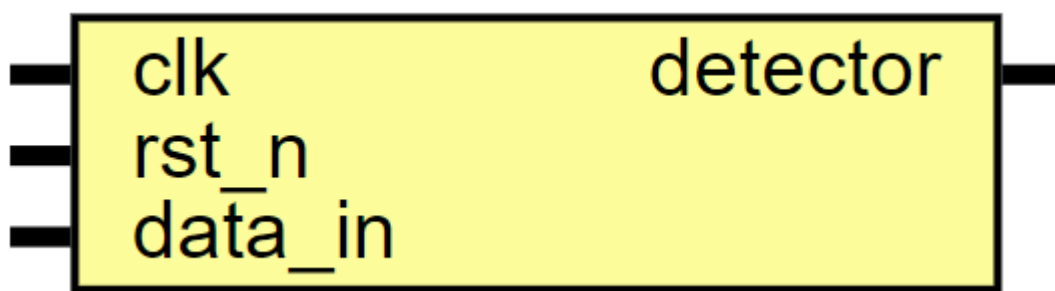


图 1. 系统端口图

1.2 状态机设计流程

首先确定状态编码，通常采用格雷码格式进行状态机的编码设定，完成状态转换图，根据状态转换图进行代码实现。

整体流程：确定状态机的基本行为；建立状态、输出表；优化状态机状态数目（optional）；给状态机编码；以状态、输出表为基础建立转换、输出表（图）；输出表达式；完成完整的状态机设计。

典型的状态机分为两段式和三段式。

两段式：用两个 always 块来描述状态机，一个 always 块用来描述新状态的产生（组合逻辑），一个 always 块用来描述状态迁移（时序逻辑）。

三段式：状态机第一段，时序逻辑，非阻塞赋值，传递寄存器的状态。状态

机第二段，组合逻辑，阻塞赋值，根据当前状态和当前输入，确定下一个状态机的状态。状态机第三段，可以采用阻塞赋值或者非阻塞赋值，取决于设计需求。

1.3 Testbench 部分

获取对应的仿真波形。

1.4 原有功能修改

将原示例代码中的不重叠序列检测改为重叠序列检测。

二 求余计算

2.1 实验介绍

现有 16 位寄存器。初始值为 0。每个时钟周期寄存器的值会左移 1 位，并且将输入的数据 `data_in` 作为寄存器的最低位，寄存器原来的最高位将被丢弃。要求每个周期实时输出该 16 位寄存器对 7 求余的余数 `data_out[2:0]`。

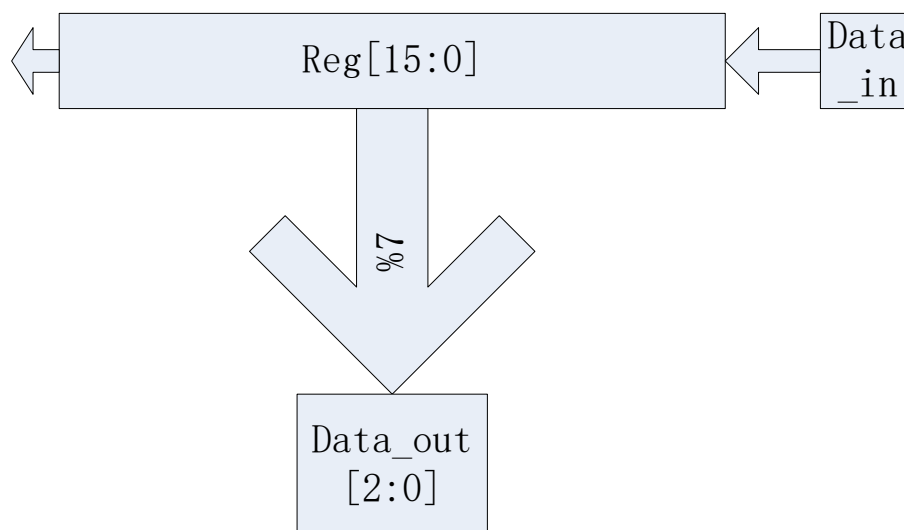


图 2. 实现功能示例

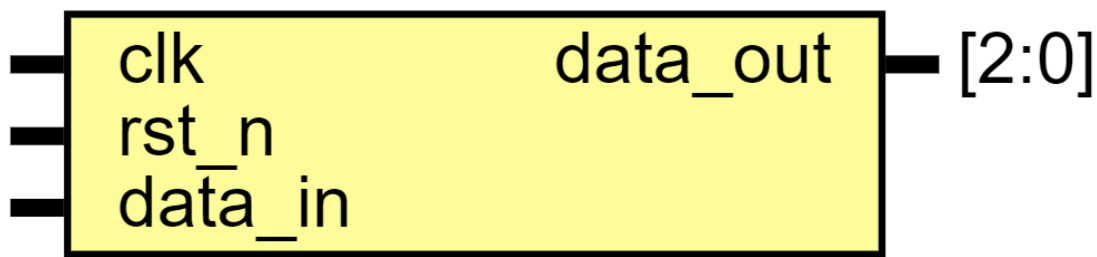


图 3. 端口示例

2.2 Testbench 设计

自行完成 testbench，需包含所有可能的情况。

三 实验报告要求

报告中需含有第一部分示例代码的仿真波形，以及 testbench 的设计思路，以及重叠序列检测的修改部分与仿真结果图。第二部分需要附上 verilog 源代码(包含适当注释)，testbench 代码以及仿真波形图。需要在报告中说明自己使用的是三段式还是两段式状态机。实验报告将是实验成绩的主要依据，请尽量做到书写规范。

实验报告递交请压缩成一个 zip 文件，文件名为 LABn_学号_姓名，如 LAB1_5123456789_李雷)，压缩包须包括：

- 1) verilog 代码
- 2) testbench 代码
- 3) 实验报告
- 4) 波形图

实验报告提交到 Canvas 上，截止日期：5 月 12 号 23: 59.