

**本科实验报告**

|  |  |
| --- | --- |
| 课程名称： | 计算机逻辑电路设计 |
| 姓 名： | 汪珉凯 |
| 学 院： | 竺可桢学院 |
| 专 业： | 计算机科学与技术 |
| 指导教师： | 洪奇军 |
| 报告日期： | 2023年 12 月 21 日 |

**浙江大学实验报告**

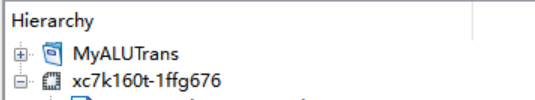
课程名称： 计算机逻辑设计基础 实验类型： 综合

实验项目名称： 寄存器和寄存器传输设计

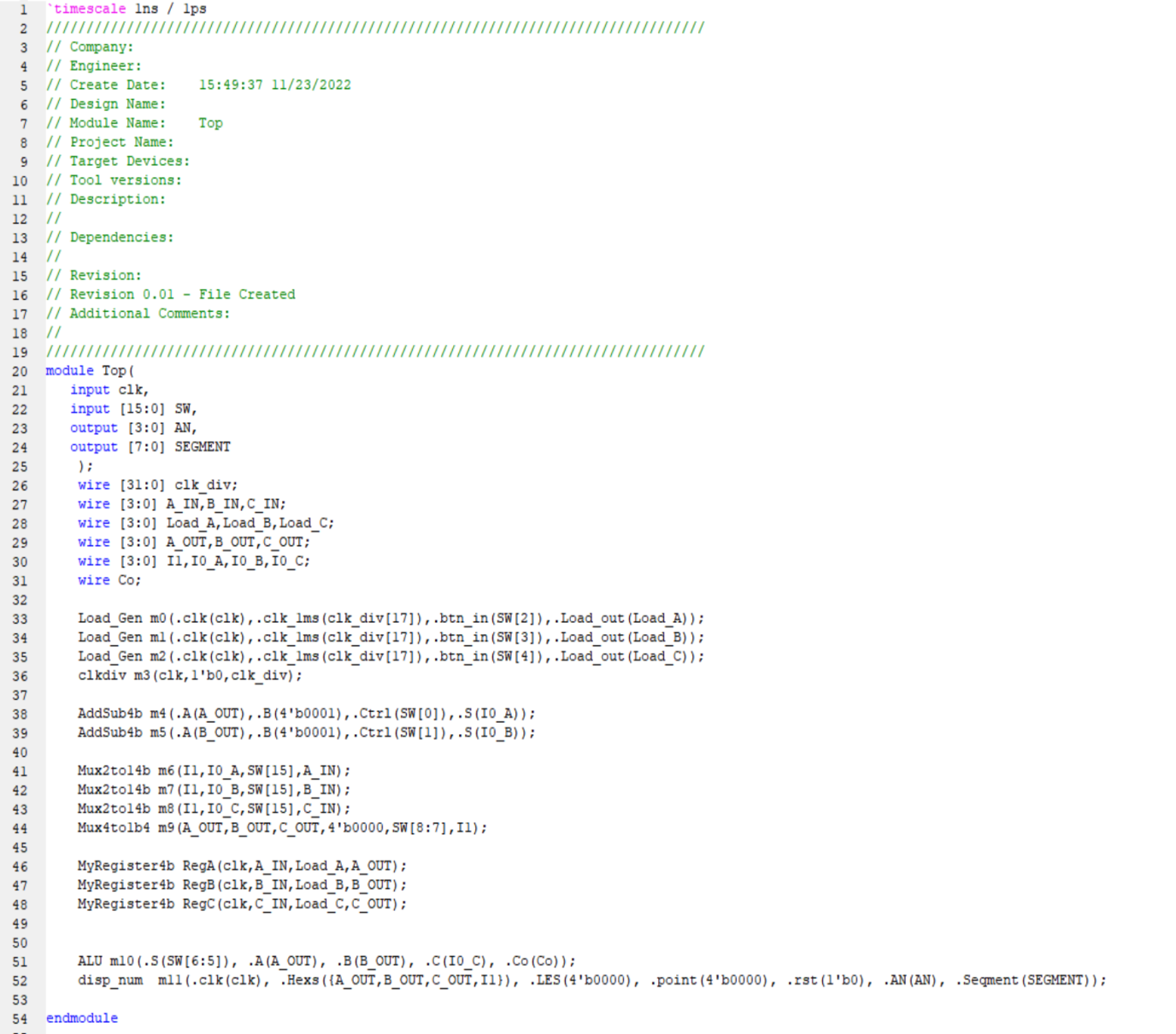
学生姓名： 汪珉凯 学号：3220100975 同组学生姓名： 岳晟稼

实验地点： 紫金港东四509室 实验日期： 2023 年 11 月 30 日

1. **操作方法与实验步骤**
   1. **建立楼道控制的工程：MyALUTrans**

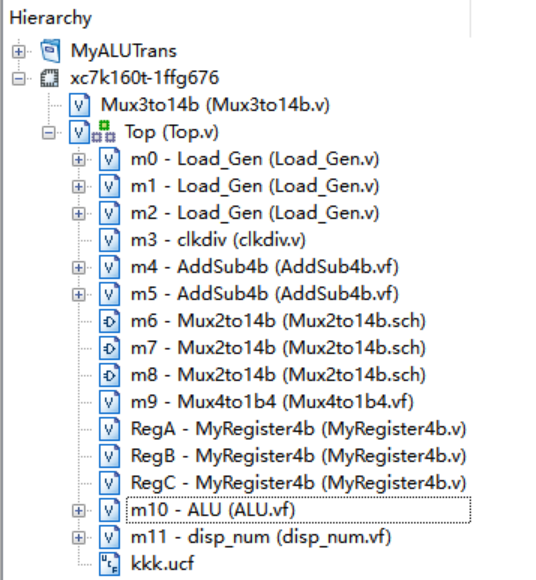


* 1. **新建源文件，类型是Verilog，命名为TOP.v 其代码如下：**

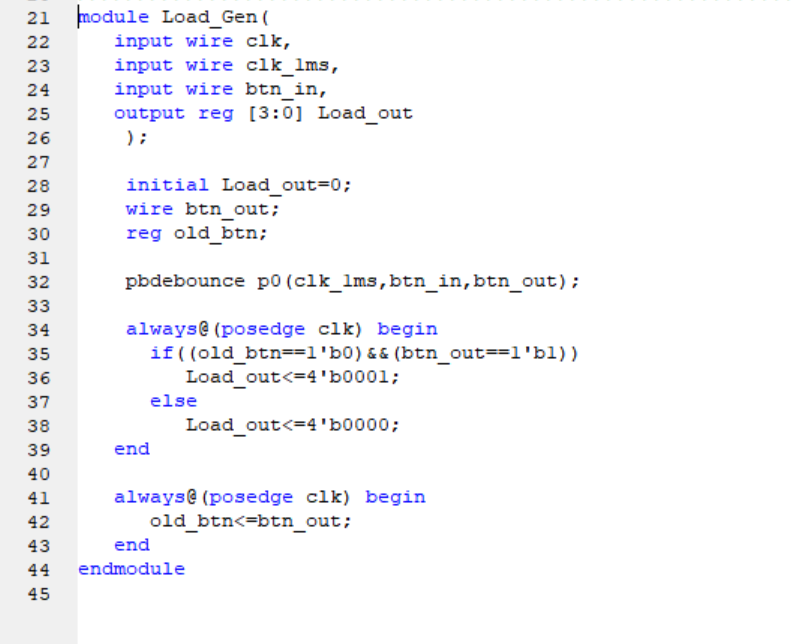


* 1. **添加调用以下模块，并补全调用模块中需要用到的其他模块。**

1. **ALU模块（2）4位4选1模块、4位2选1模块（3）防抖动模块（4）显示模块**

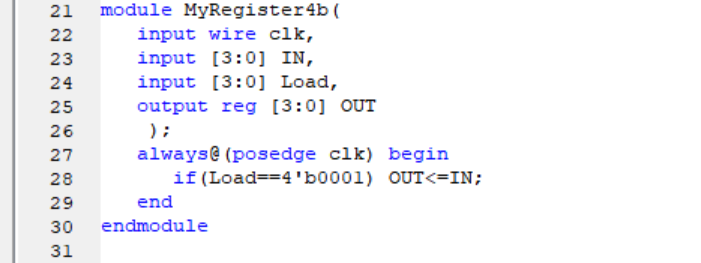


**1.4新建源文件，类型为Verilog,命名为Load\_Gen补充顶层模块中的一个函数。**该模块主要实现的功能是：检测按钮的按下状态，并在检测到按钮按下时将 Load\_out 设置为特定的值。按钮状态的稳定性可能通过 pbdebounce 模块来实现，以避免按钮抖动引起的误触发。其代码如下：



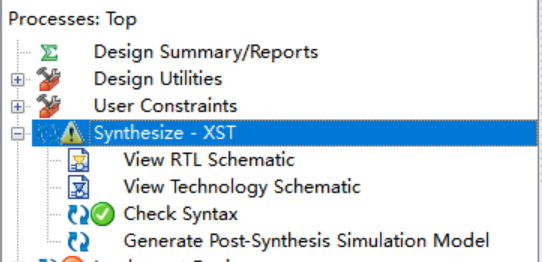
**1.5新建源文件 类型为Verilog,命名为MyRegister4b补充另一个函数。**

模块主要实现的功能是：当检测到时钟上升沿时，判断load信号。若其为1，加载信号有效，将输入值写入寄存器。在其他情况下，寄存器的值将保持不变。其代码如下：



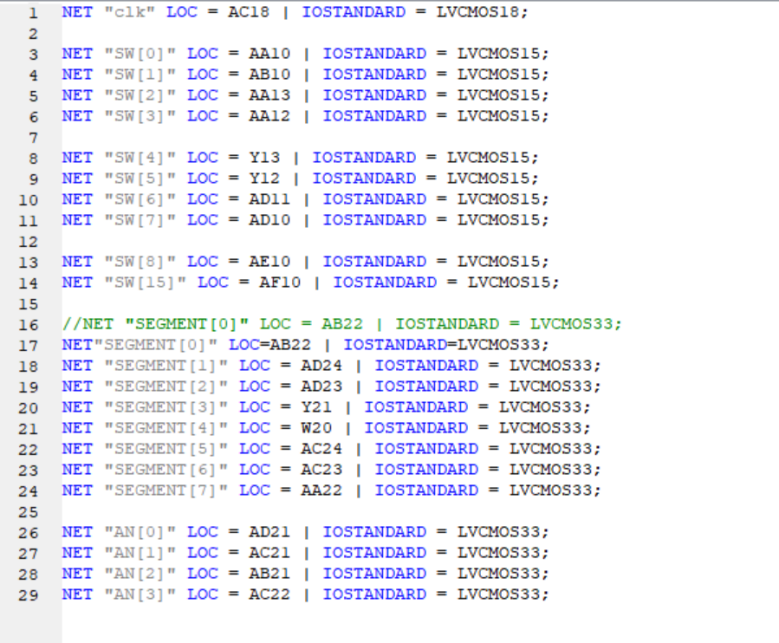
**1.6编译**

依次对该 schematic 文件运行 Check Design Rules、Synthesize。



**1.7引脚约束**

引脚约束如下：



1. **实验结果与分析**

验证：当开关15为1时，功能是将AB运算后的结果储存到C中。

具体运算（加减与或）由开关6、7决定，开关5的作用是刷新C值，开关3、4的作用分别是控制A/B的自增/自减，开关1、2的作用是具体选择加或者减。

|  |  |
| --- | --- |
| 开关67为11，执行 ‘加’ 功能 | 开关67为10，执行 ‘减’ 功能 |
| c99d51b3d2d63ccd39541d34b2ac664 | a1e079c606f82c691f535caf2129dd4 |
| 开关67为01，执行 ‘与’ 功能 | 开关67为00，执行 ‘或’ 功能 |
| 6ef7e3bf8cc2ae50d5244960ce96c65 | 5a7d4b25626e3409e0a0bf6fffd75ea |

验证：当开关15为0时，开关8、9的功能是将特定值存入寄存器中。

|  |
| --- |
| 开关89为00时，将A存入寄存器中。 |
| 58f981b5991d61412839c2143568376 |
| 开关89为10时，将B存入寄存器中。 |
| 5a7d4b25626e3409e0a0bf6fffd75ea |
| 开关89为为01时，将C存入寄存器中。 |
| f2806f2169e98b641d7636088759fdb |
| 开关89为为11时，将0值存入寄存器中。 |
| d8918847f9423ad42f6f6af3152db54 |

验证：当开关15为0时，功能是将寄存器中的数值储存到A/B/C的其中一位中（具体储存到哪一位由开关3、4、5决定。

|  |
| --- |
| 开关15、3、5均为0，开关4为1时：将寄存器中的数存到B中，符合预期： |
| 33992ccd24d94c4f93c4621b6b9f508 |
| 开关15、3、4均为0，开关5为1时：将寄存器中的数存到C中，符合预期： |
| 818d153d78e0ef6ed765b69e55ec664 |

1. **讨论、心得**

本次实验的难点在于top模块的编写。由于需要调用大量原先已经调试完成的模块。由于我对原先模块的功能并不是理解的特别透彻，我无法非常灵活地使用它们，有时候需要在同学的帮助下去完成，包括本次实验我自己设计的register模块和load\_gen模块。以后还是需要多多理解。

**浙江大学实验报告**

课程名称： 计算机逻辑设计基础 实验类型： 综合

实验项目名称： 加法器、加减法器和ALU基本原理与设计

学生姓名： 汪珉凯 学号： 3220100975 同组学生姓名： 岳晟稼

实验地点： 紫金港东四509室 实验日期： 2023 年 12 月 7 日

# 实验目的和要求

1. 掌握同步四位二进制计数器74LS161的工作原理和设计方法
2. 掌握时钟/定时器的工作原理于设计方法

# 实验内容和原理

## 2.1实验设备

* 1. 装有Xilinx ISE 14.7的计算机 1台
  2. SWORD开发板 1套

## 2.2实验内容

任务1：采用行为描述设计同步四位二进制计数器74LS161

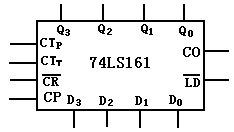
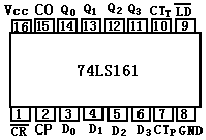
任务2：基于74LS161设计时钟应用

## 2.3实验原理

**2.3.1 同步思维二进制计数器74LS161**

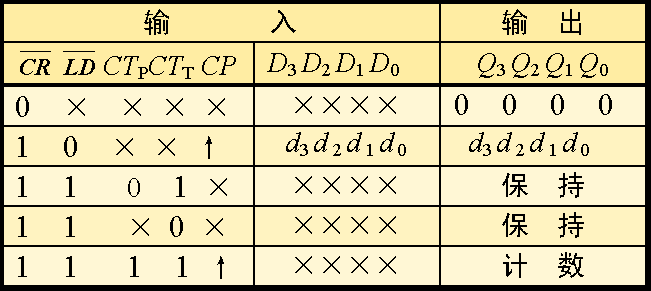
**功能描述：**

74LS161是常用的四位二进制可预置的同步加法计数器，克林或运用在各种数字电路，实现分频器灯很多重要的功能。



清零端，置数端,使能端,进位输出端CO

**功能表**

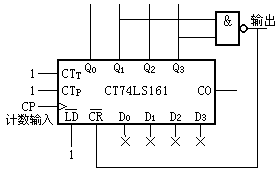


**74LS161时序图仿真结果**

****

**2.3.2 时钟应用设计**

**实现十进制计数器：**

****

**真值表**

**0 0 0 0 0**

**1 0 0 0 1**

**0 1 0 0 2**

**1 1 0 0 3**

**0 0 1 0 4**

**1 0 1 0 5**

**0 1 1 0 6**

**1 1 1 0 7**

**0 0 0 1 8**

**1 0 0 1 9**

**0 1 0 1 10->0**

利用与非门拾取状态1010

实现十进制计数（0000到1001）

改变与非门的输入信号，可以实现其他进制计数

**分钟60进制（十进制显示）**

****

# 三、实验过程和数据记录

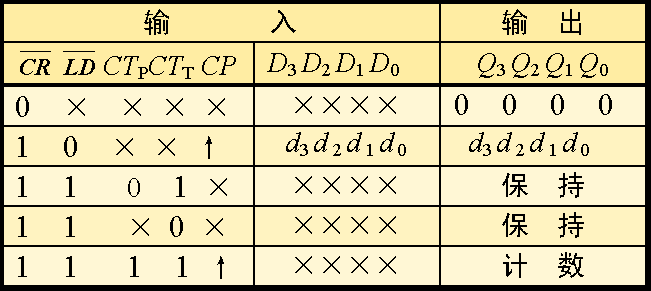
3.1实验内容：

数字时钟

* 设计一个数字钟，使用60进制和24进制计数器，实现24小时内时间的实时显示。
* 数字钟的初值通过初始化语句来实现，用数码管前两位显示小时的十位和个位，后两位显示分钟的十位和个位。

任务1： 采用行为描述设计同步四位二进制计数器74LS161

任务2： 基于74LS161设计时钟应用

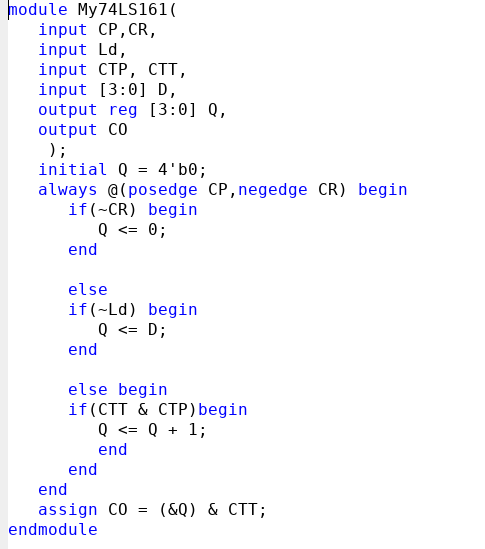


实验步骤：

1. 设计同步四位二进制计数器74LS161

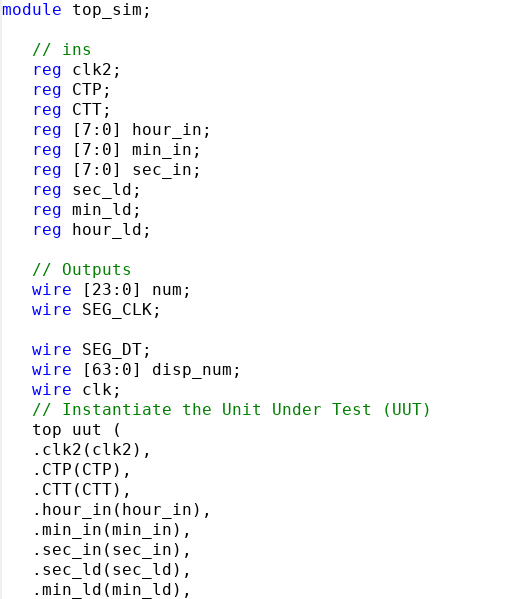
* **新建工程**
  + **工程名称用My74LS161。**
  + **Top Level Source Type用HDL**
* **用行为描述设计**
  + **CR是异步清零**
  + **LD是同步置位**

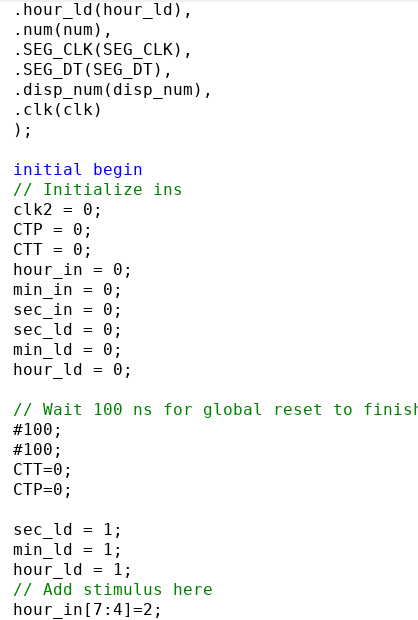
Verilog代码：

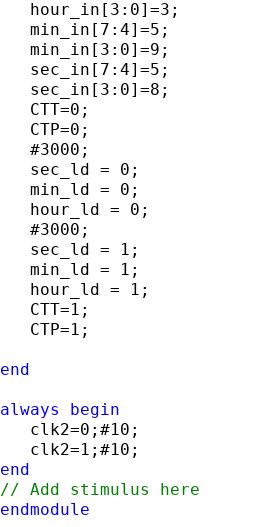


其中PS2模块如上实验所描述

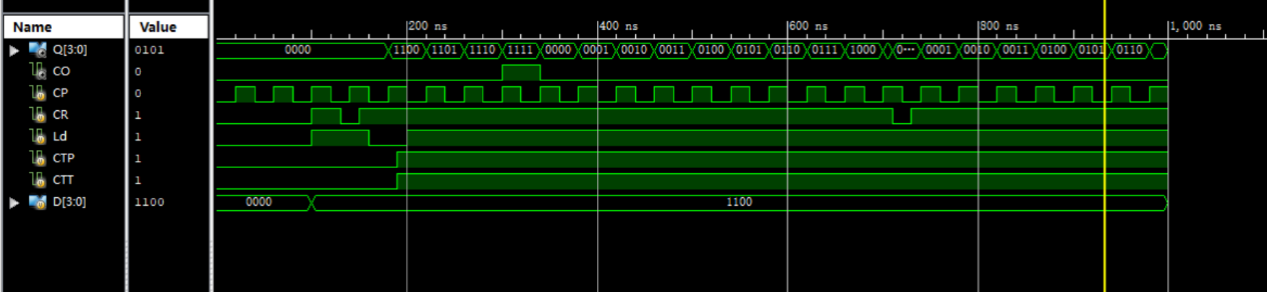
**仿真代码：**







**仿真图像：**



UCF定义：

* 输入
* 时钟: clk
* 按键控制输入: BTN11 控制 A，BTN[O] 控制 B，关联到 BTNX4Y[0:1]按键加/减方向控制: SW11] 控制 A，SW1(0] 控制 B，关联到 DSW[1:0ALU 运算控制:SW21:01,00-加，01-减，10-与，11-或，关联到 DSW115:14
* 输出
* 数码管[0]: A - num3:0
* 数码管[1]: B - num[7:4]
* 数码管 [2]: Co- Co
* 数码管3: C -C

BTNX4:按键使能输出

#七段码串行移位接口

NET "seg\_clk"        LOC = M24 | IOSTANDARD = LVCMOS33 ;

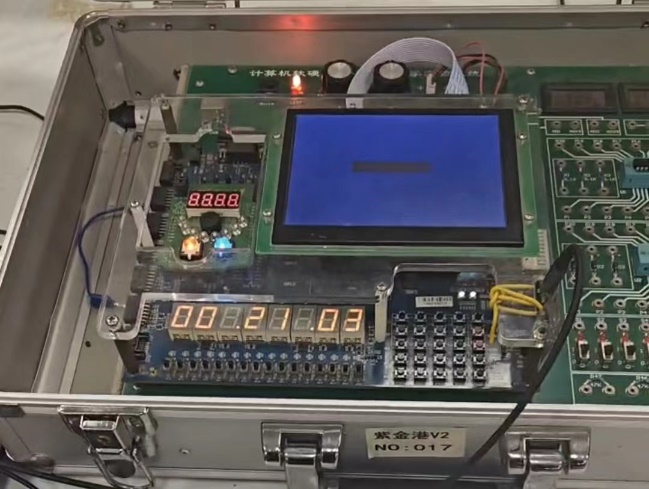
NET "seg\_clrn"       LOC = M20 | IOSTANDARD = LVCMOS33 ;

NET "seg\_sout"       LOC = L24 | IOSTANDARD = LVCMOS33 ;

NET "SEG\_PEN"        LOC = R18 | IOSTANDARD = LVCMOS33 ;

3.2实验结果：

时钟自增：



# 讨论心得：

本次实验的难点在于理解电路的复杂功能，但由于在实验前我们小组就已经做了相对充分的预习工作，因此总体来说实验难度不算太大。再接再厉！

**浙江大学实验报告**

课程名称： 计算机逻辑设计基础 实验类型： 综合

实验项目名称： 移位寄存器设计与应用

学生姓名： 汪珉凯 学号：3220100975 同组学生姓名： 岳晟稼

实验地点： 紫金港东四509室 实验日期： 2023 年 12 月 14 日

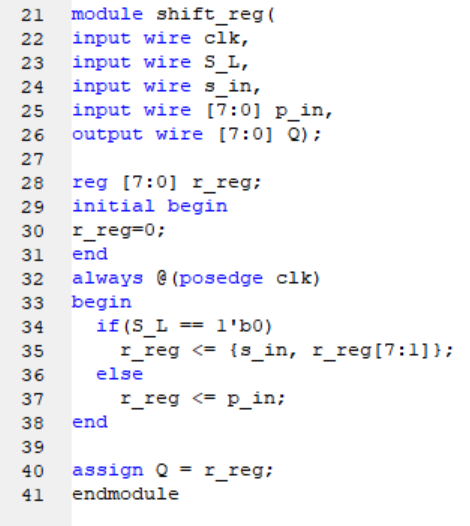
**已完成P2S功能实验！**

1. **操作方法与实验步骤**

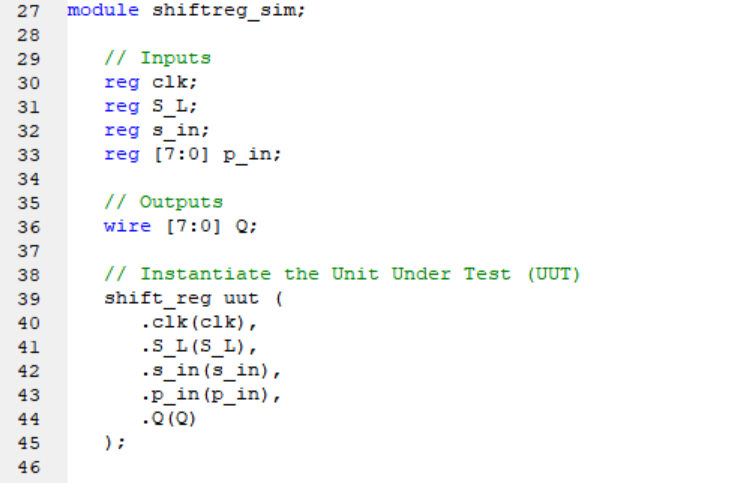
**1.设计8位带并行输入的右移移位寄存器**

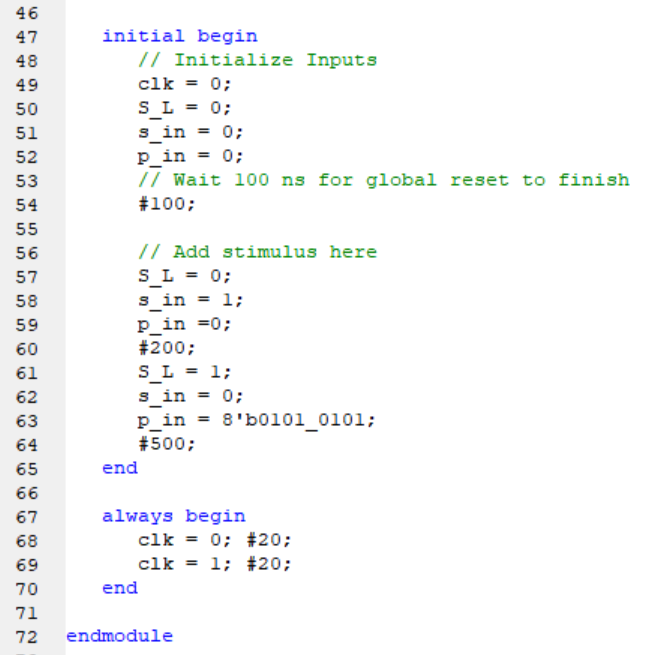
1.1新建工程ShfitReg8b，Top Level Source Type用HDL。

1.2 用结构化描述设计，代码如下：



1.3仿真仿真激励代码如下：





**2.设计跑马灯应用**

**2.1新建工程MyMarquee ，Top Level Source Type用HDL。**

**2.2调用以下模块：**

**（1）ShfitReg8b（2）显示模块（3）CreateNumber（4）分频模块**

**2.3新建源文件，格式为verilog，命名为top.v并设为顶层。**

**结构化设计顶层模块，**

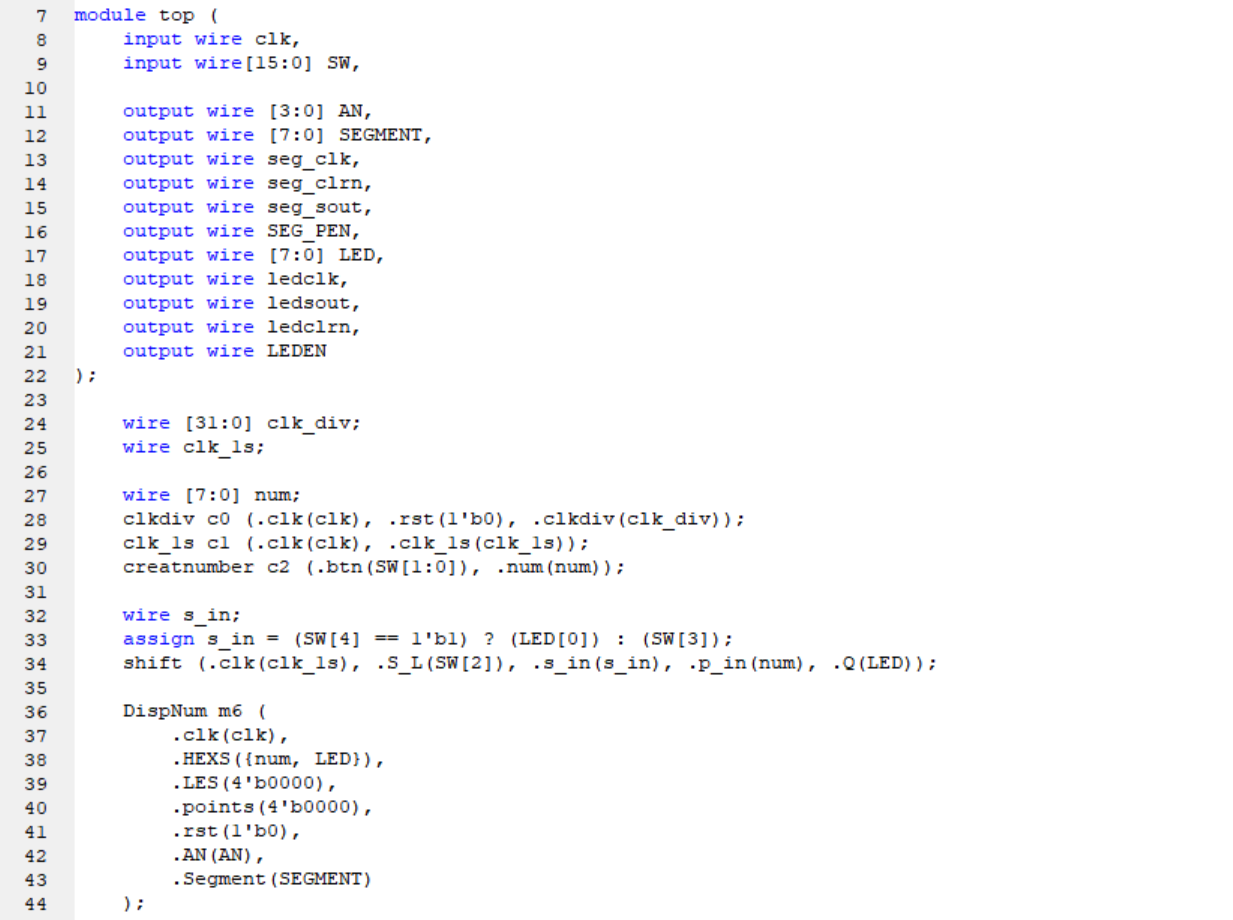
**联系加分项：**

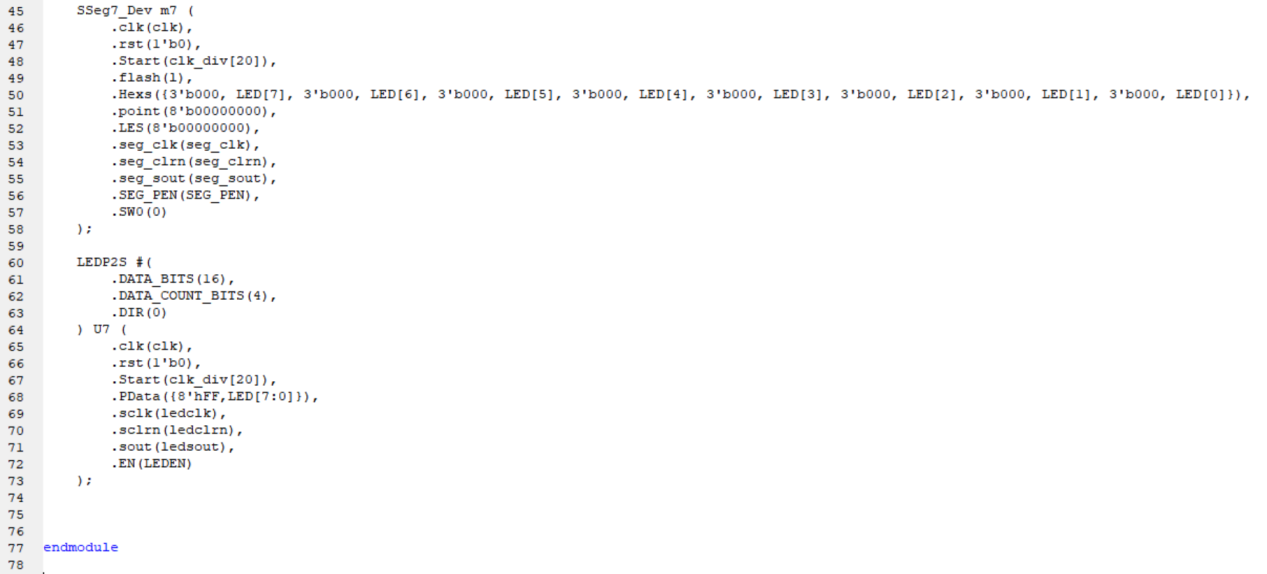
**任务一：设计8位带并行输入的右移位寄存器**

**任务二：设计主板LED灯驱动模块**

**任务三：设计主板七段数码管驱动模块**

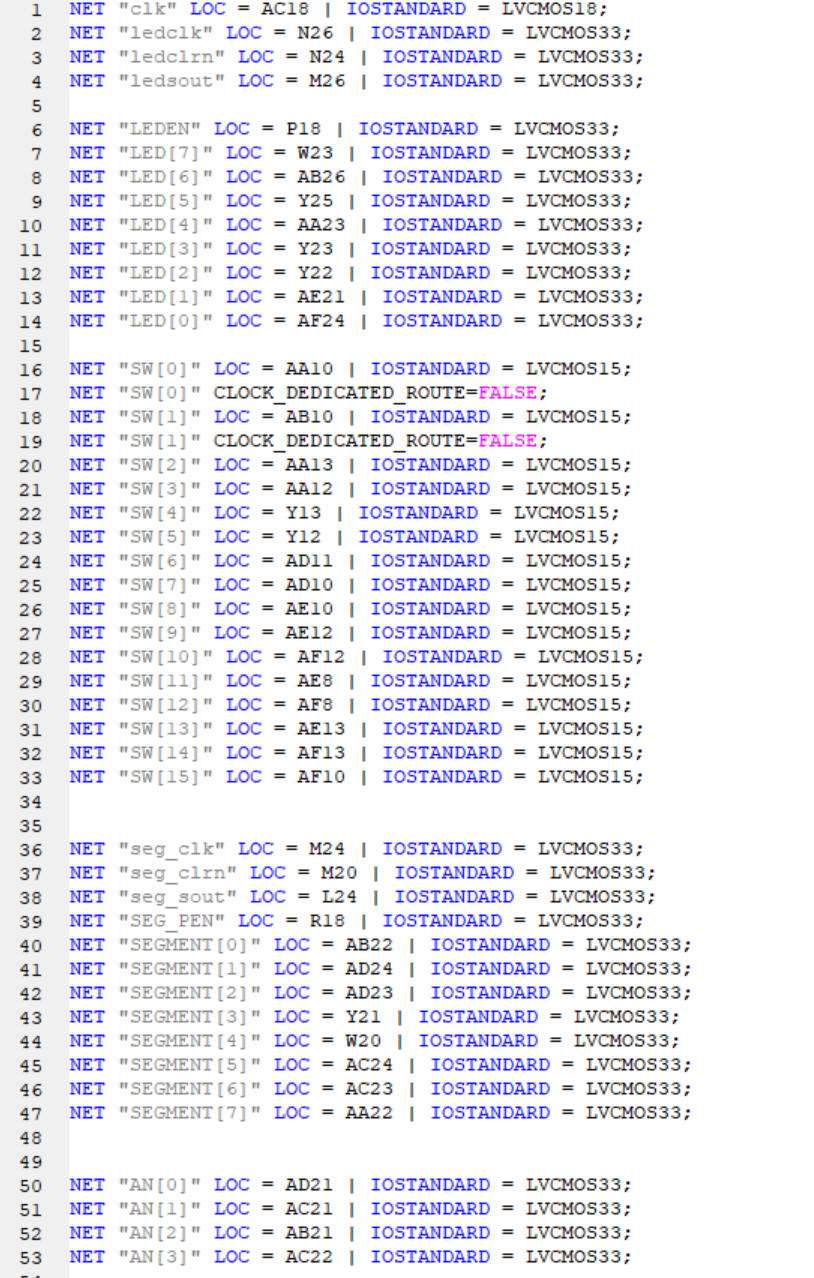
**代码如下：**





**2.4引脚约束**

**引脚约束代码如下：**



（由于后续设计主板七段数码管驱动模块的实验都放在一个工程里完成了，所以引脚约束里面有一些跑马灯模块本身不需要的代码。）

**2.5加分项实验内容和步骤**

任务一：设置八位带并行的右移移位寄存器

实验步骤：

新建工程

工程名称用ShiftReg8b

Top Level Source Type用HDL

用结构化描述

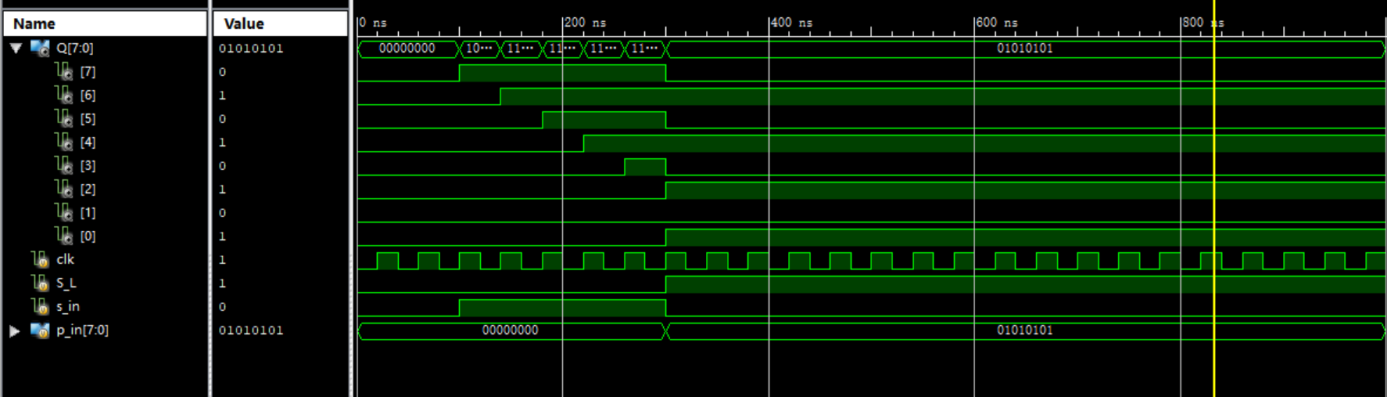
任务二：设计主板七段数码管驱动模块

实验步骤：

1. **实验结果与分析**

**（一)8位带并行输入的右移移位寄存器**

**仿真结果如下：**



S\_L=0时，每个周期右移一位。

S\_L=1或S\_L不变时，每个周期从外部读入相应数据。

综上，仿真结果符合预期。

**（二）设计跑马灯应用**

**上板结果如下：**

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |

由于当时拍的是视频，所以我们小组只能从视频中截取几个图片来表现跑马灯的变化。由以上图片可知，我们成功实现了跑马灯来回两期的功能。

1. **讨论、心得**

作为实验课的最后一个实验，跑马灯本身的设计并不复杂，只不过需要调用比较多的其他模块。在这门课上，我收获良多，对计算机的运行逻辑产生了更深的理解。希望以后有机会继续学习相关知识。青山不改、绿水长流，我们江湖再见。

1. **个人生活照片展示**

