**实验报告**

实验题目： 建模概念 日期：2018年 10 月 12 日

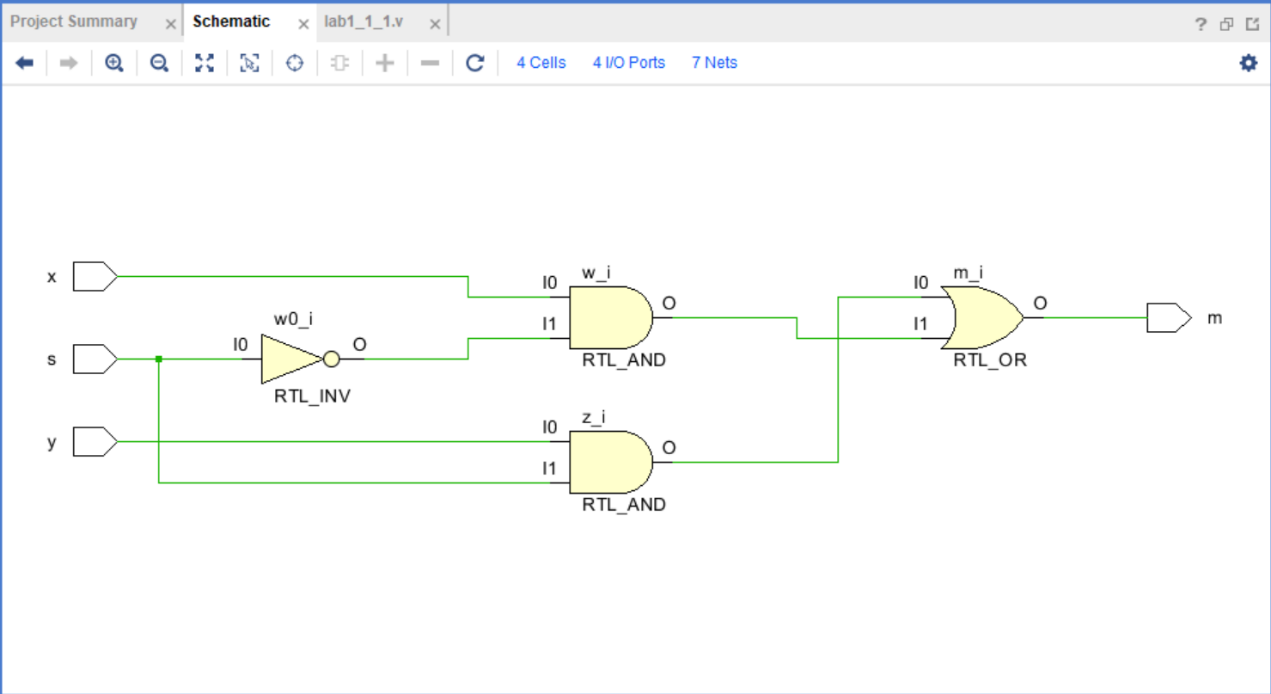
姓名: 罗晏宸 学号: PB17000297 成绩:

**实验目的：**

1. 使用门级、数据流级和行为级3种风格设计一位和多位组合逻辑电路
2. 设计模型读取拨码开关和按键开关并输出到LED和7段数码管
3. 仿真并理解设计的输出
4. 创建分层的设计
5. 综合，实现并生成比特流文件
6. 下载比特流文件到开发板，并验证功能

**实验内容**（截图、照片与代码）

**Lab1\_1\_1**



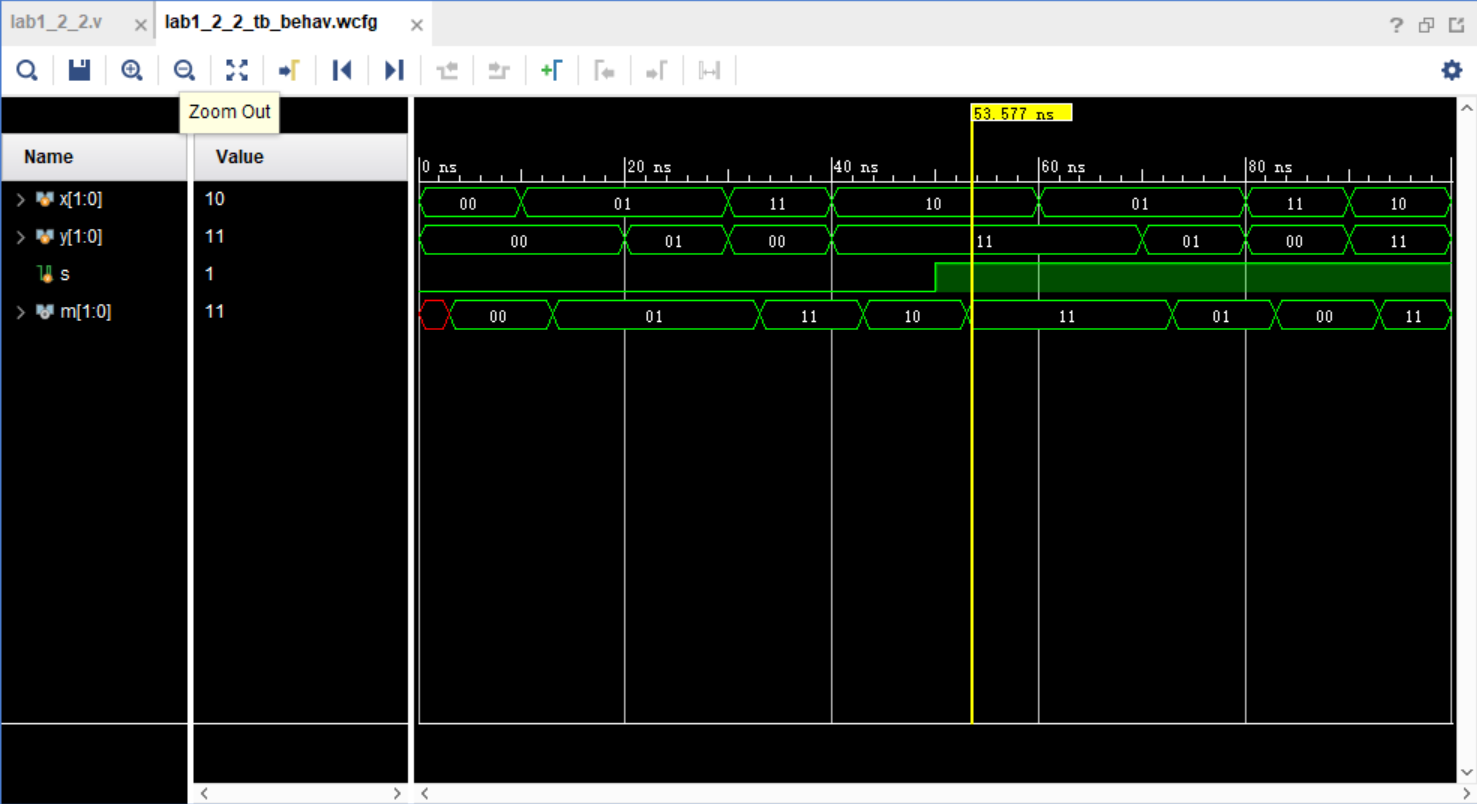
截图 1-lab1\_1\_1原理图



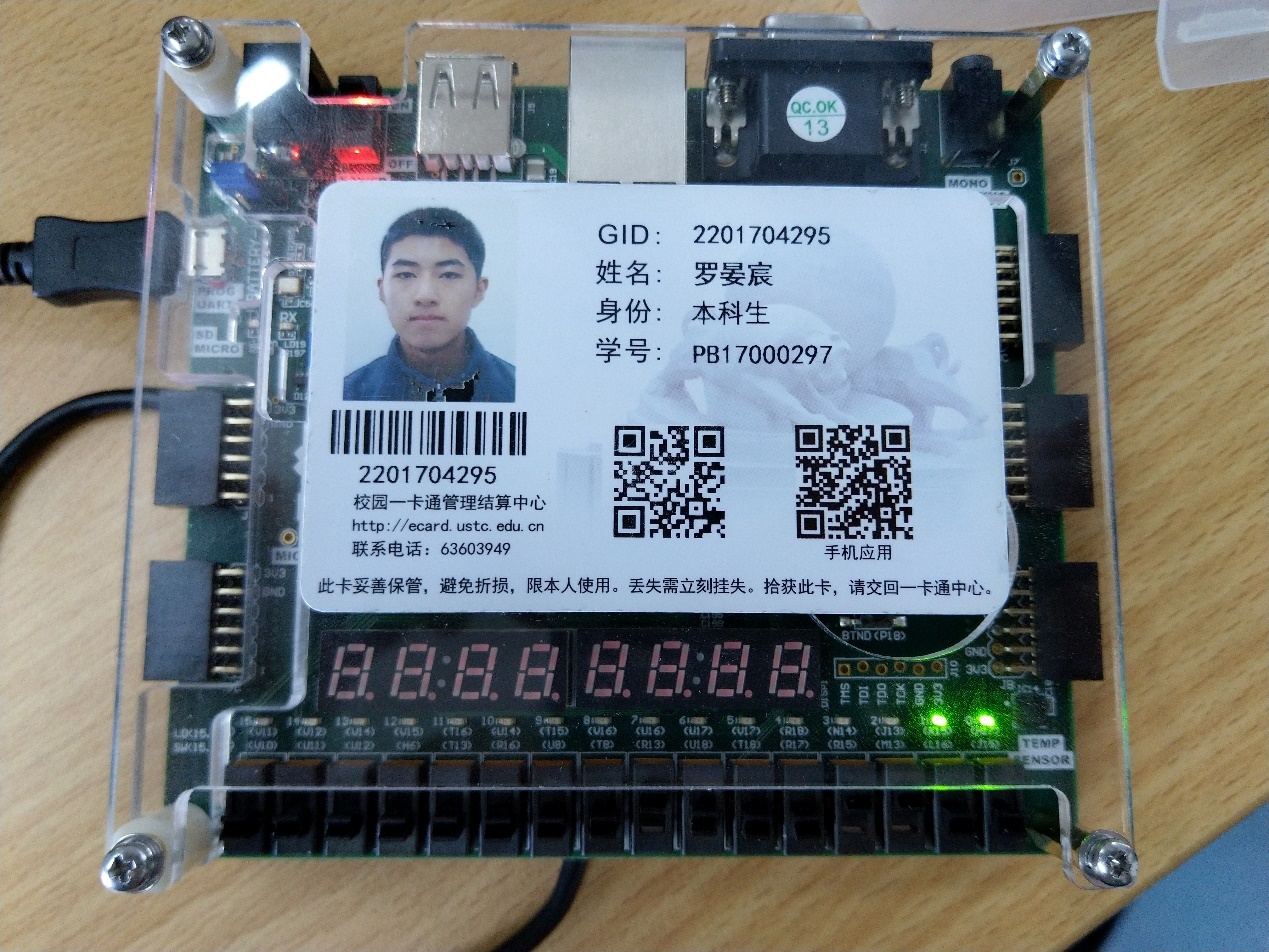
照片 1-lab1\_1\_1下载

“照片 1-lab1\_1\_1下载”的说明：本工程使用门级风格实现1位2选1数据选择器；图中开关switch[1]和switch[7]拨至开，其余开关拨至关，表示输入为x=0, y=1, s=1；灯led[0]亮，其余灯灭，表示输出为m=1即m=y.

**Lab1\_2\_2**



截图 2-lab1\_2\_2仿真图



照片 2-lab1\_2\_2下载

“照片 2-lab1\_2\_2下载”的说明：本工程使用数据流风格实现2位2选1数据选择器；图中开关switch[2], switch[3], switch[7]拨至开，其余开关拨至关，表示输入为x=(00)B, y=(11)B, s=1；灯led[0]和led[1]亮，其余灯灭，表示输出为m=(11)B即m=y.

**Lab1\_3\_2**

代码Lab1\_3\_2.v

`timescale 1ns / 1ps

//////////////////////////////////////////////////////////////////////////////////

// Company:

// Engineer:

//

// Create Date: 2018/10/11 21:26:12

// Design Name:

// Module Name: lab1\_3\_2

// Project Name:

// Target Devices:

// Tool Versions:

// Description:

//

// Dependencies:

//

// Revision:

// Revision 0.01 - File Created

// Additional Comments:

//

//////////////////////////////////////////////////////////////////////////////////

module lab1\_3\_2(

input [1:0] x,

input [1:0] y,

input s,

output reg [1:0] m

);

always @(x or y or s)

if (s == 0)

m = x;

else

m = y;

endmodule

**Lab1\_4\_2**

代码bcdto7segment\_dataflow.v

`timescale 1ns / 1ps

//////////////////////////////////////////////////////////////////////////////////

// Company:

// Engineer:

//

// Create Date: 2018/10/12 00:32:31

// Design Name:

// Module Name: bcdto7segment\_dataflow

// Project Name:

// Target Devices:

// Tool Versions:

// Description:

//

// Dependencies:

//

// Revision:

// Revision 0.01 - File Created

// Additional Comments:

//

//////////////////////////////////////////////////////////////////////////////////

module bcdto7segment\_dataflow(

input [3:0] x,

output [7:0] an,

output [6:0] seg

);

assign seg[0] = ((~x[3] && ~x[2] && ~x[1] && x[0]) | (~x[3] && x[2] && ~x[1] && ~x[0])) && ~(x[3] && (x[2] | x[1]));

assign seg[1] = (~x[3] && x[2] && ~x[1] && x[0]) | (~x[3] && x[2] && x[1] && ~x[0]) | (x[3] && (x[2] | x[1]));

assign seg[2] = (~x[3] && ~x[2] && x[1] && ~x[0]) | (x[3] && (x[2] | x[1]));

assign seg[3] = ((~x[3] && ~x[2] && ~x[1] && x[0]) | (~x[3] && x[2] && ~x[1] && ~x[0]) | (~x[3] && x[2] && x[1] && x[0])) && ~(x[3] && (x[2] | x[1]));

assign seg[4] = ((~x[3] && ~x[2] && ~x[1] && x[0]) | (~x[3] && ~x[2] && x[1] && x[0]) | (~x[3] && x[2] && ~x[1] && ~x[0]) | (~x[3] && x[2] && ~x[1] && x[0]) | (~x[3] && x[2] && x[1] && x[0]) | (x[3] && ~x[2] && ~x[1] && x[0])) && ~(x[3] && (x[2] | x[1]));

assign seg[5] = ((~x[3] && ~x[2] && ~x[1] && x[0]) | (~x[3] && ~x[2] && x[1] && ~x[0]) | (~x[3] && ~x[2] && x[1] && x[0]) | (~x[3] && x[2] && x[1] && x[0])) && ~(x[3] && (x[2] | x[1]));

assign seg[6] = ((~x[3] && ~x[2] && ~x[1] && ~x[0]) | (~x[3] && ~x[2] && ~x[1] && x[0]) | (~x[3] && x[2] && x[1] && x[0])) && ~(x[3] && (x[2] | x[1]));

assign an[0] = 0;

assign an[1] = 1;

assign an[2] = 1;

assign an[3] = 1;

assign an[4] = 1;

assign an[5] = 1;

assign an[6] = 1;

assign an[7] = 1;

endmodule



照片 3-lab1\_4\_2下载

“照片 3-lab1\_4\_2下载”的说明：本工程使用数据流风格实现一个BCD码到7段数码管的译码器；图中开关switch[2]拨至开，其余开关拨至关，表示输入为x=(0100)B；右侧七段数码管0位CA, CB, CF, CG管亮，其余管灭，其余位七段数码管全灭，灯亮的数码管笔画构成数字‘4’字样形成输出。

**实验总结：**

本次实验中，创建8个Vivado工程，分别使用了门级、数据流、行为模拟三种建模风格设计了1位与2位2选1数据选择器、开关到LED灯的直接控制装置、1位3选1数据选择器、BCD码到七段数码管的译码器并均加以综合、实现并在硬件和仿真环境中验证了功能正确性。

通过本次实验，我初步掌握了三种建模风格的语法特点，线网与寄存器类型的区别，连续数据传递中线网延迟的设置；了解了Nexys4 DDR开发板的部分硬件组成（主要为开关、LED灯与7段数码管）与工作原理；进一步练习了XDC约束文件内容的调整，Vivado软件创建工程、设计、分析、仿真、综合、实现与生成比特数据流并下载至硬件的操作。为之后的实验学习提供了知识基础。