数字电路与数字系统实验

实验二

译码器和编码器

计算机科学与技术系

191220008 陈南曈

924690736@qq.com

2020.9.14

一、实验目的

实现一个 8-3 优先编码器

查找 8-3 优先编码器相关原理和实现方法，实现一个 8-3 编码器，完成 8-3 编码器的设计、功能仿真和硬件实现。 使用拨动开关，SW7—SW0 随机输入一个 8 位二进制值，对此 8 位二进制 数进行高位优先编码成一个 3 位二进制值，并根据是否有输入增加一位输入指 示位，即 8 个开关全 0 时指示位为 0，有任何一个开关为 1 时指示位为 1。将此 编码结果以二进制形式显示在四个发光二极管上。再将此结果跟据七段数码管 的显示进行译码，将二进制的优先编码结果以十进制的形式显示在数码管上。 编码器的使能端可选实现。

二、实验原理（知识背景）

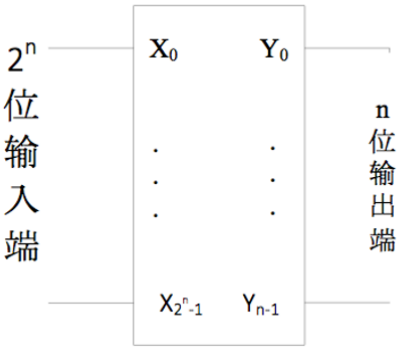
1、编码器:

对每个输入信号分配一个唯一的二进制编码,称它为编码器（encoder）
译码器的反函数电路
最常见的就是2^n-n 编码器或二进制编码器

分类

- 互斥唯一输入编码器
- 非互斥编码器
- 优先级编码器

编码器是一种与译码器功能相反的逻辑电路，编码器的输出编码比其输入 编码位数少。常用的二进制编码器把来自于2^n条输入线的信息编码转换成 n 位二进制码，如图所示。二进制编码器每次输入的2^n位信号中只能有一位为 1，其余均为 0（即独热码），编码器的输出端为一个二进制数，用来指示对应的哪一个位输入为 1。



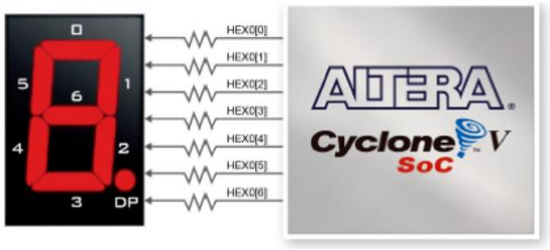
2、优先级编码器:

在任何一个特定的时刻，可能出现多于一个输入同时有效
如果出现多个输入同时有效，输出按输入优先级编码

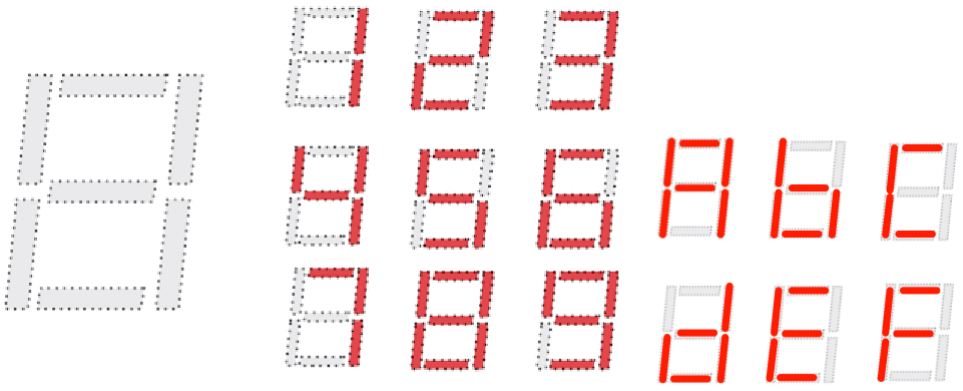
3、七段数码管:

七段 LED 数码管是一种常用的显示元件，常应用于手表、计算器等仪器中，用于显示数值。数码管分为共阴极和共阳极两种类型，共阴极就是将七个 LED 的阴极连在一起，让其接低电平。这样给任何一个 LED 的另一端高电平，它便能点亮。而共阳极就是将七个 LED 的阳极连在一起，让其接高电平。这样，给任何一个 LED 的另一端低电平，它就能点亮。

DE10-Standard 开发板上的数码管就是七段共阳极的。每个数码管的七段 LED 的一端连接在一个共同的阳极上，另一端和开发板上 Cyclone V SoC FPGA 的某一个一个引脚连接在一起，如图 2 8 所示，如果从这个引脚输出一个逻辑 0，则此段数码管被点亮。



数字 1 ~ 9 及十六进制的 a ~ f 在数码管上的显示方式如图所示。



三、实验环境/器材等

1) 软件环境：

Quartus (Quartus Prime 17.1) Lite Edition

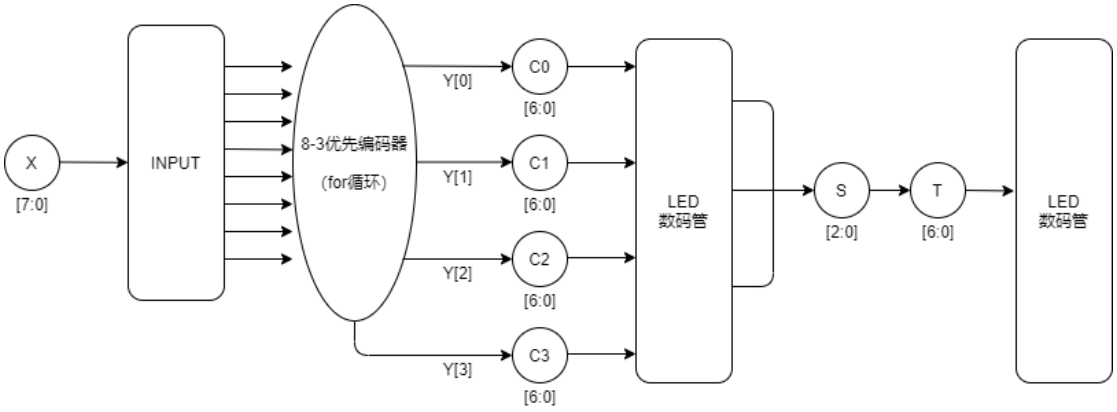
2) 硬件环境：

DE10-Standard 开发板
FPGA 部分：
Intel Cyclone V SE 5CSXFC6D6 F31C6N
- 110K 逻辑单元
- 5,761Kbit RAM

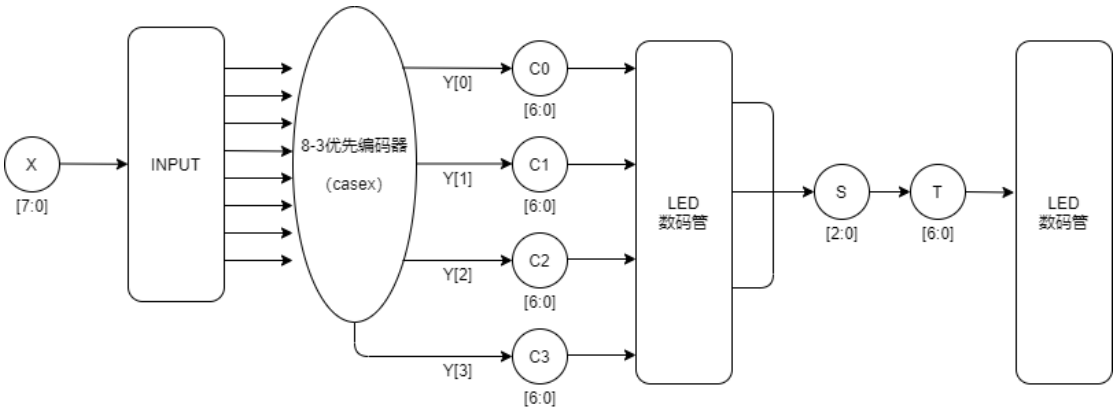
HPS 部分：
Dual-core ARM Cortex A9
- 925MHz
- 1GB DDR

四、程序代码或流程图

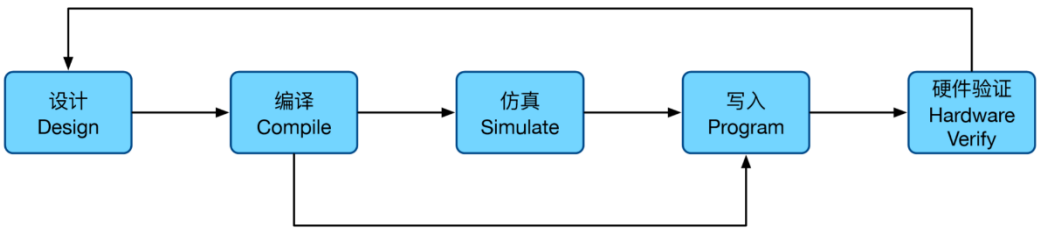
1、for 循环法：



2、casex 法：



五、实验步骤/过程



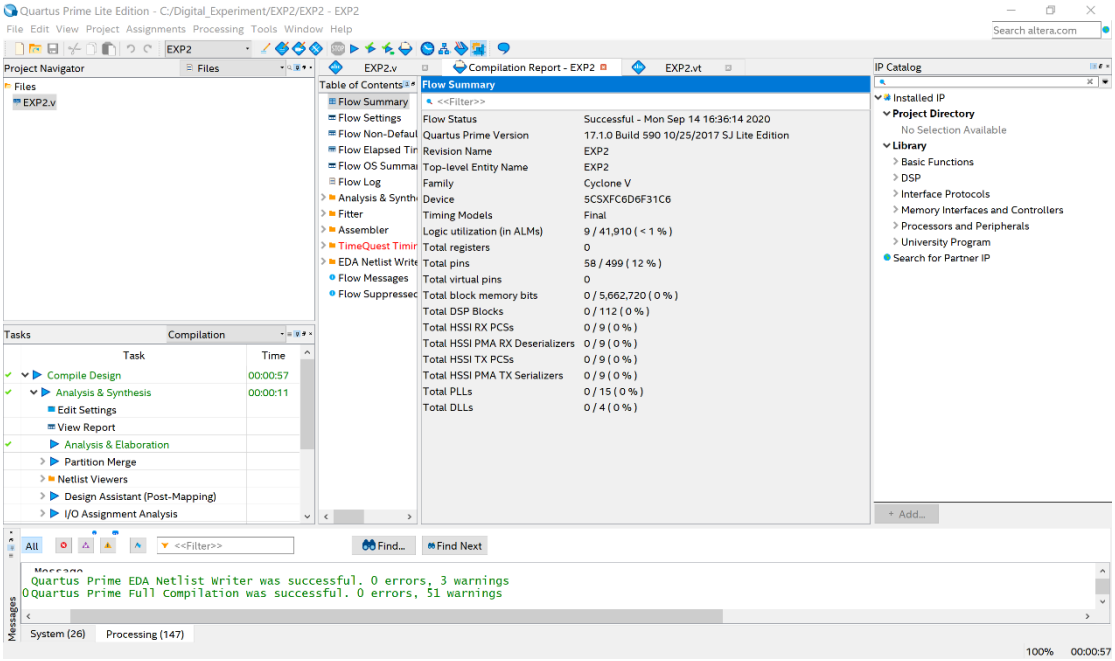
1、for 循环法：

设计：详见代码文件

测试：

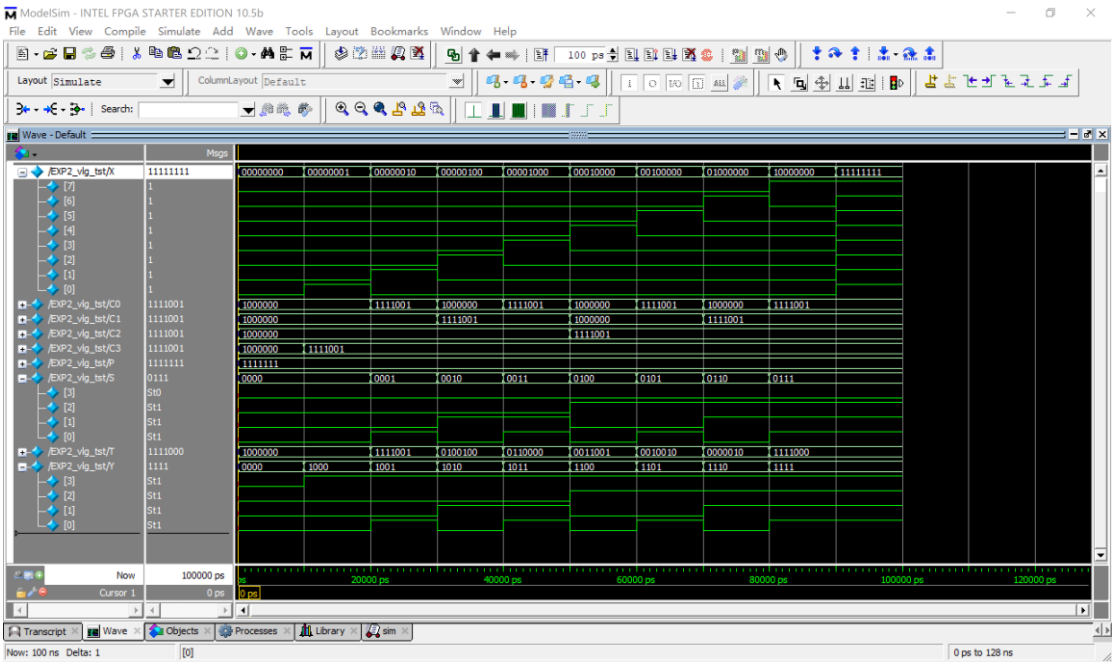
```
initial
begin
// code that executes only once
// insert code here --> begin
  X = 8'b00000000; #10;
  X = 8'b00000001; #10;
  X = 8'b00000010; #10;
  X = 8'b00000100; #10;
  X = 8'b00001000; #10;
  X = 8'b00010000; #10;
  X = 8'b00100000; #10;
  X = 8'b01000000; #10;
  X = 8'b10000000; #10;
  X = 8'b11111111; #10;
// --> end
//$display("Running testbench");
end
```

编译：

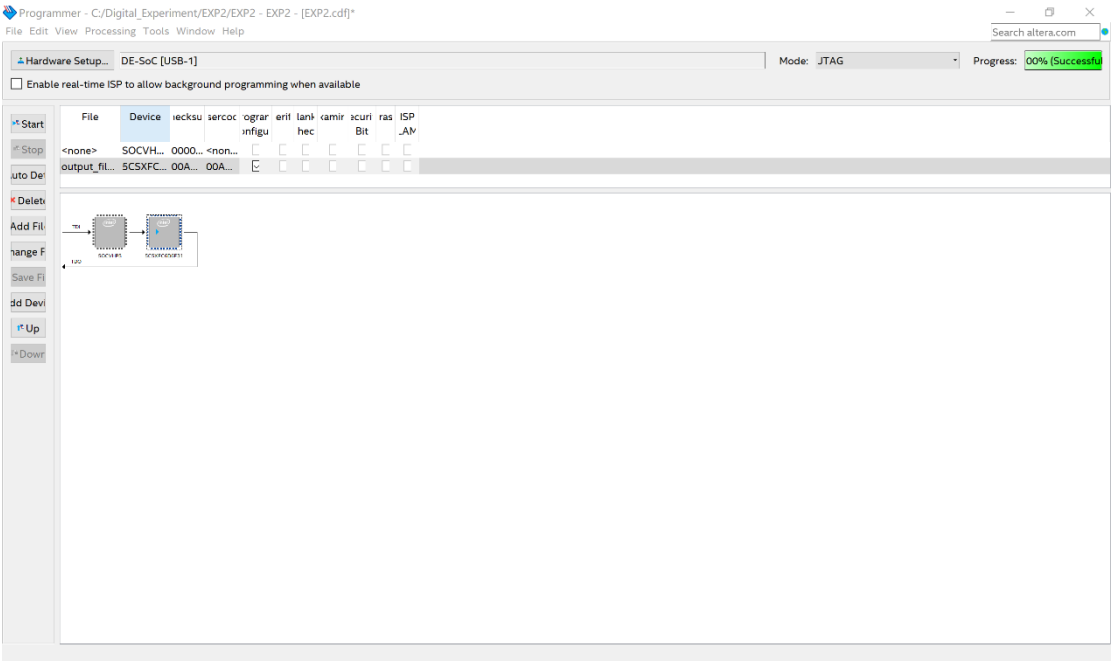


引脚分配：详见代码文件

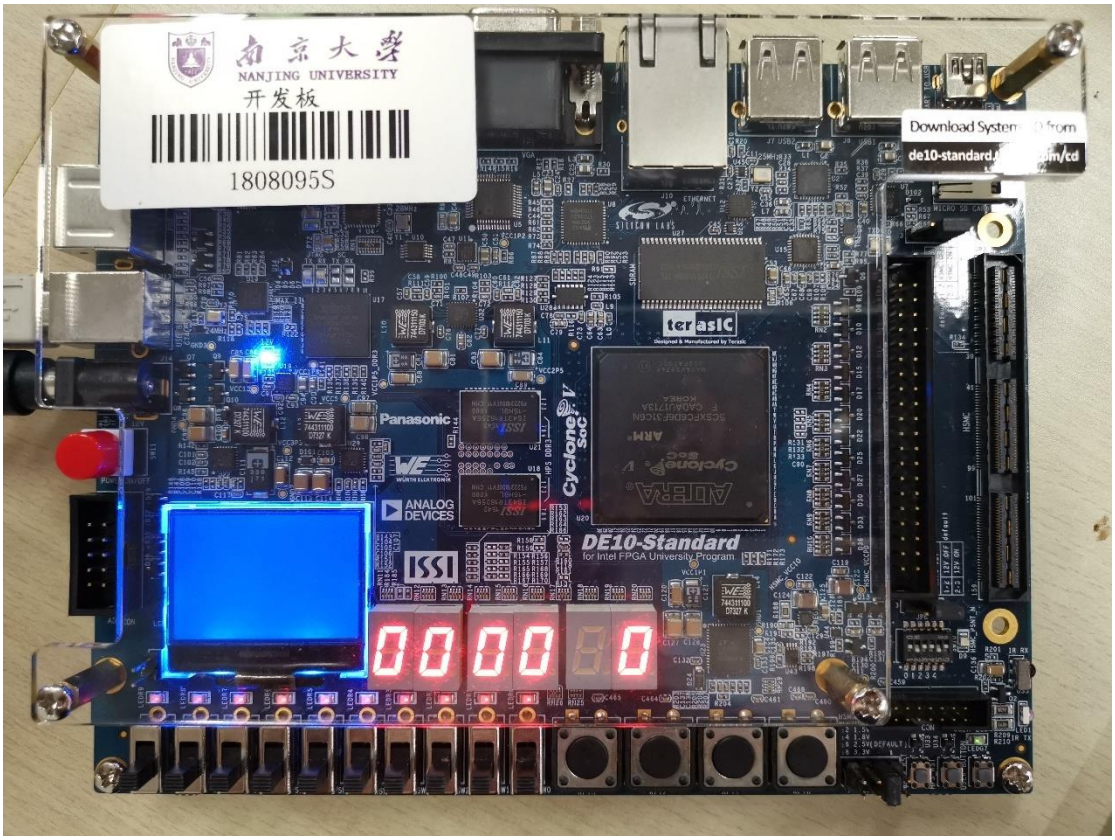
仿真：



写入：


Programmer - C:/Digital_Experiment/EXP2/EXP2 - EXP2 - [EXP2.cdf]*
File Edit View Processing Tools Window Help
Search altera.com
Hardware Setup...
DE-SoC [USB-1]
Mode: JTAG
Progress: 00% (Successful
Enable real-time ISP to allow background programming when available
Start
Stop
Add Fil
Save Fi
Up
File
Device
Bit
<none>
SOCVH...
output_fil...
5CSXFC...

硬件验证：


南京大学
NANJING UNIVERSITY
开发板
1808095S
Download System
de10-standard
terasIC
Panasonic
ANALOG DEVICES
ISSI
DE10-Standard
for Intel FPGA University Program

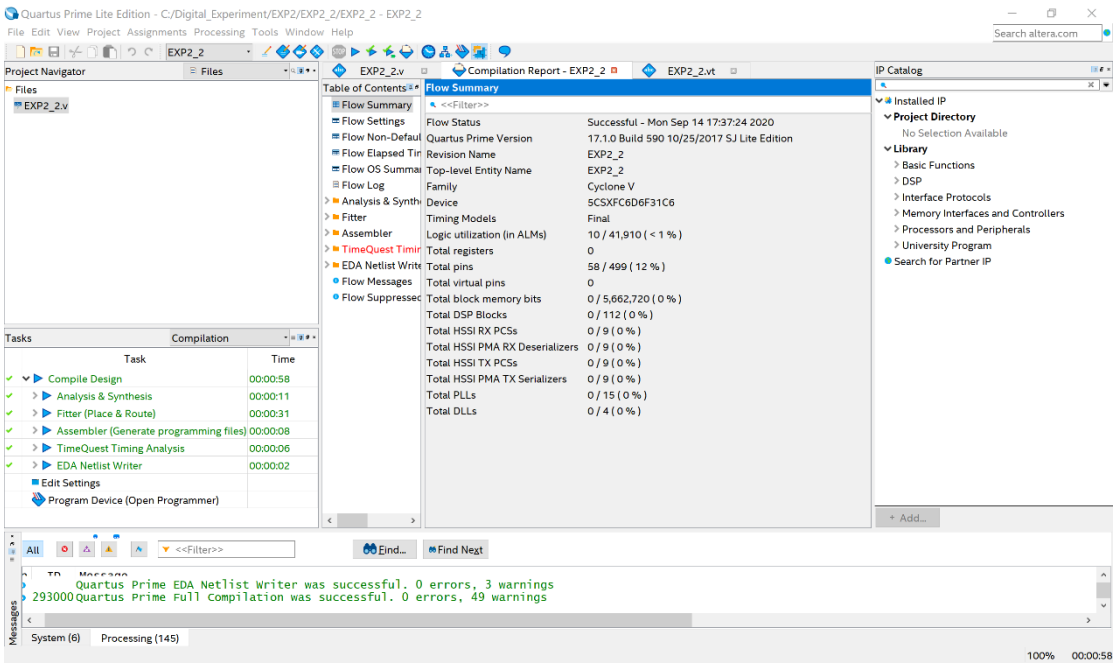
2、casex 法

设计：详见代码文件

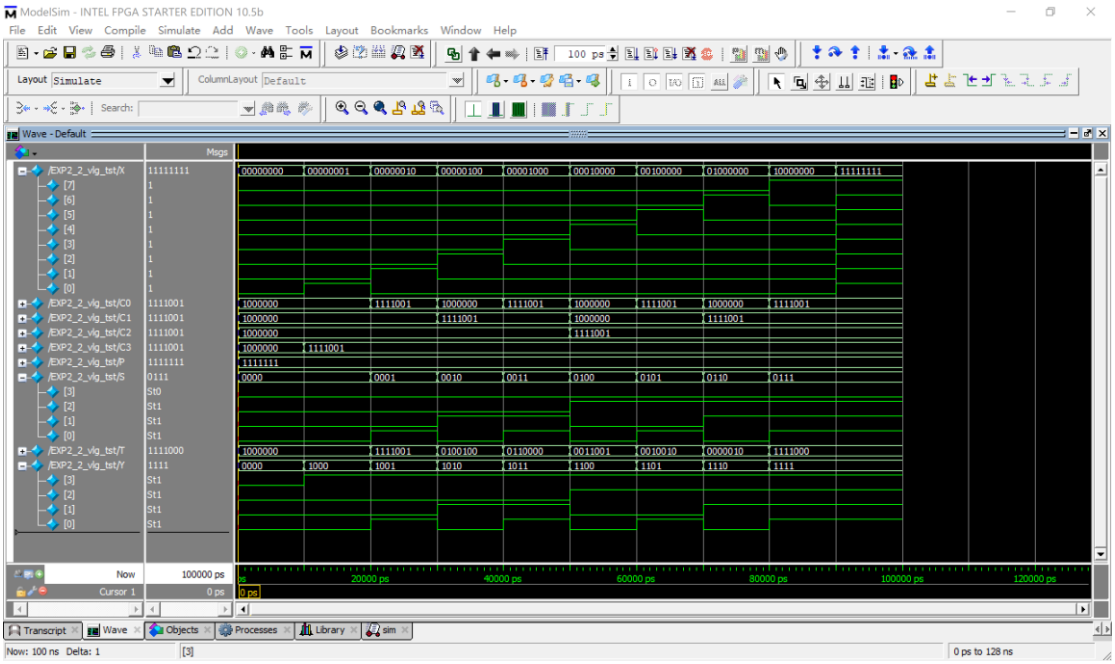
测试：

```
initial
begin
// code that executes only once
// insert code here --> begin
   X = 8'b00000000; #10;
   X = 8'b00000001; #10;
   X = 8'b00000010; #10;
   X = 8'b00000100; #10;
   X = 8'b00001000; #10;
   X = 8'b00010000; #10;
   X = 8'b00100000; #10;
   X = 8'b01000000; #10;
   X = 8'b10000000; #10;
   X = 8'b11111111; #10;
// --> end
//$display("Running testbench");
end
```

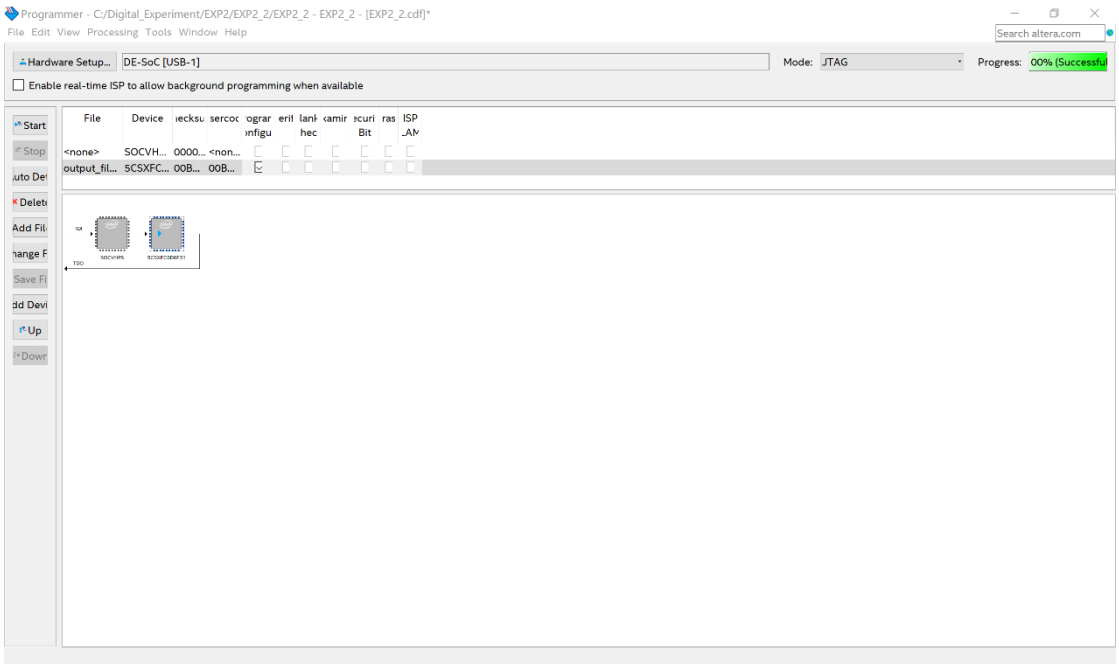
编译：



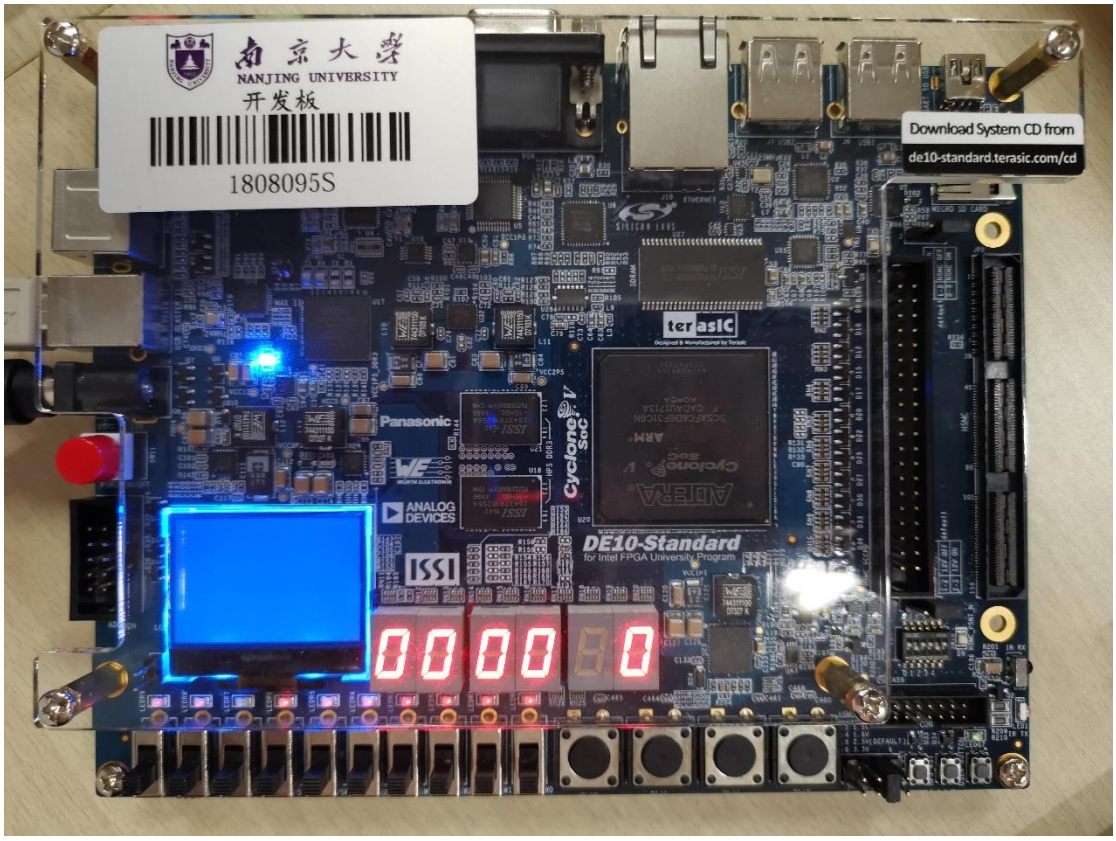
仿真：


ModelSim - INTEL FPGA STARTER EDITION 10.5b
File Edit View Compile Simulate Add Wave Tools Layout Bookmarks Window Help
Wave - Default
/EXP2_2_vlg_tst/X
/EXP2_2_vlg_tst/C0
/EXP2_2_vlg_tst/C1
/EXP2_2_vlg_tst/C2
/EXP2_2_vlg_tst/C3
/EXP2_2_vlg_tst/P
/EXP2_2_vlg_tst/S
/EXP2_2_vlg_tst/T
/EXP2_2_vlg_tst/Y
Now
Cursor 1
Transcript
Wave
Objects
Processes
Library
sim

写入：


Programmer - C:/Digital_Experiment/EXP2/EXP2_2/EXP2_2 - EXP2_2 - [EXP2_2.cdf]*
File Edit View Processing Tools Window Help
Hardware Setup...
DE-SoC [USB-1]
Mode: JTAG
Progress:
Enable real-time ISP to allow background programming when available
Start
Stop
Add Fil
Up
File
Device

硬件验证：



六、测试方法

Test Bench

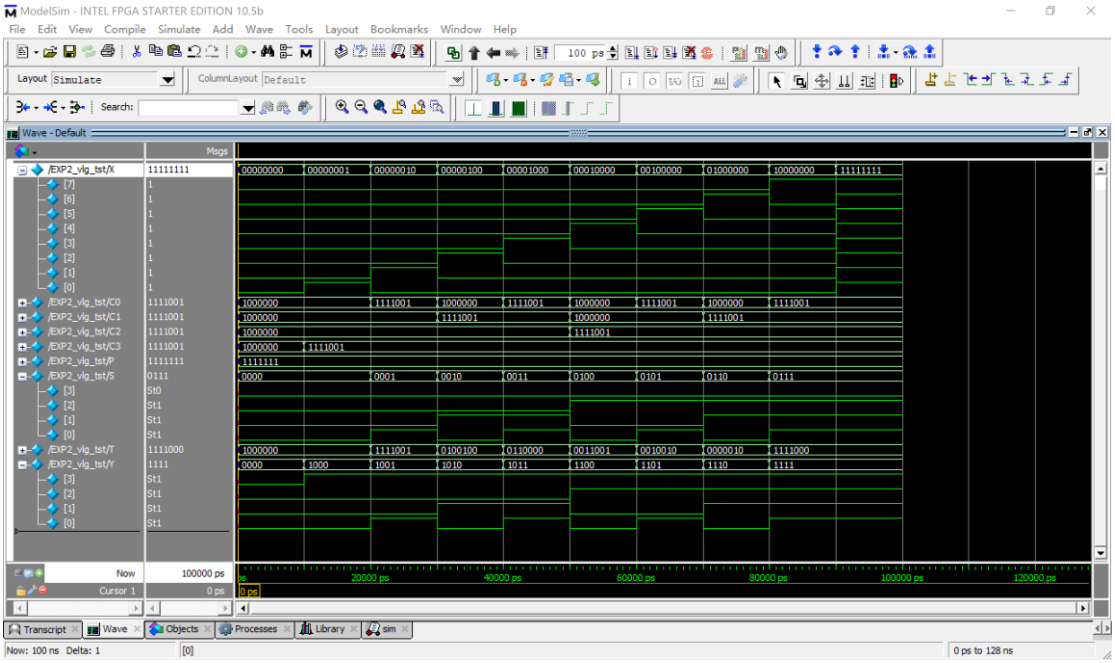
```
initial
begin
// code that executes only once
// insert code here --> begin
  X = 8'b00000000; #10;
  X = 8'b00000001; #10;
  X = 8'b00000010; #10;
  X = 8'b00000100; #10;
  X = 8'b00001000; #10;
  X = 8'b00010000; #10;
  X = 8'b00100000; #10;
  X = 8'b01000000; #10;
  X = 8'b10000000; #10;
  X = 8'b11111111; #10;
// --> end
//$display("Running testbench");
end
```

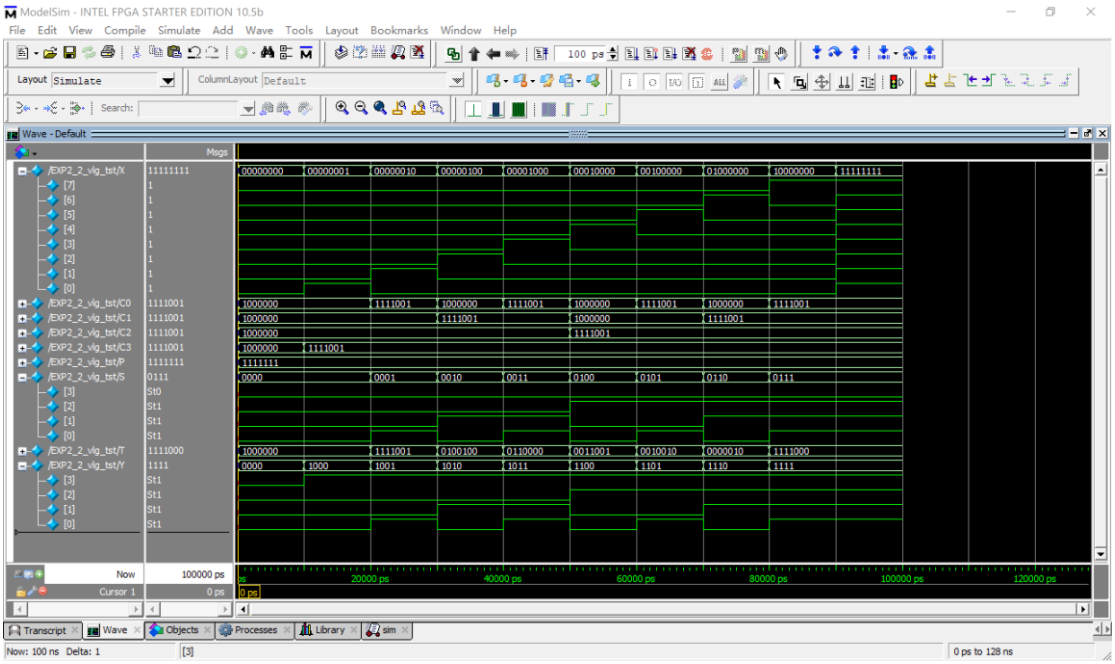
先测试只有一位输入有效时，输出结果是否正确
再测试有多位输入有效时（如：11111111），输出结果是否正确

这样，即可判断程序的正确性

七、实验结果

仿真结果：





实际结果：与仿真结果和真值表完全一致

八、实验中遇到的问题及解决办法

1、仿真图没有显示

解决办法：仔细比对 EXP0 中实验步骤，发现了两个原因导致了这种现象：①没有添加仿真测试文件；②仿真测试文件名需要和模块名不相同。当两个错误原因解决后，仿真图最后成功显示。

2、开发板上的一些不该显示的数码管（或其他显示）会自动出现奇怪的显示

解决办法：先仔细排查自己引脚分配是否出错，排查后发现自己并未出错。所以决定特别设置一个变量（如：[6:0] P）来将不该显示的数码管置为全暗，避免出现奇怪的显示（推测原因：开发板在过去可能有损坏，造成电路的连接有少量问题）

九、实验得到的启示

1、用 for 循环语句实现优先编码器是很自然的想法，但用 casex 实现节省了多次循环的时间，效率会更高，也更巧妙

2、将某一变量输出到七段数码管上时，可以用一个[6:0] P 变量作为中间变量进行转换，使得 P[i]（i=0~6）对应着数码管的每一段

十、意见和建议

希望能够明确指出“思考”部分的内容是否需要体现在实验报告中，感觉有些题意不明