

# 实验五



## 在系统编程基础篇

# 实验五

- 实验目的
- 实验器件和仪表
- 实验任务
- 实验步骤&提交成果
- 实验要求

# 实验五 | 实验目的

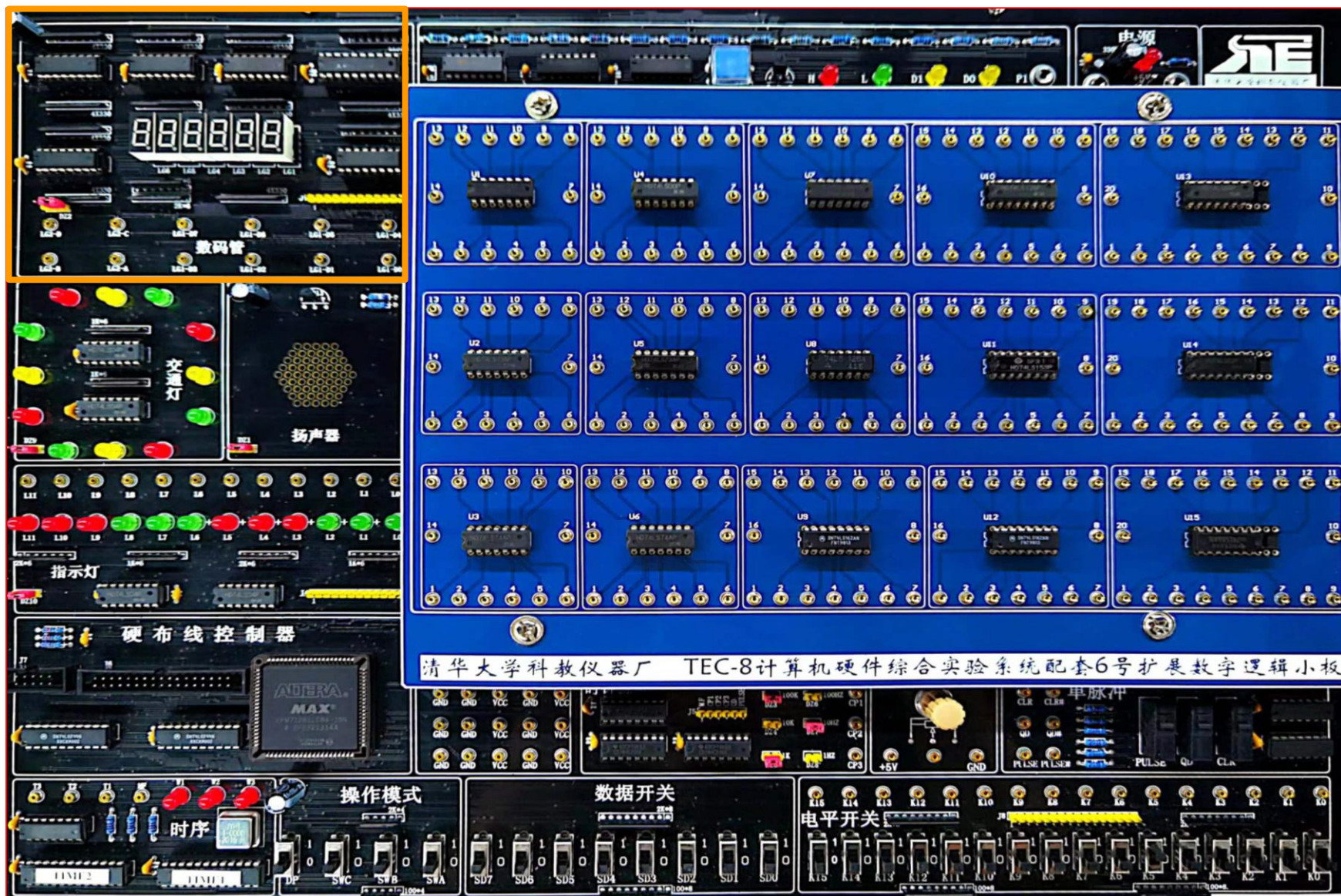
- ① 熟悉数码管的两种驱动方法及其应用；
- ② 掌握基本verilog语言的语法；
- ③ 学习使用Quartus II 软件进行设计与仿真；
- ④ 学习在系统可编程器件的下载。

## 实验五 | 实验器件和仪表

- TEC8数字电路实验系统 1台
- TBS1102B-EDU双踪示波器 1台
- QUARTUS II 软件

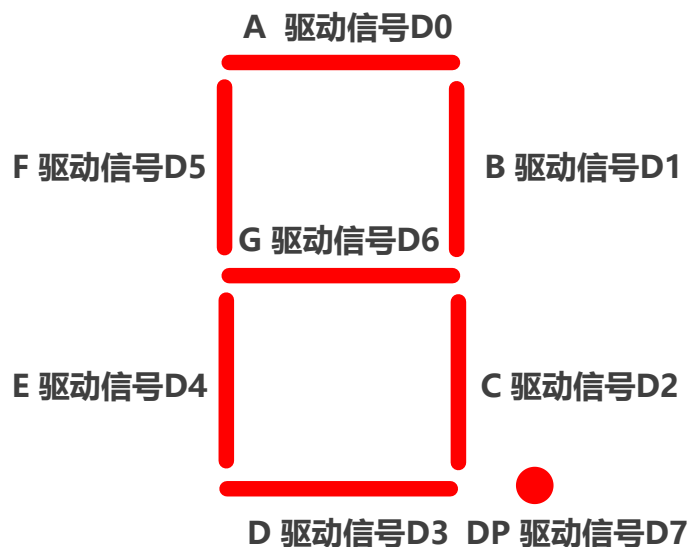
# 实验五

## 实验器件和仪表 – TEC-8 数码管



# 实验五 | 实验器件和仪表 – TEC-8 数码管

- TEC-8实验台上有6个日字型共阳极数码管;
- 每个数码管由8个单独控制的发光二极管构成;
- 8个发光二极管分别命名为a、b、c、d、e、f、g和dp, 其中dp代表小数点;





# 实验五 | 实验任务

- ① 测试数码管的功能;
- ② 利用verilog语言设计8421码十进制计数器并通过七段数码管显示该计数器数值。
  - ✓ 计数器的时钟输入为CLK的上升沿计数，具有异步复位输入端CLR，当CLR为低电平时，将计数器清零。

# 实验五

## 实验步骤&提交成果 - 任务1 - 数码管功能的测试

### □ 问题描述：测试数码管的功能；

- ✓ 将数码管LG1的驱动信号LG1-D7~LG2-D0分别接开关S7~S0，拨动开关，观察数码管LG1各段发光二极管的点亮情况；
- ✓ 将数码管LG2的驱动信号LG2-A、LG2-B、LG2-C、LG2-D分别接开关S0~S3,拨动开关形成数字0~9，观察数码管LG2对应的显示；



**注意：**使用数码管时，短路子DZ2必须短接；不使用时，最好断开DZ2。

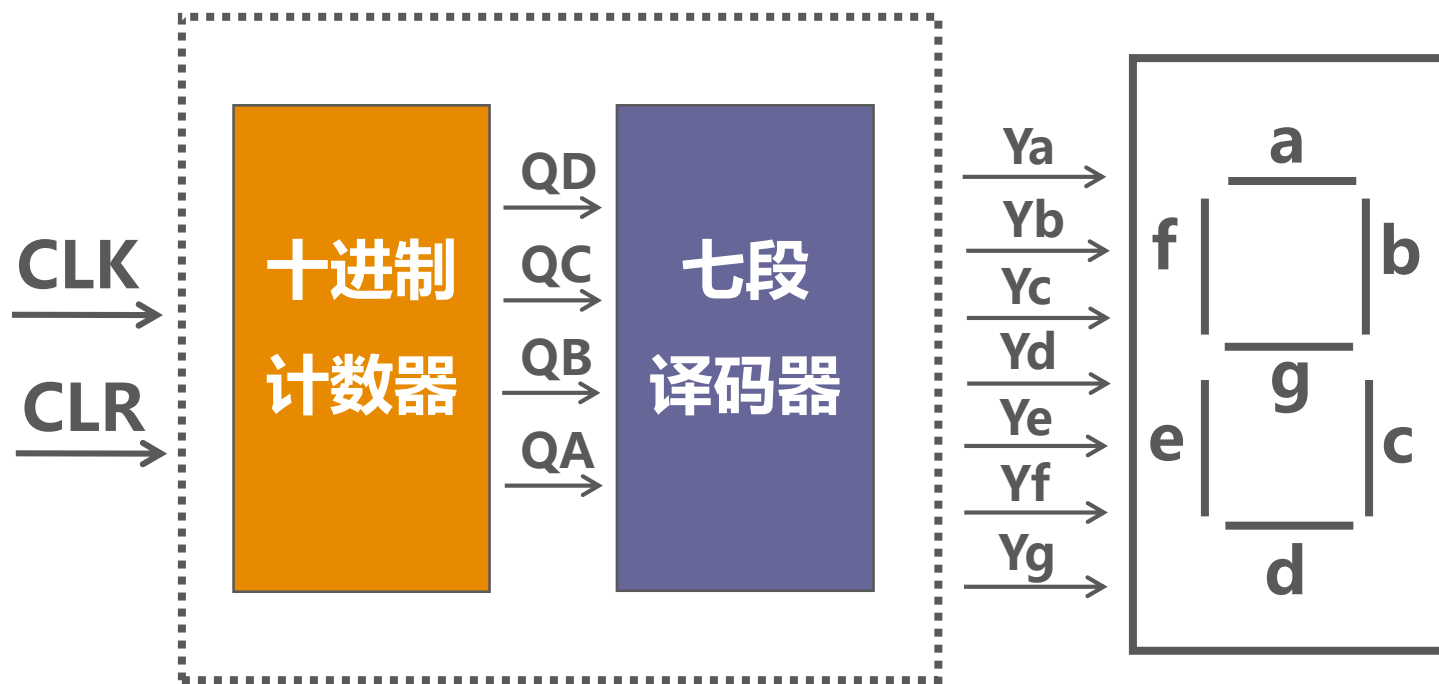


# 实验五

## 实验步骤&提交成果 - 任务2 - 十进制计数器的设计

- **问题描述：**用verilog语言设计并测试一个8421码十进制计数器及七段数码管显示系统；

- **设计框图：**

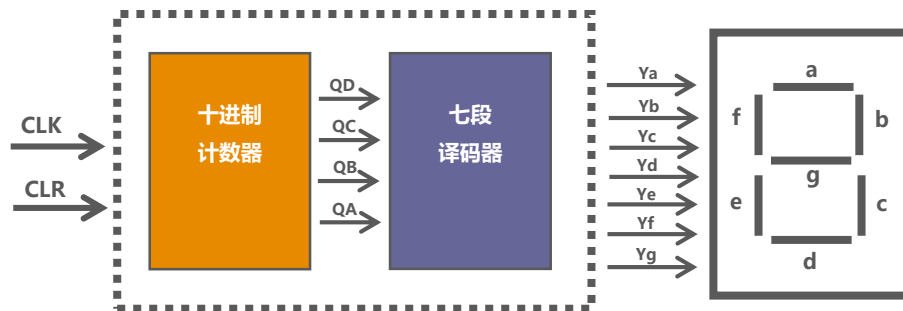


# 实验五

## 实验步骤&提交成果 -任务2 - 十进制计数器的设计

### □ 将任务2拆解为以下三个子任务：

- ① **任务2.1：** 用QUARTUS II 软件设计异步复位8421码十进制计数器，对该计数器进行功能仿真；
- ② **任务2.2：** 用QUARTUS II 软件设计七段数码管显示，对该数码管显示进行功能仿真；
- ③ **任务2.3：** 用QUARTUS II 软件实现十进制计数器并采用七段数码管显示，对该系统进行功能仿真；



# 实验五

## 实验步骤&提交成果 - 任务2.1 - 十进制计数器的设计

### □ 针对子任务2.1执行如下步骤:

- ① 设计verilog文件;
- ② 根据所选器件锁定管脚;
- ③ 对设计文件进行综合和编译;
- ④ 对设计进行功能仿真;

计数	输出			
	Q3	Q2	Q1	Q0
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1

# 实验五

## 实验步骤&提交成果 - 任务2.2- 七段译码器的设计

### □ 针对子任务2.2执行如下步骤:

- ① 写出vrellog文件;
- ② 根据所选器件锁定管脚;
- ③ 对设计文件进行综合和编译;
- ④ 对设计进行功能仿真;

# 实验五

## 实验步骤&提交成果 - 任务2.2 - 七段译码器的设计

### 七段译码器真值表：

输入				输出							字型
D	C	B	A	Fa	Fb	Fc	Fd	Fe	Ff	Fg	
0	0	0	0	1	1	1	1	1	1	0	0
0	0	0	1	0	1	0	0	0	0	0	1
0	0	1	0	1	1	0	1	1	0	1	2
0	0	1	1	1	1	1	1	0	0	1	3
0	1	0	0	0	1	1	0	0	1	1	4
0	1	0	1	1	0	1	1	0	1	1	5
0	1	1	0	1	0	1	1	1	1	1	6
0	1	1	1	1	1	1	0	0	0	0	7
1	0	0	0	1	1	1	1	1	1	1	8
1	0	0	1	1	1	1	1	0	1	1	9

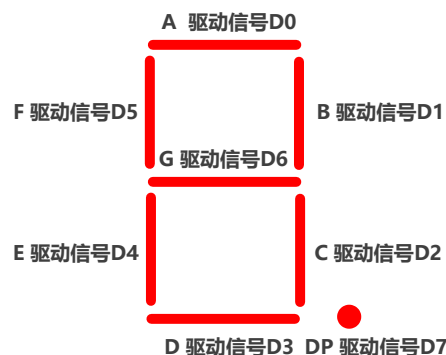
# 实验五

## 实验步骤&提交成果 - 任务2.3 - 计数器+数码管显示

### □ 针对子任务2.3执行如下步骤：

- ① 写出verilog文件；
- ② 根据所选器件锁定管脚；

信号名	引脚号	信号方向	信号意义
LG1-D0	44	OUT	a的驱动信号
LG1-D1	45	OUT	b的驱动信号
LG1-D2	46	OUT	c的驱动信号
LG1-D3	48	OUT	d的驱动信号
LG1-D4	49	OUT	e的驱动信号
LG1-D5	50	OUT	f的驱动信号
LG1-D6	51	OUT	g的驱动信号
LG1-D7	52	OUT	dp的驱动信号
QD	60	IN	单脉冲
CLR	1	IN	复位 低电平有效



# 实验五

## 实验步骤&提交成果 - 任务2.3 - 计数器+数码管显示

### □ 针对子任务2.3执行如下步骤：

- ③ 对verilog文件进行综合和编译；
- ④ 对设计进行功能仿真，并确保功能仿真的结果是正确的；



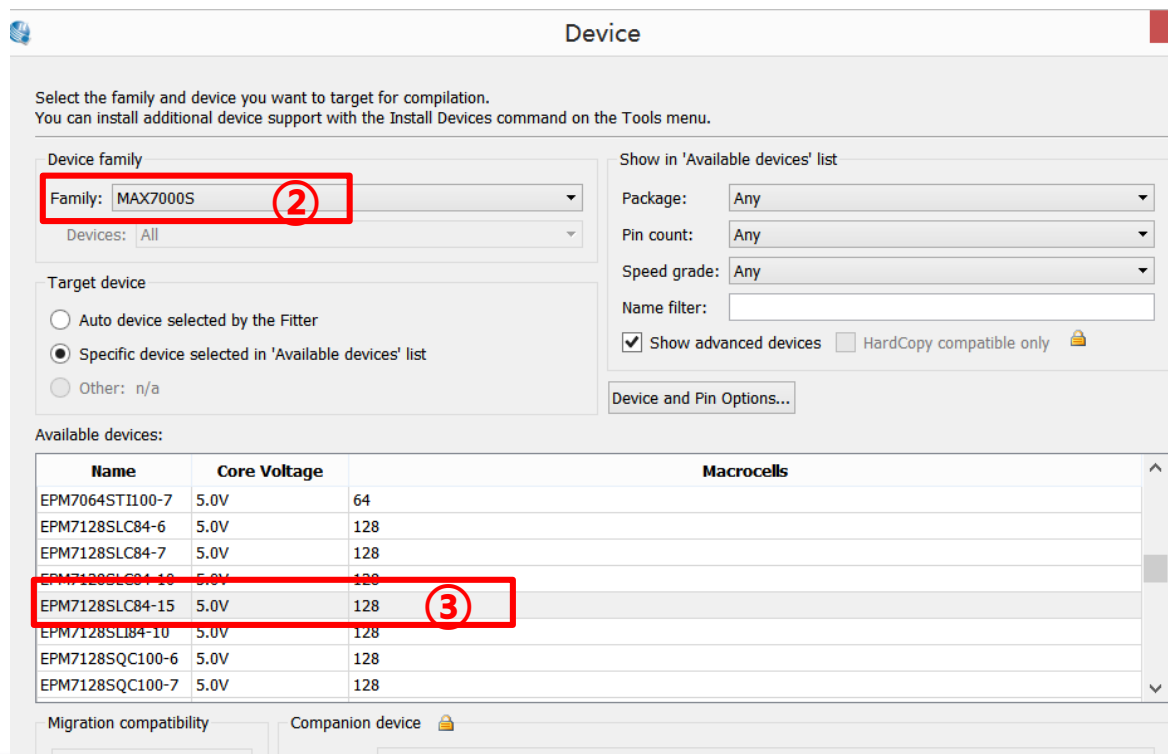
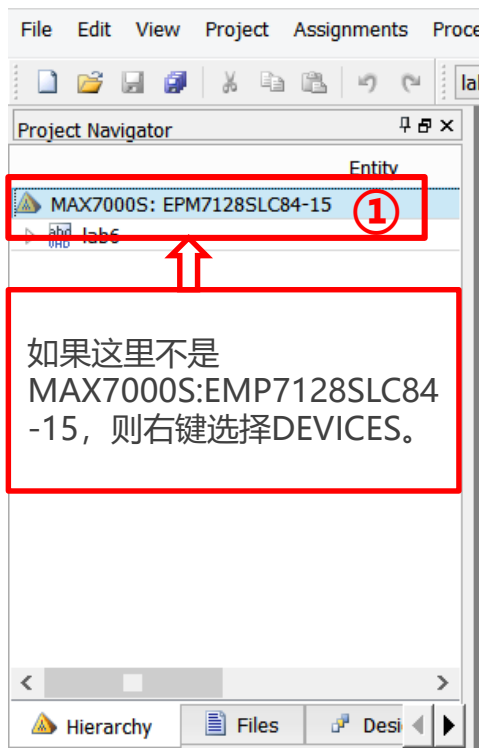
# 实验五

## 实验步骤&提交成果 -任务2.3 - 计数器+数码管显示

### □ 针对子任务2.3执行如下步骤:

#### ⑤ 进行程序下载

✓ 检查下载器件型号



# 实验五

## 实验步骤&提交成果 -任务2.3 - 计数器+数码管显示

### □ 针对子任务2.3执行如下步骤:

#### ⑤ 进行程序下载

- ✓ 连接下载线：将下载线USB端连接电脑，黑色下载端连接实验箱硬布线控制器模块下载插孔（注意插槽方向）



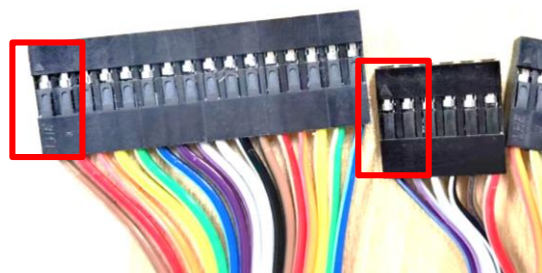
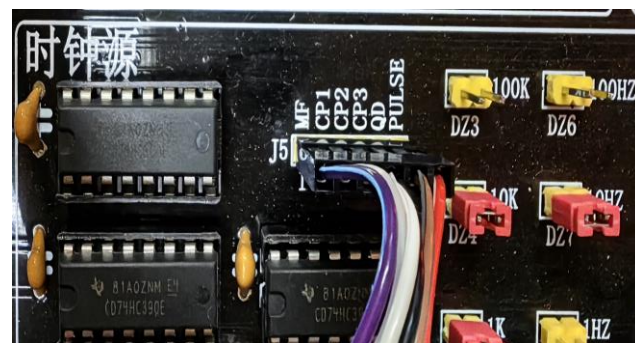
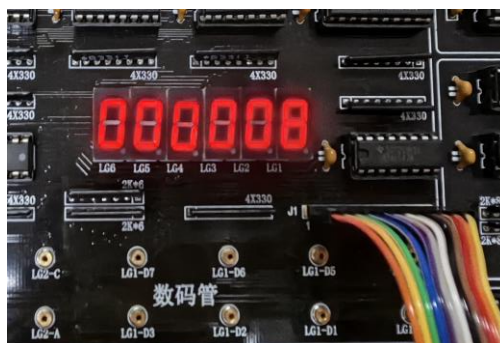
# 实验五

## 实验步骤&提交成果 -任务2.3 - 计数器+数码管显示

### □ 针对子任务2.3执行如下步骤:

#### ⑤ 进行程序下载

- ✓ 将扁平电缆的34芯端插到插座J6中（EPM7128左侧），将12芯端插到插座J1中（数码管右下），将6芯端插到插座J5中（时钟源中间）



**注意：观察12芯与6芯端的三角标志，要与34芯端的三角标志朝向同一方向**



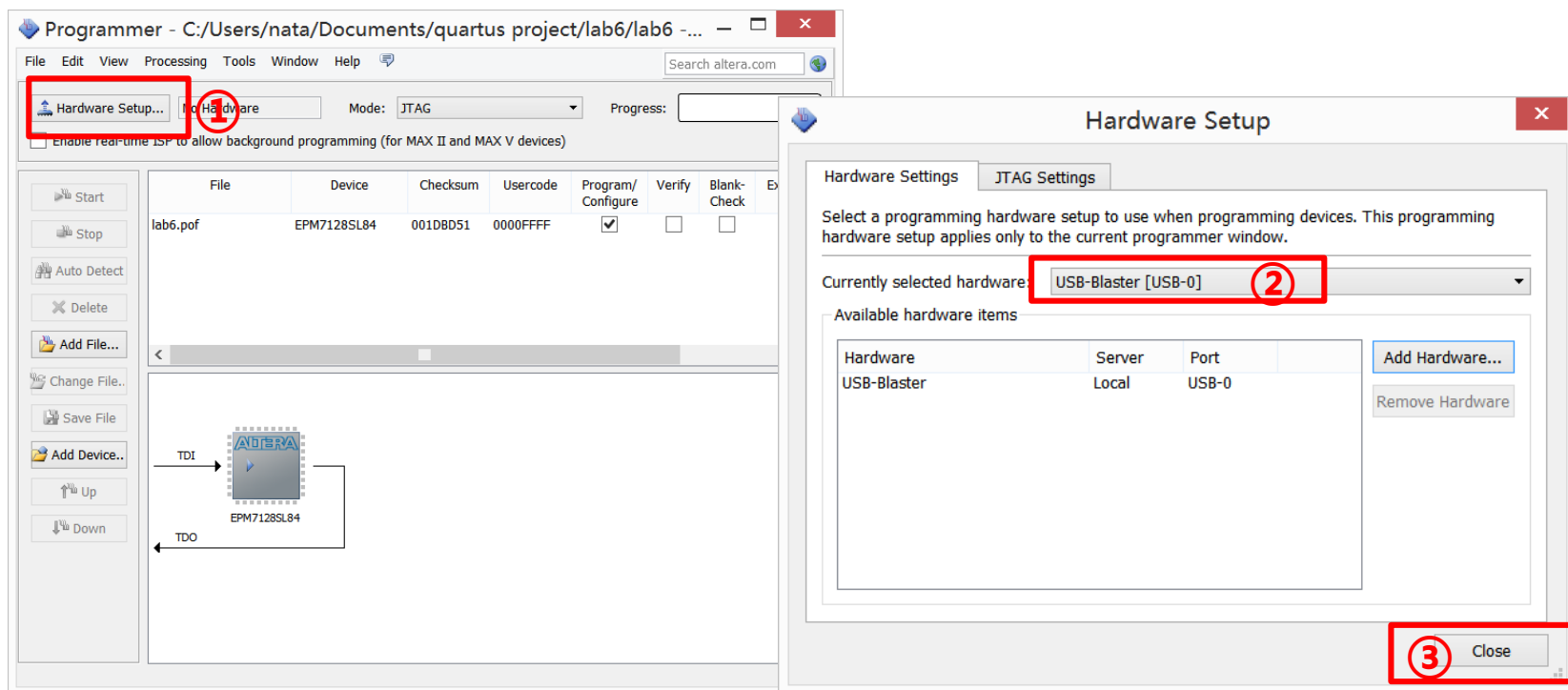
# 实验五

## 实验步骤&提交成果 -任务2.3 - 计数器+数码管显示

### □ 针对子任务2.3执行如下步骤:

#### ⑤ 进行程序下载

✓ 菜单 >> Tools >> Programmer



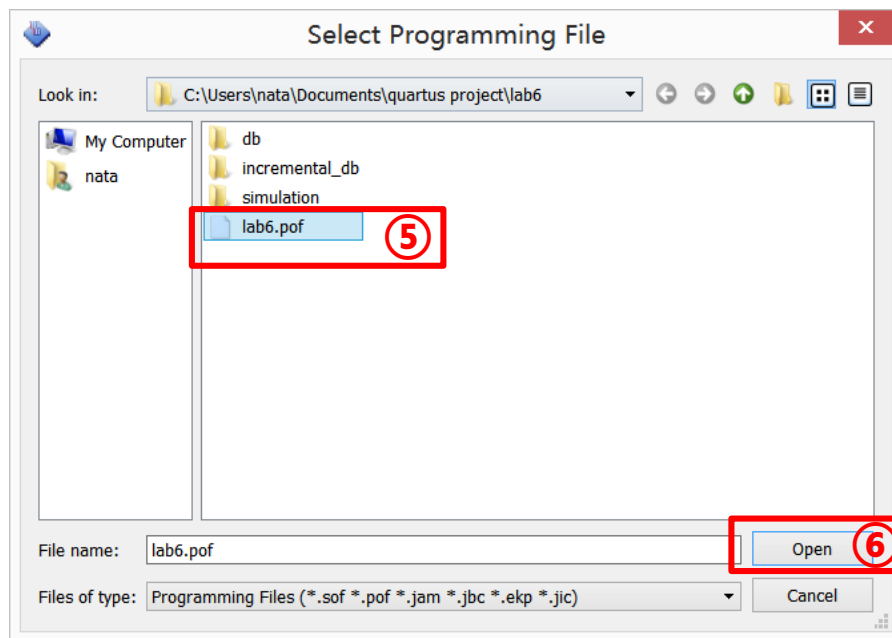
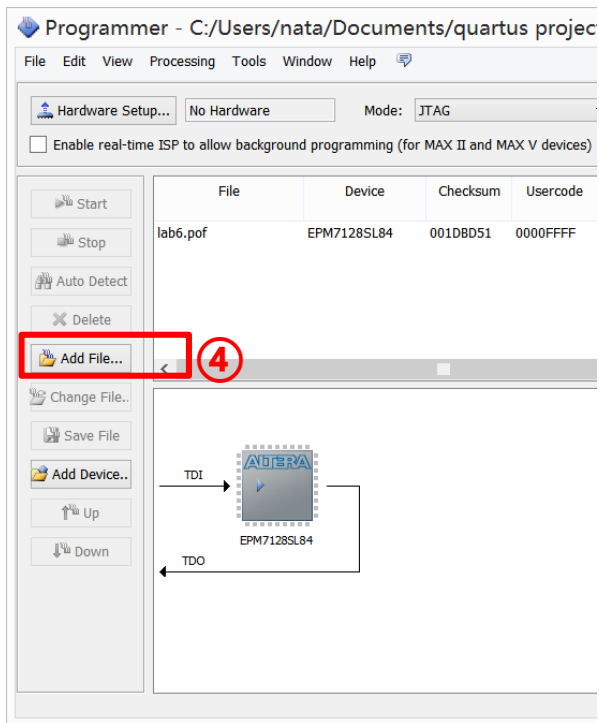
# 实验五

## 实验步骤&提交成果 -任务2.3 - 计数器+数码管显示

### □ 针对子任务2.3执行如下步骤:

#### ⑤ 进行程序下载

✓ 菜单 >> Tools >> Programmer



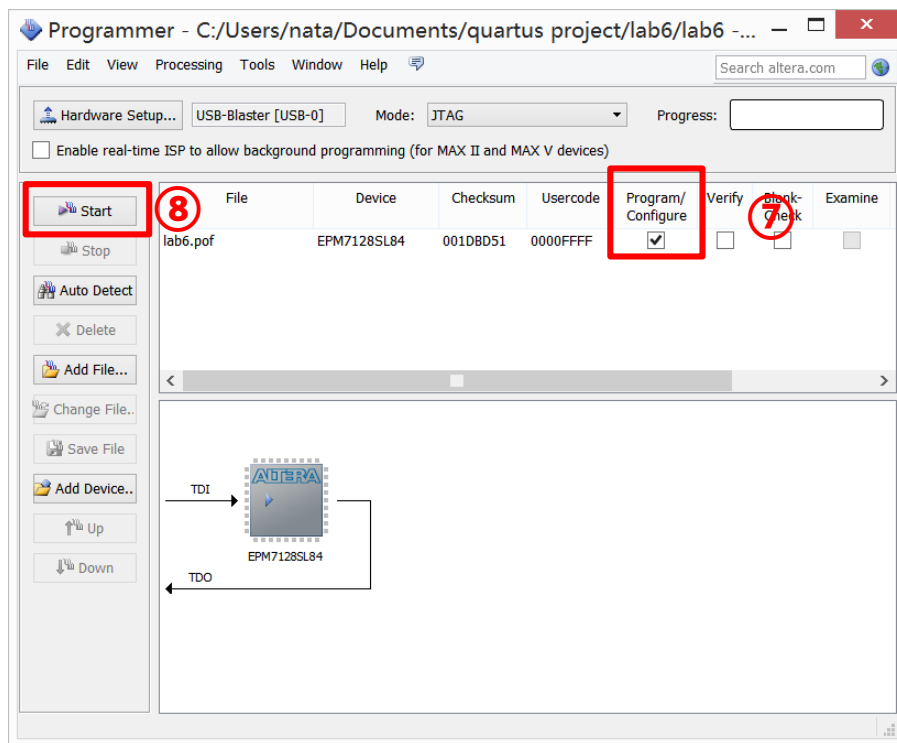
# 实验五

## 实验步骤&提交成果 -任务2.3 - 计数器+数码管显示

### □ 针对子任务2.3执行如下步骤:

#### ⑤ 进行程序下载

✓ 菜单 >> Tools >> Programmer



⑥

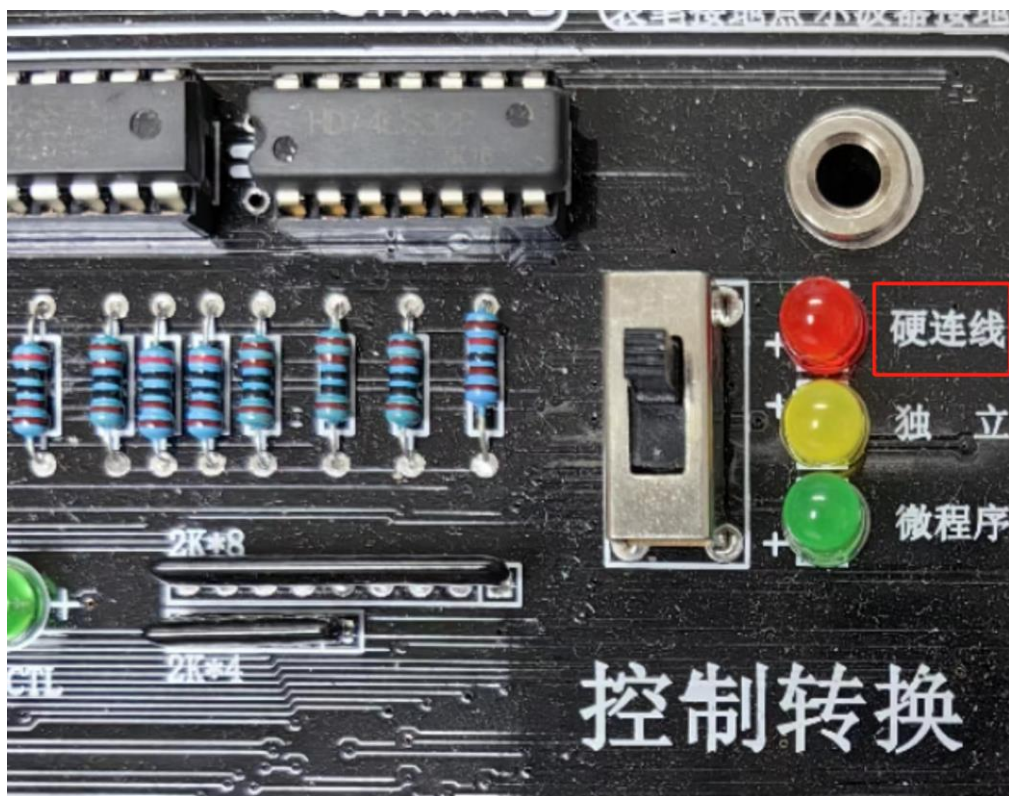


# 实验五

## 实验步骤&提交成果 - 任务2.3 - 计数器+数码管显示

### □ 针对子任务2.3执行如下步骤：

- ⑥ 将实验箱的控制转换开关设置为“硬连线”





# 实验五

## 实验步骤&提交成果 - 任务2.3 - 计数器+数码管显示

### □ 针对子任务2.3执行如下步骤：

- ⑦ 使用单脉冲QD作为计数器时钟，复位信号CLR。用实验台上最右边的LG1数码管验证设计的正确性。
  - 注意：扁平电缆进行插拔必须在关电源后进行
  - 短路子DZ2短接，实验完成后，DZ2断开。

# 实验五

## 实验步骤&提交成果 -任务2.3 - 计数器+数码管显示

### □ 针对任务提交：

- ✓ 给出子任务2.3的verilog代码；
- ✓ 给出子任务2.3的功能仿真波形图；
  - ✓ 要求计数器从0开始依次计数到最大仿真得到数码管的编码；
  - ✓ 中间需要加入异步复位信号的验证仿真；
- ✓ 在实验箱上进行下载的截图；
  - ✓ 需要包含在计算机中分配的管脚图；
  - ✓ 全编译通过的截图（整个QUARTUS II 的大窗口包含编译通过的具体信息）
- ✓ 如果有其他附加完成的设计与实现也请在报告中明确说明；

# 实验五 | 实验要求 - 总体要求

## □ 常见问题

- 常见错误:
- 程序逻辑错误;
- 没有将控制转换拨至硬连线;
- 花线插线方向出错;
- 管脚分配错误, 尤其是数码管高低位的顺序错误;
- 芯片设置错误

# 实验五 | 实验要求 - 总体要求

## □ 认真填写实验报告

- 实验报告按照要求填写；
- 内容包含：
  - 实验目的；
  - 所有任务的具体提交成果；
- 所有表格都写在实验报告内；
- 内容描述条理清晰，切忌堆砌内容；
- 实验报告一周内提交；

# 实验六



## 在系统编程提高篇

# 实验六

- 实验目的
- 实验器件和仪表
- 实验任务
- 提交成果

# 实验六 | 实验目的

- ① 熟练掌握基本verilog语言的语法;
- ② 综合设计一个略复杂的数字电路;



## 实验六 | 实验器件和仪表

- TEC8数字电路实验系统 1台
- QUARTUS II 软件

# 实验六

## 实验器件和仪表 - EPM7128引脚信号

分组	信号	管脚号	输入/输出	解释
时钟等	CLR#	1	IN	复位信号
	MF	55	IN	主时钟
	CP1	56	IN	100KHZ/10KHZ
	CP2	57	IN	1KHZ/100HZ
	CP3	58	IN	10HZ/1HZ
	QD	60	IN	启动按钮QD
	PULSE	61	IN	中断脉冲
开关	K0	54	IN	开关
	K1	81	IN	开关
	K2	80	IN	开关
	K3	79	IN	开关
	K4	77	IN	开关
	K5	76	IN	开关
	K6	75	IN	开关
	K7	74	IN	开关
	K8	73	IN	开关
	K9	70	IN	开关
	K10	69	IN	开关
	K11	68	IN	开关
	K12	67	IN	开关
	K13	65	IN	开关
	K14	64	IN	开关
	K15	63	IN	开关

# 实验六 | 实验器件和仪表 - EPM7128引脚信号

分组	信号	管脚号	输入/输出	解释
数码管1	LG1-D0/L4	44	OUT	数码管1/发光二极管L7
	LG1-D1/L5	45	OUT	数码管1/发光二极管L6
	LG1-D2/L6	46	OUT	数码管1/发光二极管L5
	LG1-D3/L7	48	OUT	数码管1/发光二极管L4
	LG1-D4/L8	49	OUT	数码管1/发光二极管L3
	LG1-D5/L9	50	OUT	数码管1/发光二极管L2
	LG1-D6/L10	51	OUT	数码管1/发光二极管L1
	LG1-D7/L11/SPEAKER	52	OUT	数码管1小数点/发光二极管L0/扬声器
数码管2	LG2-A	37	OUT	数码管LG2/发光二极管L11
	LG2-B	39	OUT	数码管LG2/发光二极管L10
	LG2-C	40	OUT	数码管LG2/发光二极管L9
	LG2-D	41	OUT	数码管LG2/发光二极管L8
数码管3	LG3-A	35	OUT	数码管LG3
	LG3-B	36	OUT	数码管LG3
	LG3-C	17	OUT	数码管LG3
	LG3-D	18	OUT	数码管LG3
数码管4	LG4-A/TL8	30	OUT	数码管LG4/控制南方绿灯
	LG4-B/TL9	31	OUT	数码管LG4/控制东方红灯
	LG4-C/TL10	33	OUT	数码管LG4/控制东方黄灯
	LG4-D/TL11	34	OUT	数码管LG4/控制东方绿灯
数码管5	LG5-A/TL4	25	OUT	数码管LG5/控制西方黄灯
	LG5-B/TL5	27	OUT	数码管LG5/控制西方绿灯
	LG5-C/TL6	28	OUT	数码管LG5/控制南方红灯
	LG5-D/TL7	29	OUT	数码管LG5/控制南方黄灯
数码管6	LG6-A/TL0	20	OUT	数码管LG6/控制北方红灯
	LG6-B/TL1	21	OUT	数码管LG6/控制北方黄灯
	LG6-C/TL2	22	OUT	数码管LG6/控制北方绿灯
	LG6-D/TL3	24	OUT	数码管LG6/控制西方红灯

# 实验六 | 实验任务

- 秒表（必做）
- 电子琴（选做）
- 寄存器堆（选做）

# 实验六

## 实验步骤&提交成果 - 任务1 - 秒表

- **问题描述：**用verilog语言设计一个模60计数器；
  - ✓ 根据实验五拓展成模60计数器；
  - ✓ 秒表可采用正计时或倒计时；
  - ✓ 秒表具备清零端；
  - ✓ 其他自由发挥。



**注意：**使用数码管时，短路子DZ2必须短接；不使用时，最好断开DZ2。

# 实验六

## 实验步骤&提交成果 - 任务2 - 电子琴 (选做)

### □ 问题描述：用verilog语言设计一个电子琴；

① 八个输入端口分别接实验箱上的k1-k8电平开关；

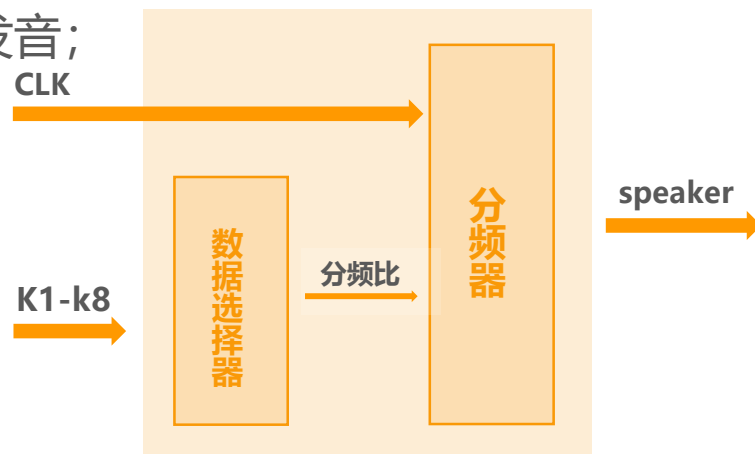
✓ K8代表1；

✓ 开关置1发出对应的音调，置0不发音；

✓ 任意时刻只有一个开关置1；

② 一个输出端口与扬声器连接；

③ 输入时钟信号为MF 1MHz；



音调	1	2	3	4	5	6	7	1
频率(Hz)	262	294	330	349	392	440	494	523

# 实验六

## 实验步骤&提交成果 - 任务3 - 寄存器堆（选做）

### ❑ 问题描述：用verilog语言设计一个寄存器堆；

#### ① 4个8位寄存器R0-R3，负责存储数据

- ✓ 输入：数据，来自电平开关k0-k7；
- ✓ 输入：写信号CRW，共用1个，来自电平开关；
- ✓ 输入：使能信号，来自2-4译码器，任何一个时刻只有一个寄存器使能有效；

#### ② 1个 2-4译码器，负责产生寄存器的使能信号

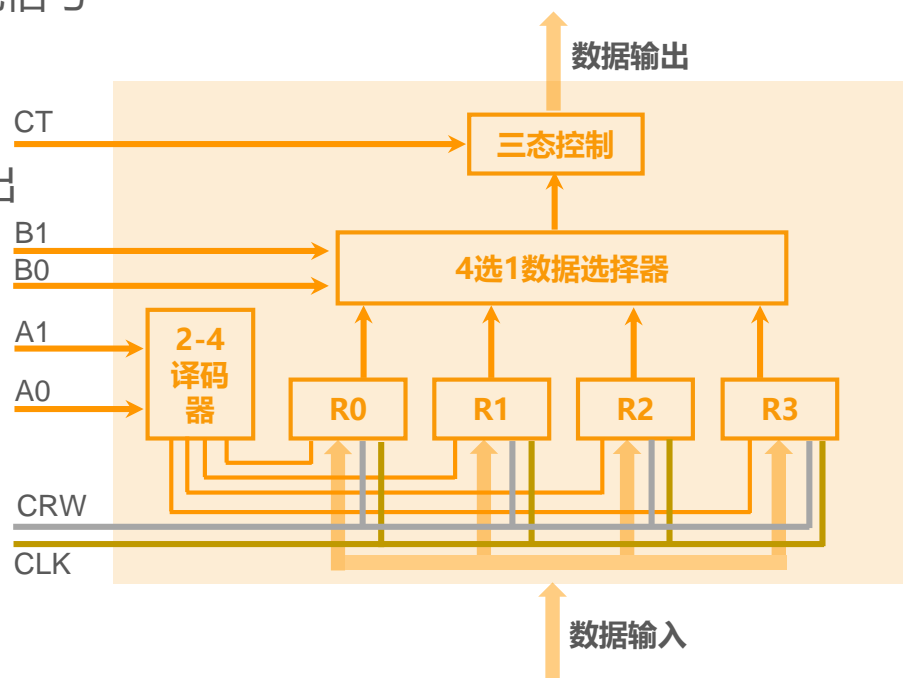
- ✓ 输入：2位地址，来自电平开关；
- ✓ 输出：1位使能信号，送给寄存器；

#### ③ 1个4选1选择器，负责选择一路数据输出

- ✓ 输入：4路数据，来自寄存器；
- ✓ 输入：2个地址选择信号，来自开关；
- ✓ 输出：输出数据；

#### ④ 1个三态门，负责控制数据的输出

- ✓ 输入：使能信号，来自开关；
- ✓ 输入：数据，来自4选1选择器；
- ✓ 输出：数据，送给LED灯；





# 实验六

## 提交成果

### □ 针对完成的任务提交：

- ✓ 给出verilog代码；
- ✓ 描述具体实现的功能；
  - ✓ 分条罗列；
  - ✓ 如果有和实验要求不一样的或者特别设计的点要明确描述清楚；
- ✓ 给出设计思路；