

# 西安电子科技大学

## 电子线路实验 (II)

## 课程实验报告

实验名称 大规模集成数字电路设计

机电工程 学院 2004031 班  
姓名 李艺超 学号 20049200519  
姓名 靳钰婷 学号 20049200170  
姓名 纪诗琪 学号 20049200061  
姓名 马雨琦 学号 20049200240  
姓名 王之韵 学号 20049200393  
姓名 杨欣睿 学号 20049200229  
实验日期 2021 年 11 月 21 日

成 绩

指导教师评语:

指导教师:

\_\_\_\_年\_\_\_\_月\_\_\_\_日

### 实验报告内容基本要求及参考格式

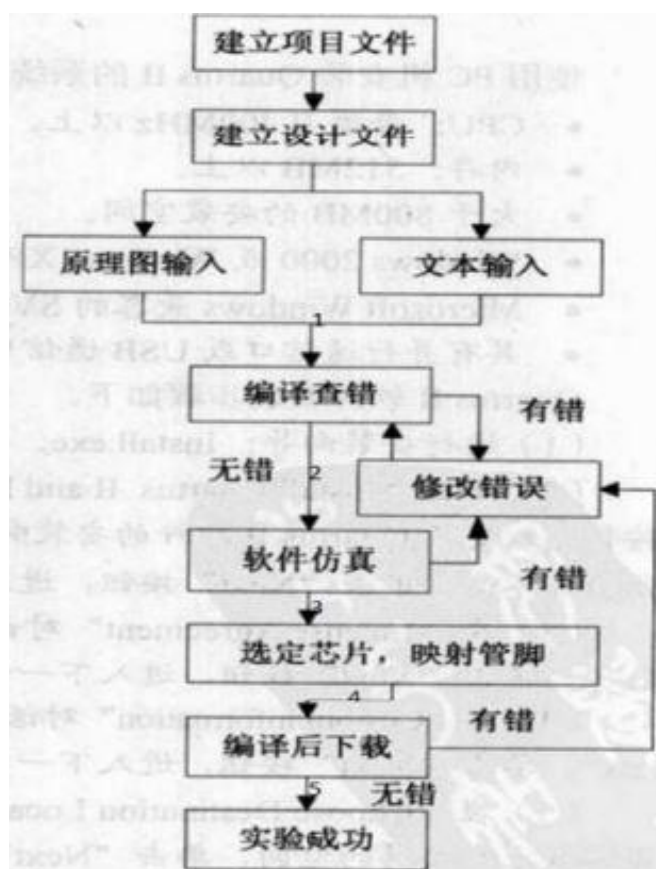
- 一、实验目的
- 二、实验所用仪器 (或实验环境)
- 三、实验基本原理及步骤 (或方案设计及理论计算)
- 四、实验数据记录 (或仿真及软件设计)
- 五、实验结果分析及回答问题 (或测试环境及测试结果)

# 实验七 大规模集成数字电路设计 ——三道交通系统控制显示器

## 一、 实验目的

- 1、熟悉大规模集成数字电路的设计方法。
- 2、熟悉数字系统调试及故障排除方法。

## 二、 实验原理



三、实验仪器

1、数字逻辑电路实验箱+CPLD 开发板 1 块

四、实验内容及步骤

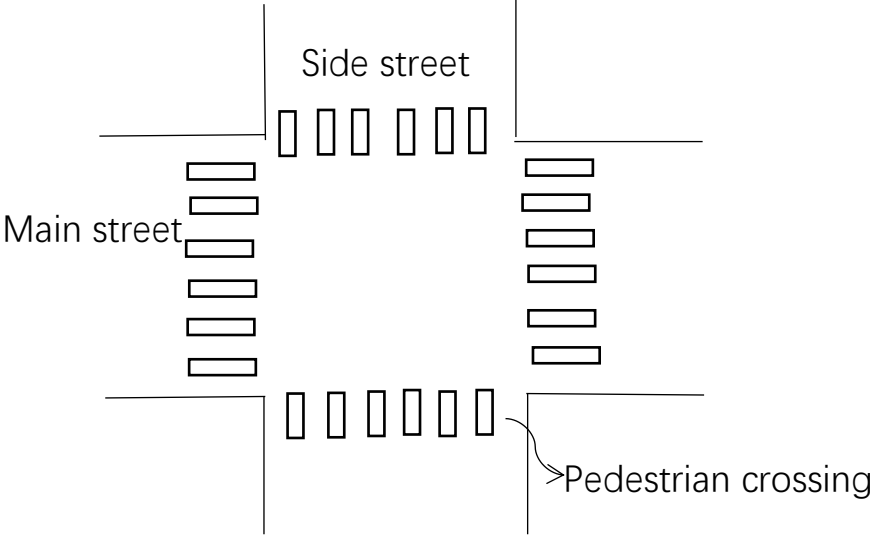
1. 实验名称

三道交通系统控制显示器

2. 设计内容

(1) 交通灯控制主干道、辅路，人行道。主干道与辅路需要红、绿、黄灯控制；人行道仅需由红绿灯控制；

(2) 设计示意图



(3) 流程表

时间	4s	2s	3s	1s	2s
主干道	绿灯亮	黄灯亮	红灯亮	红灯亮	红灯亮
辅路	红灯亮	红灯亮	绿灯亮	黄灯亮	红灯亮
人行道	红灯亮	红灯亮	红灯亮	红灯亮	绿灯亮
总时间	4s	6s	9s	10s	12s

时间	1s	1s	1s	1s	
主干道	红灯亮	红灯亮	红灯亮	红灯亮	
辅路	红灯亮	红灯亮	红灯亮	红灯亮	
人行道	红灯亮	全灭	红灯亮	全灭	
总时间	13s	14s	15s	16s	

### 3. 源代码

```

LIBRARY ieee;
USE ieee.std_logic_1164.ALL;
ENTITY jiaotong IS
PORT
(clk,reset: IN STD_LOGIC;
Ga,Ya,Ra,Gb,Yb,Rb,Gw,Rw,Fa: OUT STD_LOGIC;
    led7A:out std_logic_vector(6 downto 0);    --主干道
    led7s1:out std_logic_vector(6 downto 0);    --支路 十位
    led7B:out std_logic_vector(6 downto 0);    --支路
    led7C:out std_logic_vector(6 downto 0));    --红绿灯
END jiaotong;

ARCHITECTURE art OF jiaotong IS
type state_type is (s0,s1,s2,s3,s4,s5,s6,s7,s8);
SIGNAL state : state_type;

BEGIN

change_state:PROCESS (CLK) --CLK 为敏感信号量
VARIABLE s: integer range 0 to 255; --s 为秒定时器
VARIABLE clr,en :bit; --clr,en 为秒定时器使能信号

BEGIN
IF reset = '1' THEN state<=S0;clr:='0';    --clr 为 0，定时器清零
led7A<="1100110";    --04
led7B<="1111101";    --06
led7C<="1110110";    --x
ELSIF (clk'event and clk='1') THEN    --上升沿
IF clr='0' THEN s:=0; -- clr 为 0，定时器清零
ELSIF en='0' THEN s:=s; -- en 为 0，定时器保持不变
ELSE s:=s+1; --clr 为 1，en 为 1 时，正常计时

```

```

END IF;

CASE state IS
WHEN S0=>
IF (state<=S0 and S=0) THEN state<=S0;clr:='1';en:='1';
led7A<="1001111"; --03
led7B<="1101101"; --05
led7C<="1110111"; --09
elsif (state<=S0 and S=1) THEN state<=S0;clr:='1';en:='1';--s=1
led7A<="1011011"; --02
led7B<="1100110"; --04
led7C<="1111111"; --08
elsif (state<=S0 and S=2) THEN state<=S0;clr:='1';en:='1';--s=2
led7A<="0000110"; --01
led7B<="1001111"; --03
led7C<="0000111"; --07
ELSE
--s=3
led7A<="1011011"; --02
led7B<="1011011"; --02
led7C<="1111101"; --06
state<=S1;clr:='0';en:='0'; --s=0 且保持
END IF;
WHEN S1=>

IF (state<=S1 and S=0) THEN state<=S1;clr:='1';en:='1';--已 过 4s， 转换到 S2 状态
led7A<="0000110"; --01
led7B<="0000110"; --01
led7C<="1101101"; --05
ELSE
led7A<="1110110"; --状态 s2 x
led7B<="1001111"; --03
led7C<="1100110"; --04
state<=S2;clr:='0';en:='0';
END IF;
WHEN S2=>
IF (state<=S2 and S=0) THEN state<=S2;clr:='1';en:='1';--已 过 4s， 转换到 S2 状态
led7A<="1110111"; --9
led7B<="1011011"; --02
led7C<="1001111"; --03
elsif (state<=S2 and S=1) THEN state<=S2;clr:='1';en:='1';--已 过 4s， 转换到 S2 状态
led7A<="1111111"; --8
led7B<="0000110"; --01
led7C<="1011011"; --02
ELSE
led7A<="0000111"; --7
led7B<="0000110"; --01
led7C<="0000110"; --01

```

```

state<=S3;clr:='0';en:='0';--未到 4s, 维持 S1 状态
END IF;
WHEN S3=>
led7A<="1111101"; --6
led7s1<="0000110"; --01
led7B<="1011011"; --02
led7C<="0000110"; --01
state<=S4;clr:='0';en:='0';--未到 4s, 维持 S1 状态
WHEN S4=> IF (state<=S4 and S=0) THEN state<=S4;clr:='1';en:='1';--已过 4s, 转换到 S2 状态
led7A<="1101101"; --5
led7s1<="0000110"; --01
led7B<="0000110"; --01
led7C<="1011011"; --02
ELSE
led7A<="1100110"; --4
led7s1<="0000000"; --无
led7B<="1110110"; --x
led7C<="0000110"; --01
state<=S5;clr:='0';en:='0';--未到 4s, 维持 S1 状态
END IF;
WHEN S5=>
led7A<="1001111"; --3
led7B<="1101111"; --09
led7C<="0000110"; --01
state<=S6;clr:='0';en:='0';--未到 4s, 维持 S1 状态
WHEN S6=>
led7A<="1011011"; --2
led7s1<="0000000"; --non
led7B<="1111111"; --08
led7C<="0000000"; --00
state<=S7;clr:='0';en:='0';--未到 4s, 维持 S1 状态
WHEN S7=>
led7A<="0000110"; --1
led7B<="0000111"; --07
led7C<="0000110"; --01
state<=S8;clr:='0';en:='0';--未到 4s, 维持 S1 状态
WHEN S8=>
led7A<="0111111"; --
led7B<="1111101"; --06
led7C<="1110110"; --00
state<=S0;clr:='0';en:='0';--未到 4s, 维持 S1 状态
END CASE;
END IF;
END PROCESS change_state;

output_process:PROCESS (state)
BEGIN

```

CASE state IS

WHEN S0 => Ga<='1';Ya<='0';Ra<='0';Gb<='0';Yb<='0';Rb<='1';Gw<='0';Rw<='1';Fa<='0';--主干道绿灯亮,支道红灯亮

WHEN S1 => Ga<='0';Ya<='1';Ra<='0';Gb<='0';Yb<='0';Rb<='1';Gw<='0';Rw<='1';Fa<='0';--主干道黄灯亮,支道红灯亮

WHEN S2 => Ga<='0';Ya<='0';Ra<='1';Gb<='1';Yb<='0';Rb<='0';Gw<='0';Rw<='1';Fa<='0';--主干道红灯亮,支道绿灯亮

WHEN S3 => Ga<='0';Ya<='0';Ra<='1';Gb<='0';Yb<='1';Rb<='0';Gw<='0';Rw<='1';Fa<='0';--主干道红灯亮,支道黄灯亮

WHEN S4 => Ga<='0';Ya<='0';Ra<='1';Gb<='0';Yb<='0';Rb<='1';Gw<='1';Rw<='0';Fa<='0';--主干道红灯亮,支道黄灯亮

WHEN S5 => Ga<='0';Ya<='0';Ra<='1';Gb<='0';Yb<='0';Rb<='1';Gw<='0';Rw<='1';Fa<='0';--主干道红灯亮,支道黄灯亮

WHEN S6 => Ga<='0';Ya<='0';Ra<='1';Gb<='0';Yb<='0';Rb<='1';Gw<='0';Rw<='0';Fa<='1';--主干道红灯亮,支道黄灯亮

WHEN S7 => Ga<='0';Ya<='0';Ra<='1';Gb<='0';Yb<='0';Rb<='1';Gw<='0';Rw<='1';Fa<='1';--主干道红灯亮,支道黄灯亮

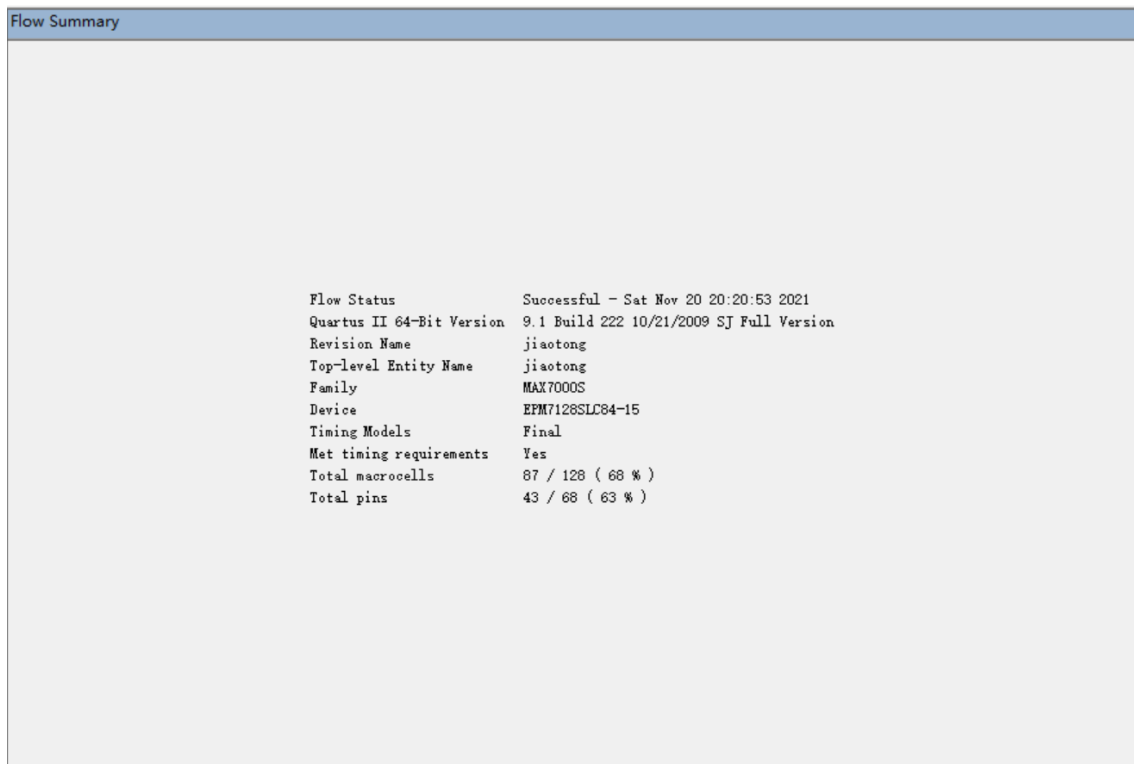
WHEN S8 => Ga<='0';Ya<='0';Ra<='1';Gb<='0';Yb<='0';Rb<='1';Gw<='0';Rw<='0';Fa<='0';--主干道红灯亮,支道黄灯亮

END CASE;

END PROCESS output\_process;

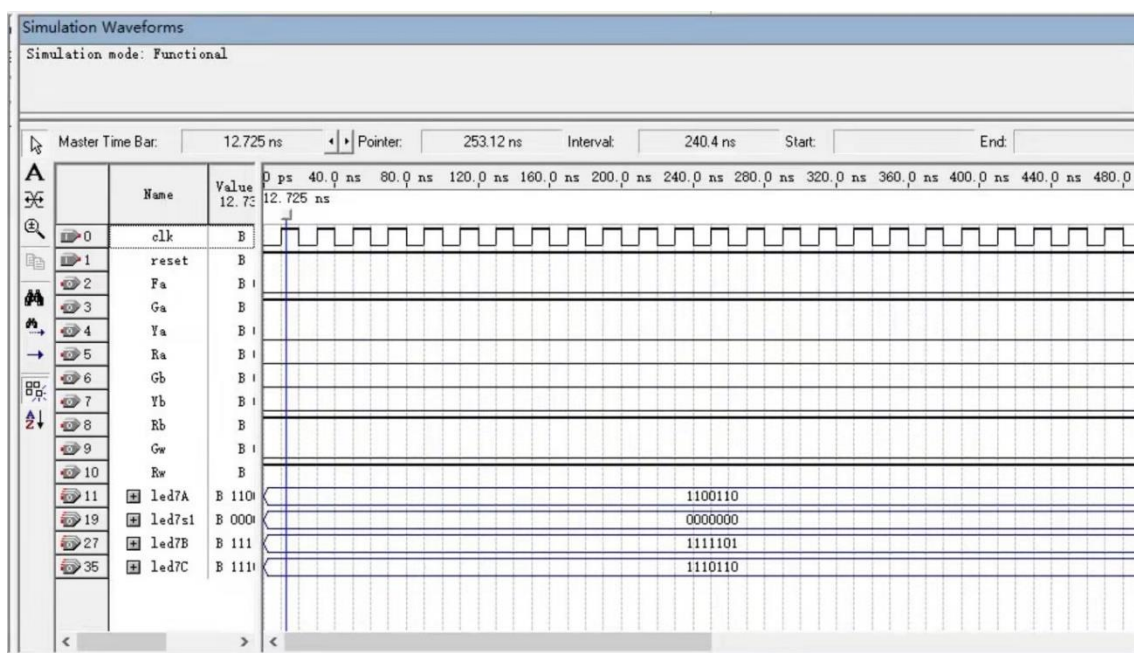
END art;

#### 4.flow Summmary 截图



## 5. 波形解释

(1) reset 为复位功能，当 reset 为高电平时，保持初状态。



(2) clk 为时钟信号，每一周期代表 1s, Ga, Ra, Ya 表示主干道的绿灯、红灯、黄灯，高电平时亮；Gb, Rb, Yb 表示辅路的绿灯、红灯、黄灯，高电平时亮；Gw, Rw 表示人行道的绿灯和红灯，高电平时亮。因为交





0000110	1011011	1001111	1100110	1101101
6	7	8	9	X (10)
1111101	0000111	1111111	1101111	1110110

## 五、设计优势

### 1. 研制成本

该交通灯系统完全采用 VHDL 语言设计，仅需购买开发板及少量杜邦线就能实现，成本较低，且能通过改动代码实现主干道（main street）、辅干道（side street）和人行道（pedestrian crossing）相应的红灯、绿灯和黄灯的持续亮灯时间或闪烁方式，拥有可编辑性，性价比较高，可操作性强，应用范围广泛。

### 2. 崭新点

该红绿灯交通系统具有复位功能，可一键重置状态，以在程序“跑飞”后能重新复位，增强了系统的稳定性。相比于传统大型红绿灯交通系统，该系统虽较为简单但基础功能均能实现，且可通过改动代码管控不同道路的交通灯的不同状态，易于操作，可用作教学用途，增强学生对交通灯系统的理解及对 VHDL 语言的掌握，也可用于用于儿童益智玩具中，寓教于乐。

## 六、自评成绩

李艺超 20049200519	100
靳钰婷 20049200170	76
纪诗琪 20049200061	76
王之韵 20049200393	76

马雨琦 20049200240	76
杨欣睿 20049200229	76