

## 目 录

一 实验要求 .....	2
参考内容 .....	3
数码管显示控制电路 .....	3
彩灯控制器电路设计 .....	5
利用 QuartusII 内建宏实现 1Hz 频率的时钟信号 .....	11
DE0 外接引脚说明 .....	20

# 一 实验要求

**要求 1:** 参照参考内容，用 QuartusII 软件内嵌宏函数 `lpm_counter` 实现 50M 分频，输出频率为 1Hz 秒脉冲信号，用实验板上绿色 LED 灯观察。

**要求 2:** 参照参考内容中数码管显示控制电路设计方法，用 74161 二进制计数器、7447 七段译码器和若干门电路，用原理图输入方法实现在一个 7 段数码管上显示序列：学号。

**要求 3:** 参照参考内容，用 74161 二进制计数器、74194 移位寄存器和若干门电路，用原理图输入方法实现彩灯控制器电路设计。

**验收要求:** 将要求 2 和要求 3 同时在实验电路上实现，验收时能够说明电路设计的原理。

**注:**如果有同学用的电脑软件出现 Megafunction 无法启用，可利用绑定按键开关作为时钟信号，验收时需要演示波形仿真结果。

# 参考内容

## 数码管显示控制电路

### 一 实验要求

能自动循环显示数字 0, 1, 2, 3, 4, 1, 3, 0, 2, 4

### 二 实验原理:

- 1. 利用 74LS90、74LS00、74LS20 实现输出序列逻辑;
- 2. 经过卡诺图化简实现码制转换实现所需序列;
- 3. 用 74LS47 驱动七段译码管，共阳极数码管显示。

### 三 实验设计过程:

1) 74LS90 产生十进制 (5421BCD) 计数器和所设计的 0, 1, 2, 3, 4, 1, 3, 0, 2, 4 序列。对应如下表所示

Q <sub>A</sub>	Q <sub>D</sub>	Q <sub>C</sub>	Q <sub>B</sub>	F <sub>D</sub>	F <sub>C</sub>	F <sub>B</sub>	F <sub>A</sub>
0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	1
0	0	1	0	0	0	1	0
0	0	1	1	0	0	1	1
0	1	0	0	0	1	0	0
1	0	0	0	0	0	0	1
1	0	0	1	0	0	1	1
1	0	1	0	0	0	0	0
1	0	1	1	0	0	1	0
1	1	0	0	0	1	0	0

2) 卡诺图如下 :

F<sub>B</sub> :

Q <sub>A</sub> Q <sub>D</sub> \ Q <sub>C</sub> Q <sub>B</sub>	00	01	11	10
00	0	0	×	0
01	0	×	0	1
11	1	×	×	1
10	1	×	×	0

$$F_B = \bar{Q}_A Q_C + Q_A \bar{Q}_D Q_B$$

F<sub>A</sub> :

$Q_A Q_D \backslash Q_C Q_B$	0 0	01	11	10
00	0	0	×	1
01	1	×	0	1
11	1	×	×	0
10	0	×	×	0

$$F_A = \bar{Q}_A Q_B + Q_A \bar{Q}_D \bar{Q}_C$$

由真值表直接可以看出：

$$F_C = Q_D$$

$$F_D = 0$$

因此实现的逻辑表达式为：

$$F_A = \bar{Q}_A Q_B + Q_A \bar{Q}_D \bar{Q}_C$$

$$F_B = \bar{Q}_A Q_C + Q_A \bar{Q}_D Q_B$$

$$F_C = Q_D$$

$$F_D = 0$$

#### 四 整体原理电路图

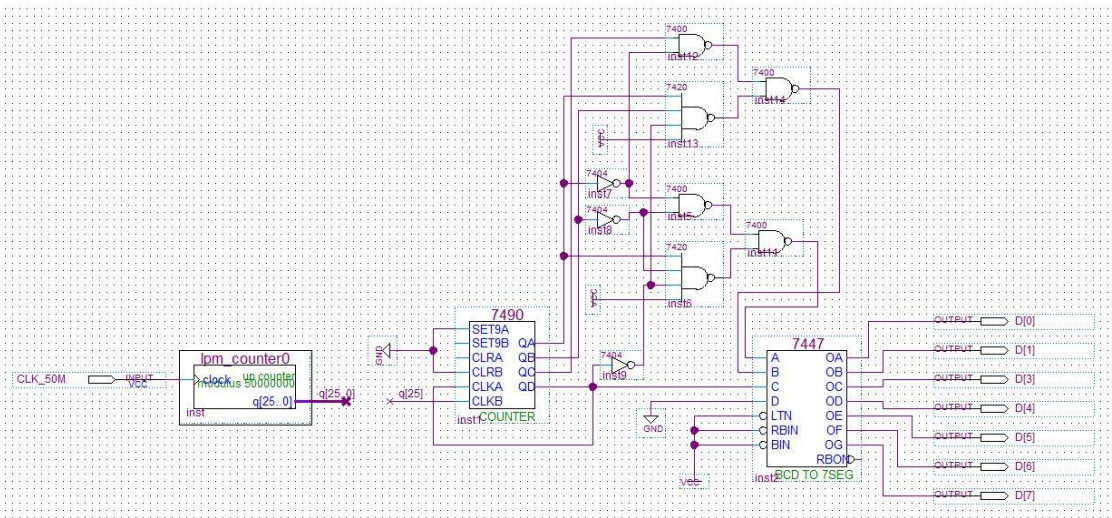


图 1 数码管显示控制电路图

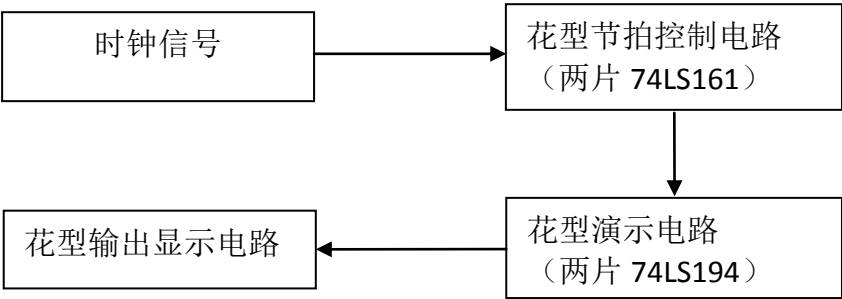
# 彩灯控制器电路设计

## 一 实验要求

设计一个能够控制八路彩灯的逻辑电路。要求彩灯组成四种种花型，花型 I ——由两边向中间对称性依次亮，全亮后仍由两边向中间依次灭；花型 II ——8 路灯分两半，从左自右顺次亮，再顺次灭；花型 III ——8 路灯分两半，从右向左顺次亮，再从右向左顺次灭；花型 IV ——由中间向两边对称性一次亮，全亮后仍由中间向两边依次灭。并且要求这四种花型循环出现。

## 二 工作原理

总体电路分为四大模块：模块一由 Quartus 宏函数 lpm\_counter 提供时钟脉冲信号；模块二花型节拍控制电路由两片 74LS161 组成一个 32 进制计数器；模块三花型演示电路由两片 74LS194 来控制花型；模块四花型输出显示电路。总体原理框图如下：



2.1 节拍控制电路：由两片 74LS161 四位二进制同步计数器完成。除了有二进制加法计数功能外, 还具有异步清零、同步并行置数、保持等功能，如表一所示

表一 74LS161 功能表

CLK	CLR $\overline{N}$ '	LD $\overline{N}$ '	ENP	ENT	工作状态
*	0	*	*	*	置零
$\uparrow$	1	0	*	*	预置数
*	1	1	0	1	保持

↑	1	1	* 0	保持（但 C = 0）
↑	1	1	1 1	计数

当 CLRN'、LDN'、ENP、ENT 等于 1 时 74LS161 实现十六进制计数功能，记录 16 个状态。两片 74LS161 级联实现从 000000 到 011111 计数功能，再利用 74LS161(1)的 Q3 对 74LS194(2)的控制端 S1,S0 进行控制，以及 74LS161（2）的 Q0 对 74LS194(1)的控制端 S1,S0 进行控制。

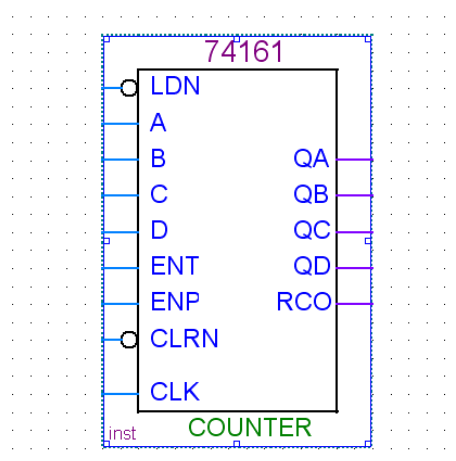


图 2 74LS161 引脚图

## 2.2 花型演示电路：

74LS194 是一个 4 位双向移位寄存器，它具有左移，右移，保持，清零等如表二所示

表二 74LS194 功能表

RD'	S1	S0	工作状态
0	*	*	置零
1	0	0	保持
1	0	1	右移
1	1	0	左移
1	1	1	并行输入

在彩灯控制电路设计中只用了 74LS194 左移右移功能，来实现彩灯的花型，双向移位寄存器 74LS194 的控制端 S1=0,S0=1 时,进行右移,S1=1,S0=0 时,进行

左移。

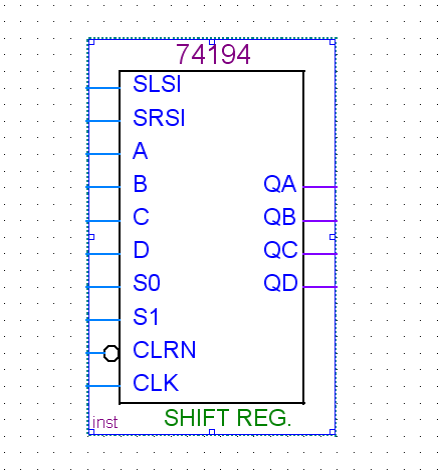


图 3 74LS194 引脚图

三 设计步骤及方法

3.1 8 位彩灯分为两部分

8 位彩灯分为 4 个一组，用两个 74LS194 来实现，花型 I ——由两边向中间对称性依次亮，全亮后仍由两边向中间依次灭；花型 II ——8 路灯分两半，从左自右顺次亮，再顺次灭；花型 III ——8 路灯分两半，从右向左顺次亮，再从右向左顺次灭；花型 IV ——由中间向两边对称性依次亮，全亮后仍由中间向两边依次灭，所以通过对花型的分析可知，其中双向移位寄存器 74LS194（1）的功能是前 16 节拍右移，后 16 节拍左移即先是 S1=0, S0=1, 后变成 S1=1, S0=0 . 而 74LS194(2)则前 16 节拍为先左移后右移后 16 节拍也是先左移后右移。根据分析，画出图表，如表三、表四所示。

3.2 低四位彩灯控制电路设计

表三 74LS194(1)控制状态表

两片 74LS161	74LS194(1)		花型
74161 (2) 74161 (1)	QA---QD	S1 S0	
QA QDQCQBQA			
0 0000	1000	0 1	从左向右亮（花型 I）
0 0001	1100	0 1	从左向右亮（花型 I）
0 0010	1110	0 1	从左向右亮（花型 I）
0 0011	1111	0 1	从左向右亮（花型 I）

0	0100	0111	0 1	从左向右灭（花型 I）
0	0101	0011	0 1	从左向右灭（花型 I）
0	0110	0001	0 1	从左向右灭（花型 I）
0	0111	0000	0 1	从左向右灭（花型 I）
0	1000	1000	0 1	从左向右亮（花型 II）
0	1001	1100	0 1	从左向右亮（花型 II）
0	1010	1110	0 1	从左向右亮（花型 II）
0	1011	1111	0 1	从左向右亮（花型 II）
0	1100	0111	0 1	从左向右灭（花型 II）
0	1101	0011	0 1	从左向右灭（花型 II）
0	1110	0001	0 1	从左向右灭（花型 II）
0	1111	0000	0 1	从左向右灭（花型 II）
1	0000	0001	1 0	从右向左亮（花型 III）
1	0001	0011	1 0	从右向左亮（花型 III）
1	0010	0111	1 0	从右向左亮（花型 III）
1	0011	1111	1 0	从右向左亮（花型 III）
1	0100	1110	1 0	从右向左灭（花型 III）
1	0101	1100	1 0	从右向左灭（花型 III）
1	0110	1000	1 0	从右向左灭（花型 III）
1	0111	0000	1 0	从右向左灭（花型 III）
1	1000	0001	1 0	从右向左亮（花型 IV）
1	1001	0011	1 0	从右向左亮（花型 IV）
1	1010	0111	1 0	从右向左亮（花型 IV）
1	1011	1111	1 0	从右向左亮（花型 IV）
1	1100	1110	1 0	从右向左灭（花型 IV）
1	1101	1100	1 0	从右向左灭（花型 IV）
1	1110	1000	1 0	从右向左灭（花型 IV）
1	1111	0000	1 0	从右向左灭（花型 IV）

电路图中 74LS161(2) 的 QA 取反与 74LS194(1) 的 S0 连在一起，74LS161(2)



的 QA 和 S1 直接连接，74LS194(1)的 QA 取反与本片的左移输入端 SL 连在一起，QD 取反与 SR 连在一起。

3.3 高四位彩灯控制电路设计

表四 74LS194（2）控制状态表

两片 74LS161	74LS194(2)		花型
74161（2）74161（1） QA        QDQCQBQA	QA-----QD	S1    S0	
0        0000	0001	1    0	从右向左亮（花型 I）
0        0001	0011	1    0	从右向左亮（花型 I）
0        0010	0111	1    0	从右向左亮（花型 I）
0        0011	1111	1    0	从右向左亮（花型 I）
0        0100	1110	1    0	从右向左灭（花型 I）
0        0101	1100	1    0	从右向左灭（花型 I）
0        0110	1000	1    0	从右向左灭（花型 I）
0        0111	0000	1    0	从右向左灭（花型 I）
0        1000	1000	0    1	从左向右亮（花型 II）
0        1001	1100	0    1	从左向右亮（花型 II）
0        1010	1110	0    1	从左向右亮（花型 II）
0        1011	1111	0    1	从左向右亮（花型 II）
0        1100	0111	0    1	从左向右灭（花型 II）
0        1101	0011	0    1	从左向右灭（花型 II）
0        1110	0001	0    1	从左向右灭（花型 II）
0        1111	0000	0    1	从左向右灭（花型 II）
1        0000	0001	1    0	从右向左亮（花型 III）
1        0001	0011	1    0	从右向左亮（花型 III）
1        0010	0111	1    0	从右向左亮（花型 III）
1        0011	1111	1    0	从右向左亮（花型 III）
1        0100	1110	1    0	从右向左灭（花型 III）
1        0101	1100	1    0	从右向左灭（花型 III）

1	0110	1000	1 0	从右向左灭（花型 III）
1	0111	0000	1 0	从右向左灭（花型 III）
1	1000	1000	0 1	从左向右亮（花型 IV）
1	1001	1100	0 1	从左向右亮（花型 IV）
1	1010	1110	0 1	从左向右亮（花型 IV）
1	1011	1111	0 1	从左向右亮（花型 IV）
1	1100	0111	0 1	从左向右灭（花型 IV）
1	1101	0011	0 1	从左向右灭（花型 IV）
1	1110	0001	0 1	从左向右灭（花型 IV）
1	1111	0000	0 1	从左向右灭（花型 IV）

所以电路图中 74LS194(2) 的 QD 取反后连在右移输入端 SR 上，74LS194(2) 的 QA 取反后连在左移输入端 SL 上，74LS161(1) 的 QD 取反后连在 74LS194(2) 的 S1 上，74LS194(2) 的 S0 直接和 74LS161(1) 的 QD 连接。

#### 四 整体原理电路图

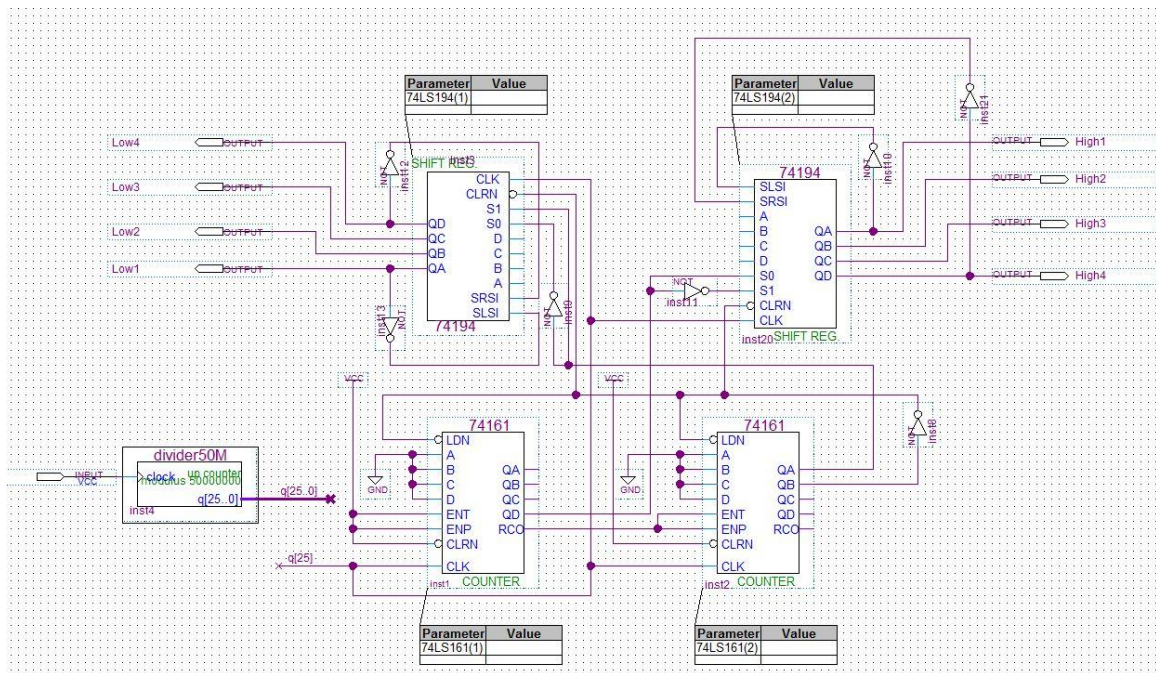
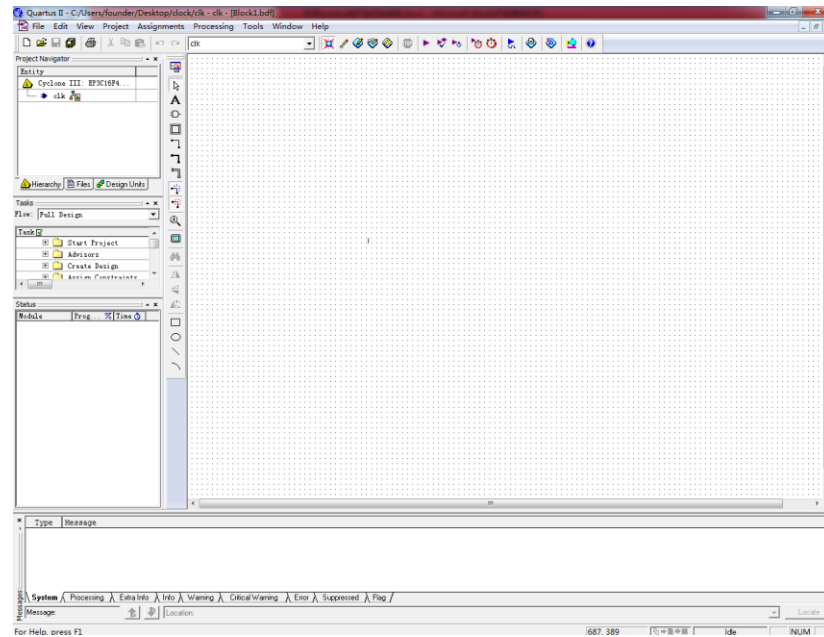


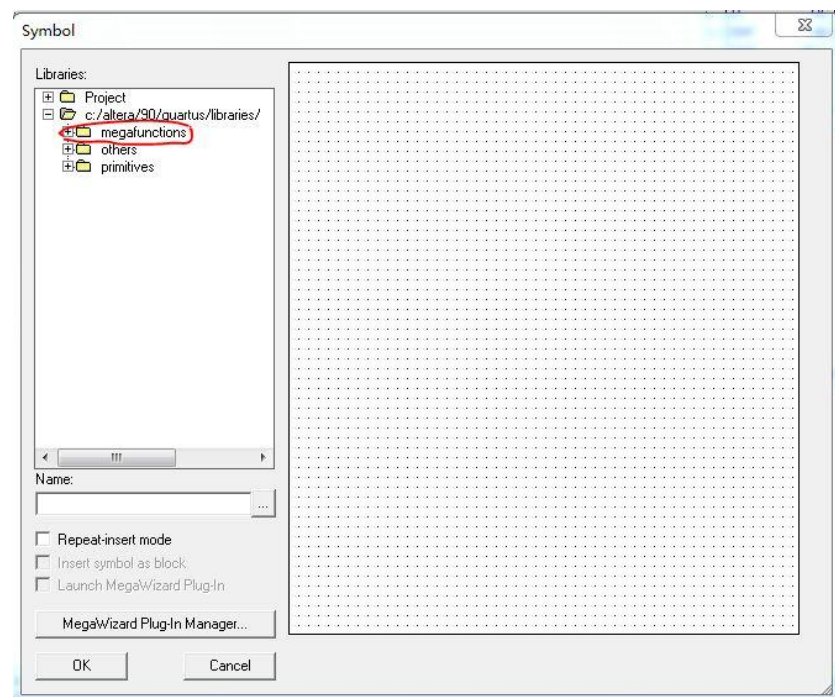
图 4 彩灯控制器电路图

# 利用 QuartusII 内建宏实现 1Hz 频率的时钟信号

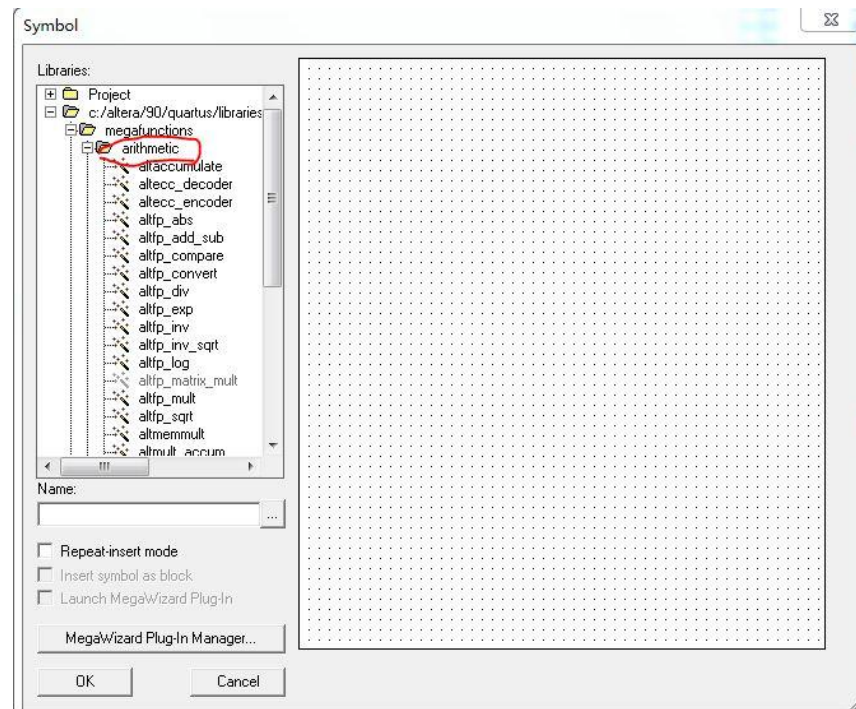
第一步：新建工程，并建立原理图文件



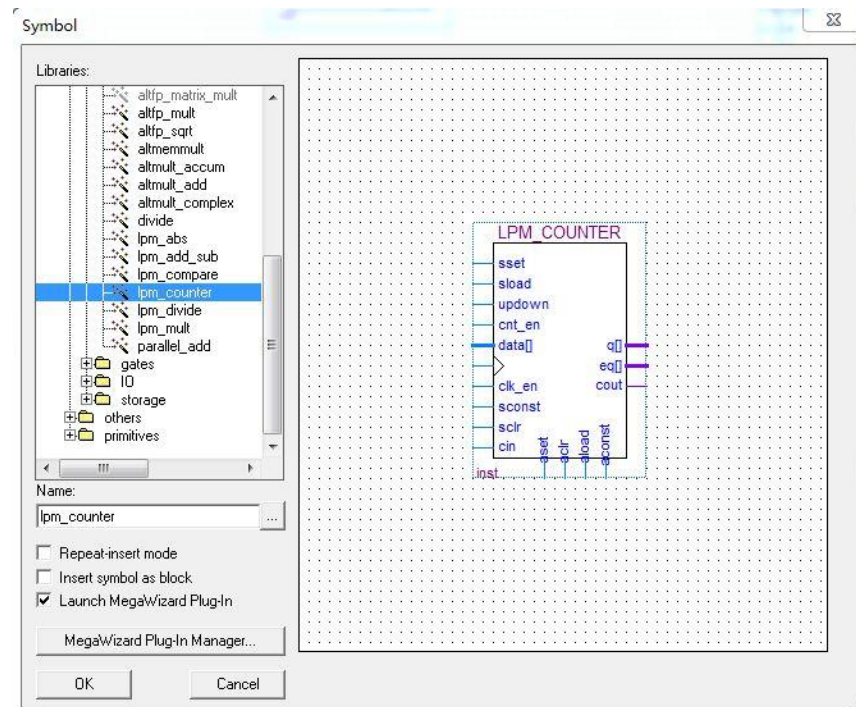
第二步：添加元件，选择“megafunction”



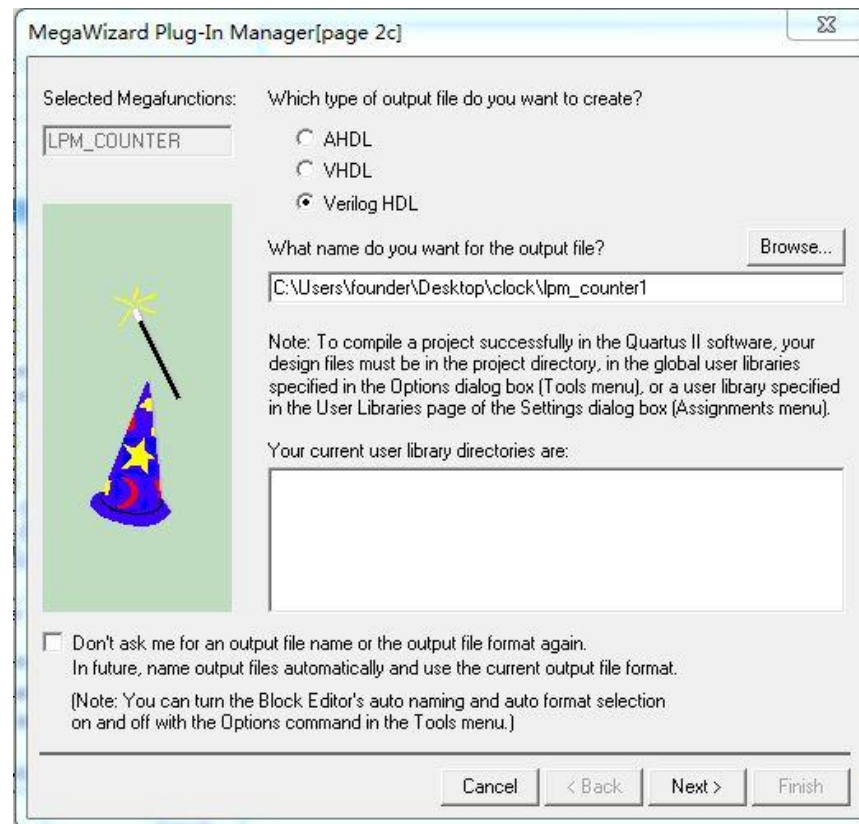
选择“arithmetic”



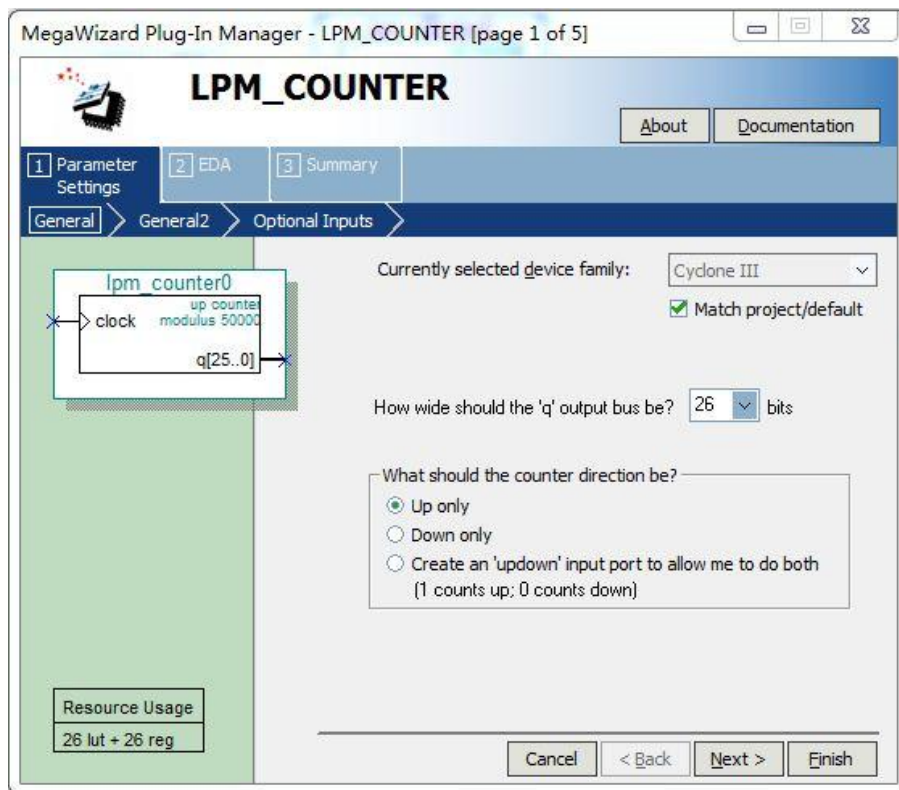
选择 lpm\_counter 宏后，单击 OK 按钮。



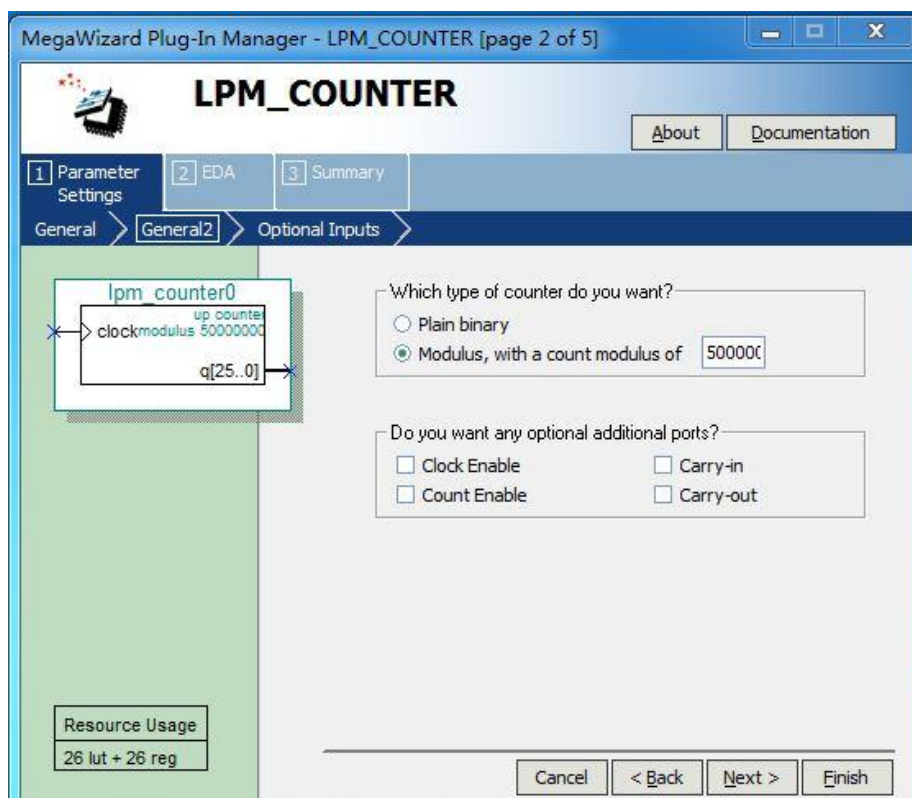
单击 “Next”



第三步：配置 lpm\_counter 宏，选择 “How wide should the ‘q’ output bus be?” 选择 **26bits**  
“What should the counter direction be?” 选择 “UP only”。

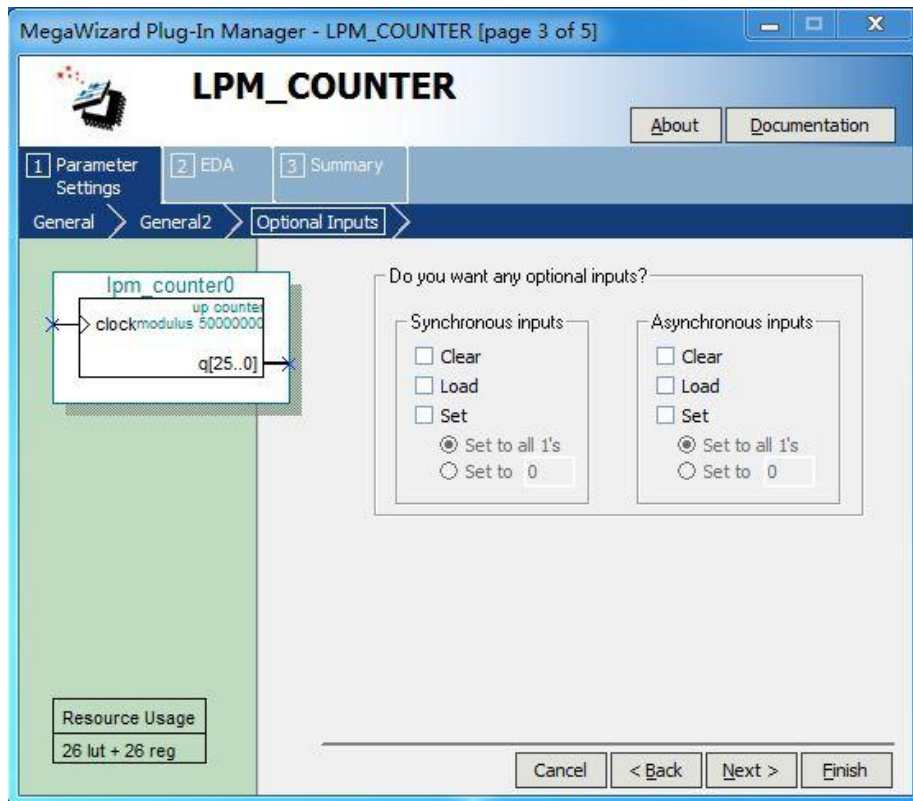


选择“Which type of counter do you want?”选择“**Module, with a count modulus of**”并输入“50,000,000”；“Do you want any optional additional ports?”不选择任何选项。

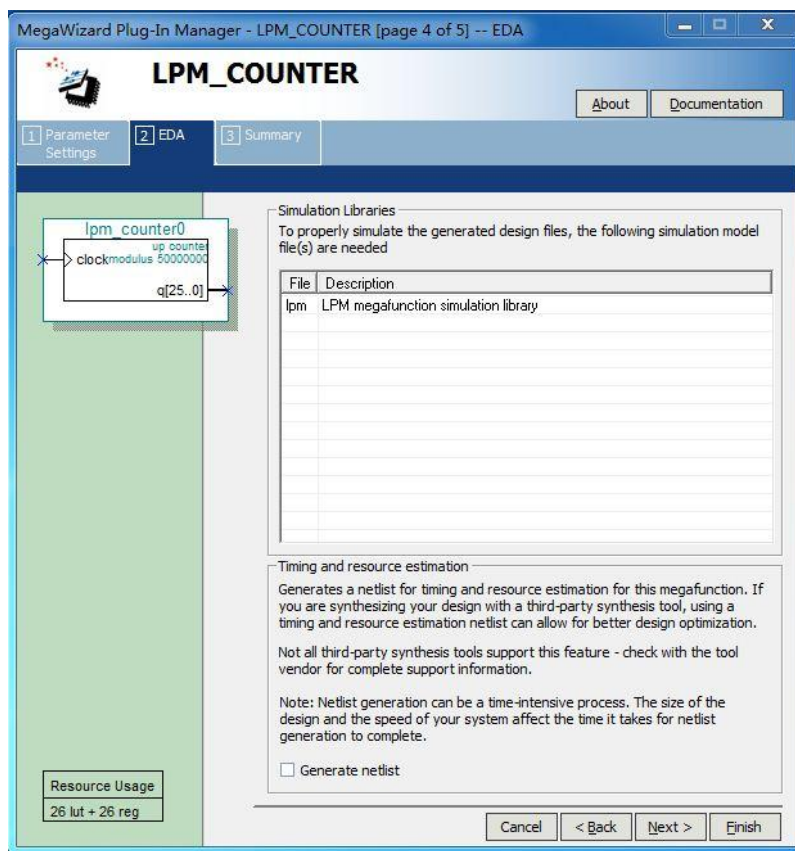


单击“Next”

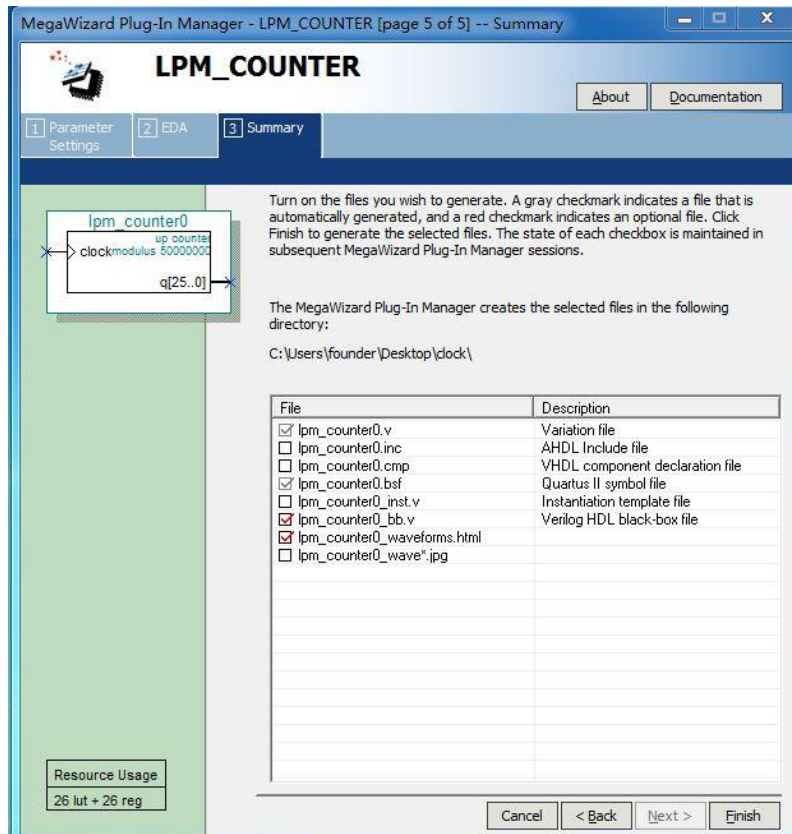




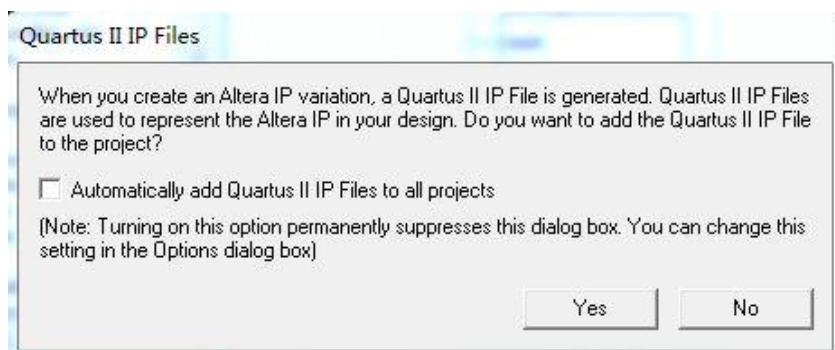
单击 “Next”



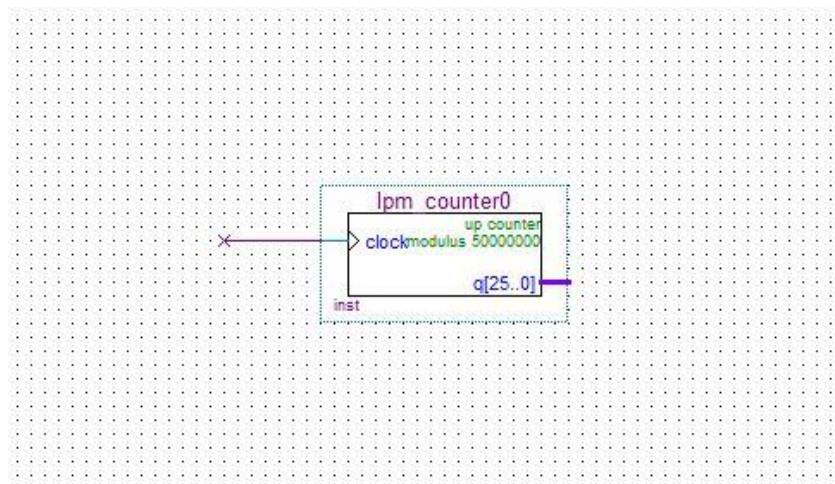
单击 “Finish”



单击 “Yes”

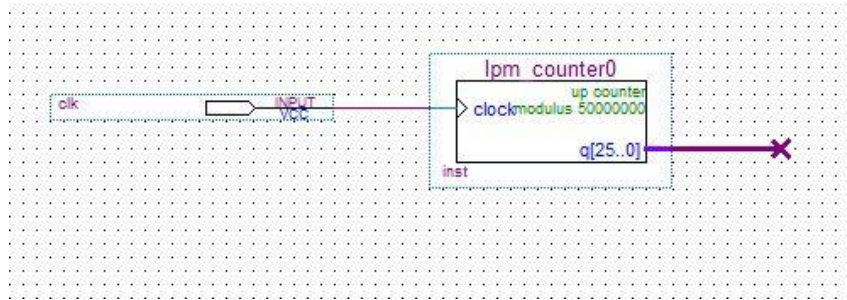


将生成的 lpm\_counter 的宏放置在原理图上。

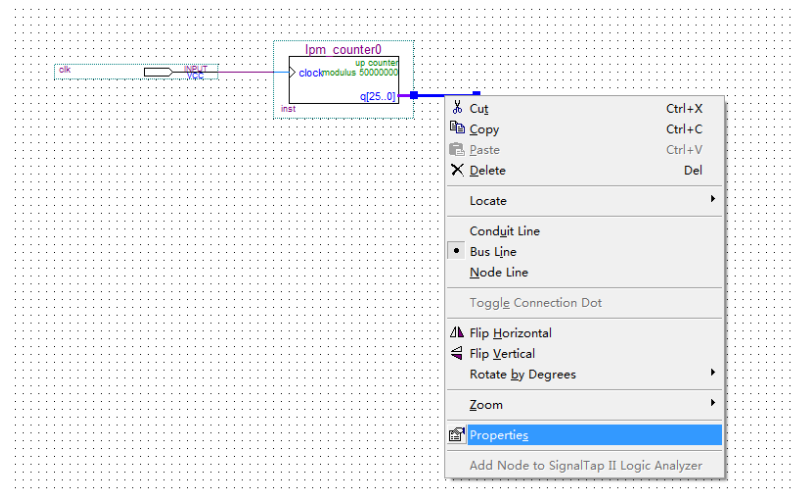




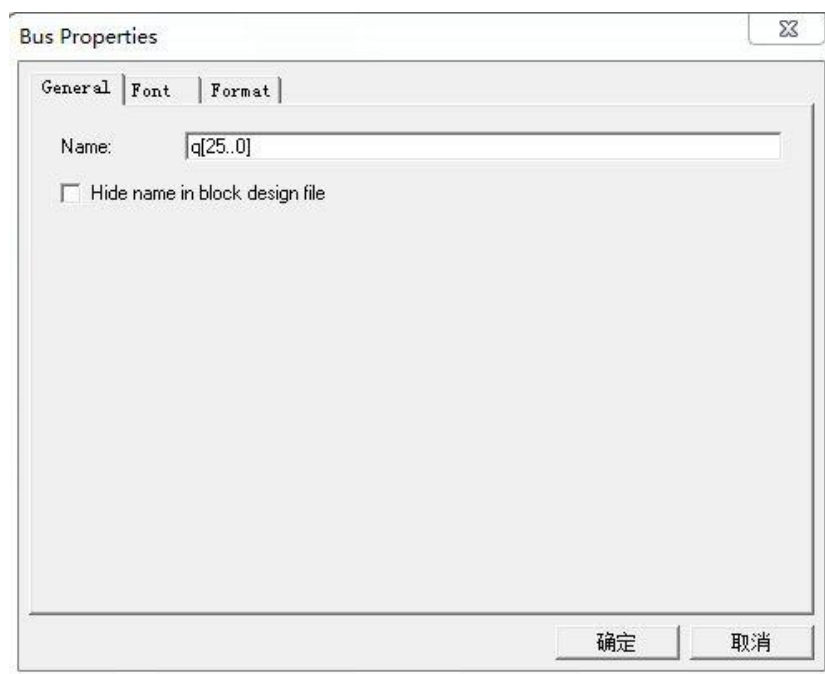
添加 Input 引脚，将 Input 端命名为 clk。在锁定引脚时将 clk 端和内部 50M 时钟（PIN\_G21 引脚）绑定。将宏 lpm\_counter0 的输出 q[25..0] 用总线连接，不用连接任何端口。

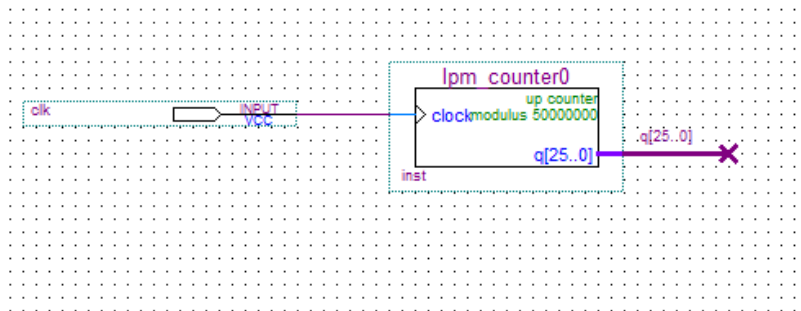


用鼠标右键单击该总线，从菜单栏中单击“Properties”选项。

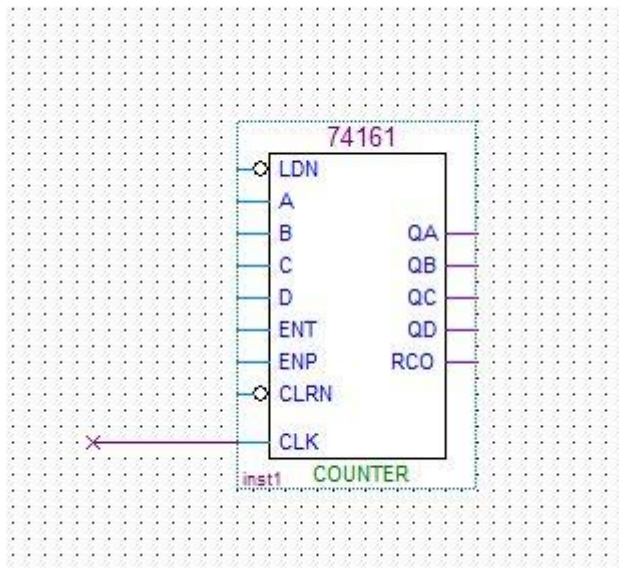


输入 q[25..0]，单击确定按钮。

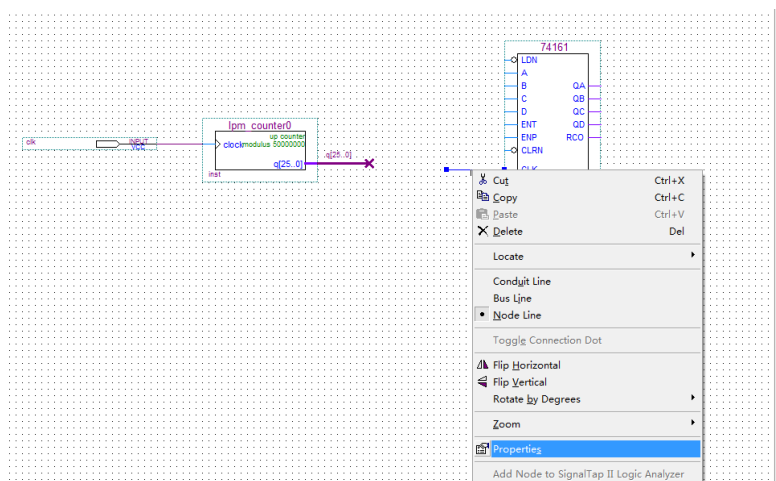




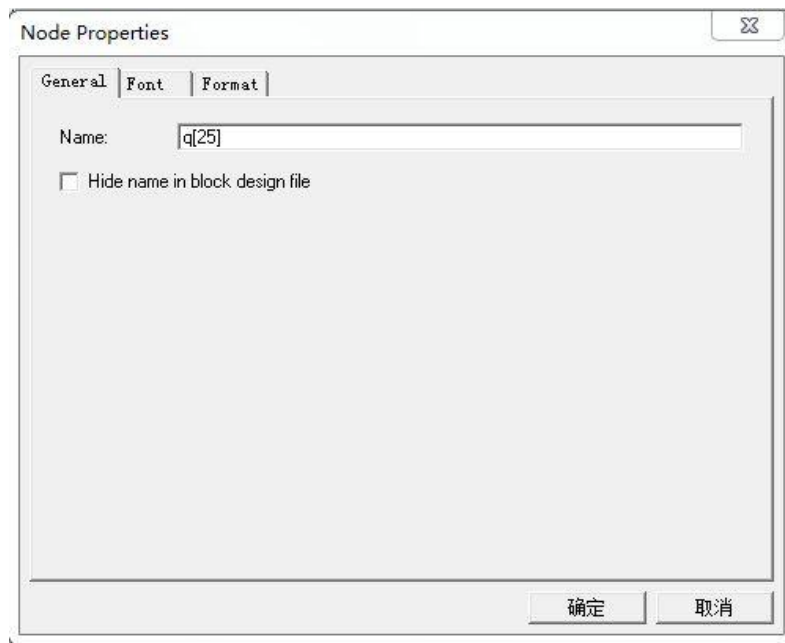
用同样的方法给下图 74161 的 CLK 端接入引线



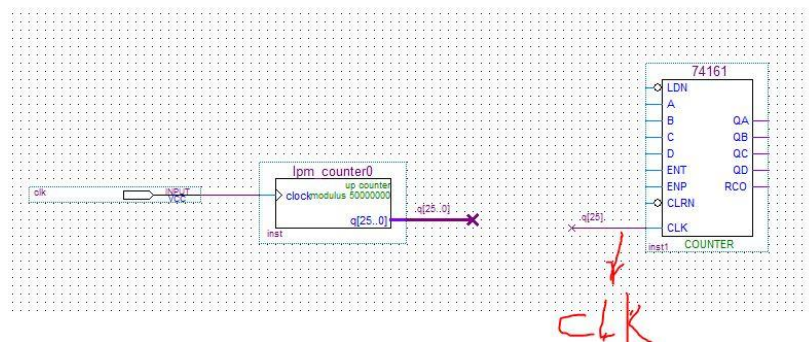
选中该引线右键单击，在弹出的菜单栏中选中“Properties”



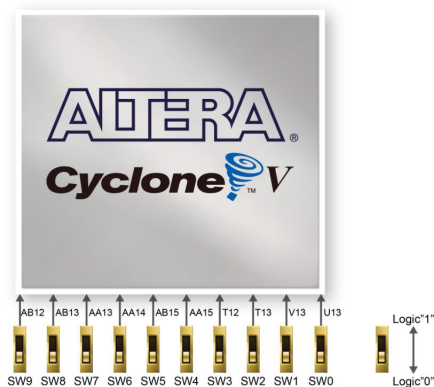
将该引线命名为 q[25]



此时 74161CLK 时钟端信号频率为 1Hz。



# DE0 外接引脚说明



拨动开关与Cyclone V FPGA之间的连接

拨动开关的引脚分配表

Signal Name	FPGA Pin No.	Description
SW0	PIN_U13	Slide Switch[0]
SW1	PIN_V13	Slide Switch[1]
SW2	PIN_T13	Slide Switch[2]
SW3	PIN_T12	Slide Switch[3]
SW4	PIN_AA15	Slide Switch[4]
SW5	PIN_AB15	Slide Switch[5]
SW6	PIN_AA14	Slide Switch[6]
SW7	PIN_AA13	Slide Switch[7]
SW8	PIN_AB13	Slide Switch[8]
SW9	PIN_AB12	Slide Switch[9]

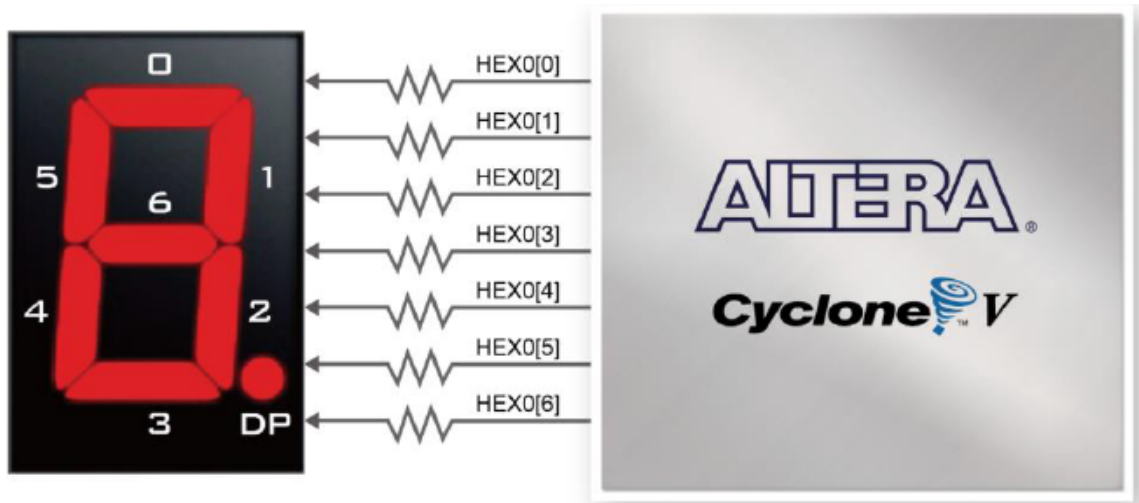


图5 发光二极管与Cyclone V FPGA之间的连接

表2 发光二极管的引脚分配表

Signal Name	FPGA Pin No.	Description
LEDR0	PIN_AA2	LED [0]
LEDR1	PIN_AA1	LED [1]
LEDR2	PIN_W2	LED [2]
LEDR3	PIN_Y3	LED [3]
LEDR4	PIN_N2	LED [4]
LEDR5	PIN_N1	LED [5]
LEDR6	PIN_U2	LED [6]
LEDR7	PIN_U1	LED [7]
LEDR8	PIN_L2	LED [8]
LEDR9	PIN_L1	LED [9]

板上提供了6个7段共阳极数码管，分别以HEX0到HEX5标注。它们如图所示方式连接到FPGA芯片上。当FPGA对应输出端口为低电平时，点亮数码管相应的段；当FPGA对应输出端口为高电平时，熄灭数码管相应的段。6个7段共阳极数码管的各段引脚与FPGA引脚引脚分配如表所示。



Signal Name	FPGA Pin No.	Description
HEX00	PIN_U21	Seven Segment Digit 0[0]
HEX01	PIN_V21	Seven Segment Digit 0[1]
HEX02	PIN_W22	Seven Segment Digit 0[2]
HEX03	PIN_W21	Seven Segment Digit 0[3]
HEX04	PIN_Y22	Seven Segment Digit 0[4]
HEX05	PIN_Y21	Seven Segment Digit 0[5]
HEX06	PIN_AA22	Seven Segment Digit 0[6]
HEX10	PIN_AA20	Seven Segment Digit 1[0]
HEX11	PIN_AB20	Seven Segment Digit 1[1]
HEX12	PIN_AA19	Seven Segment Digit 1[2]
HEX13	PIN_AA18	Seven Segment Digit 1[3]
HEX14	PIN_AB18	Seven Segment Digit 1[4]
HEX15	PIN_AA17	Seven Segment Digit 1[5]
HEX16	PIN_U22	Seven Segment Digit 1[6]
HEX20	PIN_Y19	Seven Segment Digit 2[0]
HEX21	PIN_AB17	Seven Segment Digit 2[1]
HEX22	PIN_AA10	Seven Segment Digit 2[2]
HEX23	PIN_Y14	Seven Segment Digit 2[3]
HEX24	PIN_V14	Seven Segment Digit 2[4]
HEX25	PIN_AB22	Seven Segment Digit 2[5]
HEX26	PIN_AB21	Seven Segment Digit 2[6]
HEX30	PIN_Y16	Seven Segment Digit 3[0]
HEX31	PIN_W16	Seven Segment Digit 3[1]
HEX32	PIN_Y17	Seven Segment Digit 3[2]
HEX33	PIN_V16	Seven Segment Digit 3[3]
HEX34	PIN_U17	Seven Segment Digit 3[4]
HEX35	PIN_V18	Seven Segment Digit 3[5]
HEX36	PIN_V19	Seven Segment Digit 3[6]

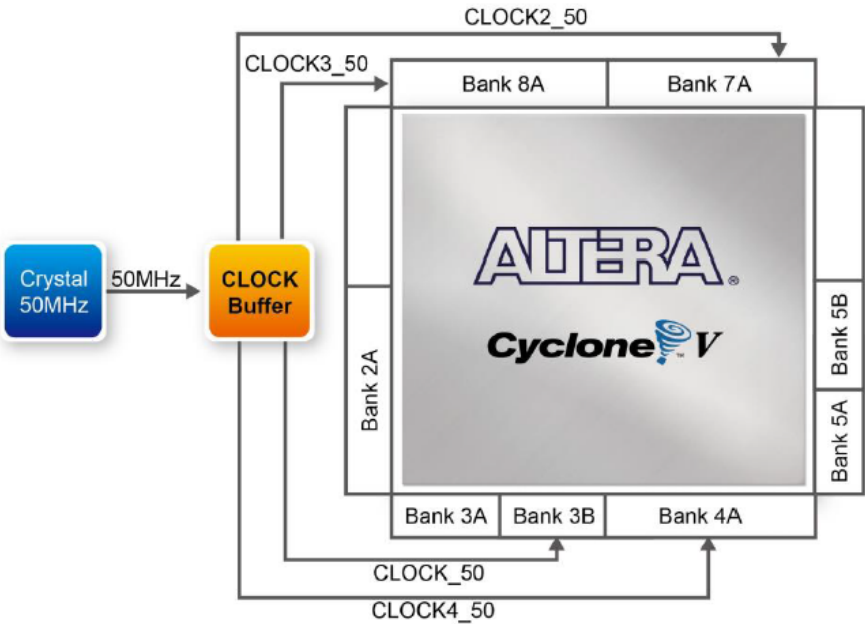


HEX40	PIN_U20	Seven Segment Digit 4[0]
HEX41	PIN_Y20	Seven Segment Digit 4[1]
HEX42	PIN_V20	Seven Segment Digit 4[2]
HEX43	PIN_U16	Seven Segment Digit 4[3]
HEX44	PIN_U15	Seven Segment Digit 4[4]
HEX45	PIN_Y15	Seven Segment Digit 4[5]
HEX46	PIN_P9	Seven Segment Digit 4[6]
HEX50	PIN_N9	Seven Segment Digit 5[0]
HEX51	PIN_M8	Seven Segment Digit 5[1]
HEX52	PIN_T14	Seven Segment Digit 5[2]
HEX53	PIN_P14	Seven Segment Digit 5[3]
HEX54	PIN_C1	Seven Segment Digit 5[4]
HEX55	PIN_C2	Seven Segment Digit 5[5]
HEX56	PIN_W19	Seven Segment Digit 5[6]

板上提供一个50MHz时钟信号，该时钟信号连接到FPGA中作为用户逻辑时钟使用。

实验板时钟分配框图如图8所示。时钟输入到FPGA的I/O引脚的相关引脚分配如表4所示。

注：实验中使用用PIN\_M9管脚。



时钟分配框图

FPGA时钟输入的引脚分配表

Signal Name	FPGA Pin No.	Description
CLOCK_50	PIN_M9	50 MHz clock input(Bank 3B)
CLOCK2_50	PIN_H13	50 MHz clock input(Bank 7A)
CLOCK3_50	PIN_E10	50 MHz clock input(Bank 8A)
CLOCK4_50	PIN_V15	50 MHz clock input(Bank 4A)





