

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 7490 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：

1) 当输入开关状态为逻辑 0 时,数码管依次显示序列本组一位组员学号的后四位；2) 当输入开关状态为逻辑 1 时,数码管依次显示序列本组另一位组员学号的后四位；（如果 1 人一组：1.输入开关状态为逻辑 0 时显示自己学号后四位，输入开关状态为逻辑 1 时显示自己学号后四位的逆序）

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 _A	Q _D	Q _C	Q _B	F _D	F _C	F _B	F _A
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_B :

Q _A Q _D \ Q _C Q _B	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_A :

$Q_A Q_D \backslash Q_C Q_B$	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

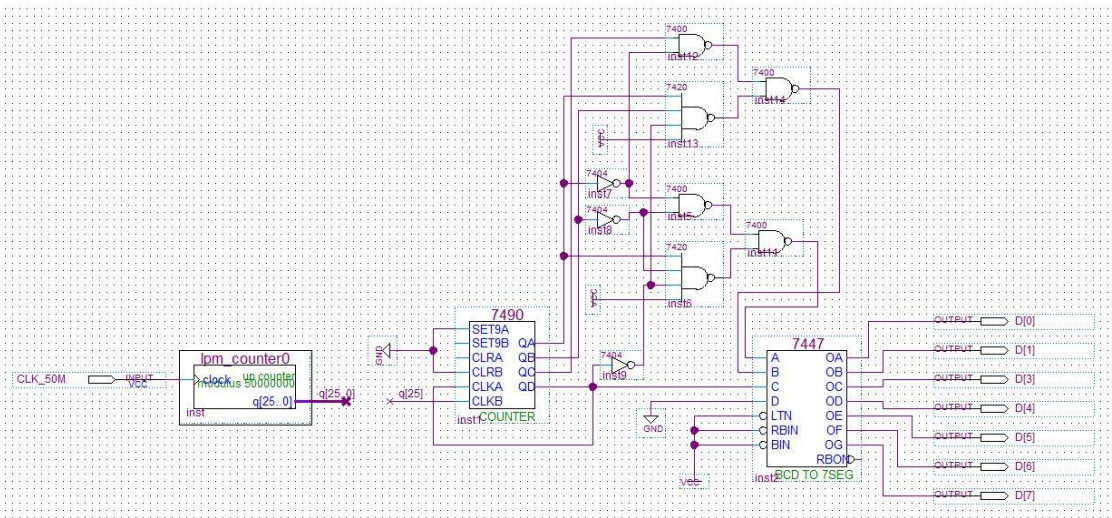


图 1 数码管显示控制电路图

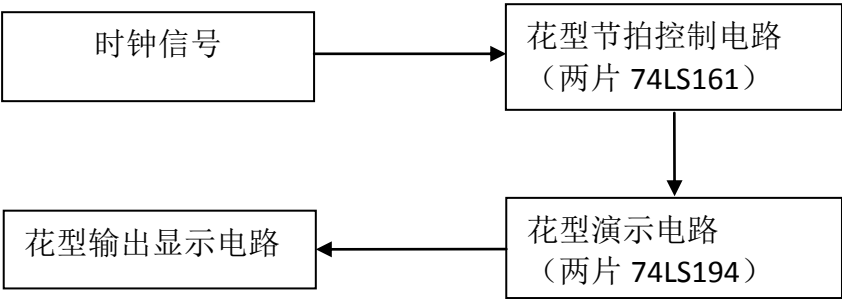
彩灯控制器电路设计

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。

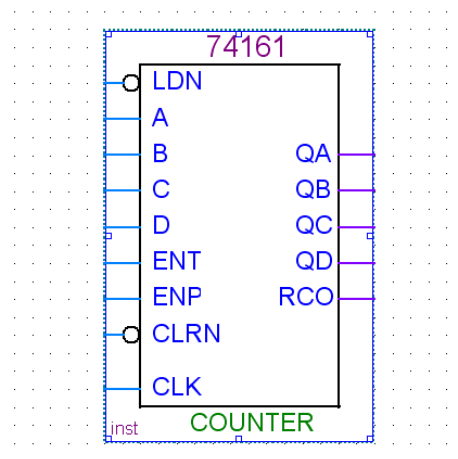


图 2 74LS161 引脚图

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。

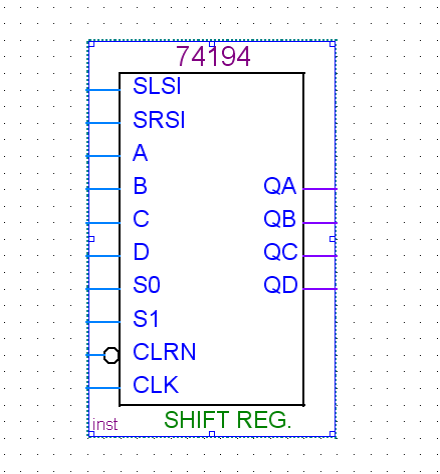


图 3 74LS194 引脚图

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

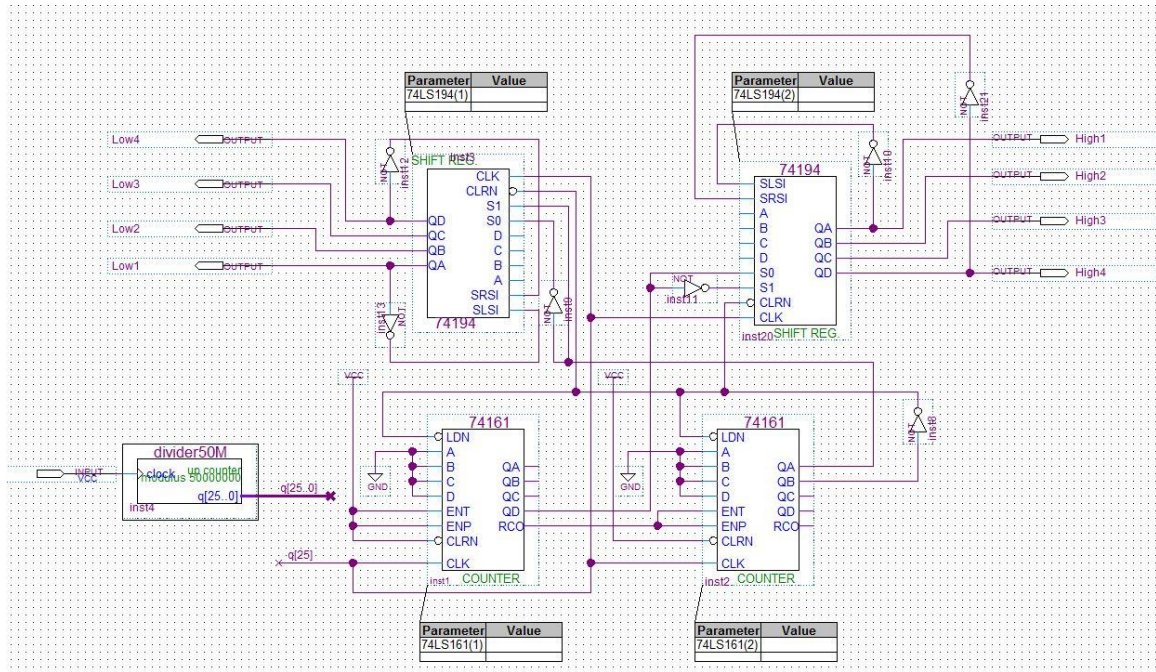
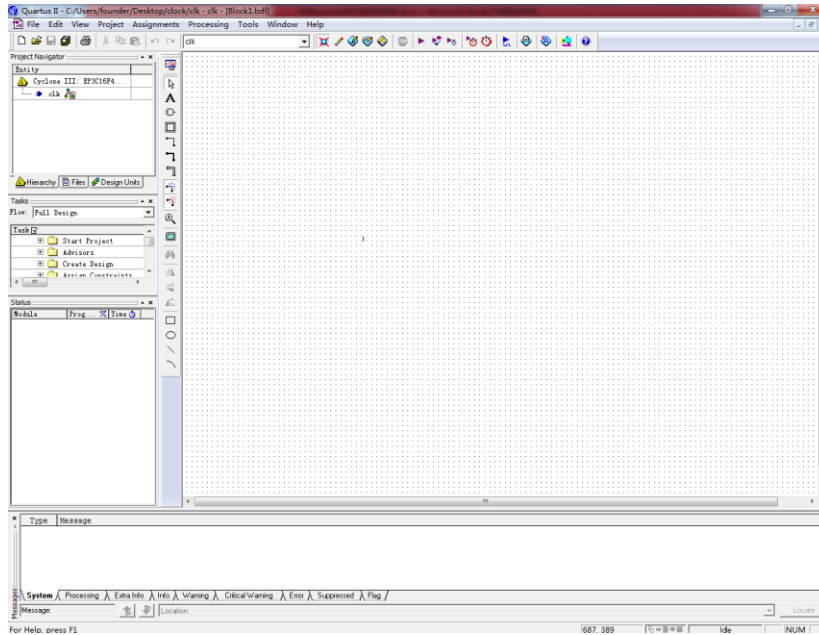


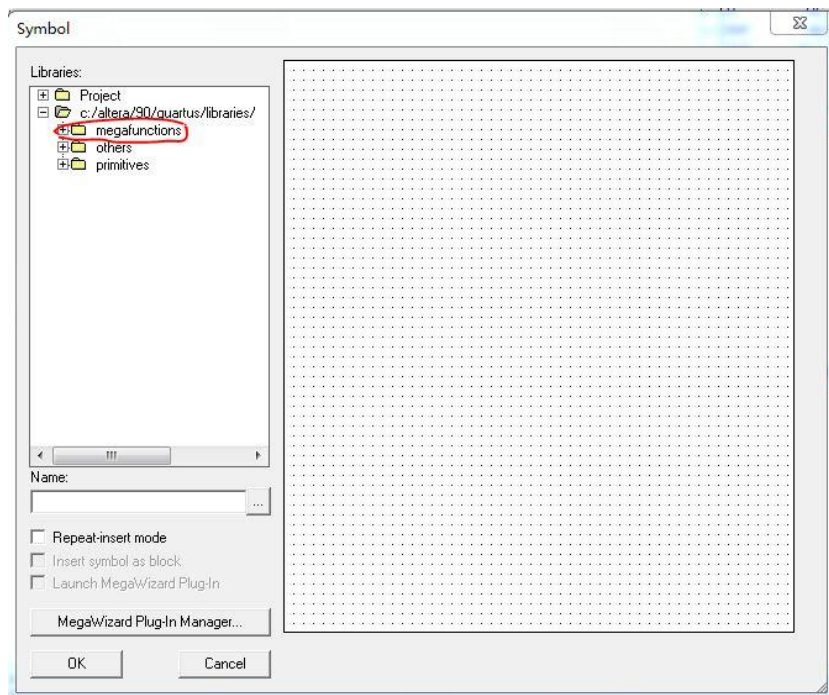
图 4 彩灯控制器电路图

利用 QuartusII 内建宏实现 1Hz 频率的时钟信号

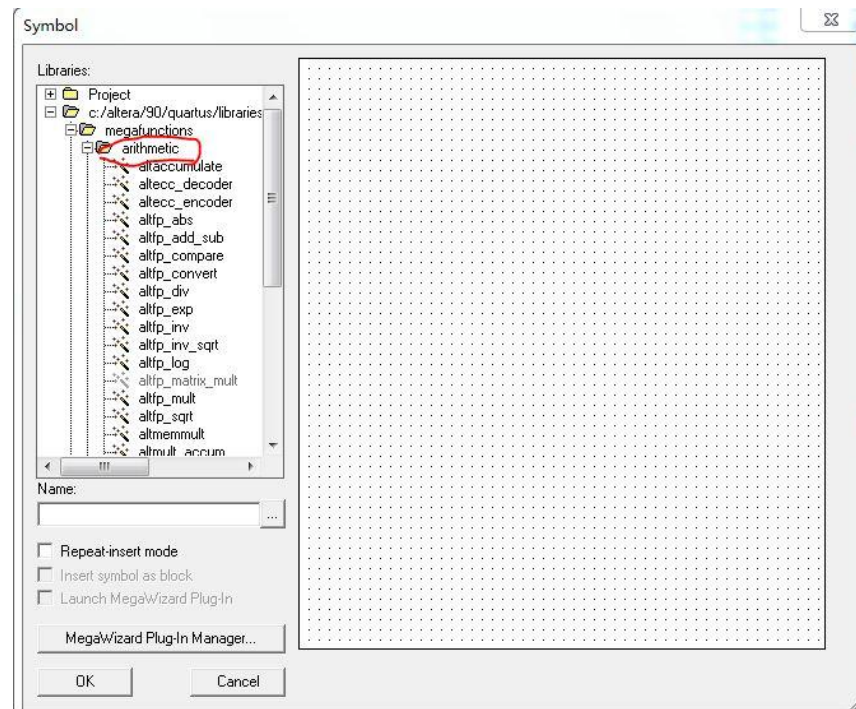
第一步：新建工程，并建立原理图文件



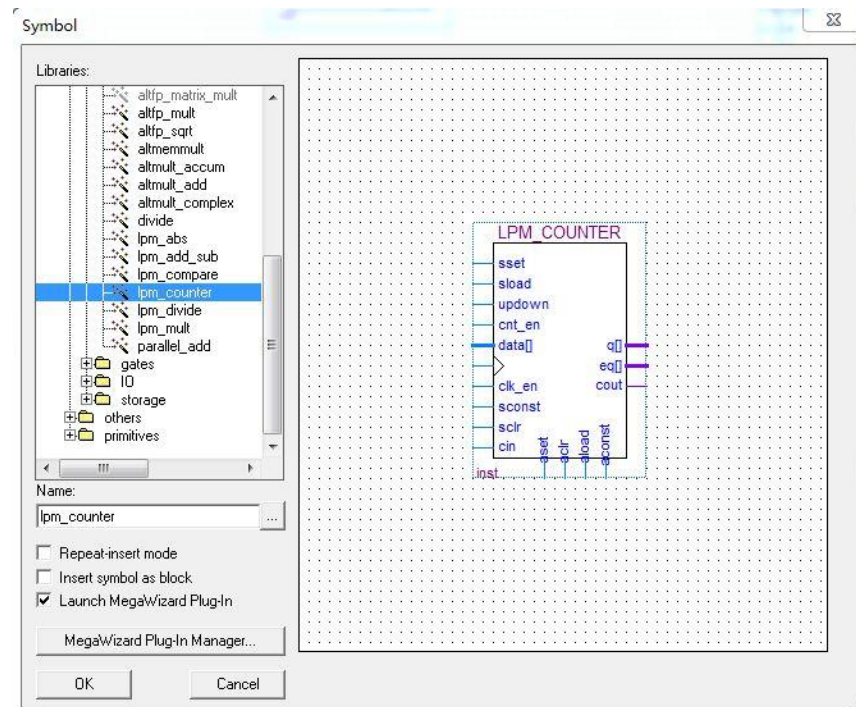
第二步：添加元件，选择“megafunction”



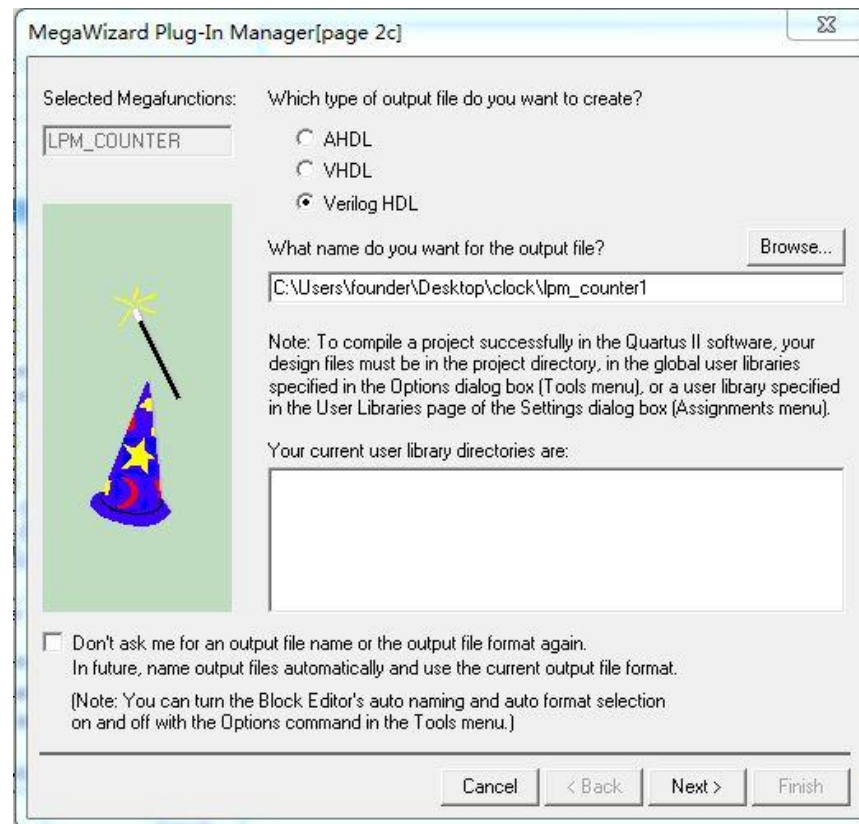
选择“arithmetic”



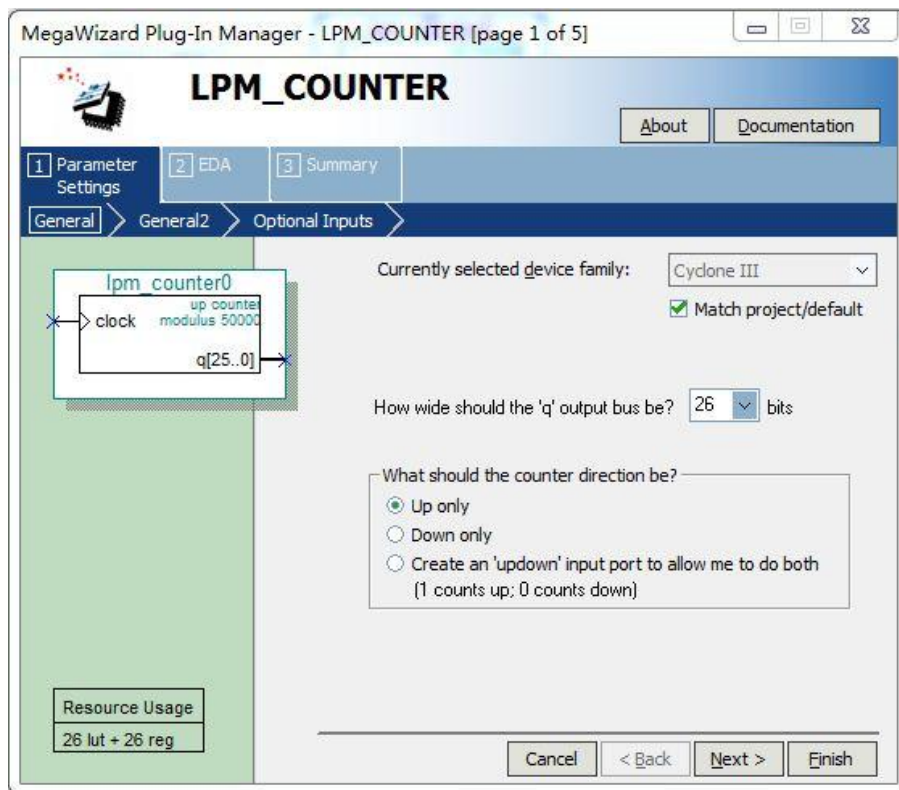
选择 lpm_counter 宏后，单击 OK 按钮。



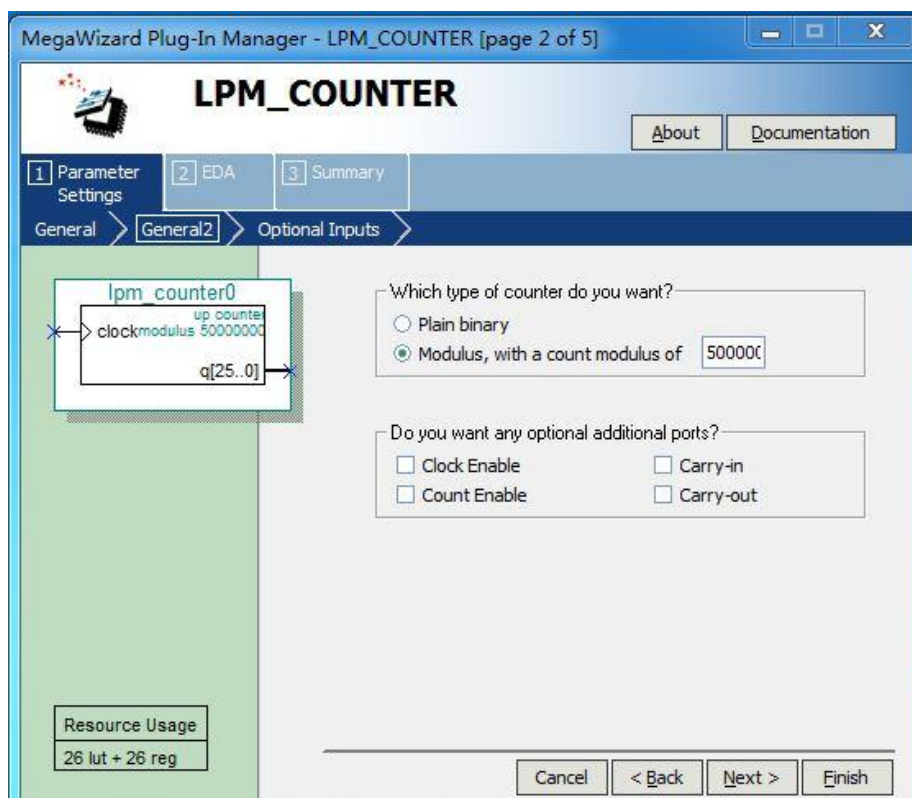
单击 “Next”



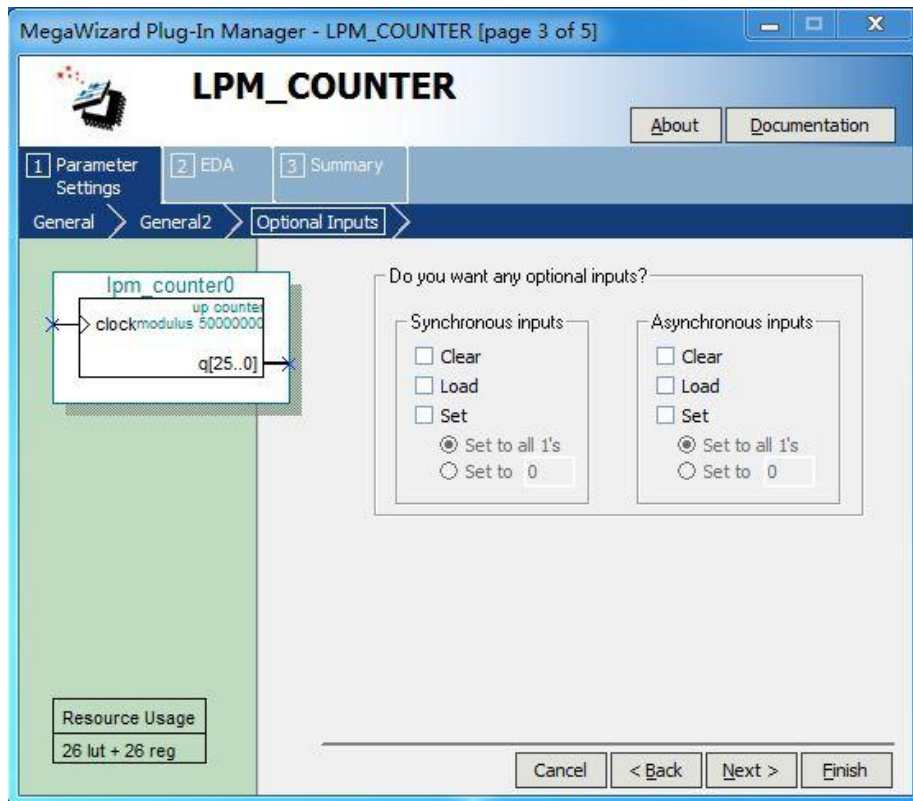
第三步：配置 lpm_counter 宏，选择 “How wide should the ‘q’ output bus be?” 选择 **26bits**
“What should the counter direction be?” 选择 “UP only”。



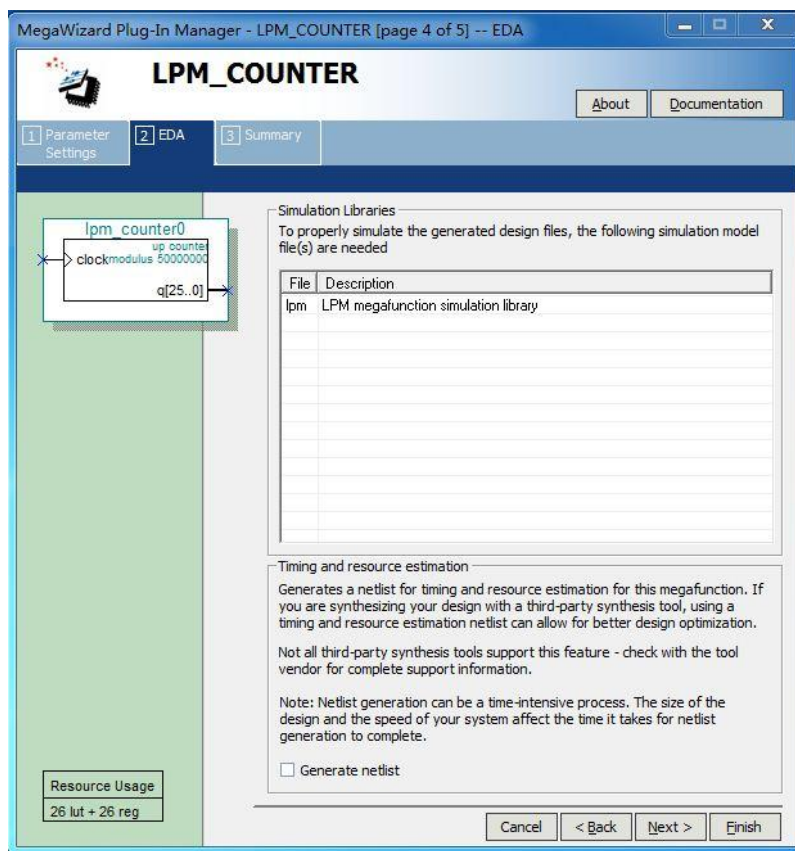
选择“Which type of counter do you want?”选择“**Module, with a count modulus of**”并输入“50,000,000”；“Do you want any optional additional ports?”不选择任何选项。



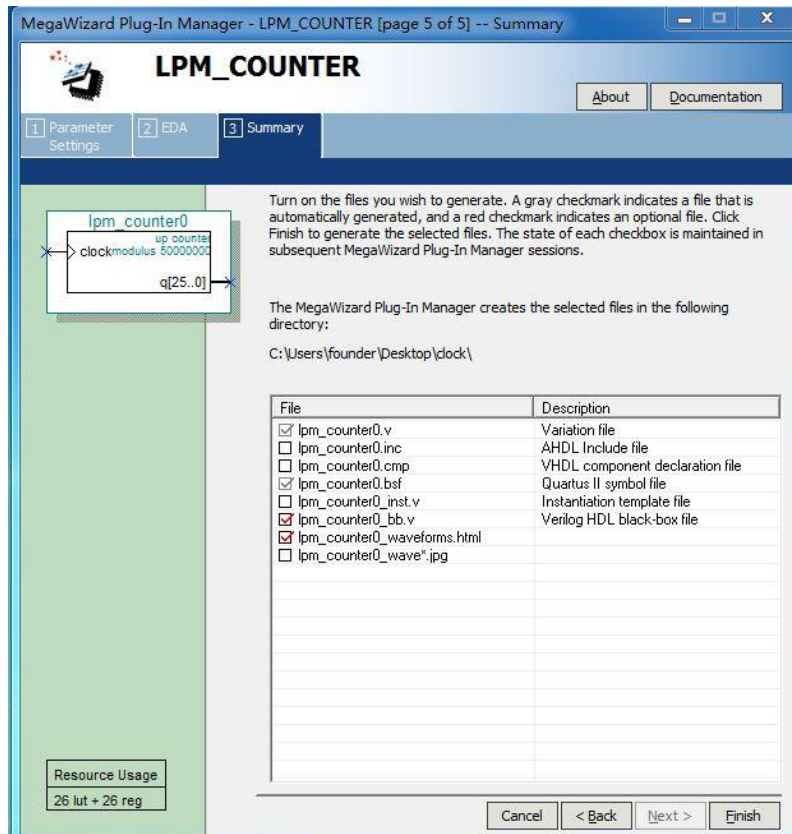
单击“Next”



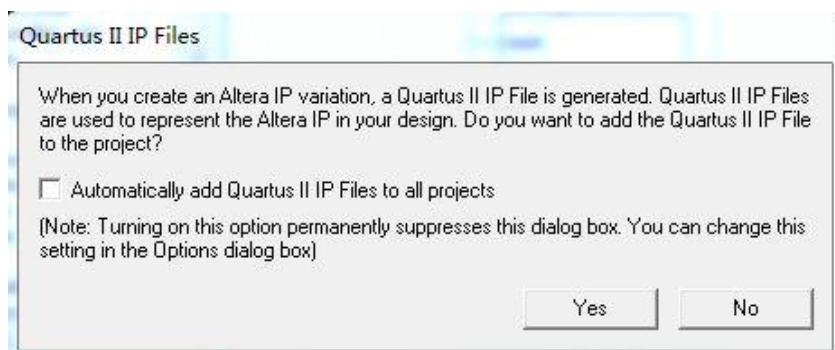
单击 “Next”



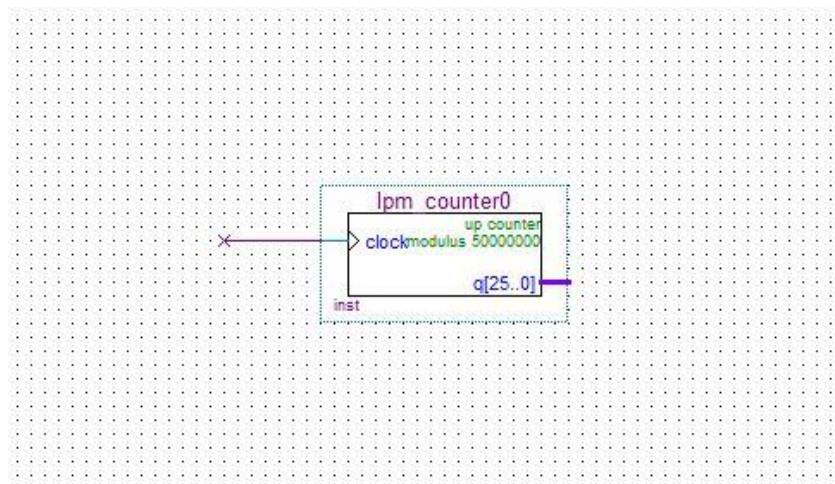
单击 “Finish”



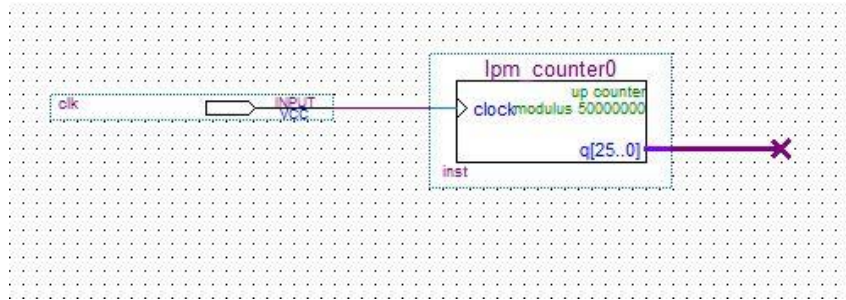
单击 “Yes”



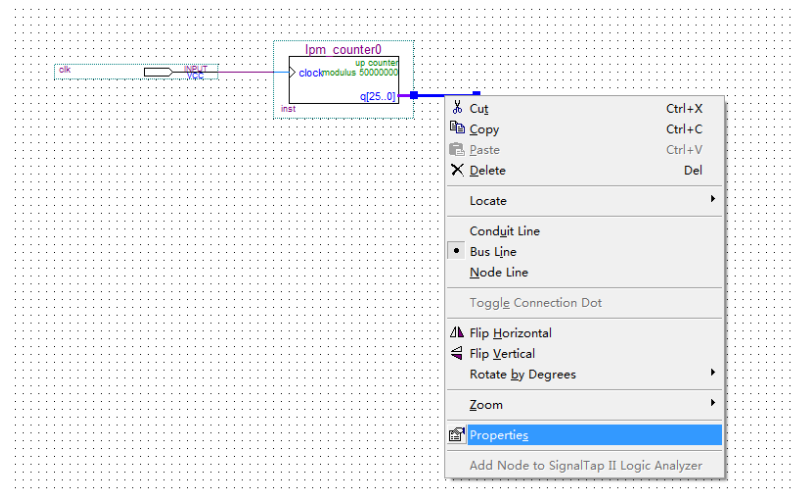
将生成的 lpm_counter 的宏放置在原理图上。



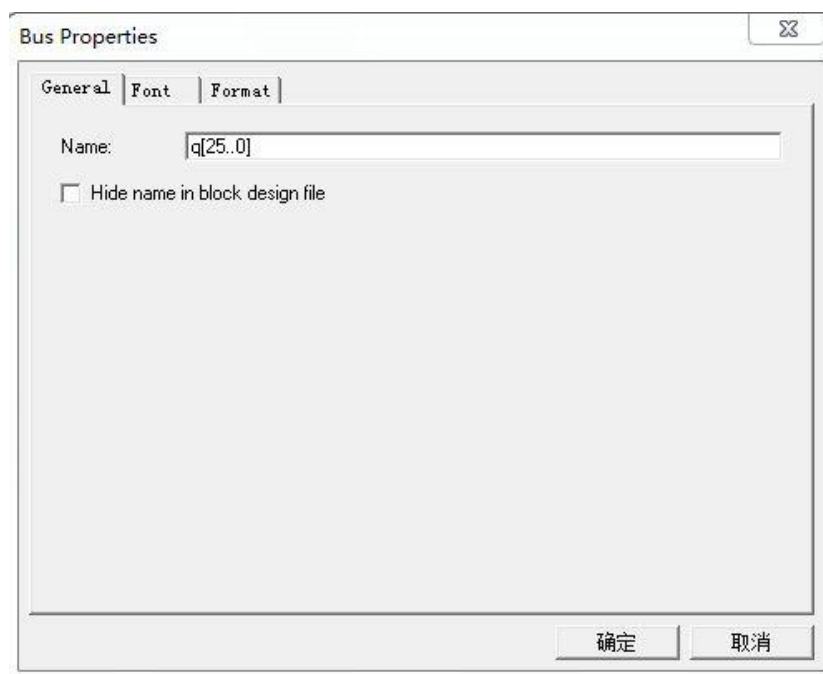
添加 Input 引脚，将 Input 端命名为 clk。在锁定引脚时将 clk 端和内部 50M 时钟（PIN_G21 引脚）绑定。将宏 lpm_counter0 的输出 q[25..0]用总线连接，不用连接任何端口。

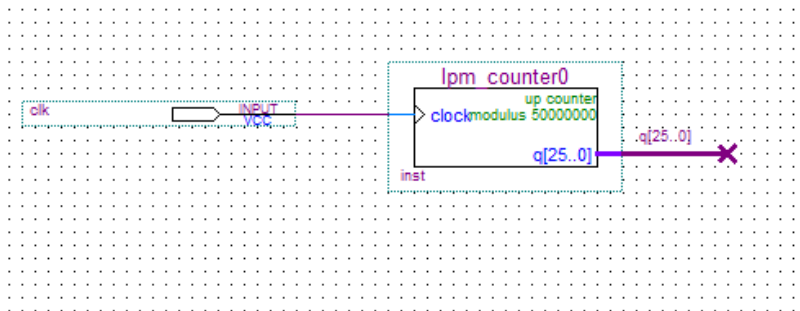


用鼠标右键单击该总线，从菜单栏中单击“Properties”选项。

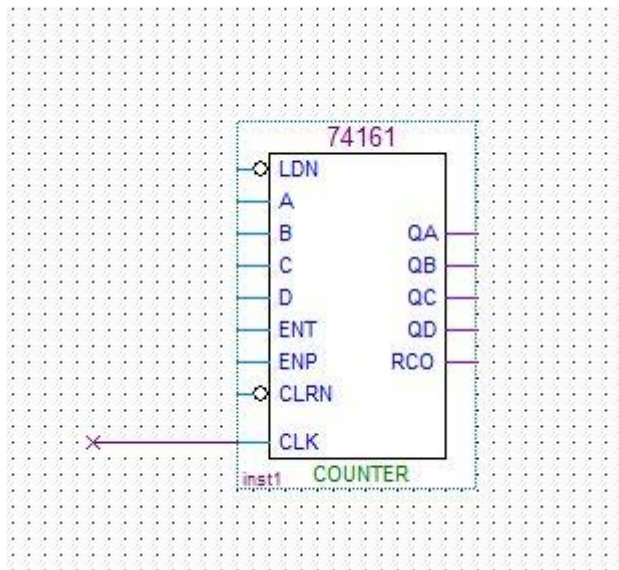


输入 q[25..0]，单击确定按钮。

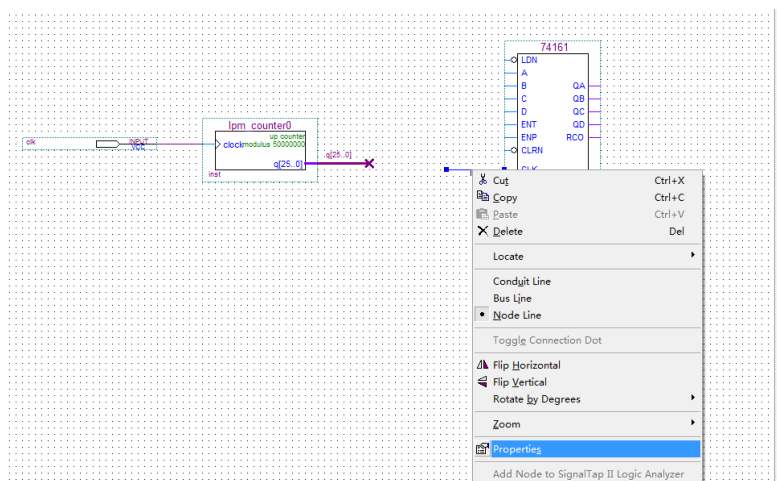




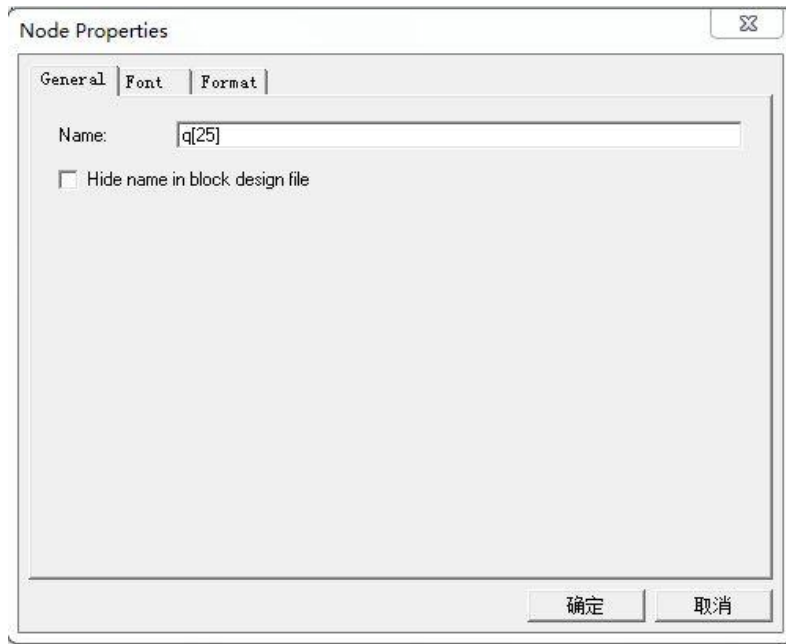
用同样的方法给下图 74161 的 CLK 端接入引线



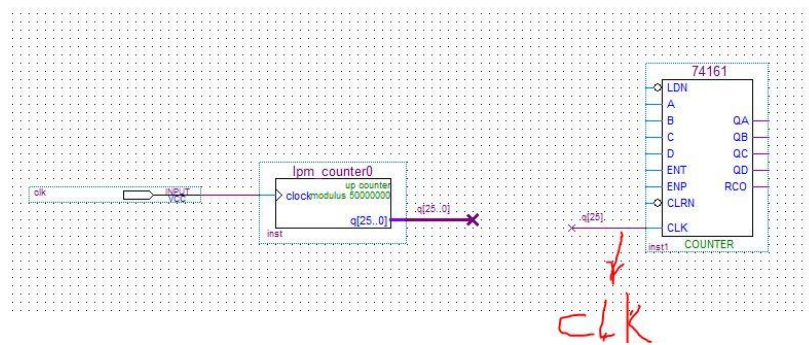
选中该引线右键单击，在弹出的菜单栏中选中“Properties”



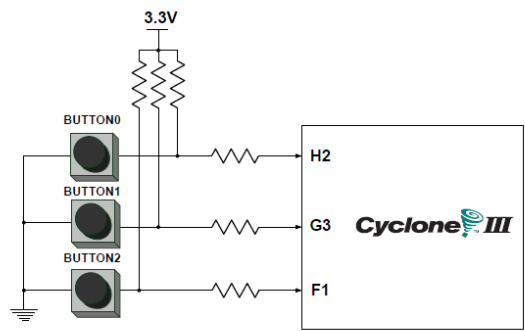
将该引线命名为 q[25]



此时 74161CLK 时钟端信号频率为 1Hz。

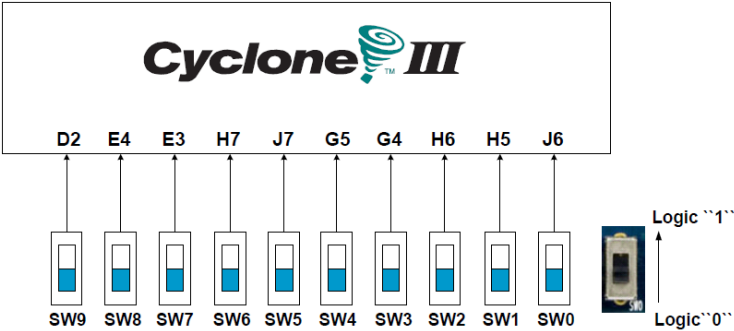


DE0 外接引脚说明



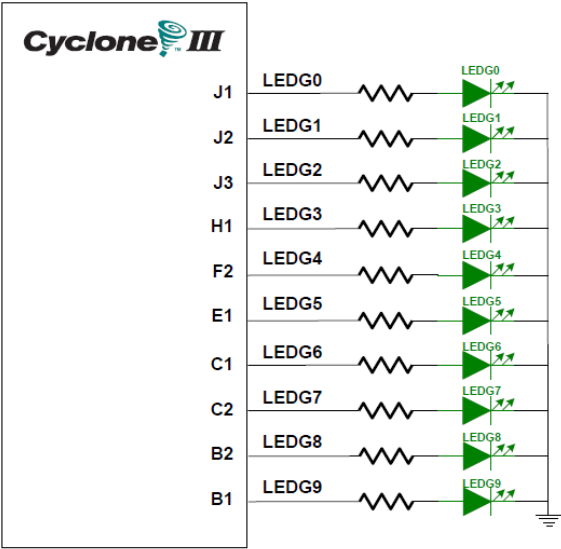
按钮开关与 Cyclone III FPGA 的连接
按钮开关的引脚分配

信号名	FPGA 引脚号	说明
BUTTON [0]	PIN_ H2	Pushbutton[0]
BUTTON [1]	PIN_ G3	Pushbutton[1]
BUTTON [2]	PIN_ F1	Pushbutton[2]



拨动开关与 CycloneIII FPGA 的连接
拨动开关的引脚分配

信号名	FPGA 引脚号.	说明
SW[0]	PIN_ J6	Slide Switch[0]
SW[1]	PIN_ H5	Slide Switch[1]
SW[2]	PIN_ H6	Slide Switch[2]
SW[3]	PIN_ G4	Slide Switch[3]
SW[4]	PIN_ G5	Slide Switch[4]
SW[5]	PIN_ J7	Slide Switch[5]
SW[6]	PIN_ H7	Slide Switch[6]
SW[7]	PIN_ E3	Slide Switch[7]
SW[8]	PIN_ E4	Slide Switch[8]
SW[9]	PIN_ D2	Slide Switch[9]

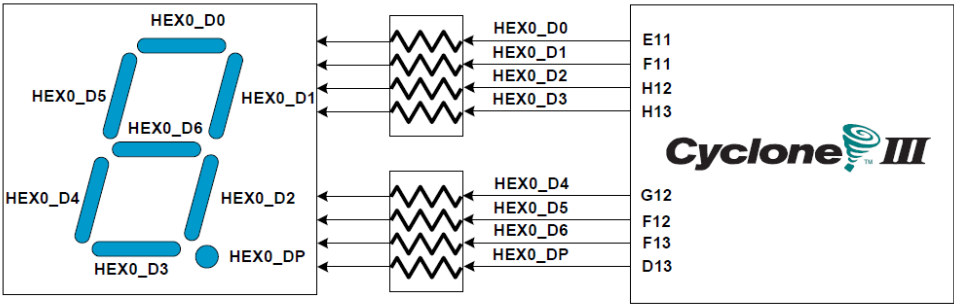


LED 灯与 Cyclone III FPGA 的连接

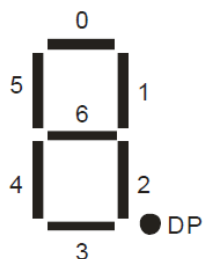
LED 灯的引脚分配

信号名	FPGA引脚号	说明
LEDG[0]	PIN_J1	LED Green[0]
LEDG[1]	PIN_J2	LED Green[1]
LEDG[2]	PIN_J3	LED Green[2]
LEDG[3]	PIN_H1	LED Green[3]
LEDG[4]	PIN_F2	LED Green[4]
LEDG[5]	PIN_E1	LED Green[5]
LEDG[6]	PIN_C1	LED Green[6]
LEDG[7]	PIN_C2	LED Green[7]
LEDG[8]	PIN_B2	LED Green[8]
LEDG[9]	PIN_B1	LED Green[9]

HEX0



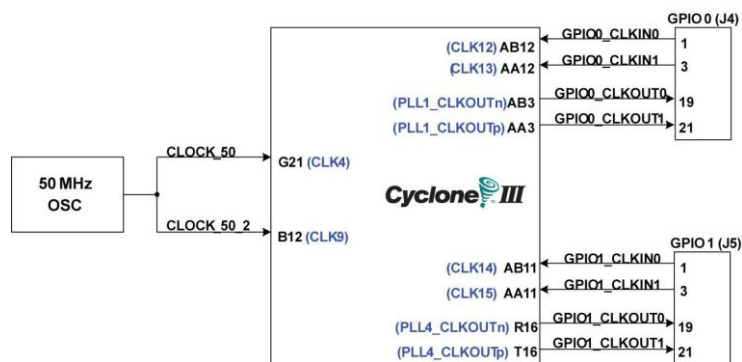
七段数码管与 Cyclone III FPGA 芯片的连接示意图



七段数码管每个字段的相应编号

七段数码管的引脚配置

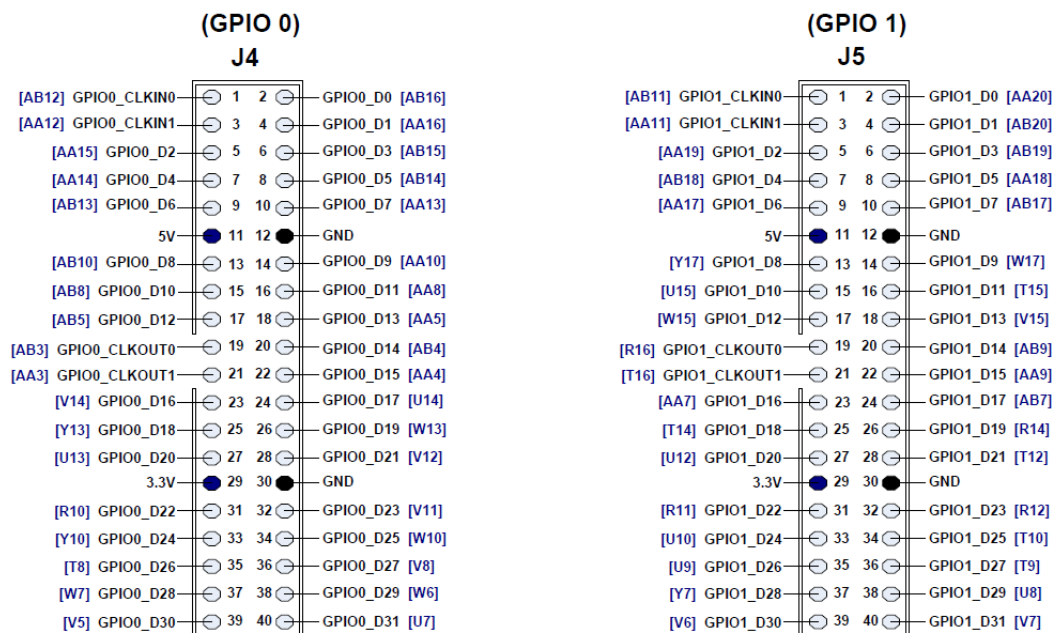
信号名	FPGA 引脚号	说明
HEX0_D[0]	PIN_E11	Seven Segment Digit 0[0]
HEX0_D[1]	PIN_F11	Seven Segment Digit 0[1]
HEX0_D[2]	PIN_H12	Seven Segment Digit 0[2]
HEX0_D[3]	PIN_H13	Seven Segment Digit 0[3]
HEX0_D[4]	PIN_G12	Seven Segment Digit 0[4]
HEX0_D[5]	PIN_F12	Seven Segment Digit 0[5]
HEX0_D[6]	PIN_F13	Seven Segment Digit 0[6]
HEX0_DP	PIN_D13	Seven Segment Decimal Point 0
HEX1_D[0]	PIN_A13	Seven Segment Digit 1[0]
HEX1_D[1]	PIN_B13	Seven Segment Digit 1[1]
HEX1_D[2]	PIN_C13	Seven Segment Digit 1[2]
HEX1_D[3]	PIN_A14	Seven Segment Digit 1[3]
HEX1_D[4]	PIN_B14	Seven Segment Digit 1[4]
HEX1_D[5]	PIN_E14	Seven Segment Digit 1[5]
HEX1_D[6]	PIN_A15	Seven Segment Digit 1[6]
HEX1_DP	PIN_B15	Seven Segment Decimal Point 1
HEX2_D[0]	PIN_D15	Seven Segment Digit 2[0]
HEX2_D[1]	PIN_A16	Seven Segment Digit 2[1]
HEX2_D[2]	PIN_B16	Seven Segment Digit 2[2]
HEX2_D[3]	PIN_E15	Seven Segment Digit 2[3]
HEX2_D[4]	PIN_A17	Seven Segment Digit 2[4]
HEX2_D[5]	PIN_B17	Seven Segment Digit 2[5]
HEX2_D[6]	PIN_F14	Seven Segment Digit 2[6]
HEX2_DP	PIN_A18	Seven Segment Decimal Point 2
HEX3_D[0]	PIN_B18	Seven Segment Digit 3[0]
HEX3_D[1]	PIN_F15	Seven Segment Digit 3[1]
HEX3_D[2]	PIN_A19	Seven Segment Digit 3[2]
HEX3_D[3]	PIN_B19	Seven Segment Digit 3[3]
HEX3_D[4]	PIN_C19	Seven Segment Digit 3[4]
HEX3_D[5]	PIN_D19	Seven Segment Digit 3[5]
HEX3_D[6]	PIN_G15	Seven Segment Digit 3[6]
HEX3_DP	PIN_G16	Seven Segment Decimal Point 3



时钟分配电路的方块图

时钟输入的引脚分配

信号名	FPGA 引脚号.	说明
CLOCK_50	PIN_G21	50 MHz clock input



扩展接头的 I/O 分配

扩展接头的引脚配置

信号名	FPGA 引脚号.	说明
GPIO0_D[0]	PIN_AB16	GPIO Connection 0 IO[0]
GPIO0_D[1]	PIN_AA16	GPIO Connection 0 IO[1]
GPIO0_D[2]	PIN_AA15	GPIO Connection 0 IO[2]
GPIO0_D[3]	PIN_AB15	GPIO Connection 0 IO[3]
GPIO0_D[4]	PIN_AA14	GPIO Connection 0 IO[4]
GPIO0_D[5]	PIN_AB14	GPIO Connection 0 IO[5]
GPIO0_D[6]	PIN_AB13	GPIO Connection 0 IO[6]
GPIO0_D[7]	PIN_AA13	GPIO Connection 0 IO[7]
GPIO0_D[8]	PIN_AB10	GPIO Connection 0 IO[8]

GPIO0_D[9]	PIN_AA10	GPIO Connection 0 IO[9]
GPIO0_D[10]	PIN_AB8	GPIO Connection 0 IO[10]
GPIO0_D[11]	PIN_AA8	GPIO Connection 0 IO[11]
GPIO0_D[12]	PIN_AB5	GPIO Connection 0 IO[12]
GPIO0_D[13]	PIN_AA5	GPIO Connection 0 IO[13]
GPIO0_D[14]	PIN_AB4	GPIO Connection 0 IO[14]
GPIO0_D[15]	PIN_AA4	GPIO Connection 0 IO[15]
GPIO0_D[16]	PIN_V14	GPIO Connection 0 IO[16]
GPIO0_D[17]	PIN_U14	GPIO Connection 0 IO[17]
GPIO0_D[18]	PIN_Y13	GPIO Connection 0 IO[18]
GPIO0_D[19]	PIN_W13	GPIO Connection 0 IO[19]
GPIO0_D[20]	PIN_U13	GPIO Connection 0 IO[20]
GPIO0_D[21]	PIN_V12	GPIO Connection 0 IO[21]
GPIO0_D[22]	PIN_R10	GPIO Connection 0 IO[22]
GPIO0_D[23]	PIN_V11	GPIO Connection 0 IO[23]
GPIO0_D[24]	PIN_Y10	GPIO Connection 0 IO[24]
GPIO0_D[25]	PIN_W10	GPIO Connection 0 IO[25]
GPIO0_D[26]	PIN_T8	GPIO Connection 0 IO[26]
GPIO0_D[27]	PIN_V8	GPIO Connection 0 IO[27]
GPIO0_D[28]	PIN_W7	GPIO Connection 0 IO[28]
GPIO0_D[29]	PIN_W6	GPIO Connection 0 IO[29]
GPIO0_D[30]	PIN_V5	GPIO Connection 0 IO[30]
GPIO0_D[31]	PIN_U7	GPIO Connection 0 IO[31]
GPIO0_CLKIN[0]	PIN_AB12	GPIO Connection 0 PLL In
GPIO0_CLKIN[1]	PIN_AA12	GPIO Connection 0 PLL In
GPIO0_CLKOUT[0]	PIN_AB3	GPIO Connection 0 PLL Out
GPIO0_CLKOUT[1]	PIN_AA3	GPIO Connection 0 PLL Out
GPIO1_D[0]	PIN_AA20	GPIO Connection 1 IO[0]
GPIO1_D[1]	PIN_AB20	GPIO Connection 1 IO[1]
GPIO1_D[2]	PIN_AA19	GPIO Connection 1 IO[2]
GPIO1_D[3]	PIN_AB19	GPIO Connection 1 IO[3]
GPIO1_D[4]	PIN_AB18	GPIO Connection 1 IO[4]
GPIO1_D[5]	PIN_AA18	GPIO Connection 1 IO[5]
GPIO1_D[6]	PIN_AA17	GPIO Connection 1 IO[6]
GPIO1_D[7]	PIN_AB17	GPIO Connection 1 IO[7]
GPIO1_D[8]	PIN_Y17	GPIO Connection 1 IO[8]
GPIO1_D[9]	PIN_W17	GPIO Connection 1 IO[9]
GPIO1_D[10]	PIN_U15	GPIO Connection 1 IO[10]
GPIO1_D[11]	PIN_T15	GPIO Connection 1 IO[11]
GPIO1_D[12]	PIN_W15	GPIO Connection 1 IO[12]
GPIO1_D[13]	PIN_V15	GPIO Connection 1 IO[13]
GPIO1_D[14]	PIN_AB9	GPIO Connection 1 IO[14]
GPIO1_D[15]	PIN_AA9	GPIO Connection 1 IO[15]

GPIO1_D[16]	PIN_AA7	GPIO Connection 1 IO[16]
GPIO1_D[17]	PIN_AB7	GPIO Connection 1 IO[17]
GPIO1_D[18]	PIN_T14	GPIO Connection 1 IO[18]
GPIO1_D[19]	PIN_R14	GPIO Connection 1 IO[19]
GPIO1_D[20]	PIN_U12	GPIO Connection 1 IO[20]
GPIO1_D[21]	PIN_T12	GPIO Connection 1 IO[21]
GPIO1_D[22]	PIN_R11	GPIO Connection 1 IO[22]
GPIO1_D[23]	PIN_R12	GPIO Connection 1 IO[23]
GPIO1_D[24]	PIN_U10	GPIO Connection 1 IO[24]
GPIO1_D[25]	PIN_T10	GPIO Connection 1 IO[25]
GPIO1_D[26]	PIN_U9	GPIO Connection 1 IO[26]
GPIO1_D[27]	PIN_T9	GPIO Connection 1 IO[27]
GPIO1_D[28]	PIN_Y7	GPIO Connection 1 IO[28]
GPIO1_D[29]	PIN_U8	GPIO Connection 1 IO[29]
GPIO1_D[30]	PIN_V6	GPIO Connection 1 IO[30]
GPIO1_D[31]	PIN_V7	GPIO Connection 1 IO[31]
GPIO1_CLKIN[0]	PIN_AB11	GPIO Connection 1 PLL In
GPIO1_CLKIN[1]	PIN_AA11	GPIO Connection 1 PLL In
GPIO1_CLKOUT[0]	PIN_R16	GPIO Connection 1 PLL Out
GPIO1_CLKOUT[1]	PIN_T16	GPIO Connection 1 PLL Out