

《计算机数字逻辑设计》

第二次实验

实验三	组合逻辑电路设计——表决器	3
一、	实验目的.....	3
二、	实验环境与仪器设备	3
三、	实验要求	3
四、	实验原理	3
五、	实验任务	5
六、	实验数据处理及其他要求.....	7
1.	附录 A Verilog 简明手册	8
2.	附录 B EGO1 板硬件约束资源	11
3.	附录 C 操作常见问题及注意事项.....	19

实验板安全使用规范

更改跳帽位置或使用扩展接口扩展电路应用前请关闭电路板总开关，避免损坏器件。

电路板建议在绝缘平台上使用，不用金属或其他导电物触碰电路板线路，否则可能引起电路板损坏。

电路使用时应防止静电。秋冬季时，操作电路板前，手或纤维手套应先接触接地金属物。

液晶显示器件或模块结雾时，不要通电工作，防止电极化学反应，产生断线。

电源正负极、输入/输出端口定义时需谨慎，避免因接反引起开发板的损坏。

保持电路板的表面清洁。

小心轻放，避免不必要的硬件损伤

实验三 组合逻辑电路设计——表决器

一、 实验目的

- 1、深化对表决器、数码管工作原理和逻辑功能的理解，会在电路中使用数码管作为显示设备。
- 2、掌握组合逻辑电路的设计方法，能利用 Verilog HDL，采用结构描述法、数据流描述法和行为描述法这三种进行数据选择器的电路建模。
- 3、掌握多人表决器（3人以上）的设计方法与数码管的使用方法。
- 4、掌握在 Vivado 环境下，用 Verilog HDL 进行电路建模、仿真验证并利用 FPGA 进行板级验证的硬件电路设计方法。

二、 实验环境与仪器设备

- 1、一台内存 4GB 以上，装有 64 位 Windows 操作系统和的 PC 机。
- 2、Xilinx 实验板（EGO1）一个。
- 3、Vivado 2017.4 以上版本软件。

三、 实验要求

1、实验预习阶段

预习实验内容及准备工作，并提交预习报告。复习多人表决器和数码管的工作原理；拟定实验步骤及操作流程（利用 Visio 等绘图工具绘制），整理逻辑电路设计思路，画出真值表、卡诺图，写出逻辑表达式等；利用 Verilog HDL 设计源文件和仿真文件；设计实验验证方法和数据分析等。

2、实验实施阶段

利用 Verilog HDL 设计相关电路，在 Vivado 中完成设计与仿真，记录并分析讨论实验结果的正确性，最后给出测试结论。

3、实验工作总结阶段

实验完毕，评价实验结果的正确性。分析实验数据，提出优化思路与设想，总结实验体会写出实验报告。

四、 实验原理

1、表决器

以三人表决器为例说明多人表决电路的原理。设三人表决器中输入为 A、B、C，同意用 1 表示，不同意用 0 表示。输出为 F，A、B、C 三者中多数同意，提案通过，否则提案不被通过；通过用 1 表示，不通过用 0 表示。有真值表如下：

表 3.1

输入			输出
A	B	C	F
0	0	0	0
0	0	1	0
0	1	0	0

0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

可得逻辑表达式如下：

$$\begin{aligned}
 F &= AB + BC + CA \\
 &= \overline{\overline{AB} \cdot \overline{BC} \cdot \overline{CA}} = \overline{\overline{\overline{AB} \cdot \overline{BC} \cdot \overline{CA}}}
 \end{aligned} \quad \text{式 3.1}$$

2、七段数码管

七段数码管内部结构如图 3.1(a)所示。7 个线形 LED 灯 a~g 构成七段，也可支持小数点 DP，即由 8 段构成。8 个 LED 灯的位置与段编号的对应关系如图 3.1(a)所示。8 个 LED 灯的一端并接在一起，称为位选信号。如果七个数码管的阳极并接在一起，称为共阳极数码管，如图 3.2(b)所示。如果七个数码管的阴极并接在一起，称为共阴极数码管，如图 3.1(c)所示。每个 LED 的另一端分别接收段选信号以控制灯的亮灭，称为 8 个段选信号组合得到的编码，称为段码。

数码管的点亮条件为：对于共阴极数码管，“位选端”接低电平，需点亮的“段选端”接高电平。而对于共阳极数码管，则“位选端”接高电平，需点亮的“段选端”接低电平。以图 3.1(c)所示的共阴极数码管为例，其“位选端”接低电平 0 时，段码与显示值的对应关系如表 3.2 所示。

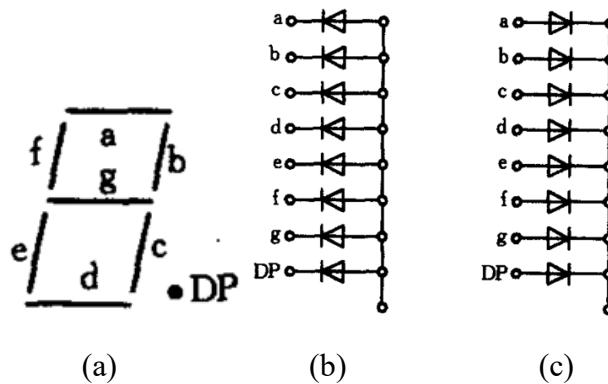


图 3.1 七段数码管结构示意图

表 3.2 段码与显示值的对应关系

显示字形	段码						
	a	b	c	d	e	f	g
0	1	1	1	1	1	1	0
1	0	1	1	0	0	0	0
2	1	1	0	1	1	0	1
3	1	1	1	1	0	0	1
4	0	1	1	0	0	1	1
5	1	0	1	1	0	1	1
6	1	0	1	1	1	1	1
7	1	1	1	0	0	0	0
8	1	1	1	1	1	1	1
9	1	1	1	1	0	1	1
A	1	1	1	0	1	1	1
b	0	0	1	1	1	1	1
C	1	0	0	1	1	1	0
d	0	1	1	1	1	0	1
E	1	0	0	1	1	1	1
F	1	0	0	0	1	1	1

2、实验板 EG01 上的数码管

EG01 实验板上配置了 8 个共阴极带小数点的七段数码管。这 8 个数码管分为两组，每组 4 个数码管。实验板上左边 4 个数码管属于第 0 组，右边 4 个数码管是第 1 组。同一组的 4 个数码管的段选信号并接再一起，利用位选信号完成数码管的选择控制。数码管管脚约束如下：

表 3.3

组号	名称	FPGA IO 引脚	说明
0	A	B4	第 0 组数码管的段选信号-段 A
	B	A4	第 0 组数码管的段选信号-段 B
	C	A3	第 0 组数码管的段选信号-段 C
	D	B1	第 0 组数码管的段选信号-段 D
	E	A1	第 0 组数码管的段选信号-段 E
	F	B3	第 0 组数码管的段选信号-段 F
	G	B2	第 0 组数码管的段选信号-段 G
	DP	D5	第 0 组数码管的段选信号-小数点
	DK1	G2	左起第 1 个数码管的位选
	DK2	C2	左起第 2 个数码管的位选
	DK3	C1	左起第 3 个数码管的位选
	DK4	H1	左起第 4 个数码管的位选
1	A	D4	第 1 组数码管的段选信号-段 A
	B	E3	第 1 组数码管的段选信号-段 B
	C	D3	第 1 组数码管的段选信号-段 C
	D	F4	第 1 组数码管的段选信号-段 D
	E	F3	第 1 组数码管的段选信号-段 E
	F	E2	第 1 组数码管的段选信号-段 F
	G	D2	第 1 组数码管的段选信号-段 G
	DP	H2	第 1 组数码管的段选信号-小数点
	DK5	G1	左起第 5 个数码管的位选
	DK6	F1	左起第 6 个数码管的位选
	DK7	E1	左起第 7 个数码管的位选
	DK8	G6	左起第 8 个数码管的位选

EG01 实验板上内部采用驱动电路控制数码管，要点亮某段，位选信号和段选信号都要接高电平。以下图 3.2 所示的一组数码管为例。若要让图中左起第 1、3 个数码管显示“0”，其他 2、4 数码管全灭，可以给 a~g 送段码 8'b11111100，给 1~4 送位码 4'b1010。



图 0.2

五、实验任务

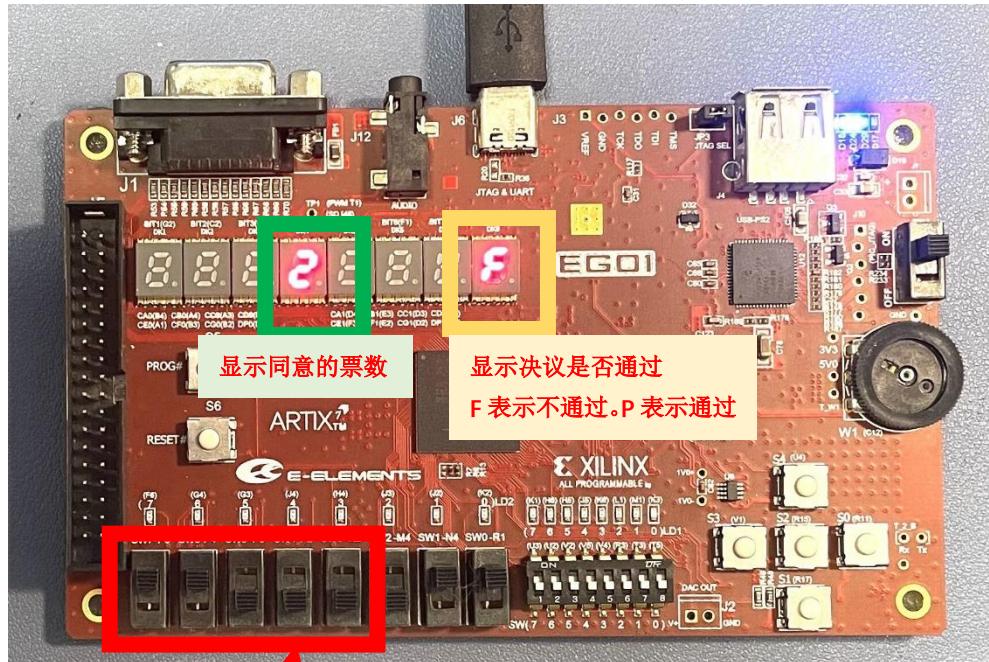
1、在结构描述法、数据流描述法或行为描述法三种方法中，选择一种方法进行电路建模，设计 5 人表决器，具体功能是：当有 3 个及以上的评委投“同意”票时，决议通过。要求：完成功能仿真验证和 FPGA

在线验证。

2、在结构描述法、数据流描述法或行为描述法三种方法中，选择一种方法进行电路建模，设计数码管显示译码控制电路。该模块可以驱动 2 个数码管分别显示不同的内容。具体要求是：1 个数码管显示 0~5 之间的数值，另一个数码管显示用字母 P 或 F。要求完成电路的建模和 FPGA 在线验证。

3、采用迭代法（模块化/层次化设计方法），利用前两步的电路模块，设计带数码管显示的 5 人表决器。实现的功能时：当有 3 个及以上的评委投“同意”票时，决议通过。要求利用 5 个拨码开关作为 5 个投票按键，利用数码管显示同意的票数和决议是否通过（决议通过时，显示用字母 P，决议不通过时，显示字母 F）。要求完成电路的建模和 FPGA 在线验证。

实验效果如下图 3.3 所示。



拨码开关：用作投票按键。

开关往上拨时，输入为高电平，表示同意。

开关往下拨时，输入为低电平，表示不同意

带数码管显示的 5 人表决器的原理框图，如下图 3.4 所示。

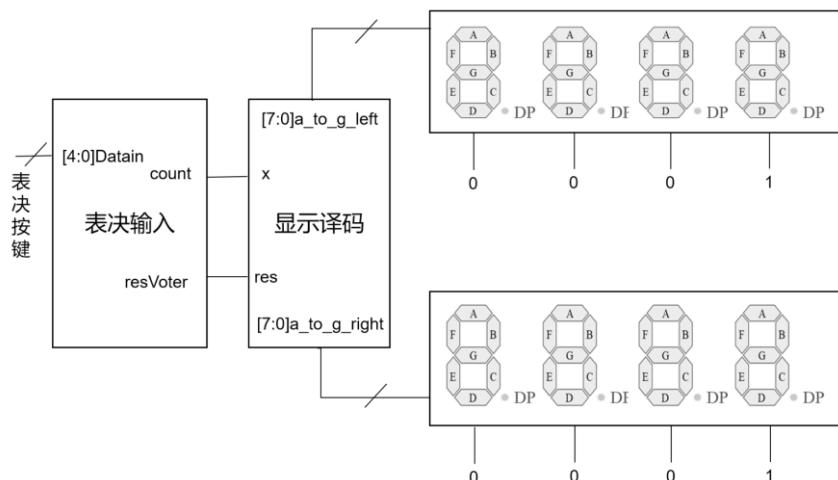


图 3.4 带数码管显示的 5 人表决器原理框图

六、实验数据处理及其他要求

- 1、编写 Verilog HDL 源代码，完成电路建模。
- 2、编译文件并查看 RTL 视图，说明其与原理框图的差异。
- 3、编写有关电路模块的 Test Bench，进行仿真测试，观察输出波形，整理实验数据，分析实验结果验证电路功能是否满足设计目标。
- 4、编写硬件约束文件，利用实验板 EG01 验证电路功能。
- 5、分析实验中出现的问题的原因。

1. 附录A Verilog 简明手册

Verilog Cheat Sheet —— S Winberg and J Taylor

Comments

```
// One-liner  
/* Multiple  
lines */
```

Numeric Constants

```
// The 8-bit decimal number 106:  
8'b_0110_1010 // Binary  
8'o_152 // Octal  
8'd_106 // Decimal  
8'h_6A // Hexadecimal  
"j" // ASCII  
78'bZ // 78-bit high-impedance
```

Too short constants are padded with zeros on the left. Too long constants are truncated from the left.

Nets and Variables

```
wire [3:0]w; // Assign outside always blocks  
reg [1:7]r; // Assign inside always blocks  
reg [7:0]mem[31:0];  
integer j; // Compile-time variable  
genvar k; // Generate variable
```

Parameters

```
parameter N = 8;  
localparam State = 2'd3;
```

Assignments

```
assign Output = A * B;  
assign {C, D} = {D[5:2], C[1:9], E};
```

Operators

// These are in order of precedence...

// Select

```
A[N] A[N:M]  
// Reduction  
&A ~&A |A ~|A ^A ~^A
```

// Compliment

```
!A ~A  
// Unary  
+A -A  
// Concatenate  
{A, ..., B}  
// Replicate  
{N{A}}
```

// Arithmetic

```
A*B A/B A%B  
A+B A-B
```

// Shift

```
A<<B A>>B
```

// Relational

```
A>B A<B A>=B A<=B
```

```
A==B A!=B
```

// Bit-wise

```
A&B  
A^B A~^B  
A|B
```

// Logical

```
A&&B  
A||B  
// Conditional  
A ? B : C
```

Module

```
module MyModule  
#(parameter N = 8) // Optional parameter  
(input Reset, Clk,  
output [N-1:0]Output);
```

```

// Module implementation
endmodule

Module Instantiation
// Override default parameter: setting N = 13
MyModule #(13) MyModule1(Reset, Clk, Result);

Case
always @(*) begin
  case(Mux)
    2'd0: A = 8'd9;
    2'd1,
    2'd3: A = 8'd103;
    2'd2: A = 8'd2;
  default:;
  endcase
end
always @(*) begin
  casex(Decoded)
    4'b1xxx: Encoded = 2'd0;
    4'b01xx: Encoded = 2'd1;
    4'b001x: Encoded = 2'd2;
    4'b0001: Encoded = 2'd3;
  default: Encoded = 2'd0;
  endcase
end

Synchronous
always @(posedge Clk) begin
  if(Reset) B <= 0;
  else B <= B + 1'b1;
end

Loop
always @(*) begin
  Count = 0;
  for(j = 0; j < 8; j = j+1)
    Count = Count + Input[j];
end

```

Function

```

function [6:0]F;
  input [3:0]A;
  input [2:0]B;
begin
  F = {A+1'b1, B+2'd2};
end
endfunction

```

Generate

```

genvar j;
wire [12:0]Output[19:0];
generate
  for(j = 0; j < 20; j = j+1)
    begin: Gen_Modules
      MyModule #(13) MyModule_Instance(
        Reset, Clk,
        Output[j]
      );
    end
  endgenerate

```

State Machine

```

reg [1:0]State;
localparam Start = 2'b00;
localparam Idle = 2'b01;
localparam Work = 2'b11;
localparam Done = 2'b10;
reg tReset;

```

```

always @(posedge Clk) begin
  tReset <= Reset;
  if(tReset) begin
    State <= Start;
  end else begin
    case(State)
      Start: begin

```

```
State <= Idle;  
end  
Idle: begin  
    State <= Work;  
end  
Work: begin  
    State <= Done;  
end  
  
Done: begin  
    State <= Idle;  
end  
default:;  
endcase  
end
```

2. 附录 B EGO1 板硬件约束资源

Vivado 的约束文件编写采用 Tcl 语言规则。Tcl（读作 tickle）诞生于 80 年代的加州大学伯克利分校，作为一种简单高效可移植性好的脚本语言，目前已经广泛应用在几乎所有的 EDA 工具中。Tcl 的最大特点就是其语法格式极其简单甚至可以说僵化，采用纯粹的 [命令 选项 参数] 形式，是名副其实的“工具命令语言”（Tcl 的全称 Tool Command Language）。

```
set myVar "Hello World!" //设置一个名为 myVar 的变量，其值为 Hello World!
```

在 Vivado 中使用 Tcl 最基本的场景就是对网表上的目标进行遍历、查找和定位，这也是对网表上的目标进行约束的基础。要掌握这些则首先需要理解 Vivado 对目标的分类。

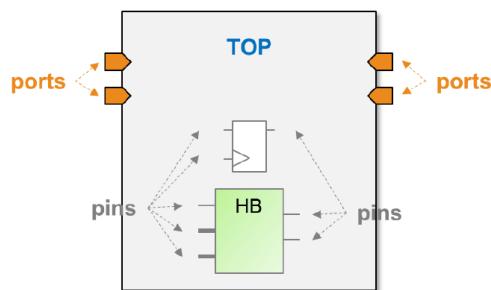


图 2.1

如上图所示，设计顶层的 I/O 称作 ports，其余底层模块或是门级网表上的元件端口都称作 pins。而包括顶层在内的各级模块，blackbox 以及门级元件，都称作 cells。连线称作 nets，加上 XDC 中定义的 clocks，在 Vivado 中一共将网表文件中的目标定义为五类。要选取这五类目标，则需用相应的 get_*命令，例如 get_pins 等等。

Tcl 在搜索网表中的目标时，除了上述根据名字条件直接搜索的方式，还可以利用目标间的关系，使用 -of_objects (Tcl 中可以简写为-of) 来间接搜索特定目标。Vivado 中定义的五类目标间关系如下图所示。

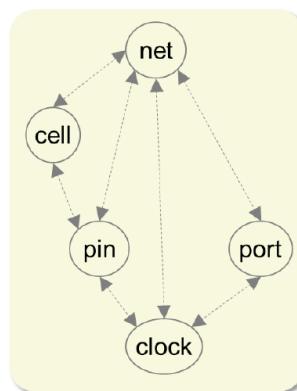


图 2.2

以下图的设计来举例，

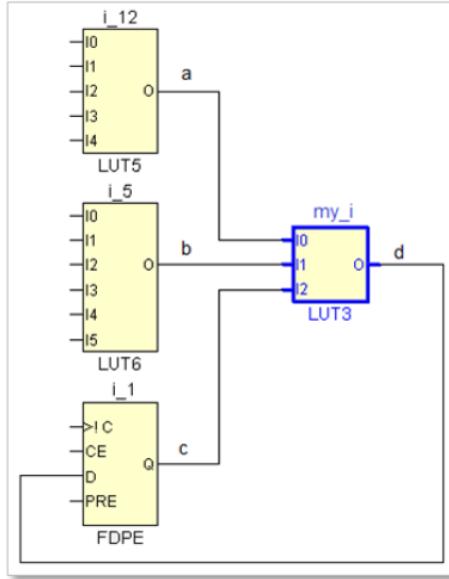


图 2.3

```

get_pins -of [get_cells my_i]           //返回 my_i/I0 my_i/I1 my_i/I2 my_i/O
get_nets -of [get_cells my_i]           //返回 a b c d
get_cells -of [get_nets -of [get_cells my_i]] //返回 i_12 i_5 i_1 my_i

```

需要注意的是：

XDC 在本质上就是 Tcl 语言，但其仅支持基本的 Tcl 语法如变量、列表和运算符等等，对其它复杂的循环以及文件 I/O 等语法可以通过在 Vivado 中 source 一个 Tcl 文件的方式来补充。XDC 的基本语法可以分为时钟约束、I/O 约束以及时序例外约束三大类。

以下为实验板自带的系统引脚 I/O 约束，设计者可以结合自己的电路修改引脚命名。

```

//////////系统时钟和复位//////////
set_property -dict {PACKAGE_PIN P17 IOSTANDARD LVCMOS33} [get_ports sys_clk_in]
set_property -dict {PACKAGE_PIN P15 IOSTANDARD LVCMOS33} [get_ports sys_rst_n ]

```

```

//////////串口/////////

```

```

set_property -dict {PACKAGE_PIN N5 IOSTANDARD LVCMOS33} [get_ports PC_Uart_rxd]
set_property -dict {PACKAGE_PIN T4 IOSTANDARD LVCMOS33} [get_ports PC_Uart_txd]

```

```

//////////蓝牙/////////

```

```

set_property -dict {PACKAGE_PIN L3 IOSTANDARD LVCMOS33} [get_ports BT_Uart_rxd]
set_property -dict {PACKAGE_PIN N2 IOSTANDARD LVCMOS33} [get_ports BT_Uart_txd]

```

```

set_property -dict {PACKAGE_PIN D18 IOSTANDARD LVCMOS33} [get_ports {bt_ctrl_o[0]}]
set_property -dict {PACKAGE_PIN M2 IOSTANDARD LVCMOS33} [get_ports {bt_ctrl_o[1]}]
set_property -dict {PACKAGE_PIN H15 IOSTANDARD LVCMOS33} [get_ports {bt_ctrl_o[2]}]

```

```
set_property -dict {PACKAGE_PIN C16 IOSTANDARD LVCMOS33} [get_ports {bt_ctrl_o[3]}]
set_property -dict {PACKAGE_PIN E18 IOSTANDARD LVCMOS33} [get_ports {bt_ctrl_o[4]}]
```

```
set_property -dict {PACKAGE_PIN C17 IOSTANDARD LVCMOS33} [get_ports bt_mcu_int_i]
```

//////////////////音频接口//////////////////

```
set_property -dict {PACKAGE_PIN T1 IOSTANDARD LVCMOS33} [get_ports audio_pwm_o]
set_property -dict {PACKAGE_PIN M6 IOSTANDARD LVCMOS33} [get_ports audio_sd_o]
```

//////////////////iic//////////////////

```
set_property -dict {PACKAGE_PIN F18 IOSTANDARD LVCMOS33} [get_ports pw_iic_scl_io]
set_property -dict {PACKAGE_PIN G18 IOSTANDARD LVCMOS33} [get_ports pw_iic_sda_io]
```

//////////////////XADC 模数转换//////////////////

```
set_property -dict {PACKAGE_PIN B12 IOSTANDARD LVCMOS33} [get_ports XADC_AUX_v_n ]
set_property -dict {PACKAGE_PIN C12 IOSTANDARD LVCMOS33} [get_ports XADC_AUX_v_p ]
set_property -dict {PACKAGE_PIN K9 IOSTANDARD LVCMOS33} [get_ports XADC_VP_VN_v_n]
set_property -dict {PACKAGE_PIN J10 IOSTANDARD LVCMOS33} [get_ports XADC_VP_VN_v_p]
```

//////////////////5 个按键//////////////////位置以印刷板丝印为准//////////////////

```
set_property -dict {PACKAGE_PIN R11 IOSTANDARD LVCMOS33} [get_ports {btn_pin[0]}]
set_property -dict {PACKAGE_PIN R17 IOSTANDARD LVCMOS33} [get_ports {btn_pin[1]}]
set_property -dict {PACKAGE_PIN R15 IOSTANDARD LVCMOS33} [get_ports {btn_pin[2]}]
set_property -dict {PACKAGE_PIN V1 IOSTANDARD LVCMOS33} [get_ports {btn_pin[3]}]
set_property -dict {PACKAGE_PIN U4 IOSTANDARD LVCMOS33} [get_ports {btn_pin[4]}]
```

//////////////////拨码开关 sw0~sw7//////////////////位置以印刷板丝印为准//////////////////

```
set_property -dict {PACKAGE_PIN P5 IOSTANDARD LVCMOS33} [get_ports {sw_pin[0]}]
set_property -dict {PACKAGE_PIN P4 IOSTANDARD LVCMOS33} [get_ports {sw_pin[1]}]
set_property -dict {PACKAGE_PIN P3 IOSTANDARD LVCMOS33} [get_ports {sw_pin[2]}]
set_property -dict {PACKAGE_PIN P2 IOSTANDARD LVCMOS33} [get_ports {sw_pin[3]}]
set_property -dict {PACKAGE_PIN R2 IOSTANDARD LVCMOS33} [get_ports {sw_pin[4]}]
set_property -dict {PACKAGE_PIN M4 IOSTANDARD LVCMOS33} [get_ports {sw_pin[5]}]
set_property -dict {PACKAGE_PIN N4 IOSTANDARD LVCMOS33} [get_ports {sw_pin[6]}]
set_property -dict {PACKAGE_PIN R1 IOSTANDARD LVCMOS33} [get_ports {sw_pin[7]}]
```

//////////////////拨码开关 sw8~sw15//////////////////位置以印刷板丝印为准//////////////////

```
set_property -dict {PACKAGE_PIN U3 IOSTANDARD LVCMOS33} [get_ports {dip_pin[0]}]
```

```
set_property -dict {PACKAGE_PIN U2 IOSTANDARD LVCMOS33} [get_ports {dip_pin[1]}]
set_property -dict {PACKAGE_PIN V2 IOSTANDARD LVCMOS33} [get_ports {dip_pin[2]}]
set_property -dict {PACKAGE_PIN V5 IOSTANDARD LVCMOS33} [get_ports {dip_pin[3]}]
set_property -dict {PACKAGE_PIN V4 IOSTANDARD LVCMOS33} [get_ports {dip_pin[4]}]
set_property -dict {PACKAGE_PIN R3 IOSTANDARD LVCMOS33} [get_ports {dip_pin[5]}]
set_property -dict {PACKAGE_PIN T3 IOSTANDARD LVCMOS33} [get_ports {dip_pin[6]}]
set_property -dict {PACKAGE_PIN T5 IOSTANDARD LVCMOS33} [get_ports {dip_pin[7]}]
```

//////////////////LED0~LED15////////////////位置以印刷板丝印为准////////////////

```
set_property -dict {PACKAGE_PIN F6 IOSTANDARD LVCMOS33} [get_ports {led_pin[0]}]
set_property -dict {PACKAGE_PIN G4 IOSTANDARD LVCMOS33} [get_ports {led_pin[1]}]
set_property -dict {PACKAGE_PIN G3 IOSTANDARD LVCMOS33} [get_ports {led_pin[2]}]
set_property -dict {PACKAGE_PIN J4 IOSTANDARD LVCMOS33} [get_ports {led_pin[3]}]
set_property -dict {PACKAGE_PIN H4 IOSTANDARD LVCMOS33} [get_ports {led_pin[4]}]
set_property -dict {PACKAGE_PIN J3 IOSTANDARD LVCMOS33} [get_ports {led_pin[5]}]
set_property -dict {PACKAGE_PIN J2 IOSTANDARD LVCMOS33} [get_ports {led_pin[6]}]
set_property -dict {PACKAGE_PIN K2 IOSTANDARD LVCMOS33} [get_ports {led_pin[7]}]
```

```
set_property -dict {PACKAGE_PIN K1 IOSTANDARD LVCMOS33} [get_ports {led_pin[8]}]
set_property -dict {PACKAGE_PIN H6 IOSTANDARD LVCMOS33} [get_ports {led_pin[9]}]
set_property -dict {PACKAGE_PIN H5 IOSTANDARD LVCMOS33} [get_ports {led_pin[10]}]
set_property -dict {PACKAGE_PIN J5 IOSTANDARD LVCMOS33} [get_ports {led_pin[11]}]
set_property -dict {PACKAGE_PIN K6 IOSTANDARD LVCMOS33} [get_ports {led_pin[12]}]
set_property -dict {PACKAGE_PIN L1 IOSTANDARD LVCMOS33} [get_ports {led_pin[13]}]
set_property -dict {PACKAGE_PIN M1 IOSTANDARD LVCMOS33} [get_ports {led_pin[14]}]
set_property -dict {PACKAGE_PIN K3 IOSTANDARD LVCMOS33} [get_ports {led_pin[15]}]
```

//////////////////8 个数码管位选信号////////////////

```
set_property -dict {PACKAGE_PIN G2 IOSTANDARD LVCMOS33} [get_ports {seg_cs_pin[0]}]
set_property -dict {PACKAGE_PIN C2 IOSTANDARD LVCMOS33} [get_ports {seg_cs_pin[1]}]
set_property -dict {PACKAGE_PIN C1 IOSTANDARD LVCMOS33} [get_ports {seg_cs_pin[2]}]
set_property -dict {PACKAGE_PIN H1 IOSTANDARD LVCMOS33} [get_ports {seg_cs_pin[3]}]
set_property -dict {PACKAGE_PIN G1 IOSTANDARD LVCMOS33} [get_ports {seg_cs_pin[4]}]
set_property -dict {PACKAGE_PIN F1 IOSTANDARD LVCMOS33} [get_ports {seg_cs_pin[5]}]
set_property -dict {PACKAGE_PIN E1 IOSTANDARD LVCMOS33} [get_ports {seg_cs_pin[6]}]
set_property -dict {PACKAGE_PIN G6 IOSTANDARD LVCMOS33} [get_ports {seg_cs_pin[7]}]
```

//////////////////////////////数码管第 0 组数码管的段选信号/////////////////////////////

```
set_property -dict {PACKAGE_PIN B4 IOSTANDARD LVCMOS33} [get_ports {seg_data_0_pin[0]}]
set_property -dict {PACKAGE_PIN A4 IOSTANDARD LVCMOS33} [get_ports {seg_data_0_pin[1]}]
set_property -dict {PACKAGE_PIN A3 IOSTANDARD LVCMOS33} [get_ports {seg_data_0_pin[2]}]
set_property -dict {PACKAGE_PIN B1 IOSTANDARD LVCMOS33} [get_ports {seg_data_0_pin[3]}]
set_property -dict {PACKAGE_PIN A1 IOSTANDARD LVCMOS33} [get_ports {seg_data_0_pin[4]}]
set_property -dict {PACKAGE_PIN B3 IOSTANDARD LVCMOS33} [get_ports {seg_data_0_pin[5]}]
set_property -dict {PACKAGE_PIN B2 IOSTANDARD LVCMOS33} [get_ports {seg_data_0_pin[6]}]
set_property -dict {PACKAGE_PIN D5 IOSTANDARD LVCMOS33} [get_ports {seg_data_0_pin[7]}]
```

```
set_property -dict {PACKAGE_PIN D4 IOSTANDARD LVCMOS33} [get_ports {seg_data_1_pin[0]}]
set_property -dict {PACKAGE_PIN E3 IOSTANDARD LVCMOS33} [get_ports {seg_data_1_pin[1]}]
set_property -dict {PACKAGE_PIN D3 IOSTANDARD LVCMOS33} [get_ports {seg_data_1_pin[2]}]
set_property -dict {PACKAGE_PIN F4 IOSTANDARD LVCMOS33} [get_ports {seg_data_1_pin[3]}]
set_property -dict {PACKAGE_PIN F3 IOSTANDARD LVCMOS33} [get_ports {seg_data_1_pin[4]}]
set_property -dict {PACKAGE_PIN E2 IOSTANDARD LVCMOS33} [get_ports {seg_data_1_pin[5]}]
set_property -dict {PACKAGE_PIN D2 IOSTANDARD LVCMOS33} [get_ports {seg_data_1_pin[6]}]
set_property -dict {PACKAGE_PIN H2 IOSTANDARD LVCMOS33} [get_ports {seg_data_1_pin[7]}]
```

/////////////////////////////VGA 行同步场同步信号/////////////////////////////

```
set_property -dict {PACKAGE_PIN D7 IOSTANDARD LVCMOS33} [get_ports vga_hs_pin]
set_property -dict {PACKAGE_PIN C4 IOSTANDARD LVCMOS33} [get_ports vga_vs_pin]
```

/////////////////////////////VGA 红绿蓝信号/////////////////////////////

```
set_property -dict {PACKAGE_PIN F5 IOSTANDARD LVCMOS33} [get_ports {vga_data_pin[0]}]
set_property -dict {PACKAGE_PIN C6 IOSTANDARD LVCMOS33} [get_ports {vga_data_pin[1]}]
set_property -dict {PACKAGE_PIN C5 IOSTANDARD LVCMOS33} [get_ports {vga_data_pin[2]}]
set_property -dict {PACKAGE_PIN B7 IOSTANDARD LVCMOS33} [get_ports {vga_data_pin[3]}]
set_property -dict {PACKAGE_PIN B6 IOSTANDARD LVCMOS33} [get_ports {vga_data_pin[4]}]
set_property -dict {PACKAGE_PIN A6 IOSTANDARD LVCMOS33} [get_ports {vga_data_pin[5]}]
set_property -dict {PACKAGE_PIN A5 IOSTANDARD LVCMOS33} [get_ports {vga_data_pin[6]}]
set_property -dict {PACKAGE_PIN D8 IOSTANDARD LVCMOS33} [get_ports {vga_data_pin[7]}]
set_property -dict {PACKAGE_PIN C7 IOSTANDARD LVCMOS33} [get_ports {vga_data_pin[8]}]
set_property -dict {PACKAGE_PIN E6 IOSTANDARD LVCMOS33} [get_ports {vga_data_pin[9]}]
set_property -dict {PACKAGE_PIN E5 IOSTANDARD LVCMOS33} [get_ports {vga_data_pin[10]}]
set_property -dict {PACKAGE_PIN E7 IOSTANDARD LVCMOS33} [get_ports {vga_data_pin[11]}]
```

/////////////////////////////DAC 数模转换/////////////////////////////

```

set_property -dict {PACKAGE_PIN R5 IOSTANDARD LVCMOS33} [get_ports dac_ile]
set_property -dict {PACKAGE_PIN N6 IOSTANDARD LVCMOS33} [get_ports dac_cs_n]
set_property -dict {PACKAGE_PIN V6 IOSTANDARD LVCMOS33} [get_ports dac_wr1_n]
set_property -dict {PACKAGE_PIN R6 IOSTANDARD LVCMOS33} [get_ports dac_wr2_n]
set_property -dict {PACKAGE_PIN V7 IOSTANDARD LVCMOS33} [get_ports dac_xfer_n]

set_property -dict {PACKAGE_PIN T8 IOSTANDARD LVCMOS33} [get_ports {dac_data[0]}]
set_property -dict {PACKAGE_PIN R8 IOSTANDARD LVCMOS33} [get_ports {dac_data[1]}]
set_property -dict {PACKAGE_PIN T6 IOSTANDARD LVCMOS33} [get_ports {dac_data[2]}]
set_property -dict {PACKAGE_PIN R7 IOSTANDARD LVCMOS33} [get_ports {dac_data[3]}]
set_property -dict {PACKAGE_PIN U6 IOSTANDARD LVCMOS33} [get_ports {dac_data[4]}]
set_property -dict {PACKAGE_PIN U7 IOSTANDARD LVCMOS33} [get_ports {dac_data[5]}]
set_property -dict {PACKAGE_PIN V9 IOSTANDARD LVCMOS33} [get_ports {dac_data[6]}]
set_property -dict {PACKAGE_PIN U9 IOSTANDARD LVCMOS33} [get_ports {dac_data[7]}]

```

//////////PS2//////////

```

set_property -dict {PACKAGE_PIN K5 IOSTANDARD LVCMOS33} [get_ports ps2_clk ]
set_property -dict {PACKAGE_PIN L4 IOSTANDARD LVCMOS33} [get_ports ps2_data ]

```

//////////SDRAM//////////

```

set_property -dict {PACKAGE_PIN L15 IOSTANDARD LVCMOS33} [get_ports {sram_addr[18]}]
set_property -dict {PACKAGE_PIN L16 IOSTANDARD LVCMOS33} [get_ports {sram_addr[17]}]
set_property -dict {PACKAGE_PIN L18 IOSTANDARD LVCMOS33} [get_ports {sram_addr[16]}]
set_property -dict {PACKAGE_PIN M18 IOSTANDARD LVCMOS33} [get_ports {sram_addr[15]}]
set_property -dict {PACKAGE_PIN R12 IOSTANDARD LVCMOS33} [get_ports {sram_addr[14]}]
set_property -dict {PACKAGE_PIN R13 IOSTANDARD LVCMOS33} [get_ports {sram_addr[13]}]
set_property -dict {PACKAGE_PIN M13 IOSTANDARD LVCMOS33} [get_ports {sram_addr[12]}]
set_property -dict {PACKAGE_PIN R18 IOSTANDARD LVCMOS33} [get_ports {sram_addr[11]}]
set_property -dict {PACKAGE_PIN T18 IOSTANDARD LVCMOS33} [get_ports {sram_addr[10]}]
set_property -dict {PACKAGE_PIN N14 IOSTANDARD LVCMOS33} [get_ports {sram_addr[9]}]
set_property -dict {PACKAGE_PIN P14 IOSTANDARD LVCMOS33} [get_ports {sram_addr[8]}]
set_property -dict {PACKAGE_PIN N17 IOSTANDARD LVCMOS33} [get_ports {sram_addr[7]}]
set_property -dict {PACKAGE_PIN P18 IOSTANDARD LVCMOS33} [get_ports {sram_addr[6]}]
set_property -dict {PACKAGE_PIN M16 IOSTANDARD LVCMOS33} [get_ports {sram_addr[5]}]
set_property -dict {PACKAGE_PIN M17 IOSTANDARD LVCMOS33} [get_ports {sram_addr[4]}]
set_property -dict {PACKAGE_PIN N15 IOSTANDARD LVCMOS33} [get_ports {sram_addr[3]}]
set_property -dict {PACKAGE_PIN N16 IOSTANDARD LVCMOS33} [get_ports {sram_addr[2]}]

```

```

set_property -dict {PACKAGE_PIN T14 IOSTANDARD LVCMOS33} [get_ports {sram_addr[1]}]
set_property -dict {PACKAGE_PIN T15 IOSTANDARD LVCMOS33} [get_ports {sram_addr[0]}]

set_property -dict {PACKAGE_PIN V15 IOSTANDARD LVCMOS33} [get_ports sram_ce_n]
set_property -dict {PACKAGE_PIN R10 IOSTANDARD LVCMOS33} [get_ports sram_lb_n]
set_property -dict {PACKAGE_PIN T16 IOSTANDARD LVCMOS33} [get_ports sram_oe_n]
set_property -dict {PACKAGE_PIN R16 IOSTANDARD LVCMOS33} [get_ports sram_ub_n]
set_property -dict {PACKAGE_PIN V16 IOSTANDARD LVCMOS33} [get_ports sram_we_n]

set_property -dict {PACKAGE_PIN T10 IOSTANDARD LVCMOS33} [get_ports {sram_data[15]}]
set_property -dict {PACKAGE_PIN T9 IOSTANDARD LVCMOS33} [get_ports {sram_data[14]}]
set_property -dict {PACKAGE_PIN U13 IOSTANDARD LVCMOS33} [get_ports {sram_data[13]}]
set_property -dict {PACKAGE_PIN T13 IOSTANDARD LVCMOS33} [get_ports {sram_data[12]}]
set_property -dict {PACKAGE_PIN V14 IOSTANDARD LVCMOS33} [get_ports {sram_data[11]}]
set_property -dict {PACKAGE_PIN U14 IOSTANDARD LVCMOS33} [get_ports {sram_data[10]}]
set_property -dict {PACKAGE_PIN V11 IOSTANDARD LVCMOS33} [get_ports {sram_data[9]}]
set_property -dict {PACKAGE_PIN V10 IOSTANDARD LVCMOS33} [get_ports {sram_data[8]}]
set_property -dict {PACKAGE_PIN V12 IOSTANDARD LVCMOS33} [get_ports {sram_data[7]}]
set_property -dict {PACKAGE_PIN U12 IOSTANDARD LVCMOS33} [get_ports {sram_data[6]}]
set_property -dict {PACKAGE_PIN U11 IOSTANDARD LVCMOS33} [get_ports {sram_data[5]}]
set_property -dict {PACKAGE_PIN T11 IOSTANDARD LVCMOS33} [get_ports {sram_data[4]}]
set_property -dict {PACKAGE_PIN V17 IOSTANDARD LVCMOS33} [get_ports {sram_data[3]}]
set_property -dict {PACKAGE_PIN U16 IOSTANDARD LVCMOS33} [get_ports {sram_data[2]}]
set_property -dict {PACKAGE_PIN U18 IOSTANDARD LVCMOS33} [get_ports {sram_data[1]}]
set_property -dict {PACKAGE_PIN U17 IOSTANDARD LVCMOS33} [get_ports {sram_data[0]}]

```

/////////////////////////////32 个 pmod 接口/////////////////////////////

```

set_property -dict {PACKAGE_PIN B16 IOSTANDARD LVCMOS33} [get_ports {exp_io[0]} ]
set_property -dict {PACKAGE_PIN A15 IOSTANDARD LVCMOS33} [get_ports {exp_io[1]} ]
set_property -dict {PACKAGE_PIN A13 IOSTANDARD LVCMOS33} [get_ports {exp_io[2]} ]
set_property -dict {PACKAGE_PIN B18 IOSTANDARD LVCMOS33} [get_ports {exp_io[3]} ]
set_property -dict {PACKAGE_PIN F13 IOSTANDARD LVCMOS33} [get_ports {exp_io[4]} ]
set_property -dict {PACKAGE_PIN B13 IOSTANDARD LVCMOS33} [get_ports {exp_io[5]} ]
set_property -dict {PACKAGE_PIN D14 IOSTANDARD LVCMOS33} [get_ports {exp_io[6]} ]
set_property -dict {PACKAGE_PIN B11 IOSTANDARD LVCMOS33} [get_ports {exp_io[7]} ]
set_property -dict {PACKAGE_PIN E15 IOSTANDARD LVCMOS33} [get_ports {exp_io[8]} ]
set_property -dict {PACKAGE_PIN D15 IOSTANDARD LVCMOS33} [get_ports {exp_io[9]} ]
set_property -dict {PACKAGE_PIN H16 IOSTANDARD LVCMOS33} [get_ports {exp_io[10]}]

```

```
set_property -dict {PACKAGE_PIN F15 IOSTANDARD LVCMOS33} [get_ports {exp_io[11]}]
set_property -dict {PACKAGE_PIN H14 IOSTANDARD LVCMOS33} [get_ports {exp_io[12]}]
set_property -dict {PACKAGE_PIN E17 IOSTANDARD LVCMOS33} [get_ports {exp_io[13]}]
set_property -dict {PACKAGE_PIN K13 IOSTANDARD LVCMOS33} [get_ports {exp_io[14]}]
set_property -dict {PACKAGE_PIN H17 IOSTANDARD LVCMOS33} [get_ports {exp_io[15]}]

set_property -dict {PACKAGE_PIN B17 IOSTANDARD LVCMOS33} [get_ports {exp_io[16]}]
set_property -dict {PACKAGE_PIN A16 IOSTANDARD LVCMOS33} [get_ports {exp_io[17]}]
set_property -dict {PACKAGE_PIN A14 IOSTANDARD LVCMOS33} [get_ports {exp_io[18]}]
set_property -dict {PACKAGE_PIN A18 IOSTANDARD LVCMOS33} [get_ports {exp_io[19]}]
set_property -dict {PACKAGE_PIN F14 IOSTANDARD LVCMOS33} [get_ports {exp_io[20]}]
set_property -dict {PACKAGE_PIN B14 IOSTANDARD LVCMOS33} [get_ports {exp_io[21]}]
set_property -dict {PACKAGE_PIN C14 IOSTANDARD LVCMOS33} [get_ports {exp_io[22]}]
set_property -dict {PACKAGE_PIN A11 IOSTANDARD LVCMOS33} [get_ports {exp_io[23]}]
set_property -dict {PACKAGE_PIN E16 IOSTANDARD LVCMOS33} [get_ports {exp_io[24]}]
set_property -dict {PACKAGE_PIN C15 IOSTANDARD LVCMOS33} [get_ports {exp_io[25]}]
set_property -dict {PACKAGE_PIN G16 IOSTANDARD LVCMOS33} [get_ports {exp_io[26]}]
set_property -dict {PACKAGE_PIN F16 IOSTANDARD LVCMOS33} [get_ports {exp_io[27]}]
set_property -dict {PACKAGE_PIN G14 IOSTANDARD LVCMOS33} [get_ports {exp_io[28]}]
set_property -dict {PACKAGE_PIN D17 IOSTANDARD LVCMOS33} [get_ports {exp_io[29]}]
set_property -dict {PACKAGE_PIN J13 IOSTANDARD LVCMOS33} [get_ports {exp_io[30]}]
set_property -dict {PACKAGE_PIN G17 IOSTANDARD LVCMOS33} [get_ports {exp_io[31]}]
```

3. 附录 C 操作常见问题及注意事项

1、新建工程时没有选择合适的型号怎么办？

答：可以在 Flow Navigator 的 Project Manager 下 settings 中重新选择。

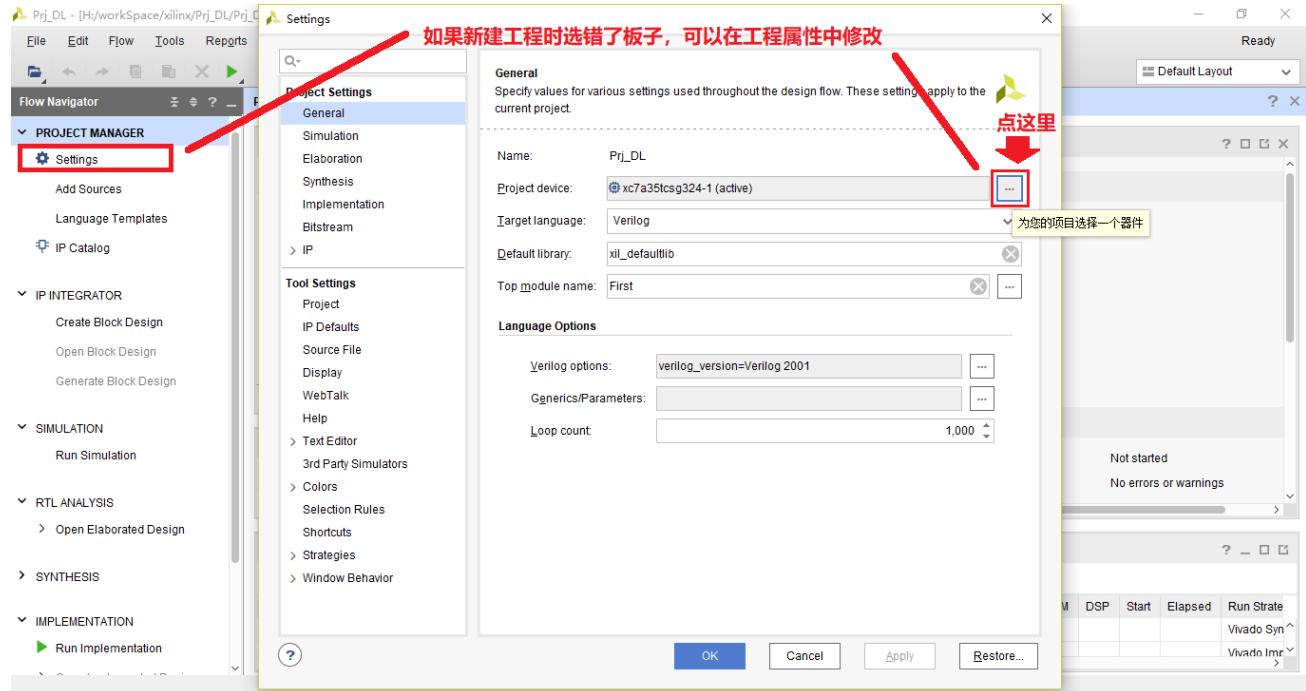


图 3.1