



学号 19335286 姓名 郑有为

【实验题目】组合逻辑实验

【实验目的】掌握 FPGA 实现组合逻辑的基本方法。

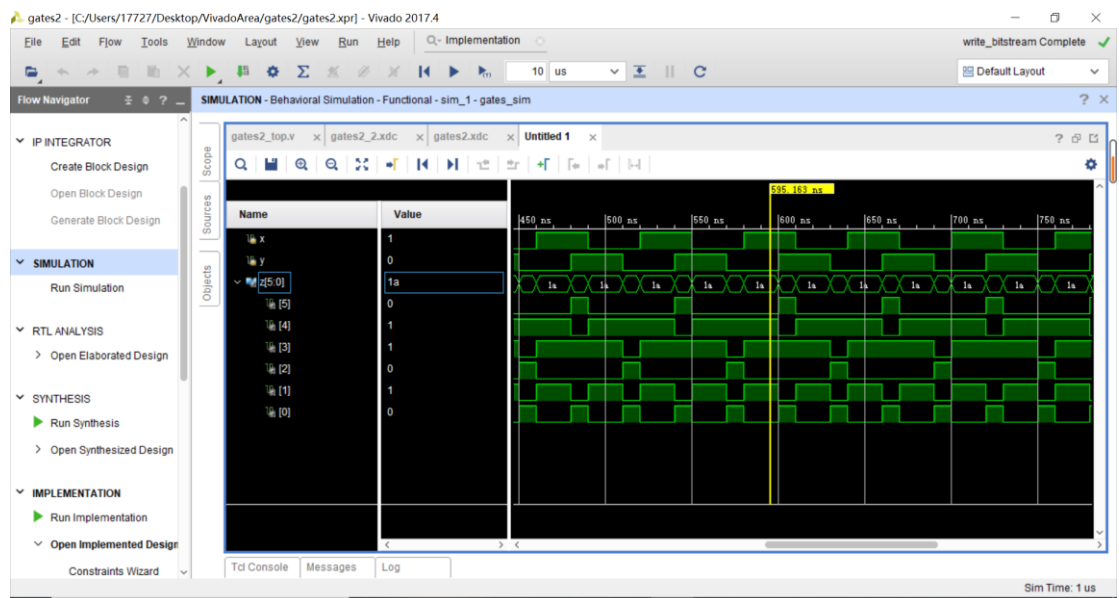
【实验说明】

- ◆ 安装[教学网站](#)的软件下载中下载 Vivado 进行安装。
- ◆ 所有的板上运行拍照都采用自选运行案例拍照（一般选择 4 张），并对案例进行说明。

【实验内容】

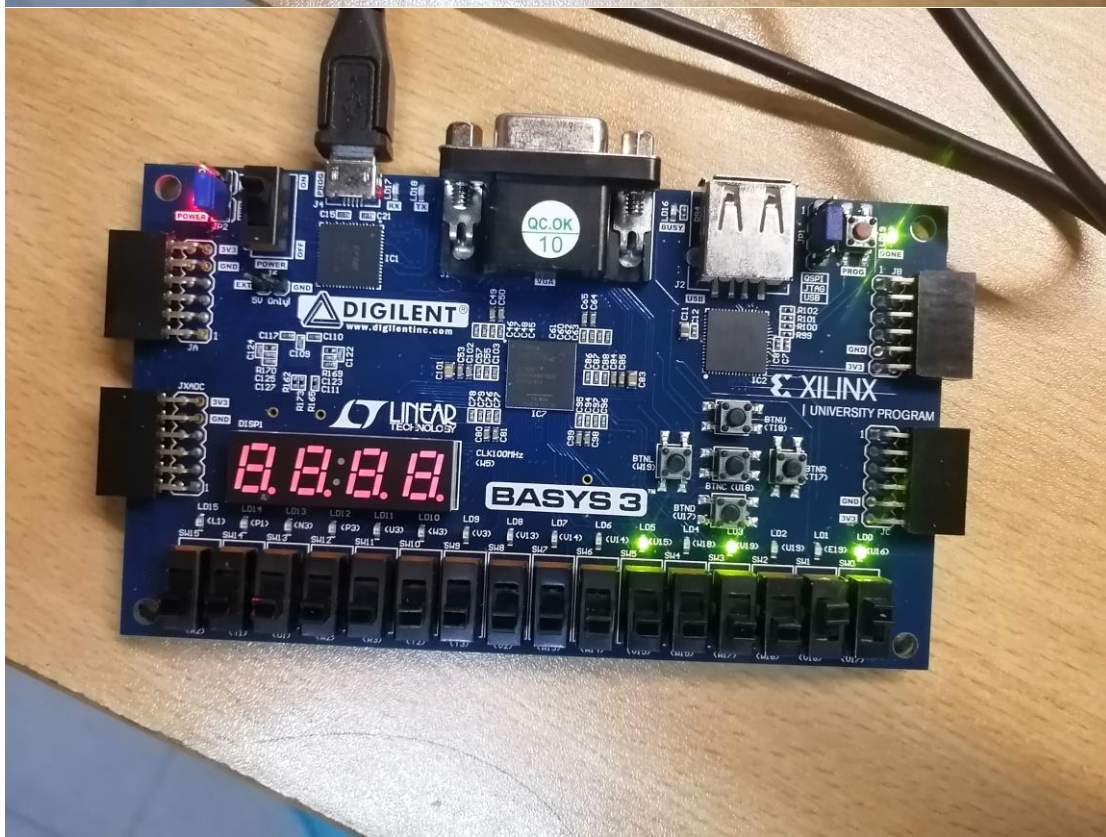
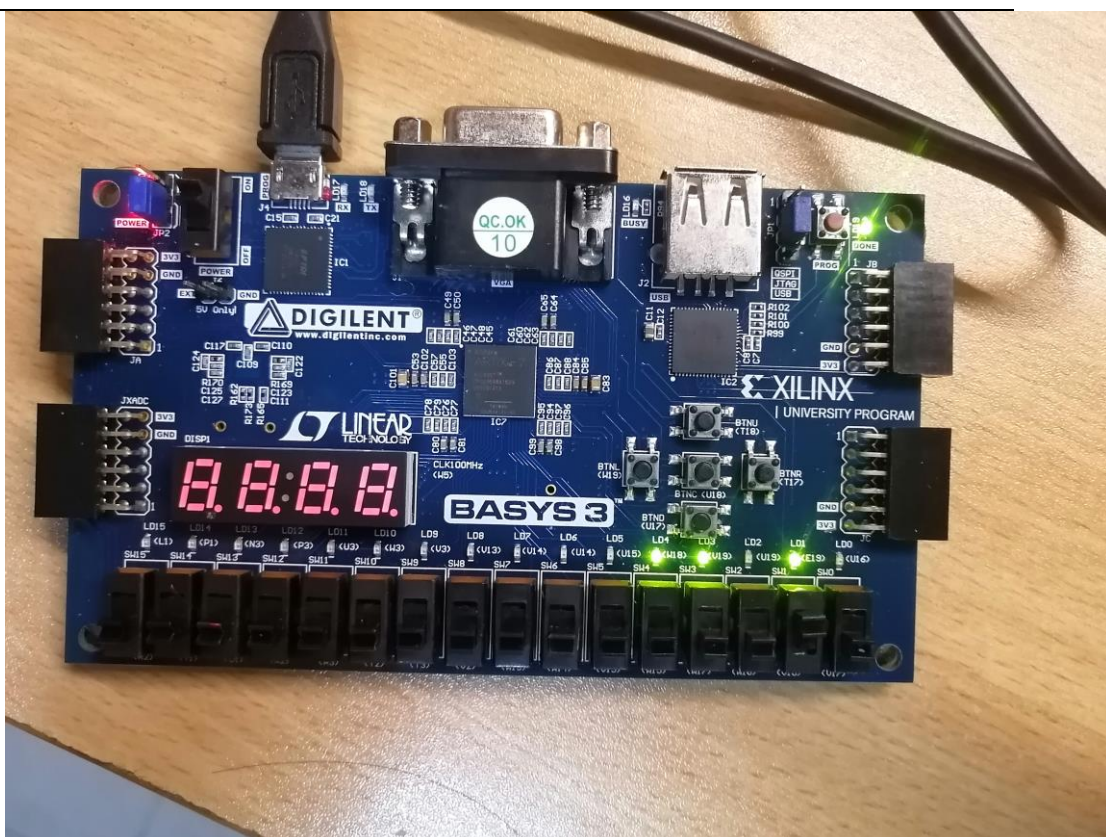
- (1) 按照“Vivado 工程 1.pdf”中的案例进行实验。最后把实验结果拍照粘贴。

仿真结果截图：



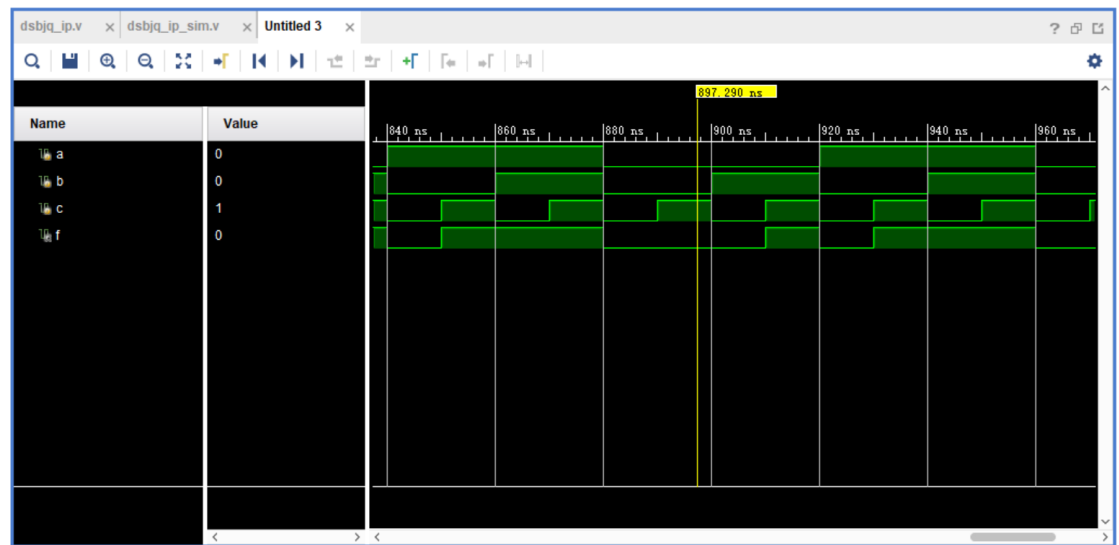
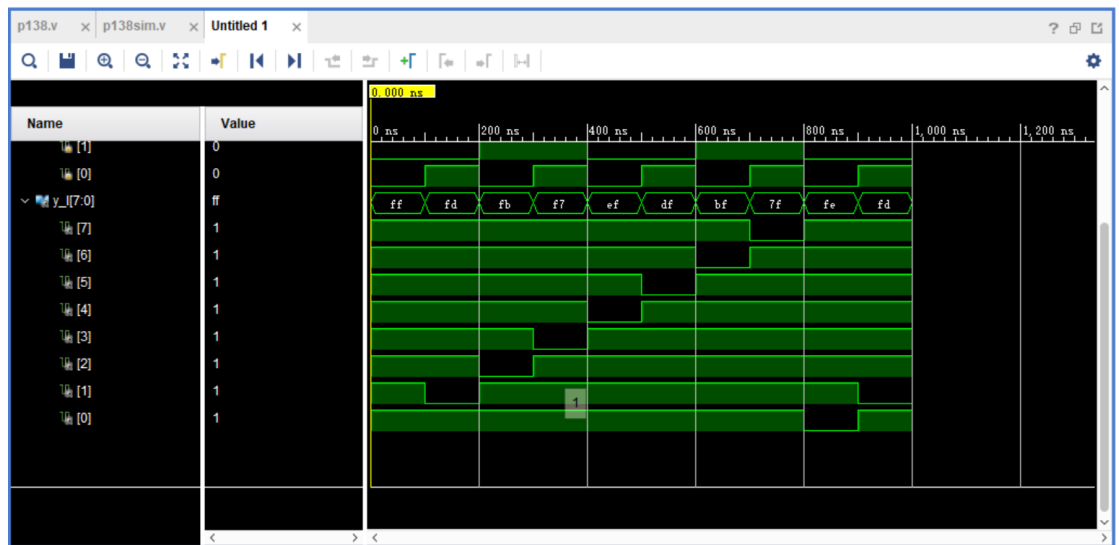
板上运行拍照：



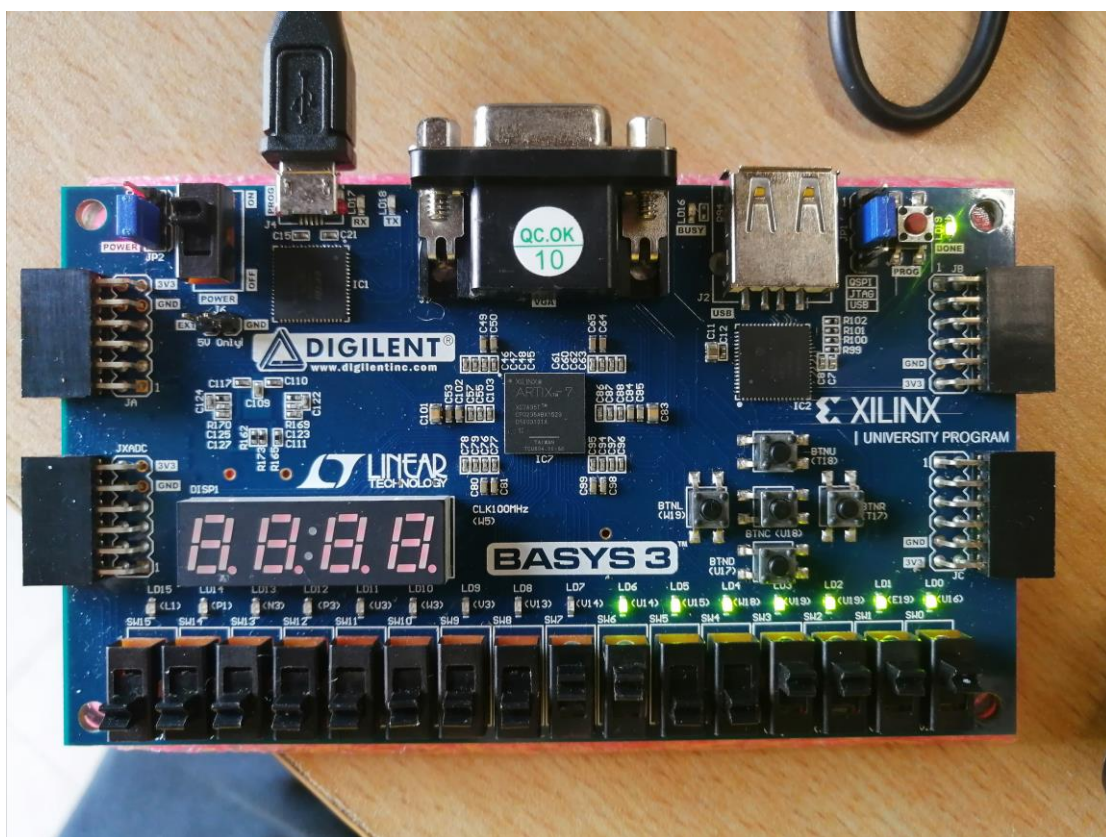
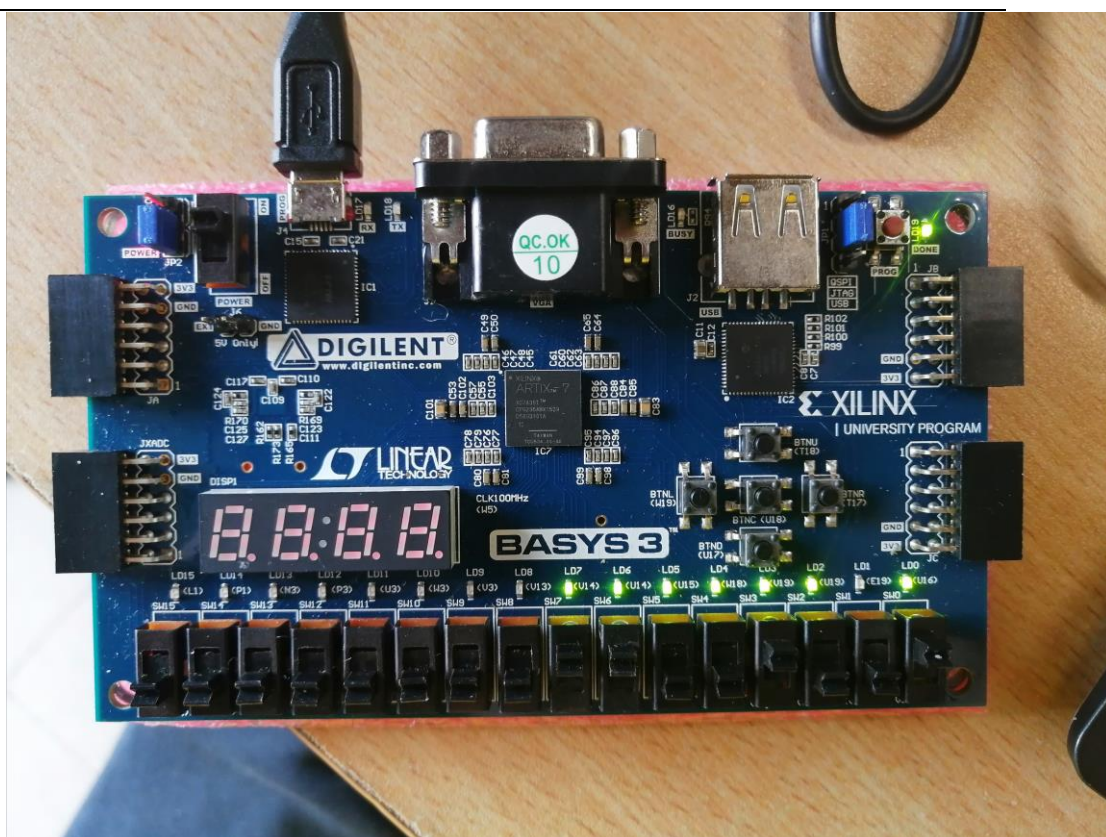


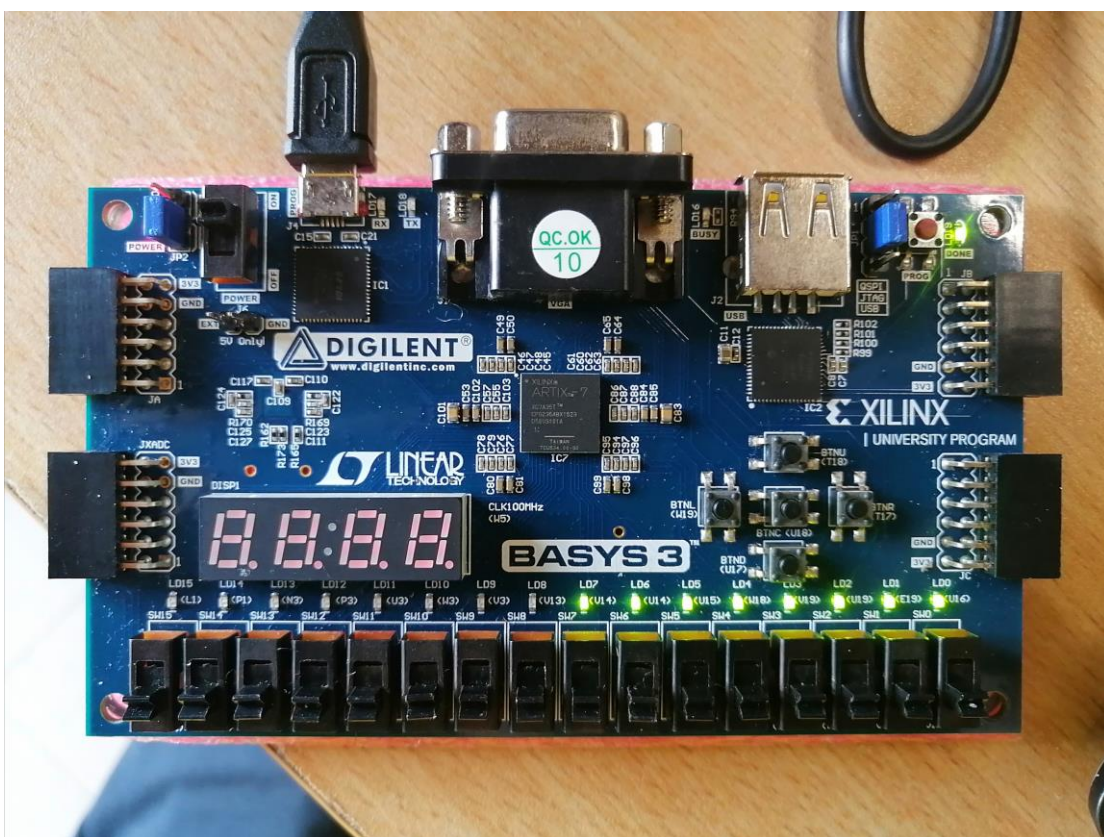
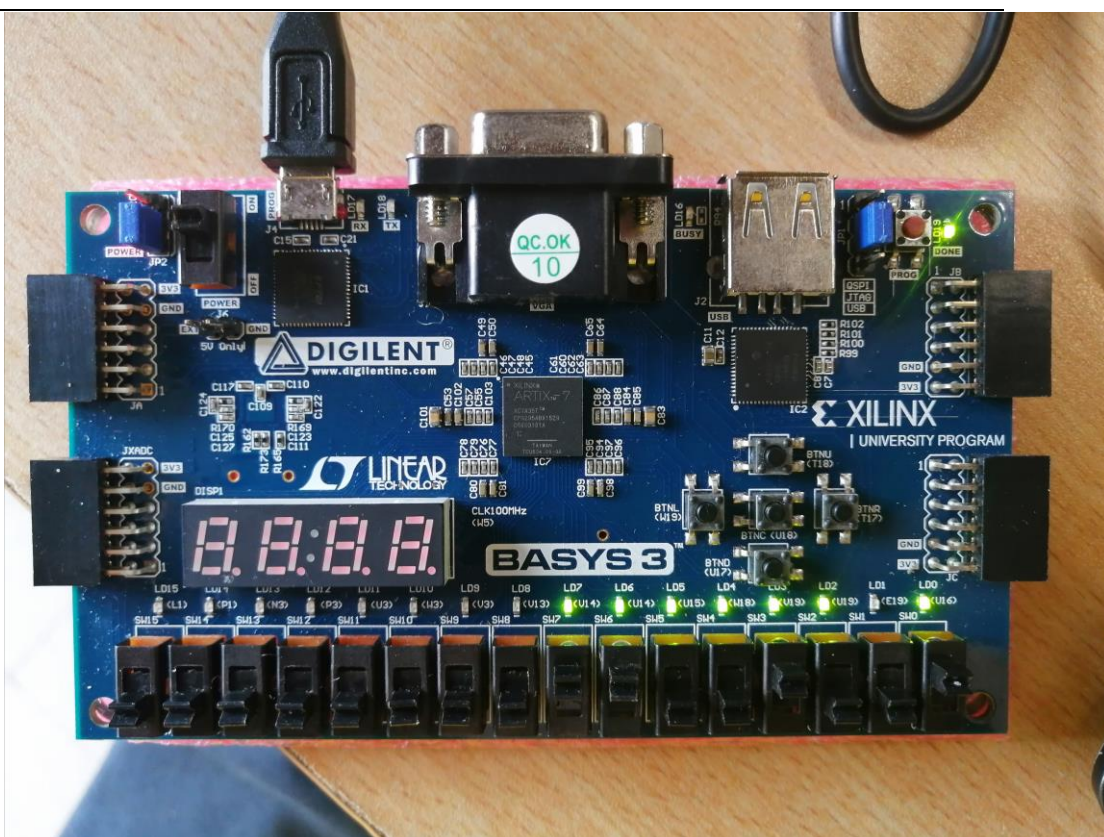
- (2) 按照“Vivado 工程 2. pdf”中的案例进行实验，先生成 38 译码器模块的 IP 核，再利用该 IP 核实现多数表决器（要求参考附录写入 flash 使得重启后可以直接运行）。

仿真结果截图：

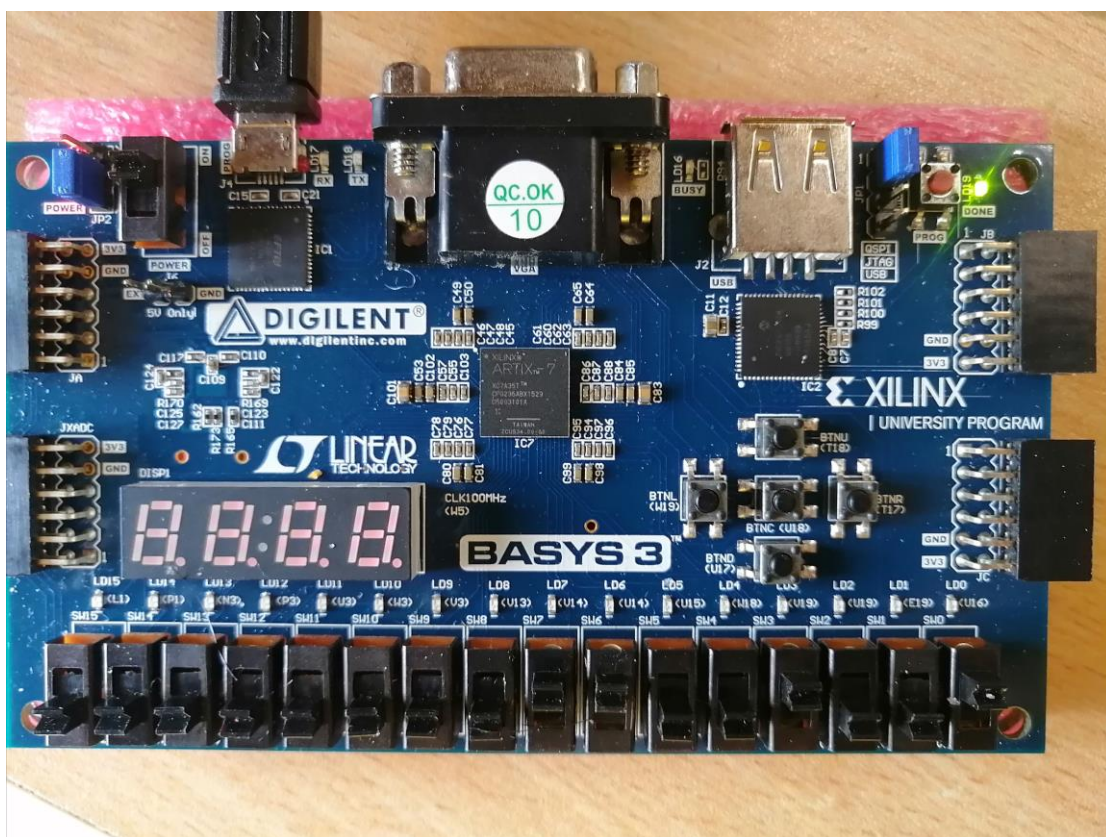
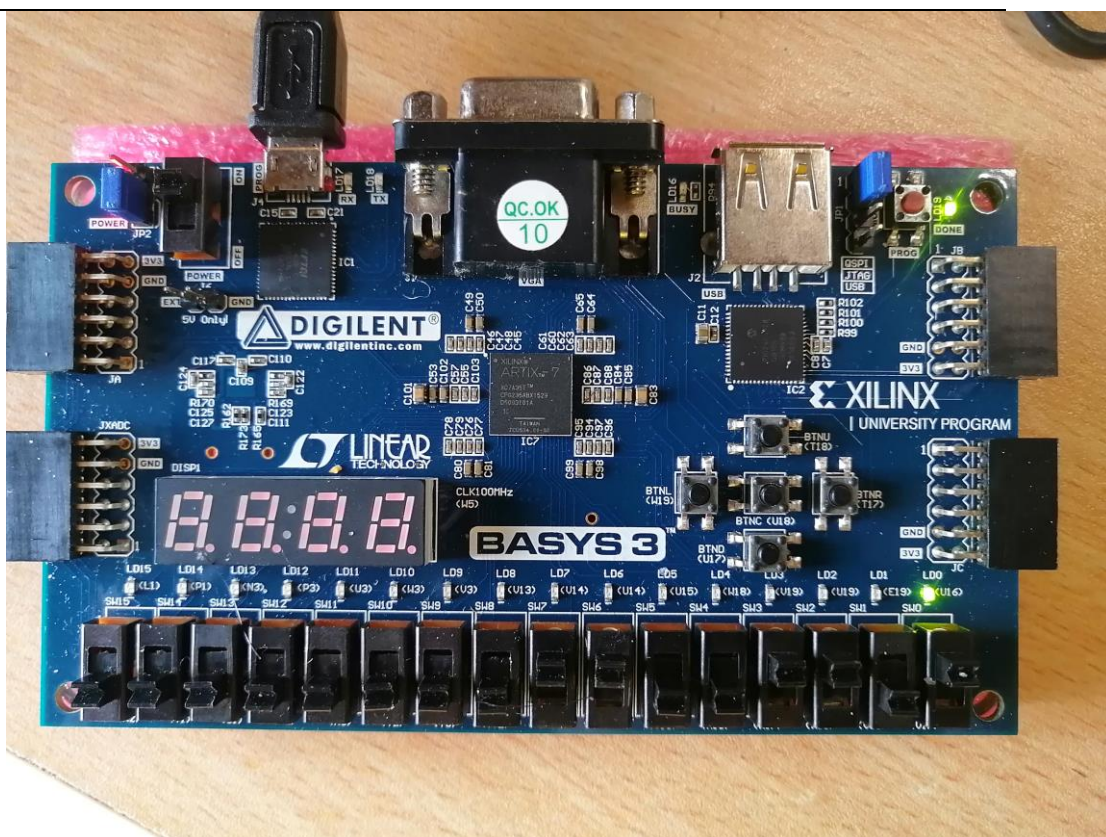


板上运行拍照：

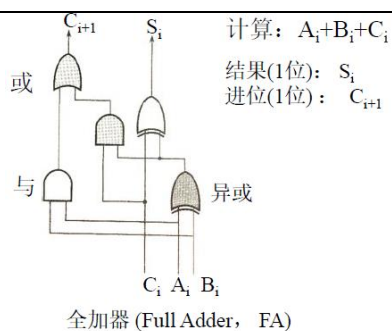




多数表决器:



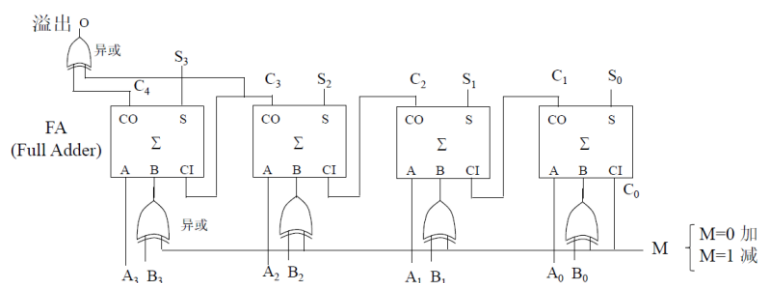
- (3) 先实现一位全加器模块 full_adder: A_i 、 B_i 、 C_i 、 S_i 分别用 A_i 、 B_i 、 C_i 和 S_i 表示, C_{i+1} 用 C_{i_1} 表示。



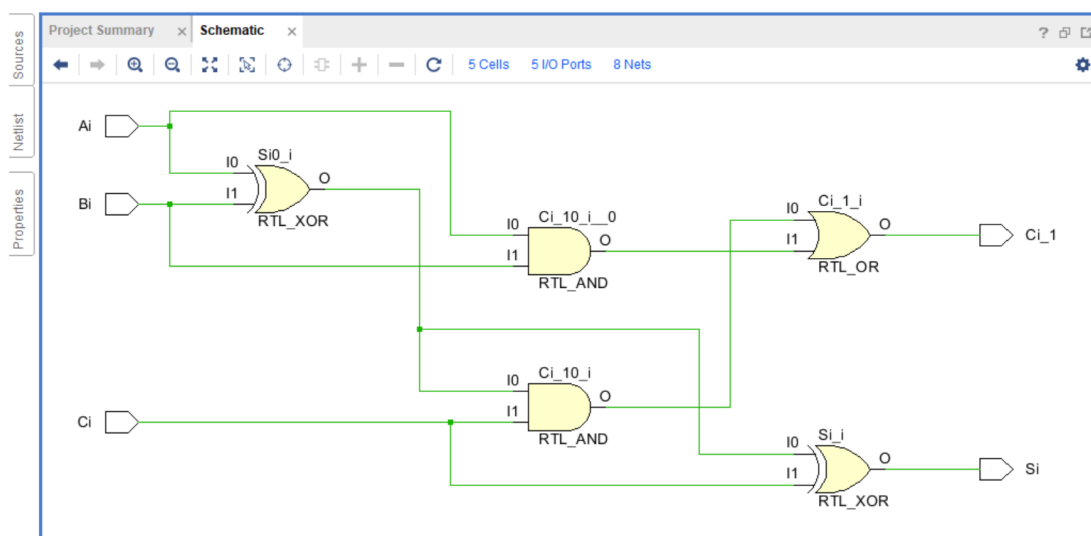
再利用一位全加器模块 adder 实现如下的四位加减法器。 以下是全加器模块 Adder 变量与外设的对应关系:

A0~A3 --- sw0~sw3
B0~B3 --- sw4~sw7
M --- sw8
S0~S3 --- led0~led3
C4 --- led4
O --- led5 (溢出位)

为了方便写约束源文件,可以再加一个的设计源代码模块 adder_top, 直接使用 sw 和 led 作为参数。

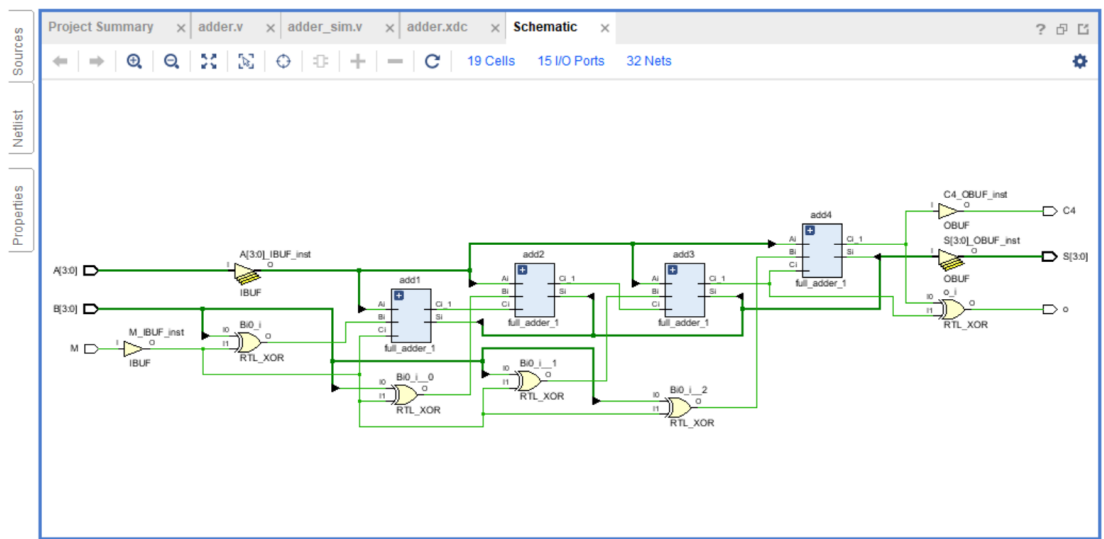


RTL ANALYSIS/Schematic --- full_adder (要点击+展开细节):

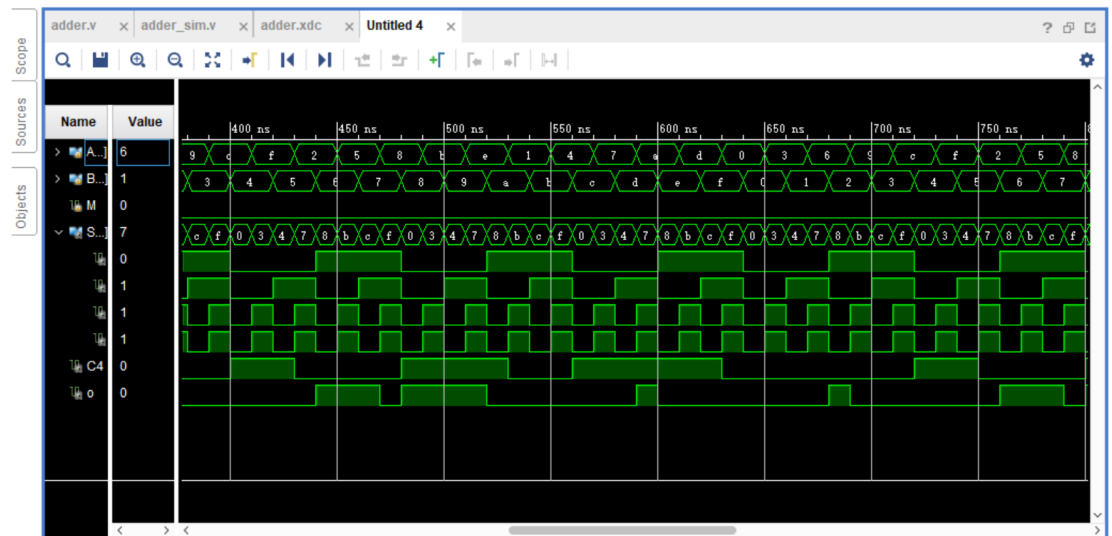




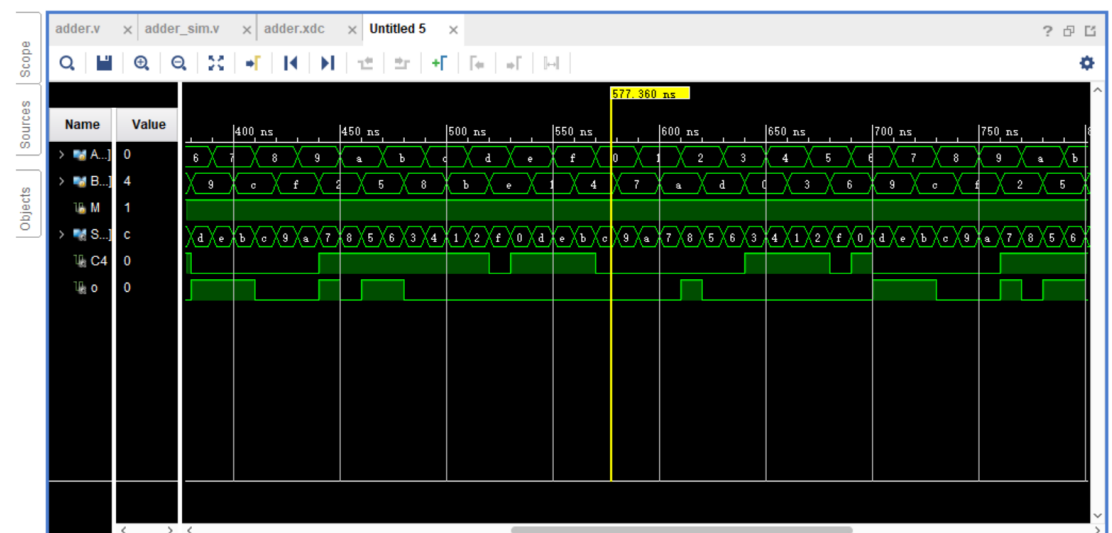
RTL ANALYSIS/Schematic --- adder (要点击+展开细节):



adder 仿真结果截图--加法 (a 从 0 开始每隔 10 秒加 3, 同时 b 从 0 开始每隔 10 秒加 1):



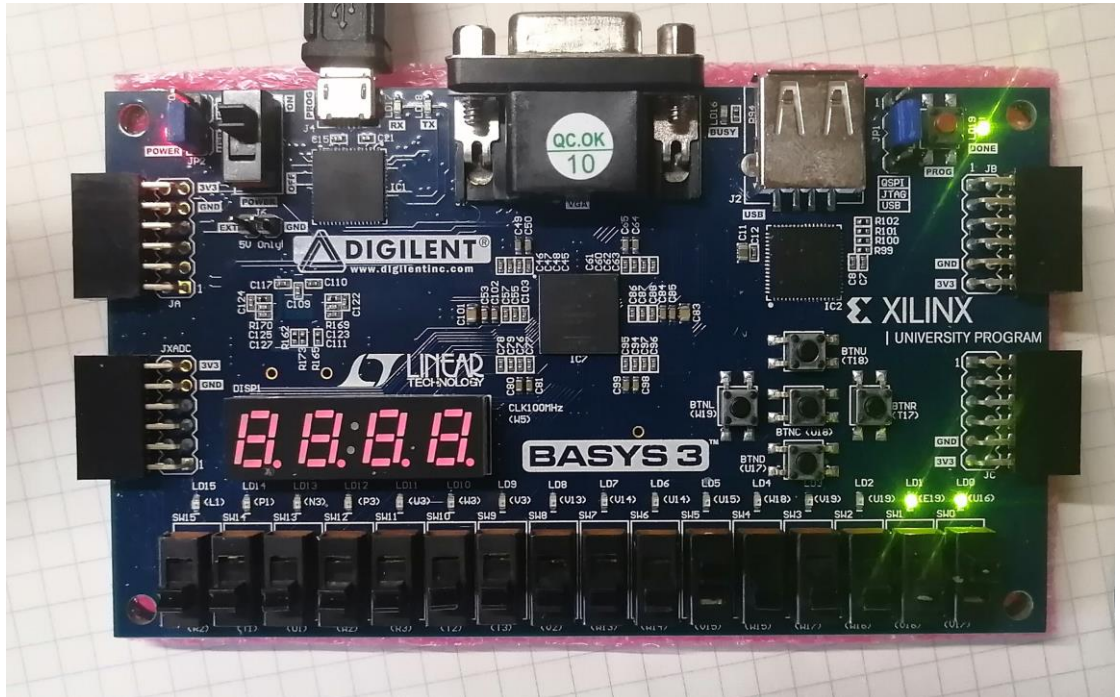
adder 仿真结果截图--减法 (a 从 3 开始每隔 10 秒加 1, 同时 b 从 0 开始每隔 10 秒加 3):



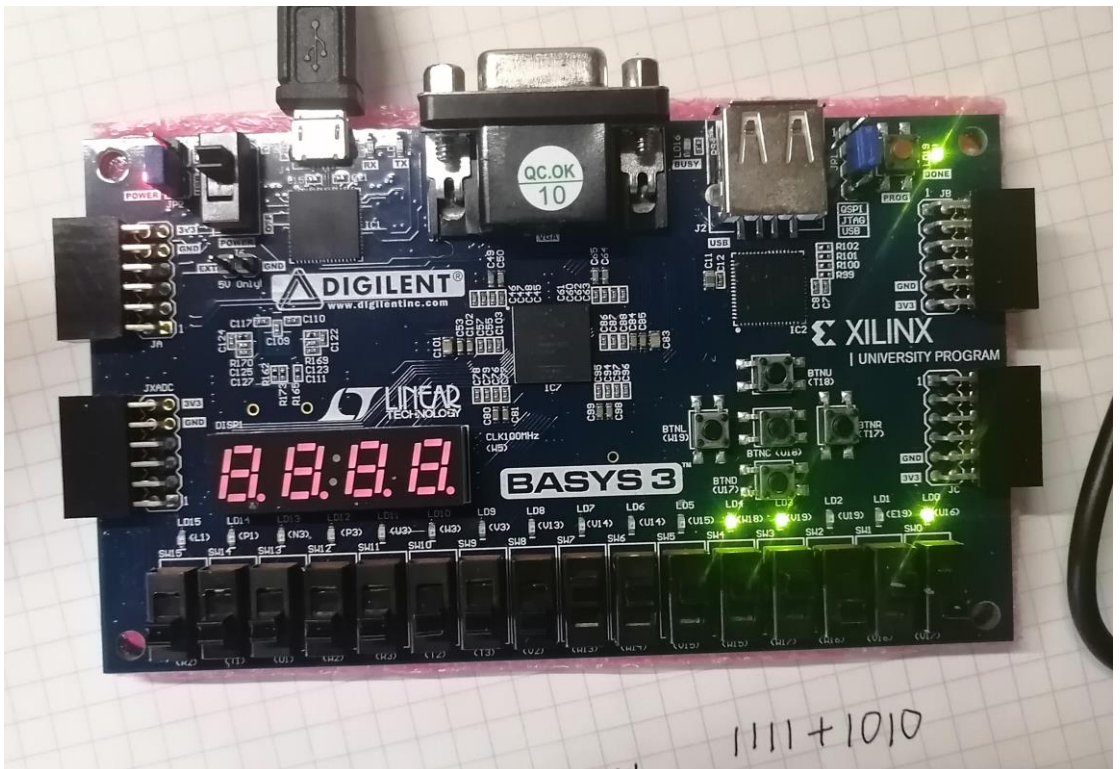


板上运行拍照：

$$0001 + 0010 = 0011:$$

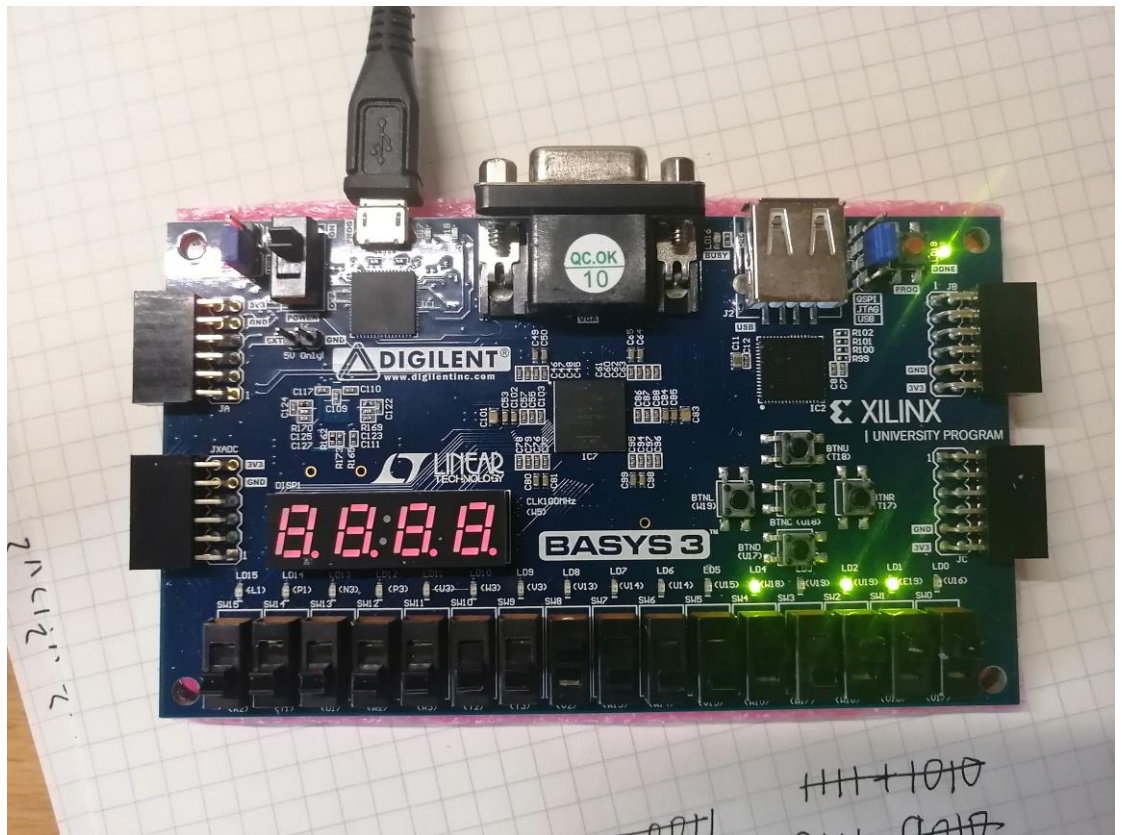


$$1111 + 1010 = 11001 \text{ 即 } -1 + -6 = -7$$

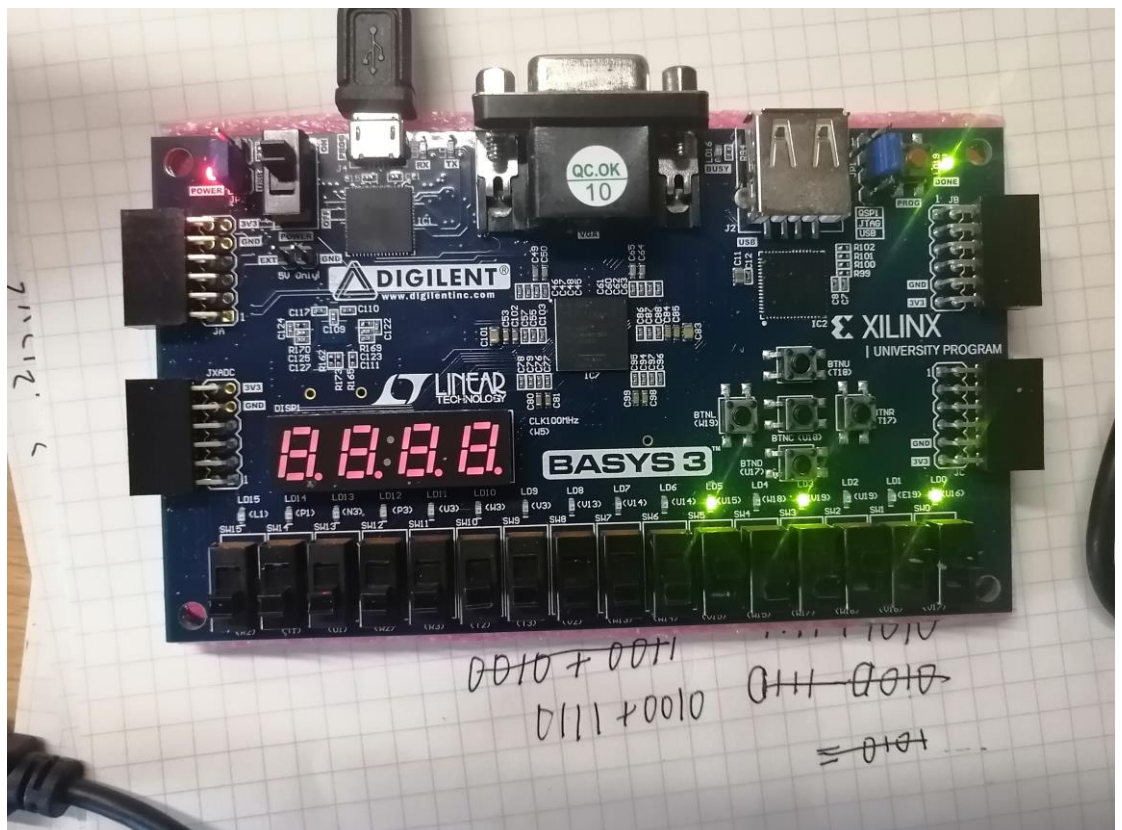




0111 - 0001 = 0110 即 $7 - 1 = 6$



0111 + 0010 = 1001 溢出



设计源代码 (design source):



```
`timescale 1ns / 1ps

module full_adder(
    input Ai,
    input Bi,
    input Ci,
    output Si,
    output Ci_1
);
    assign Si = (Ai^Bi)^Ci;
    assign Ci_1 = ((Ai^Bi)&Ci)|(Ai&Bi);
endmodule

`timescale 1ns / 1ps

module adder(
    input [3:0]A,
    input [3:0]B,
    input M,
    output [3:0]S,
    output C4,
    output o
);
    wire C0,C1,C2;
    full_adder_1 add1(A[0],B[0]^M,M,S[0],C0);
    full_adder_1 add2(A[1],B[1]^M,C0,S[1],C1);
    full_adder_1 add3(A[2],B[2]^M,C1,S[2],C2);
    full_adder_1 add4(A[3],B[3]^M,C2,S[3],C4);
    assign o = C4^C2;
endmodule
```

仿真源代码(simulation source):

```
`timescale 1ns / 1ps
module adder_sim();
    reg [3:0] A;
    reg [3:0] B;
    reg M;
    wire [3:0] S;
    wire C4;
    wire o;
    /*
    adder add(.A(A),.B(B),.M(M),.S(S),.C4(C4),.o(o));
    initial begin
        A = 0;
        B = 0;
```




```
M = 0;

end

always begin

    #10 {A} = {A} + 3;

    #10 {B} = {B} + 1;

end

*/

adder add(. A(A),. B(B),. M(M),. S(S),. C4(C4),. o(o));

    initial begin

        A = 3;

        B = 0;

        M = 1;

    end

    always begin

        #10 {A} = {A} + 1;

        #10 {B} = {B} + 3;

    end

endmodule
```

约束源代码(constraints source):

```
set_property PACKAGE_PIN V17 [get_ports A[0]]
set_property IOSTANDARD LVCMOS33 [get_ports A[0]]
set_property PACKAGE_PIN V16 [get_ports A[1]]
set_property IOSTANDARD LVCMOS33 [get_ports A[1]]
set_property PACKAGE_PIN W16 [get_ports A[2]]
set_property IOSTANDARD LVCMOS33 [get_ports A[2]]
set_property PACKAGE_PIN W17 [get_ports A[3]]
set_property IOSTANDARD LVCMOS33 [get_ports A[3]]
set_property PACKAGE_PIN W15 [get_ports B[0]]
set_property IOSTANDARD LVCMOS33 [get_ports B[0]]
set_property PACKAGE_PIN V15 [get_ports B[1]]
set_property IOSTANDARD LVCMOS33 [get_ports B[1]]
set_property PACKAGE_PIN W14 [get_ports B[2]]
set_property IOSTANDARD LVCMOS33 [get_ports B[2]]
set_property PACKAGE_PIN W13 [get_ports B[3]]
set_property IOSTANDARD LVCMOS33 [get_ports B[3]]
set_property PACKAGE_PIN V2 [get_ports M]
set_property IOSTANDARD LVCMOS33 [get_ports M]
set_property PACKAGE_PIN U16 [get_ports S[0]]
set_property IOSTANDARD LVCMOS33 [get_ports S[0]]
set_property PACKAGE_PIN E19 [get_ports S[1]]
set_property IOSTANDARD LVCMOS33 [get_ports S[1]]
set_property PACKAGE_PIN U19 [get_ports S[2]]
```



```
set_property IOSTANDARD LVCMOS33 [get_ports S[2]]  
set_property PACKAGE_PIN V19 [get_ports S[3]]  
set_property IOSTANDARD LVCMOS33 [get_ports S[3]]  
set_property PACKAGE_PIN W18 [get_ports C4]  
set_property IOSTANDARD LVCMOS33 [get_ports C4]  
set_property PACKAGE_PIN U15 [get_ports o]  
set_property IOSTANDARD LVCMOS33 [get_ports o]
```

【完成情况】

是否完成以下步骤？（√完成 ×未做）

(1) [√] (2) [√] (3) [√]

【实验体会】

写出实验过程中遇到的问题，解决方法和自己的思考；并简述实验体会（如果有的话）。

遇到的问题：仿真实验结果输出非稳态（红线），思考：多线输入冲突，通过观察 RTL，发现问题出现在仿真文件对输出赋予了初值 0，导致接地线与加法器输出线冲突。解决方法：删除仿真文件对输出线的初始赋值。

【交实验报告】

每位同学单独完成本实验内容并填写实验报告。

交作业地点：<http://172.18.187.251/netdisk/default.aspx?vm=19org>

Vivado 实验/实验 1、组合逻辑设计

截止日期：2020 年 10 月 30 日 23:00（周五）

上传文件：学号_姓名_组合逻辑设计.doc

学号_姓名_组合逻辑设计.rar （步骤 3 的 Vivado 项目）