**1 实验目的和要求**

**1.1 实验目的**

（分点简要说明本次实验需要进行的工作和最终的目的）

Design an ALU with the following functions.

1. Design the modules of 4-bit 2-1 Mux, 4-bit 3-1 Mux, 4-bit Adder, 4-bit Subtracter, and 4-bitMultiplier. You do not need to consider the situation of overflow.
2. Design the top module for the ALU with hierarchy structure of the modules defined in step 1).
3. Implement the design in FPGA with the data input A[3:0], B[3:0], and the control input F[2:0] connected to the switches on Basys3 board, and the output R[3:0] to the LEDs onboard.
4. (optional) Try to solve the overflow problem in this design with your own method.
5. (optional) Try to extend the Function table with new functions.

**1.2 实验要求**

（说明本次实验的要求与任务）

Design an ALU with the following functions.

1)Design the modules of 4-bit 2-1 Mux, 4-bit 3-1 Mux, 4-bit Adder, 4-bit Subtracter, and 4-bitMultiplier. You do not need to consider the situation of overflow.

2)Design the top module for the ALU with hierarchy structure of the modules defined in step 1).

3)Implement the design in FPGA with the data input A[3:0], B[3:0], and the control input F[2:0] connected to the switches on Basys3 board, and the output R[3:0] to the LEDs onboard.

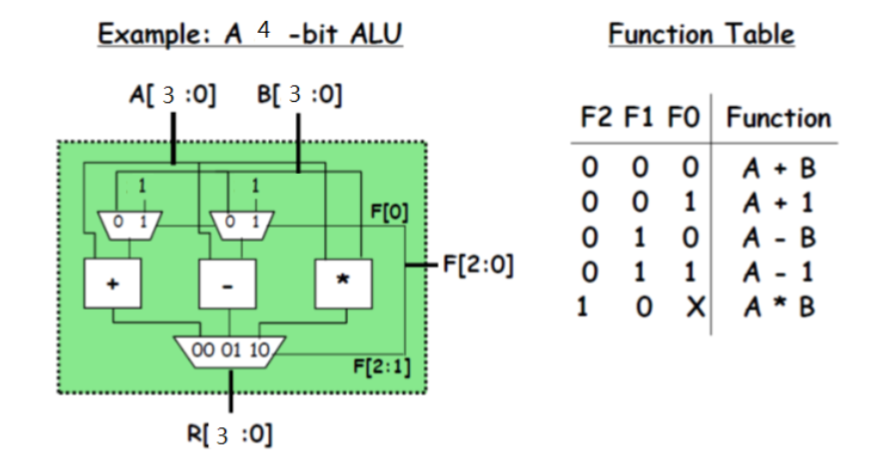
4) (optional) Try to solve the overflow problem in this design with your own method.

5) (optional) Try to extend the Function table with new functions.

**2 实验原理**

（简要说明本次实验的理论，包括但不限于物理、数学或是算法方面的理论，电路原理图、算法框图等示意图也可以在此处给出）

HDL语言和算法原理

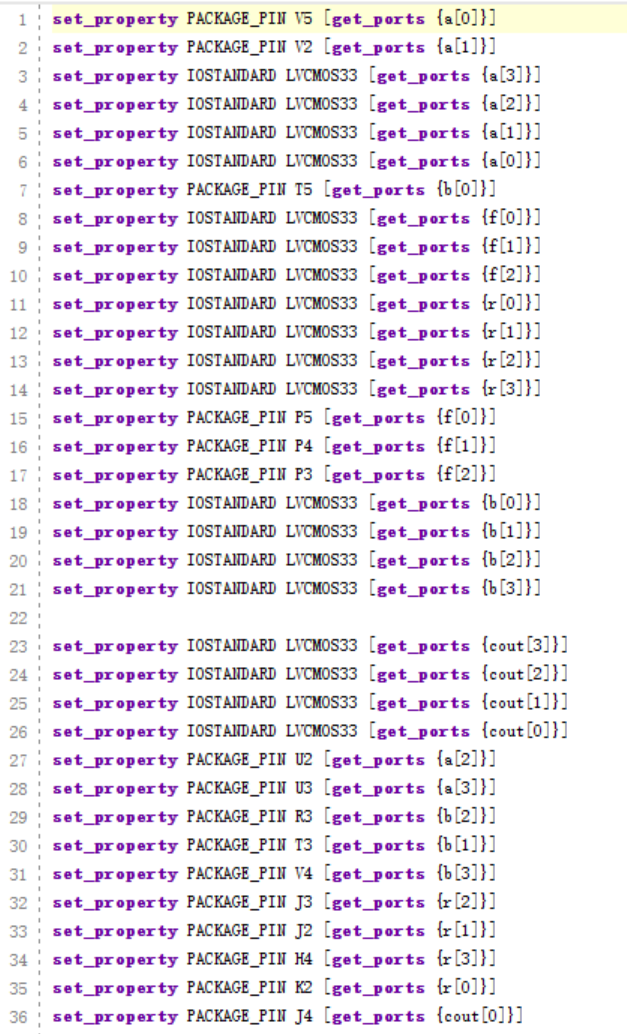
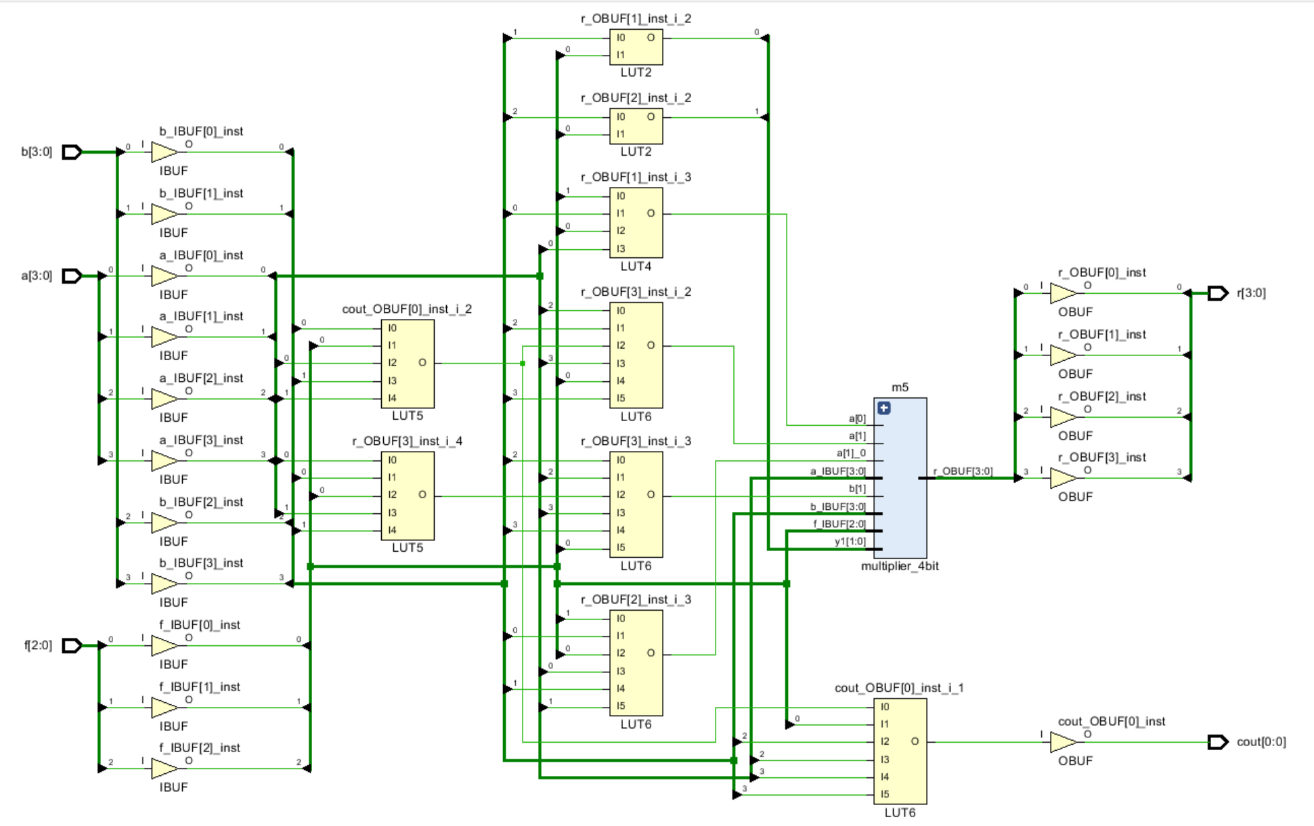


**3 实验内容**

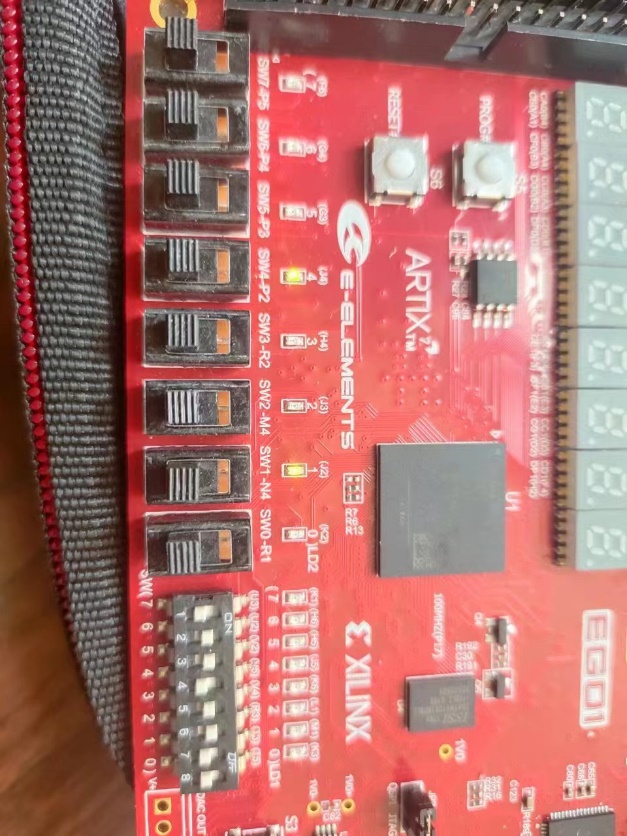
（分点阐述实验步骤）

1. 使用vivado软件，分别编写4-bit 2-1 Mux, 4-bit 3-1 Mux, 4-bit Adder, 4-bit Subtracter, and 4-bitMultiplier模块的HDL语言代码。
2. Design the top module for the ALU with hierarchy structure of the modules defined in step 1).
3. 进行综合，管脚约束，执行，生成比特流，并烧写到电路板上。
4. 使用电路板，测试ALU
5. 进行综合，管脚约束，执行，生成比特流，并烧写到电路板上。使用电路板，测试4选1选择器。

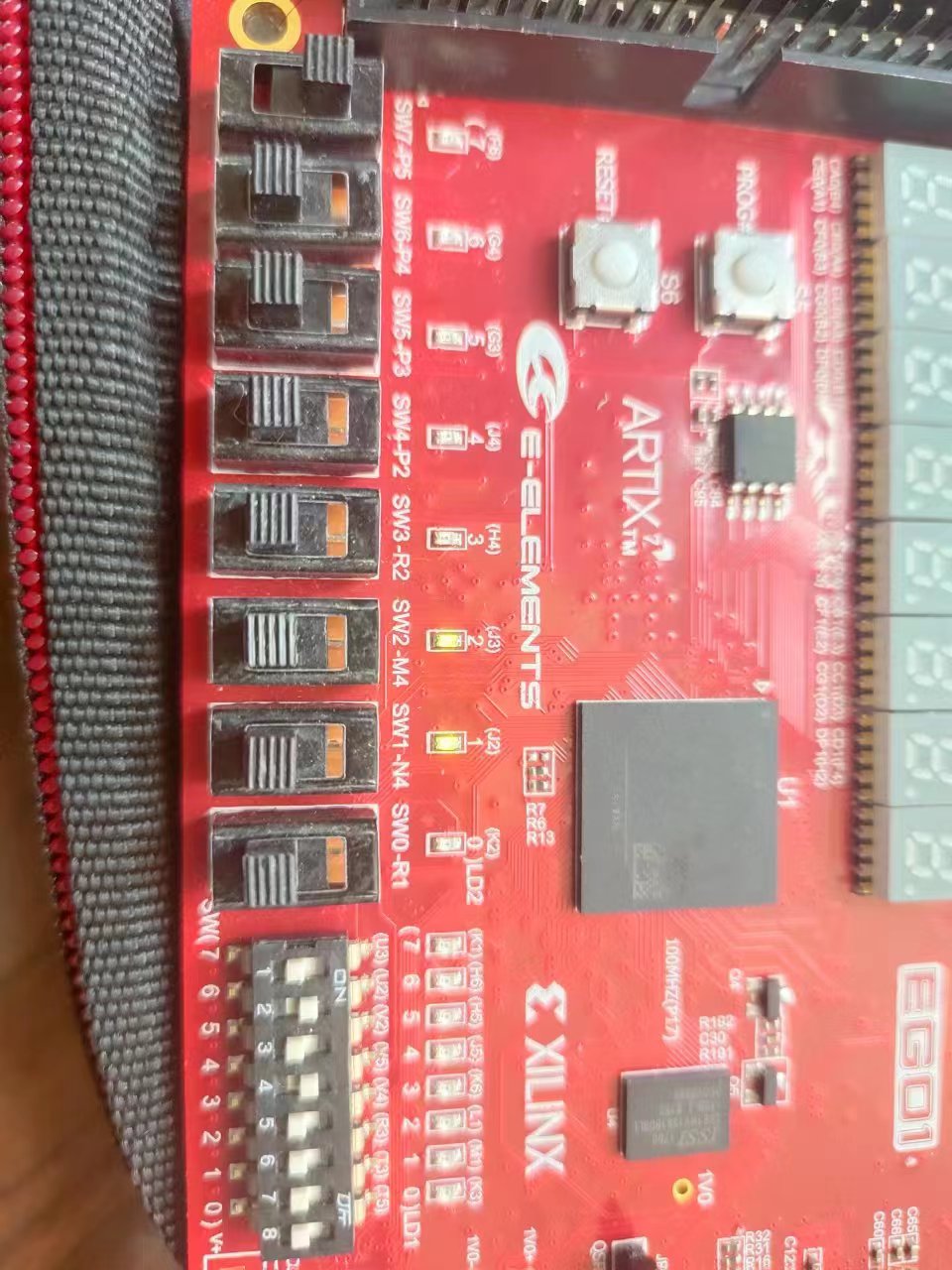
**4 实验结果和分析**



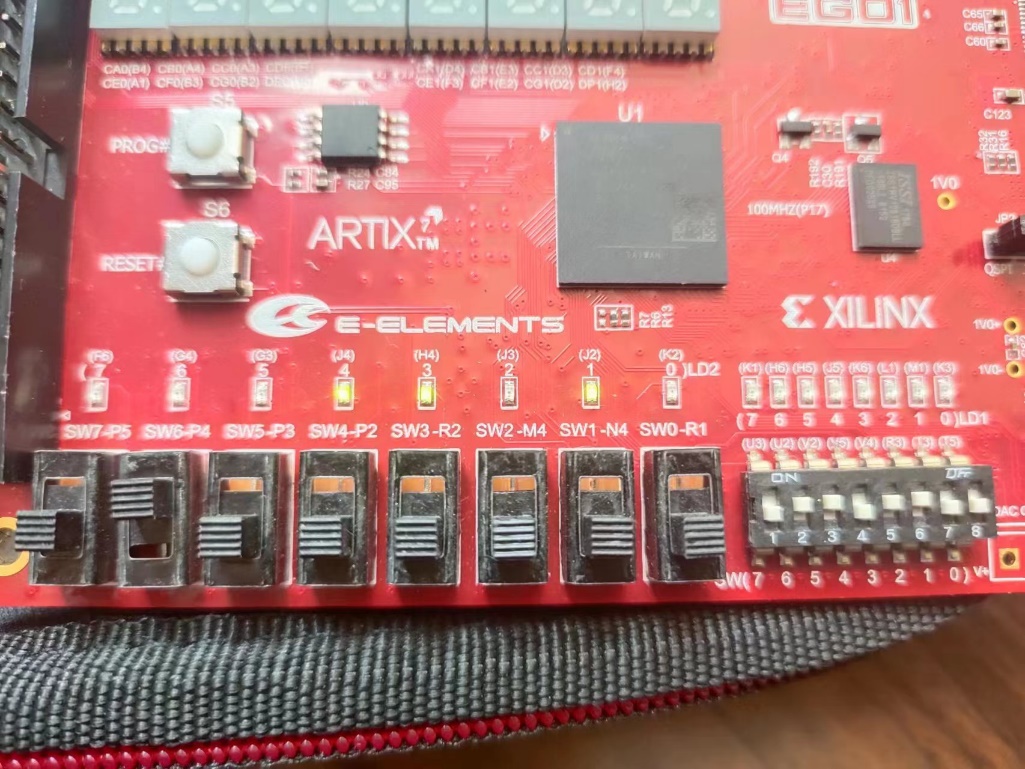
图表 1综合结果与管脚约束



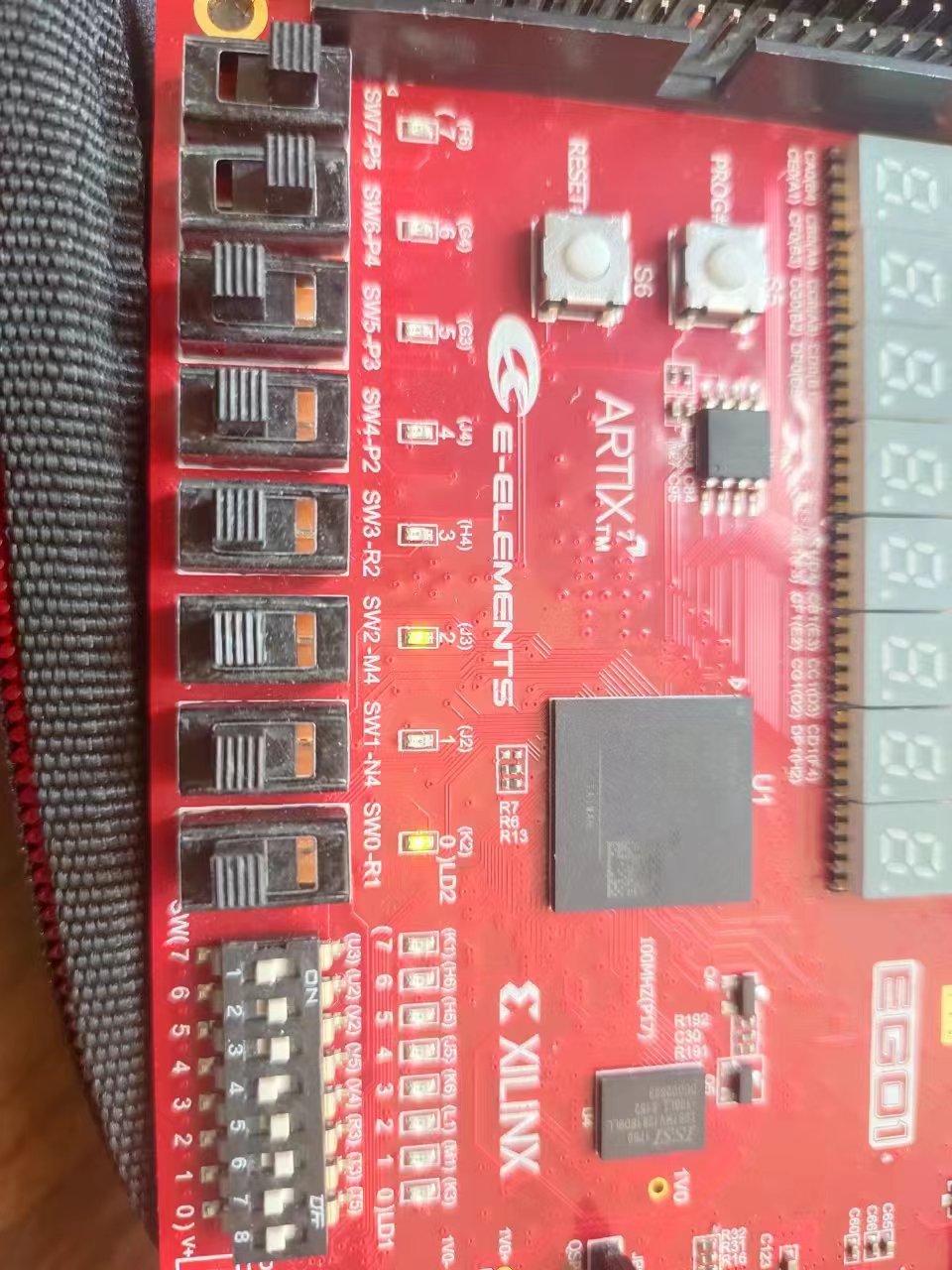
图表 2 2 A+B测试效果:0110+1100=10010



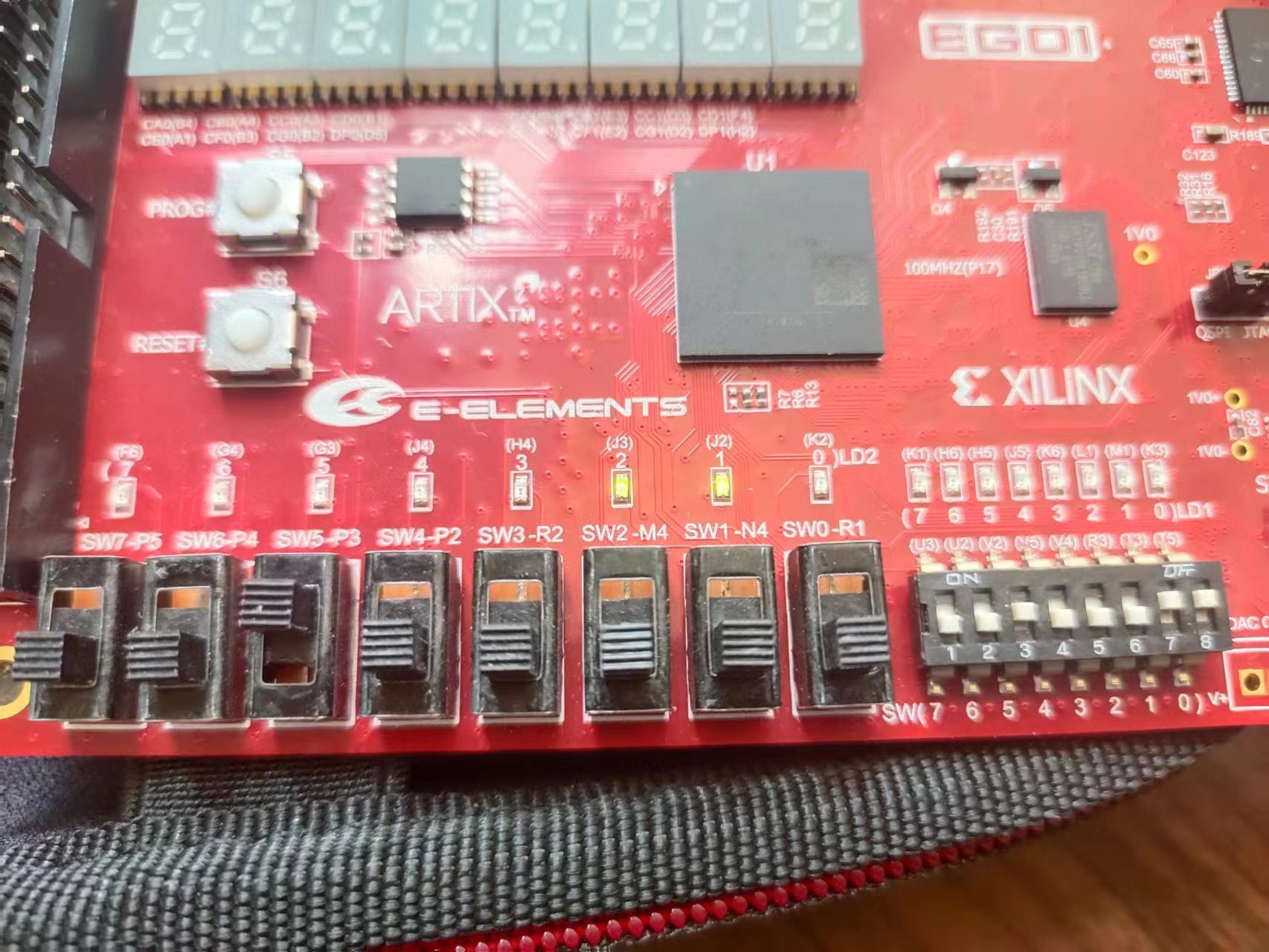
图表 3 2 A+1测试效果:0101+0001=0110



图表 4 2 A-B测试效果:0110-1100=11010



图表 3 4 A-1测试效果:0110-0001=0101



图表 3 4 A\*B测试效果:0010\*0011=0110

**5 实验结论**

（基于实验结果和分析，得出结论）

成功实现了所要求的ALU的设计，运行和测试。

**6 源代码与分析**

module mux21( //2选1选择器

input [2:0]f,

input [3:0] b,

output [3:0]y

);

assign y= f[0]? 4'b0001:b;

endmodule

module mux31( //3选1选择器

input [2:0]f,

input [3:0]a,b,c,

output reg [3:0]r

);

always@(\*) begin

if (f[2:1]==2'b00) r=a;

else if (f[2:1]==2'b01) r=b;

else if (f[2:1]==2'b10) r=c;

else r=4'b0000;

end

endmodule

module adder\_4bit( //四位加法器

input [3:0]a,b,

output [3:0]y,[0:0]cout //cout用作溢出进位

);

assign {cout,y}=a+b;

endmodule

module subtract\_4bit( //四位减法器

input [3:0]a, b,

output [3:0]y

);

assign y=a-b;

endmodule

module multiplier\_4bit( //四位乘法器

input [3:0]a, b,

output [3:0]y

);

assign y=a\*b;

endmodule

module ALUdesign( //顶层模块，ALU

input [2:0]f,

input[3:0]a,b,

output[3:0]r,[0:0]cout

);

wire [3:0]y1,y2,s1,s2,s3;

mux21 m1(f,b,y1); //模块化结构搭建

mux21 m2(f,b,y2);

adder\_4bit m3(a,y1,s1,cout);

subtract\_4bit m4(a,y2,s2);

multiplier\_4bit m5(a,b,s3);

mux31 m6(f,s1,s2,s3,r);

endmodule