

实验报告

2021 年 4 月 9 日

成绩: _____

| | | | | | |
|------|----------|------|-------------|-------------|----------|
| 姓名 | 汪凌峰 | 学号 | 19270824 | 班级 | 19052321 |
| 专业 | 计算机科学与技术 | | 课程名称 | 计算机组成原理课程设计 | |
| 任课老师 | 曾虹 | 指导老师 | 曾虹 | 机位号 | 31 |
| 实验序号 | 2 | 实验名称 | 超前进位加法器设计实验 | | |
| 实验时间 | 2021.4.9 | 实验地点 | 一教 225 | 实验设备号 | 31 |

一、实验程序源代码

```
`timescale 1ns / 1ps
module Adder4(
    input [3:0] X,
    input [3:0] Y,
    input Cin,
    output [3:0] F,
    output Cout
);

    wire c1, c2, c3;
    wire [3:0] G, P;
    assign G[0]=X[0]&Y[0];
    assign P[0]=X[0]|Y[0];
    assign F[0]=G[0]^P[0]^Cin;
    assign c1=G[0]|(P[0]&Cin);
    assign G[1]=X[1]&Y[1];
    assign P[1]=X[1]|Y[1];
    assign F[1]=G[1]^P[1]^c1;
    assign c2=G[1]|(P[1]&c1);
    assign G[2]=X[2]&Y[2];
    assign P[2]=X[2]|Y[2];
    assign F[2]=G[2]^P[2]^c2;
    assign c3=G[2]|(P[2]&c2);
    assign G[3]=X[3]&Y[3];
    assign P[3]=X[3]|Y[3];
    assign F[3]=G[3]^P[3]^c3;
    assign Cout=G[3]|(P[3]&c3);

endmodule
```

二、仿真测试代码

```
`timescale 1ns / 1ps

module Adder4Test;

    // Inputs
    reg [3:0] X;
    reg [3:0] Y;
    reg Cin;

    // Outputs
    wire [3:0] F;
    wire Cout;

    // Instantiate the Unit Under Test (UUT)
    Adder4 uut (
        .X(X),
        .Y(Y),
        .Cin(Cin),
        .F(F),
        .Cout(Cout)
    );

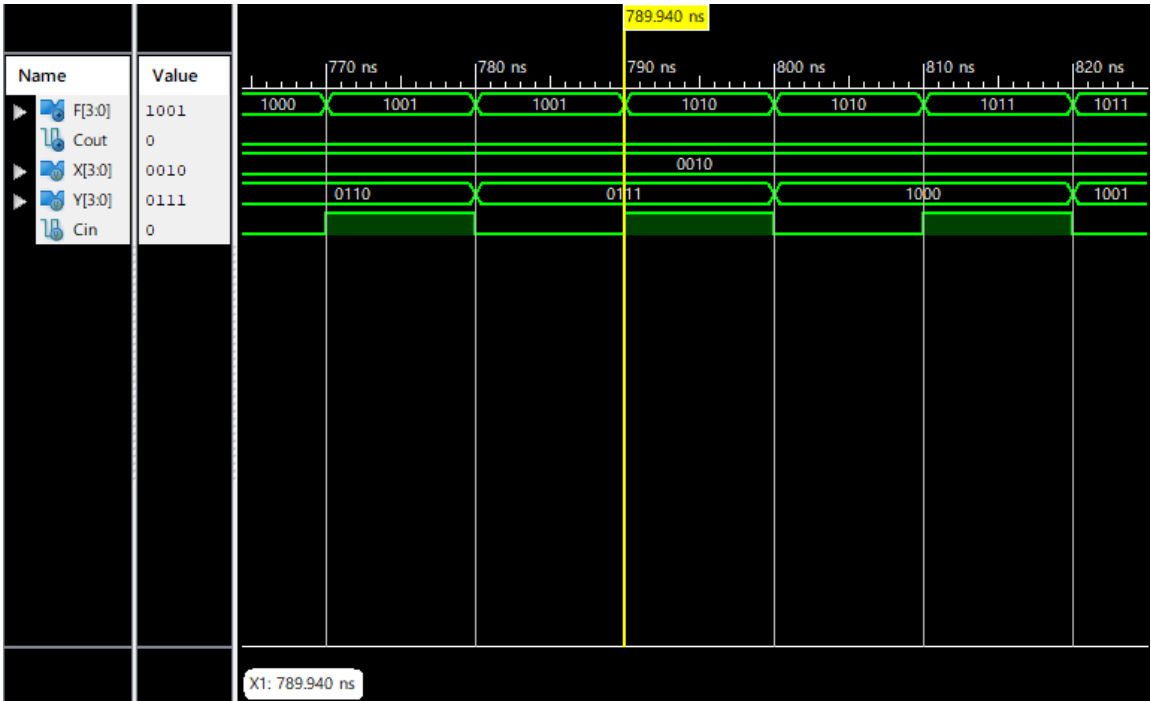
    initial begin
        // Initialize Inputs
        X = 0;
        Y = 0;
        Cin = 0;

        // Wait 100 ns for global reset to finish
        #10;

        // Add stimulus here
        repeat(99)
            begin
                {X, Y, Cin} = {X, Y, Cin} + 1;
                #10;
            end
        end

endmodule
```

三、仿真波形



四、思考与探索

这个实验源代码实现的方法是利用实验指导书中的公式进行逻辑运算而不进行加法计算来完成，其实还可以把 Cout 与 F 绑定赋值 $X+Y+Cin$ 的值，溢出的内容给 Cout 而本位和给 F，更简单且只需一行代码就可以实现相同功能：

```
assign {Cout, F} = X + Y + Cin;
```

或者还有另一个方法，使用芯片级联的方式把实验一做的 1 位全加器连起来，虽然速度可能不如这个，但是也可以实现相同效果。