中山大学数据科学与计算机学院本科生实验报告

（2019 学年秋季学期）

课程名称：**计算机组成原理实验**　　任课教师：郭雪梅　　助教：汪庭葳、刘洋旗

年级&班级	2018 级计科一班	专业(方向)	计算机类
学号	18340013	姓名	陈琮昊
电话	15734867907	Email	1062080021@qq.com
开始日期	2019.10.23	完成日期	2019.11.5

一、实验题目： VIVADO 实验一

二、实验目的：用 VIVADO 实现加减法器

三、实验内容

1. 实验步骤：

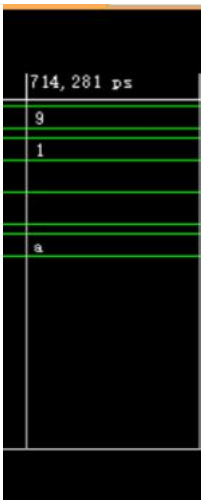
（i）实现 4 位超前进位加法器设计并仿真实现

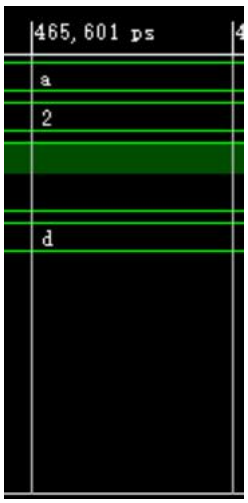
（ii）设计 8 位的加减法器

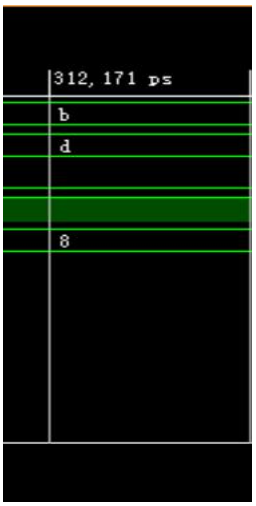
2. 实验原理：加法器的原理

四、实验结果：

（I）4 位超前进位加法器仿真结果：



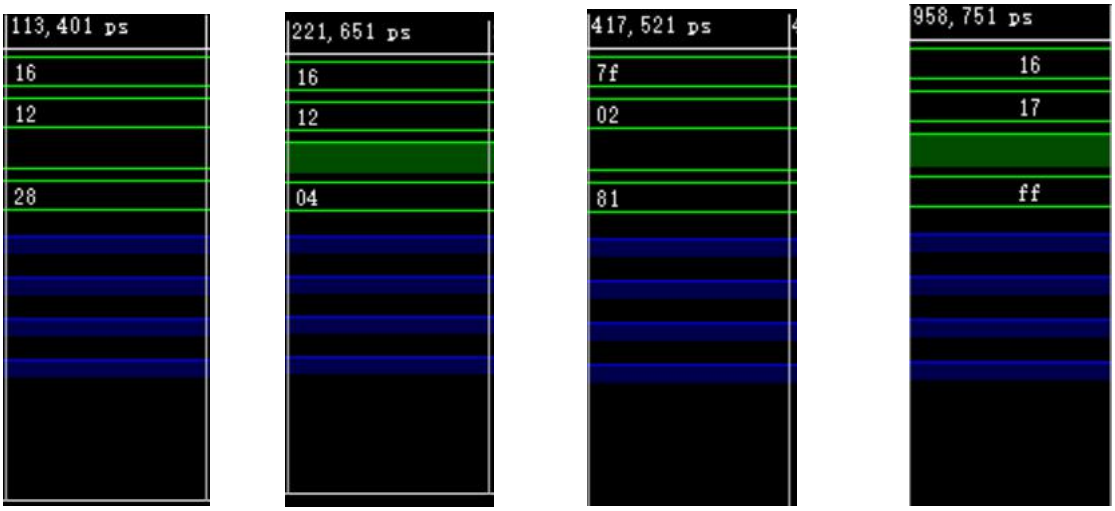




如图：9+1+0=0a　　a+2+1=0d　　b+d+0=18（均为 16 进制，第三行为进位）

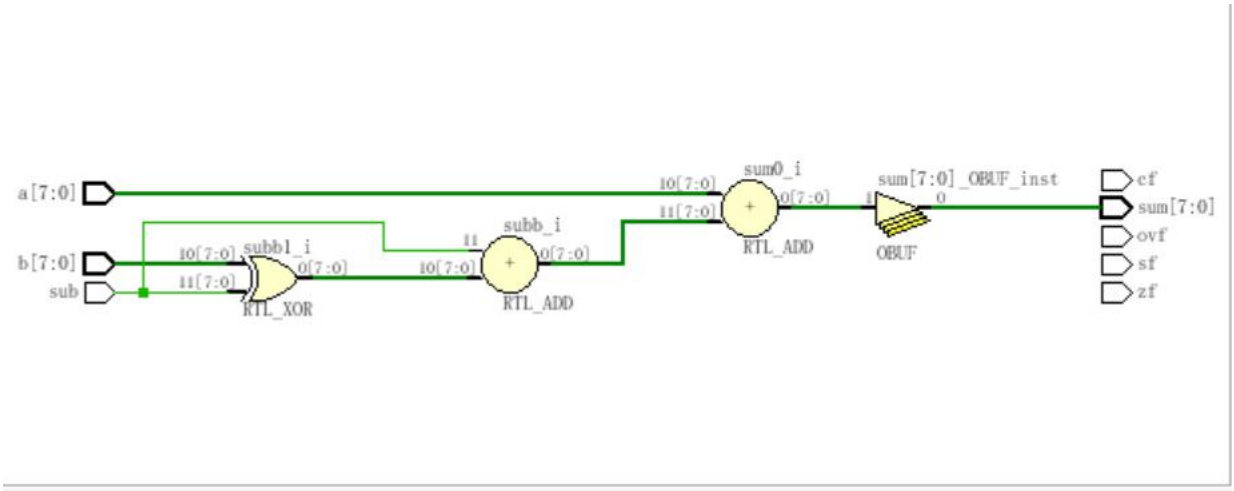
（II）8 位的加减法器结果：

1.Run Simulation:



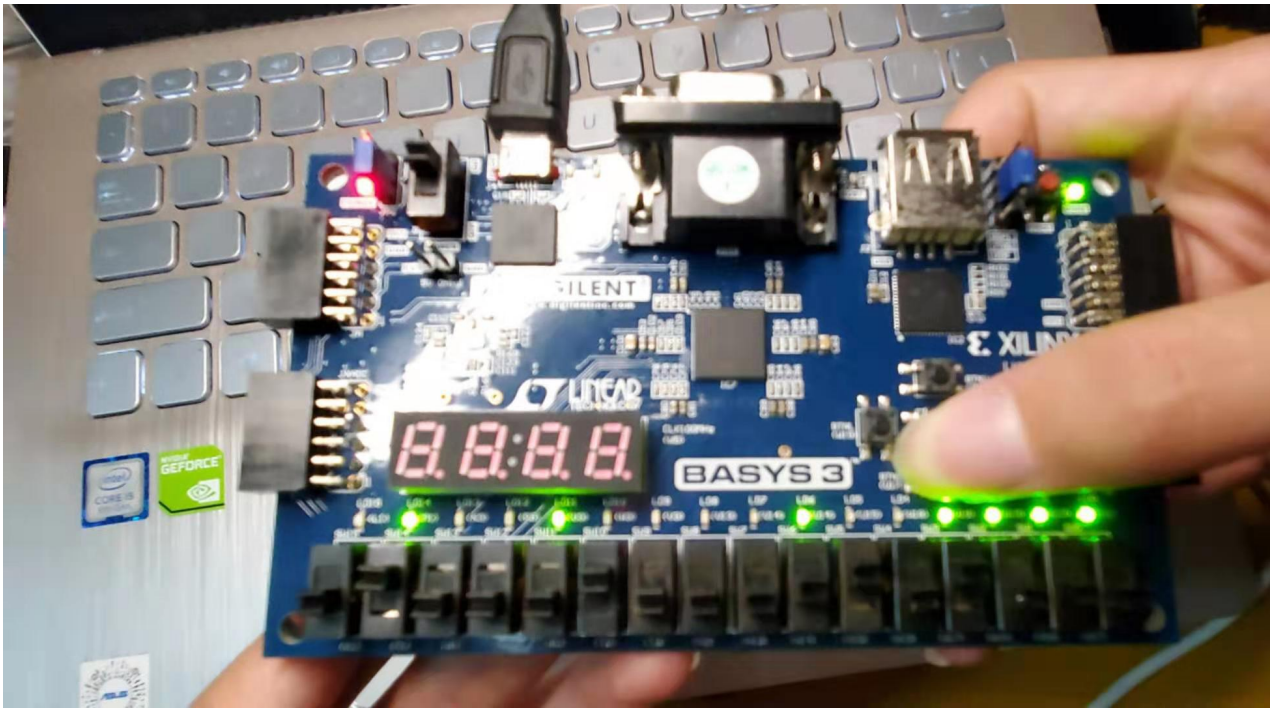
如图：16+12=28，16-12=04，7f+02=81，16-17=ff（16 进制，第三行 0 加 1 减）。

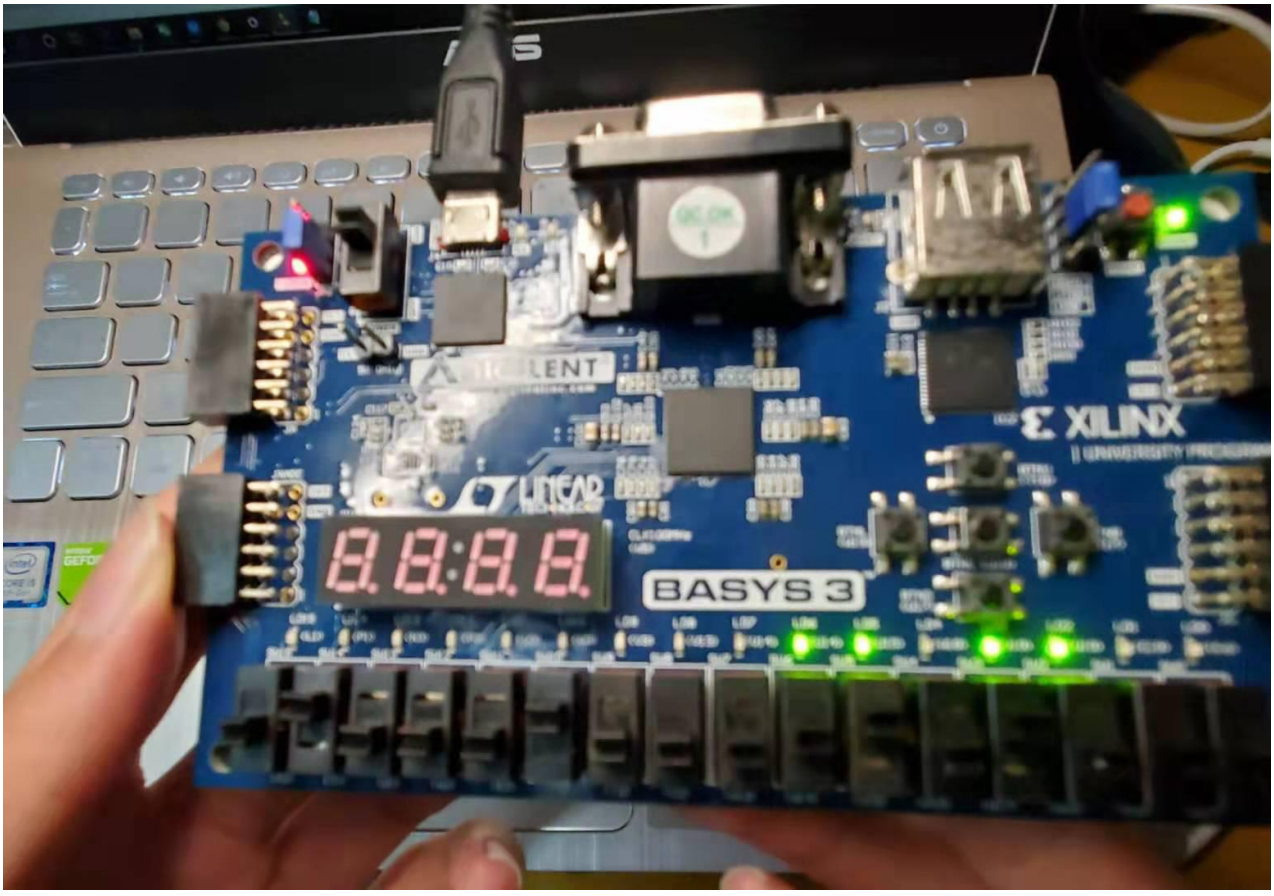
2. RTL Analysis:



3. Run Implementation&&Generate Bitstream

4. **烧 FPGA 板，结果如下：**





如图所示：最左面 8 位为第二个数的从低到高位，右面 8 位为第一个数的从低到高位。第一张为减法（20-34=-14=-128+64+32+16+2），第二张为加法（20+34=54=32+16+4+2），核对可知结果正确，实验成功。

五、实验感想：本次实验主要是用 VIVADO 设计加减法器，其中超前进位的实验只需要仿真，比较简单；重点在于第二个实验：8 位的加减法器，该实验是按照完整的流程来的。第一次使用 VIVADO 遇到了许多状况，比如需要 valid license，在网上找到 license 后可以正常使用 VIVADO 了；解决该问题后遇到的又一个瓶颈是烧板，该过程有时会不稳定，需要耐心；还有个小问题就是连接引脚，这倒不是很难解决的问题，但真的需要特别细心，才能保证不出错并且顺利连接板，从而显示正确的结果。不过最后看到板上显示正确结果的时候还是很开心的。

附录（流程图，注释过的代码）：

（1）4 位超前进位加法器代码：

```
`timescale 1ns / 1ps
//////////////////////////////////////////////////////////////////////////////////
//////////////////////////////////////////////////////////////////////////////////

module chaoadder(A,B,C0,C4,F);
   input[3:0] A,B;
   input          C0;
   output[3:0] F;
   output          C4;
   wire[3:0]      A,B;
   reg [3:0]      F,G,P;
   reg              C4;
   reg    C1,C2,C3;
   always@(*)
   begin
         G[0]=A[0]&B[0];
         G[1]=A[1]&B[1];
         G[2]=A[2]&B[2];
         G[3]=A[3]&B[3];

         P[0]=A[0]|B[0];
         P[1]=A[1]|B[1];
         P[2]=A[2]|B[2];
         P[3]=A[3]|B[3];

         C1=G[0]|(P[0]&C0);
         C2=G[1]|(P[1]&C1);
         C3=G[2]|(P[2]&C2);
         C4=G[3]|(P[3]&C3);

         F[0]=A[0]^B[0]^C0;
         F[1]=A[1]^B[1]^C1;
         F[2]=A[2]^B[2]^C2;
```

```
        F[3]=A[3]^B[3]^C3;

end
endmodule
```

仿真实现：

```
`timescale 1ns / 1ps
module simadder(
    );
    // Inputs
        reg [3:0] A;
        reg [3:0] B;
        reg C0;
    // Outputs
        wire C4;
        wire [3:0] F;
    // Instantiate the Unit Under Test (UUT)
       chaoadder uut
                    (.A(A),
                   .B(B),
                   .C0(C0),
                   .C4(C4),
                   .F(F)
                        );
     initial begin
                // Initialize Inputs
                        C0 = 0;
                // Wait 100 ns for global reset to finish
                #100;
    // Add stimulus here
        begin A = 'B1100;B='B1011;C0 = 0;end
```

```
        #100;
        begin A = 'B1011;B='B0010;C0 = 1;end
        #100;
        begin A = 'B1011;B='B1101;C0 = 0;end
        #100;
        begin A = 'B1010;B='B0010;C0 = 1;end
        #100;
        A = 'B0111;B='B1000;C0= 0;
        #100;
        begin A = 'B0011;B='B0100;C0 = 1;end
        #100;
        begin A = 'B1001;B='B0001;C0 = 0;end
        #100;
    end
endmodule
```

(2)8 位的加减法器代码：

```
module addsub
#(parameter WIDTH=8)                    //指定数据宽度参数，缺省值是8
(
    input [(WIDTH-1):0] a,          // 缺省位数由参数WIDTH决定
    input [(WIDTH-1):0] b,
    input   sub,                    // =1为减法
    output [(WIDTH-1):0] sum,
    output cf,                      // 进位标志
    output ovf,                     // 溢出标志
    output sf,                      // 符号标志
    output zf                       // 为0标志
    );

    wire [(WIDTH-1):0] subb,subb1;
        wire cf2;       // 进位
        assign subb1 = b ^ {WIDTH{sub}};   // 对于减法是取反
        assign subb   =   subb1 + sub;     // 对于减法是加1，sub=1（减法）sub=0（加法）

        assign {cf2,sum} = a + subb;
```

```
    assign {cf2,sum} = a + subb;

    assign sf = sum[WIDTH-1];

    assign zf = (sum == 0) ? 1 : 0 ;

    assign cf = cf2 ^ sub;

    assign ovf = (a[WIDTH-1] ^ sum[WIDTH-1]) & (subb[WIDTH-1] ^ sum[WIDTH-1]);

endmodule
```

仿真实现：

```
`timescale 1ns / 1ps

module addsub_sim(      );
    // input
    reg [7:0] a = 8'h16;
    reg [7:0] b = 8'h12;
    reg sub = 0;

    //output
    wire [7:0] sum;
    wire cf;
    wire ovf;
    wire sf;
    wire zf;

    // initial
    addsub U (a,b,sub,sum,cf,ovf,sf,zf);
    initial begin
    #200 sub = 1;
    #200 begin a = 8'h7f; b = 8'h2; sub = 0; end
    #200 begin a = 8'hff; b = 8'h2; sub = 0; end
    #200 begin a = 8'h16; b = 8'h17; sub = 1; end
    #200 begin a = 8'hfe; b = 8'hff; sub = 1; end

    end

endmodule
```

约束文件：

```
set_property IOSTANDARD LVCMOS33 [get_ports {a[7]}]
set_property IOSTANDARD LVCMOS33 [get_ports {a[6]}]
set_property IOSTANDARD LVCMOS33 [get_ports {a[5]}]
set_property IOSTANDARD LVCMOS33 [get_ports {a[4]}]
set_property IOSTANDARD LVCMOS33 [get_ports {a[3]}]
set_property IOSTANDARD LVCMOS33 [get_ports {a[2]}]
set_property IOSTANDARD LVCMOS33 [get_ports {a[1]}]
set_property IOSTANDARD LVCMOS33 [get_ports {a[0]}]
set_property IOSTANDARD LVCMOS33 [get_ports {b[7]}]
set_property IOSTANDARD LVCMOS33 [get_ports {b[6]}]
set_property IOSTANDARD LVCMOS33 [get_ports {b[5]}]
set_property IOSTANDARD LVCMOS33 [get_ports {b[4]}]
set_property IOSTANDARD LVCMOS33 [get_ports {b[3]}]
set_property IOSTANDARD LVCMOS33 [get_ports {b[2]}]
set_property IOSTANDARD LVCMOS33 [get_ports {b[1]}]
set_property IOSTANDARD LVCMOS33 [get_ports {b[0]}]
set_property IOSTANDARD LVCMOS33 [get_ports {sum[7]}]
set_property IOSTANDARD LVCMOS33 [get_ports {sum[6]}]
set_property IOSTANDARD LVCMOS33 [get_ports {sum[5]}]
set_property IOSTANDARD LVCMOS33 [get_ports {sum[4]}]
set_property IOSTANDARD LVCMOS33 [get_ports {sum[3]}]
set_property IOSTANDARD LVCMOS33 [get_ports {sum[2]}]
set_property IOSTANDARD LVCMOS33 [get_ports {sum[1]}]
set_property IOSTANDARD LVCMOS33 [get_ports {sum[0]}]
set_property IOSTANDARD LVCMOS33 [get_ports cf]
set_property IOSTANDARD LVCMOS33 [get_ports ovf]
set_property IOSTANDARD LVCMOS33 [get_ports sf]
set_property IOSTANDARD LVCMOS33 [get_ports sub]
set_property IOSTANDARD LVCMOS33 [get_ports zf]
set_property PACKAGE_PIN V17 [get_ports {a[7]}]
set_property PACKAGE_PIN V16 [get_ports {a[6]}]
set_property PACKAGE_PIN W16 [get_ports {a[5]}]
set_property PACKAGE_PIN W17 [get_ports {a[4]}]
set_property PACKAGE_PIN W15 [get_ports {a[3]}]
set_property PACKAGE_PIN V15 [get_ports {a[2]}]
set_property PACKAGE_PIN W14 [get_ports {a[1]}]
set_property PACKAGE_PIN W13 [get_ports {a[0]}]
```