

浙江大学

电路与电子技术 实验报告

实验名称: 数字电路设计基础
实验人员: 潘谷雨、杨骐恺
报告撰写: 潘谷雨
学号: 3220102382

实验日期: 2023 年 10 月 9 日

地点: 东三 406

【实验目的】

1. 了解数字电路基础知识
2. 掌握简单数字逻辑电路的设计
3. 学习 FPGA 开发工具 Modelsim 与 Quartus 的使用

【实验内容】（测试方案，含仿真与硬件测试两种类型）

用 Modelsim、Quartus 软件和 DEMO 板设计并实现如下逻辑功能

一、输入是三位二进制数 A, B, C, 要求当输入是 2 或 3 的倍数时输出等于逻辑 1, 其它情况, 输出等于 0

1. 用 Modelsim 软件编程, test.vhd 代码如下:

```
library IEEE;
use IEEE.std_logic_1164.all;
use IEEE.numeric_std.all;

entity test is
    port (
        A, B, C: in STD_LOGIC;
        S: out STD_LOGIC;
        Rst: in STD_LOGIC
    );
end test ;

architecture BEHAV of test is
begin
    process(A, B, C, Rst)
        variable x: integer;
    begin
        IF (Rst = '1') then
            S <= '0';
        else
            x:=0;
            if (A='1') then x := x+4;end if;
            if (B='1') then x:=x+2;end if;
            if (C='1') then x:=x+1;end if;
            if ((x mod 2=0)or(x mod 3=0)) then S<='1';
            else S<='0';end if;
        end if;
    end process;
end;
```

```
        end IF;  
    end process;
```

```
end BEHAV;
```

2. 进行软件仿真，test_tb.vhd 代码如下：

```
library IEEE;  
use IEEE.std_logic_1164.all;
```

```
entity test_tb is  
end test_tb ;
```

```
architecture BEHAV of test_tb is
```

```
    component test is
```

```
        port (
```

```
            A, B , C: in STD_LOGIC;
```

```
            S: out STD_LOGIC;
```

```
            Rst: in STD_LOGIC);
```

```
    end component;
```

```
    signal A, B,C, S, Rst: STD_LOGIC;
```

```
begin
```

```
U1:test port map(A=>A,B=>B,C=>C,S=>S,Rst=>Rst);
```

```
    p1: process
```

```
    begin
```

```
        Rst <= '1';
```

```
        wait for 10 ns;
```

```
        Rst <= '0';
```

```
        A <= '0'; B <= '0'; C<='0';wait for 10 ns;
```

```
        A <= '0'; B <= '0'; C<='1';wait for 10 ns;
```

```
        A <= '0'; B <= '1'; C<='0';wait for 10 ns;
```

```
        A <= '0'; B <= '1'; C<='1';wait for 10 ns;
```

```
        A <= '1'; B <= '0'; C<='0';wait for 10 ns;
```

```
        A <= '1'; B <= '0'; C<='1';wait for 10 ns;
```

```
        A <= '1'; B <= '1'; C<='0';wait for 10 ns;
```

```
        A <= '1'; B <= '1'; C<='1';wait for 10 ns;
```

```
    end process;
```

```
end architecture BEHAV;
```

3. 连接 DEMO 板，进行验证。

二、设计并实现一位二进制全加器

1. 用 Modelsim 软件编程，test.vhd 代码如下：

```
library IEEE;
use IEEE.std_logic_1164.all;
use IEEE.numeric_std.all;

entity test is
    port (
        A, B , Cin: in STD_LOGIC;
        S,Cout: out STD_LOGIC;
        Rst: in STD_LOGIC
    );
end test ;
```

```
architecture BEHAV of test is
begin
    process(A, B, Cin,Rst)
        variable x,y,z:STD_LOGIC;
    begin
        IF (Rst = '1') then
            S <= '0';
        else
            x:=A xor B;
            y:=X and Cin;
            z:=A and B;
            S<=x xor Cin;
            Cout<=y or z;
        end IF;
    end process;
end BEHAV;
```

2. 进行软件仿真，test_tb.vhd 代码如下：

```
library IEEE;
use IEEE.std_logic_1164.all;

entity test_tb is
end test_tb ;

architecture BEHAV of test_tb is
    component test is
        port (
            A, B , Cin: in STD_LOGIC;
```

```

        S,Cout: out STD_LOGIC;
        Rst: in STD_LOGIC
    );
end component;

signal A, B,Cin,Cout, S, Rst: STD_LOGIC;

begin

U1:test port map(A=>A,B=>B,Cin=>Cin,Cout=>Cout,S=>S,Rst=>Rst);

    p1: process
    begin
        Rst <= '1';
        wait for 10 ns;
        Rst <= '0';
        A <= '0'; B <= '0'; Cin<='0';wait for 10 ns;
        A <= '0'; B <= '0'; Cin<='1';wait for 10 ns;
        A <= '0'; B <= '1'; Cin<='0';wait for 10 ns;
        A <= '0'; B <= '1'; Cin<='1';wait for 10 ns;
        A <= '1'; B <= '0'; Cin<='0';wait for 10 ns;
        A <= '1'; B <= '0'; Cin<='1';wait for 10 ns;
        A <= '1'; B <= '1'; Cin<='0';wait for 10 ns;
        A <= '1'; B <= '1'; Cin<='1';wait for 10 ns;
    end process;

end architecture BEHAV;

```

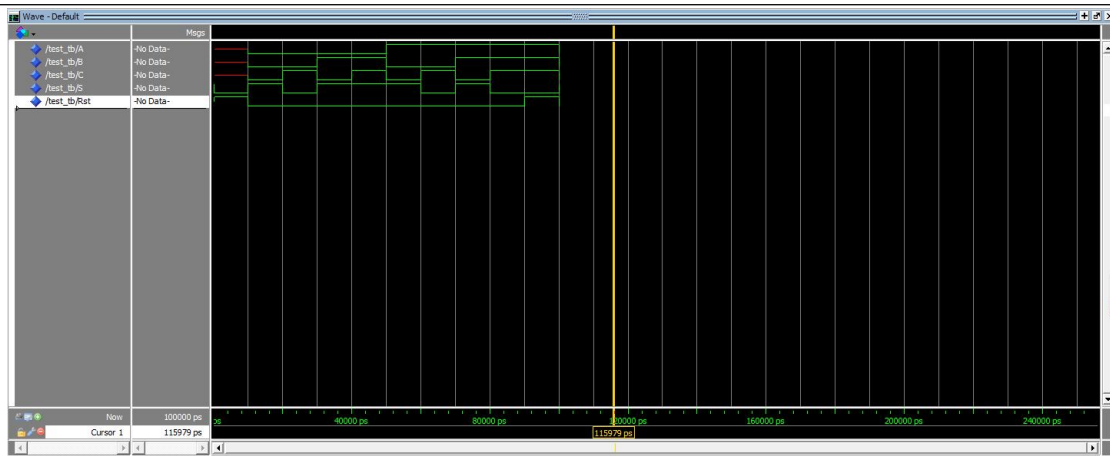
3. 连接 DEMO 板，进行验证。

【测试过程与结果】（注明测试设备、原始数据）

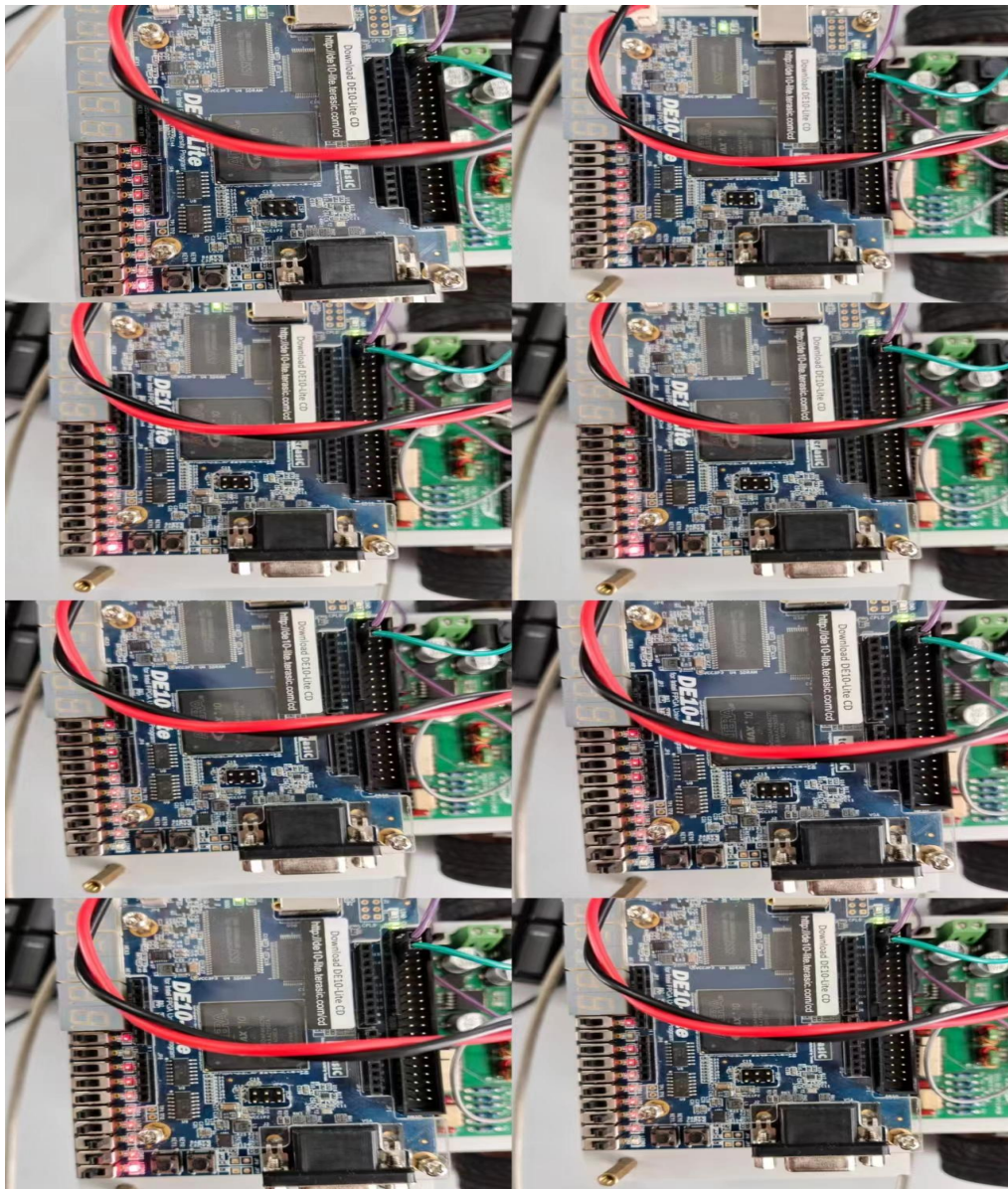
用 Modelsim、Quartus 软件和 DEMO 板设计并实现如下逻辑功能（测试设备：DE10-Lite 开发板，型号 M50DCF484C7G）：

一、输入是三位二进制数 A，B，C，要求当输入是 2 或 3 的倍数时输出等于逻辑 1，其它情况，输出等于 0

Modelsim 输出仿真波形如下：

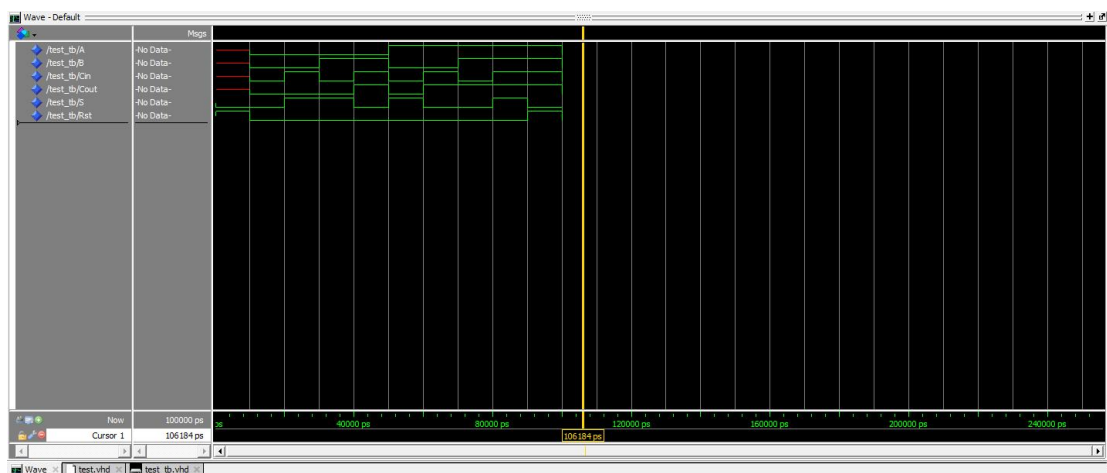


通过 Quartus 连入开发板，输入 A 对应管脚 C10，输入 B 对应管脚 C11，输入 C 对应管脚 D12，输出 S 对应 LED 灯 A8，得到如下结果：

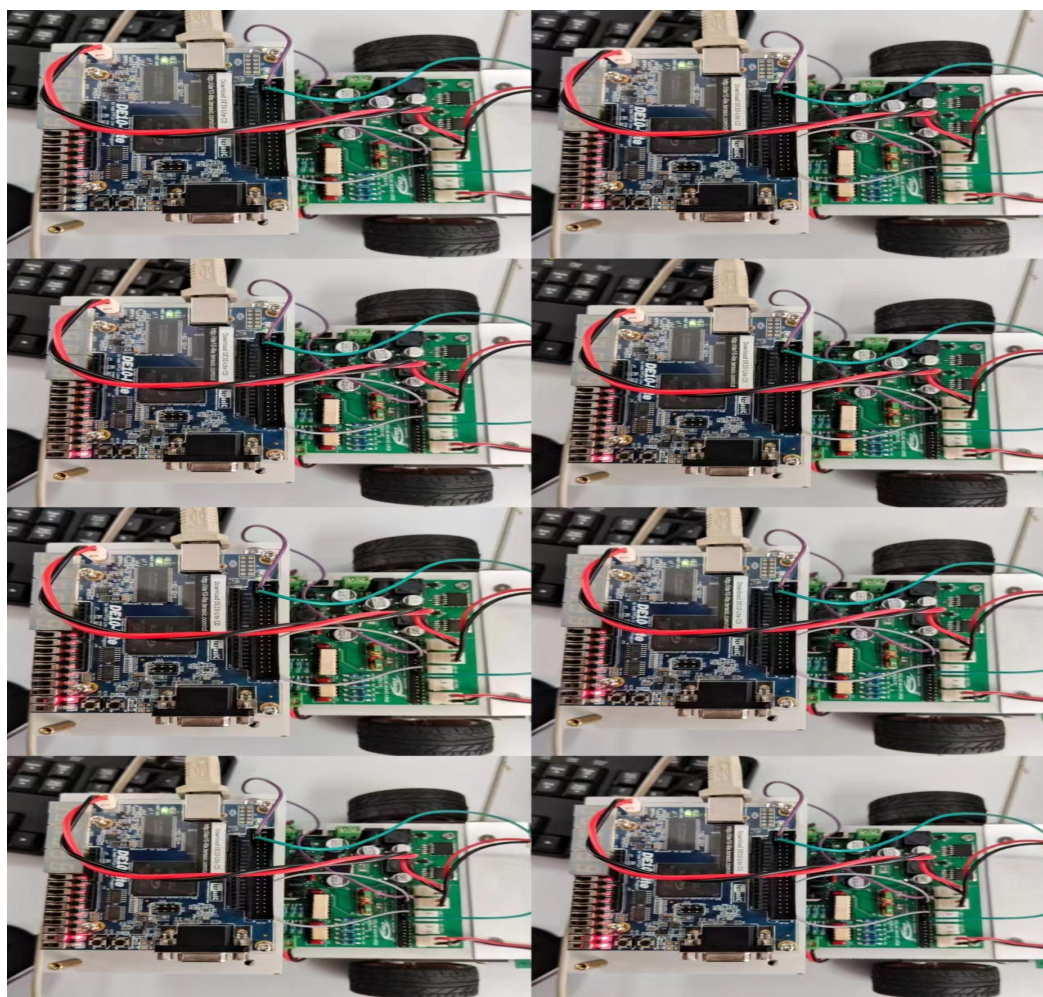


二、设计并实现一位二进制全加器

Modelsim 输出仿真波形如下:



通过 Quartus 连入开发板，输入 A 对应管脚 D12，输入 B 对应管脚 C11，输入 Cin 对应管脚 C10，输出 S 对应 LED 灯 A8，输出 Cout 对应 LED 灯 A9，得到如下结果：



【结果分析】

一、输入是三位二进制数 A, B, C, 要求当输入是 2 或 3 的倍数时输出等于逻辑 1, 其它情况, 输出等于 0

遍历所有情况, 通过仿真波形可以看出, 输入 A、B、C 与输出 S 满足如下关系:

A	B	C	ABC	S
0	0	0	0	1
0	0	1	1	0
0	1	0	2	1
0	1	1	3	1
1	0	0	4	1
1	0	1	5	0
1	1	0	6	1
1	1	1	7	0

运用开发板开关输入 A、B、C, 遍历所有情况, 结果同上, 满足设计要求。

二、采用逐点测量法测量二极管的 V_A 特性

遍历所有情况, 通过仿真波形可以看出, 输入 A、B、Cin 与输出 S 满足如下关系:

A	B	Cin	和	Cout	S
0	0	0	0	0	0
0	0	1	1	0	1
0	1	0	1	0	1
0	1	1	2	1	0
1	0	0	1	0	1
1	0	1	2	1	0
1	1	0	2	1	0
1	1	1	3	1	1

运用开发板开关输入 A、B、Cin, 遍历所有情况, 结果同上, 满足设计要求。

【探究性实验内容】

用 Modelsim 设计并仿真实现如下逻辑功能:

某客厅四周有 4 个房间, 每个房间门口有一个开关, 客厅中间有一盏灯 A。试设计一个逻辑电路, 要求每个开关都能控制灯 A 的亮灭, 并在 DE10 开发板上模拟此逻辑电路的功能。

1. 用 Modelsim 软件编程，test.vhd 代码如下：

```
library IEEE;
use IEEE.std_logic_1164.all;
use IEEE.numeric_std.all;

entity test is
    port (
        A, B, C, D: in STD_LOGIC;
        S: out STD_LOGIC;
        Rst: in STD_LOGIC
    );
end test ;

architecture BEHAV of test is
    signal ss,x,y,z,k:STD_LOGIC:='0';
begin
    process(A, B, C,D,Rst)

    begin
        IF (Rst = '1') then
            S <= '0';
        else
            if (x=not A) then ss<=not ss;end if;
            if (y=not B) then ss<=not ss;end if;
            if (z=not C) then ss<=not ss;end if;
            if (k=not D) then ss<=not ss;end if;
            x<=A;y<=B;z<=C;k<=D;
            S<=ss;
        end IF;
    end process;

end BEHAV;
```

2. 进行软件仿真，test_tb.vhd 代码如下：

```
library IEEE;
use IEEE.std_logic_1164.all;

entity test_tb is
end test_tb ;

architecture BEHAV of test_tb is
    component test is
```

```

port (
    A, B , C,D: in STD_LOGIC;
    S: out STD_LOGIC;
    Rst: in STD_LOGIC
);
end component;

signal A, B,C,D, S, Rst: STD_LOGIC;

```

```
begin
```

```
U1:test port map(A=>A,B=>B,C=>C,D=>D,S=>S,Rst=>Rst);
```

```
p1: process
```

```
begin
```

```
    Rst <= '1';
```

```
    wait for 10 ns;
```

```
    Rst <= '0';
```

```
    A <= '0'; B <= '0'; C<='0';D<='0';wait for 10 ns;
```

```
    A <= '0'; B <= '0'; C<='1';D<='0';wait for 10 ns;
```

```
    A <= '0'; B <= '1'; C<='1';D<='0';wait for 10 ns;
```

```
    A <= '1'; B <= '1'; C<='1';D<='0';wait for 10 ns;
```

```
    A <= '1'; B <= '1'; C<='0';D<='0';wait for 10 ns;
```

```
    A <= '1'; B <= '1'; C<='0';D<='0';wait for 10 ns;
```

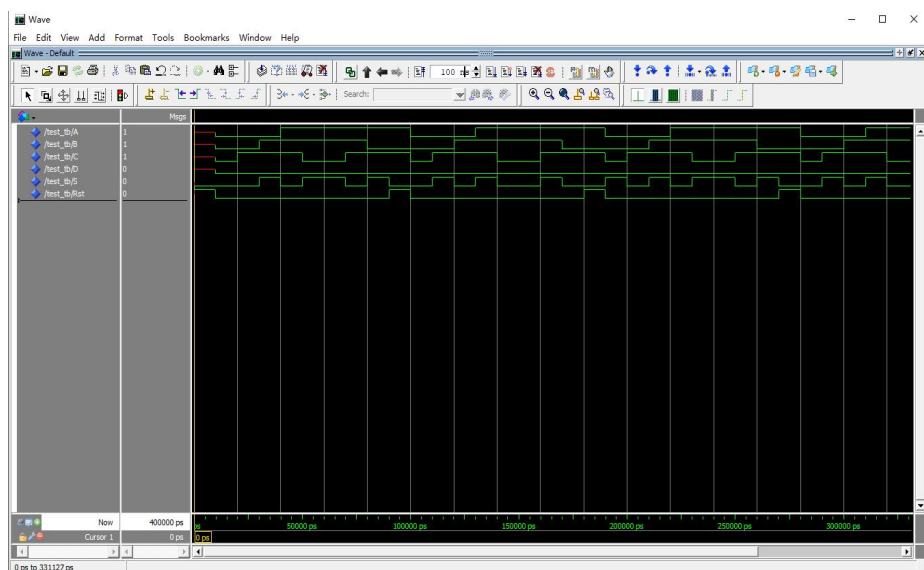
```
    A <= '1'; B <= '1'; C<='1';D<='0';wait for 10 ns;
```

```
    A <= '1'; B <= '0'; C<='1';D<='0';wait for 10 ns;
```

```
end process;
```

```
end architecture BEHAV;
```

Modelsim 仿真曲线如下：



结果分析：

由于第一个周期有初始化因素干扰，导致该周期中仅第二组输入没有引起 S 的预期变化，因此以第二个周期为例。又因为在代码中 A、B、C、D 地位完全相同，为了简化验证过程，将输入 D 初始置 0 并保持不变，观察 A、B、C 对输出 S 的影响。

状态	A	B	C	D	S
1	0	0	0	0	0
2	0	0	1	0	1
3	0	1	1	0	0
4	1	1	1	0	1
5	1	1	0	0	0
6	1	1	0	0	0
7	1	1	1	0	1
8	1	0	1	0	0

输入的每次变化代表开关闭合或打开，设 ‘1’ 为打开，‘0’ 为关闭。由测试结果看出：

- ①依次打开 C、B、A，每次试验均能够引起 S 的变化。
- ②该实验测试了输入 B、C 的完整开关过程。状态 4 到 5 中，关闭 C，S 变为 ‘0’；状态 6 到 7 中，打开 C，S 变为 ‘1’。状态 2 到 3 中，打开 B，S 变为 ‘0’；状态 7 到 8 中，关闭 B，S 变为 ‘1’。
- ③从状态 5 到 6 的过程可以看出，当输入值与上次相同，即开关状态不变时，S 输出不变。

结论：仿真满足条件，该代码可以满足设计要求。