

点亮数字人生实验报告

计 71 张程远 2017011429

一、实验目的

- 1、通过数码管点亮程序，熟悉 VHDL 语言，了解掌握硬件程序的编写规范。
- 2、掌握 EDA 软件的使用方法和工作流程。
- 3、进一步理解可编程芯片的工作原理。

二、实验内容

- 1、同时点亮一个经过译码的数码管和一个未经过译码的数码管。数码管从 0 到 f，带译码的显示 0 到 9。
- 2、设计一个数码管显示实验，要求数码管能够显示奇数列、偶数列和自然数列，尽可能多地点亮数码管。要求实验中至少使用一个不带译码的数码管。

三、代码及注释

- 1、同时点亮一个经过译码和未经过译码的数码管：

代码部分：

```
LIBRARY IEEE;
USE IEEE.STD_LOGIC_1164.ALL;
USE IEEE.STD_LOGIC_ARITH.ALL;
USE IEEE.STD_LOGIC_UNSIGNED.ALL;

entity convertor is
    port(
        inp: in std_logic_vector(3 downto 0); --输入
        display: out std_logic_vector(6 downto 0); --不带译码的输出向量
        display_trans: out std_logic_vector(3 downto 0) --带译码的输出向量
    );
end convertor;

architecture prc of convertor is
begin
    display_trans <= inp; --带译码的直接赋值

    process(inp)
    begin
        case inp is
```

```

when "0000"=>display<="1111110"; --0
when "0001"=>display<="0110000"; --1
when "0010"=>display<="1101101"; --2
when "0011"=>display<="1111001"; --3
when "0100"=>display<="0110011"; --4
when "0101"=>display<="1011011"; --5
when "0110"=>display<="0011111"; --6
when "0111"=>display<="1110000"; --7
when "1000"=>display<="1111111"; --8
when "1001"=>display<="1110011"; --9
when "1010"=>display<="1110111"; --a
when "1011"=>display<="0011111"; --b
when "1100"=>display<="1001110"; --c
when "1101"=>display<="0111101"; --d
when "1110"=>display<="1001111"; --e
when "1111"=>display<="1000111"; --f
when others=>display<="0000000"; --default
end case;
end process;
end prc;

```

工作原理：首先设置一个输入向量 inp，一个带译码器的输出向量 display_trans，以及一个不带译码器的输出向量 display。向输入向量中输入 4 个 0 或 1，带译码器的输出向量直接接受输入即可，而不带译码器的输出向量需要在进程中对 inp 进行转换，转换的代码见上，之后数码管就可以显示相应的数值。

2、令数码管显示奇数列、偶数列以及自然数列

代码部分：

```

LIBRARY IEEE ;
USE IEEE.STD_LOGIC_1164.ALL ;
USE IEEE.STD_LOGIC_ARITH.ALL;
USE IEEE.STD_LOGIC_UNSIGNED.ALL ;

entity convertor_new is
port(

```

```

    clk:in std_logic; --输入
    rst:in std_logic; --重置为 0
    display:out std_logic_vector(6 downto 0);--不带译码器
    display_odd:out std_logic_vector(3 downto 0);--带译码器
    display_even:out std_logic_vector(3 downto 0) --带译码器
);
end convertor_new;

architecture prc of convertor_new is
    signal N:std_logic_vector(3 downto 0):="0000" ;-- 自然数列
    signal O:std_logic_vector(3 downto 0):="0001" ;-- 奇数列
    signal E:std_logic_vector(3 downto 0):="0000" ;-- 偶数列
begin
    process(clk,rst)
    begin
        if(clk'event and clk='0') then
            if(N="1001") then N<="0000"; else N<=N+1 ; end if ;
            if(O="1001") then O<="0001"; else O<=O+2 ; end if ;
            if(E="1001") then E<="0000"; else E<=E+2 ; end if ;
            end if;

            if(rst='0') then
                N<="0000" ; O<="0001"; E<="0000";--偶数列和自然数列重置为 0, 奇数为 1
            end if;

            display_odd<=O ;--显示奇数列
            display_even<=E ;--显示偶数列
            case N is --译码, 显示自然数列
                when "0000"=>display<="1111110"; --0
                when "0001"=>display<="0110000"; --1
                when "0010"=>display<="1101101"; --2
                when "0011"=>display<="1111001"; --3
                when "0100"=>display<="0110011"; --4
                when "0101"=>display<="1011011"; --5
                when "0110"=>display<="0011111"; --6
            end case;
        end if;
    end process;
end architecture prc;

```

```

when "0111"=>display<="1110000"; --7
when "1000"=>display<="1111111"; --8
when "1001"=>display<="1110011"; --9
when "1010"=>display<="1110111"; --a
when "1011"=>display<="0011111"; --b
when "1100"=>display<="1001110"; --c
when "1101"=>display<="0111101"; --d
when "1110"=>display<="1001111"; --e
when "1111"=>display<="1000111"; --f
when others=>display<="0000000"; --default
end case;
end process;
end prc;

```

工作原理：首先设置 clk 和 rst 控制输入，rst 效果为重置当前数列至初始状态，而每当按下 clk 时各数列便相应显示下一项。为了显示 3 个数列，代码中设置了 3 个输出向量，其中两个长度为 4 的向量（带译码器）显示奇偶数列，而长度为 7 的向量（不带译码器）显示自然数列。三个数列分别用 3 个信号 OEN 控制，信号的值随着 clk 的每次触发而发生变化。当三个信号的值分别达到相应的最大值时，信号会自动重置为初始值，再从初始开始计数。

四、实验小结

本次实验是我第一次使用可编程器件进行实验，初次实验确实遇到了很多问题，包括 USBblaster 的下载和安装，对 clk 和 rst 的机制和语法不了解，按下 clk 时的信号易产生毛刺从而导致数码管跳变等等。通过查阅百度资料，以及向老师和同学的请教，我一步步解决了上面的问题，并最终完成了整个实验。

整个实验过程中，我熟悉了 Quartus 的使用和 VHDL 语言的语法，并动手将写好的程序烧入了芯片里，最后看到数字被点亮的时候，感到十分有成就感，收获也十分丰富。这次实验标志了我的数字人生正式被点亮，我期待能够编写更复杂的逻辑、搭建更复杂的电路，实现更多实用功能！