

点亮数字人生报告

2019011831 项晨东

实验目的:

1. 熟悉VHDL语言
2. 熟悉quartus软件使用
3. 熟悉硬件下载流程和电路搭建
4. 熟悉译码和不带译码数码管显示的逻辑, 达到课程实验要求

实验内容

同时显示三个数码管, 使得其中一个显示奇数序列, 一个显示偶数序列, 一个显示自己的学号+9208周期串, 并且支持reset功能。至少使用一个不带译码的数码管。

实验代码以及注释

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

ENTITY light IS
    PORT (
        display : OUT STD_LOGIC_VECTOR(6 DOWNTO 0);
        display_4_even : OUT STD_LOGIC_VECTOR(3 DOWNTO 0);
        display_4_odd : OUT STD_LOGIC_VECTOR(3 DOWNTO 0);
        clk, reset : IN STD_LOGIC
    );
END light;
```

```

ARCHITECTURE fire OF light IS
    SIGNAL display_4_buf : STD_LOGIC_VECTOR(3 DOWNTO 0) := "0000";
    SIGNAL display_4_buf_odd : STD_LOGIC_VECTOR(3 DOWNTO 0) := "0001";
    SIGNAL display_4_buf_even : STD_LOGIC_VECTOR(3 DOWNTO 0) := "0000";
    SIGNAL cnt : INTEGER := 0;
BEGIN
    PROCESS (clk, reset)
    BEGIN
        IF (clk'event AND clk = '1') THEN
            IF (cnt < 1000000) THEN --每一秒进行一次变动, 否则只进行计数
                cnt <= cnt + 1;
            ELSE
                cnt <= 0;
                IF (display_4_buf = "1101") THEN --14s为一个周期
                    display_4_buf <= "0000";
                ELSE
                    display_4_buf <= display_4_buf + 1;
                END IF;
                IF (display_4_buf_even = "1000") THEN
                    display_4_buf_even <= "0000";
                ELSE
                    display_4_buf_even <= display_4_buf_even + 2;
                END IF;
                IF (display_4_buf_odd = "1001") THEN
                    display_4_buf_odd <= "0001";
                ELSE
                    display_4_buf_odd <= display_4_buf_odd + 2;
                END IF;
            END IF;
        END IF;
        IF (reset = '1') THEN -- reset判断
            display_4_buf <= "0000";
            display_4_buf_odd <= "0001";
            display_4_buf_even <= "0000";
        END IF;
    END PROCESS;
END ARCHITECTURE fire;

```

```

PROCESS (display_4_buf_even) -- 直接输出
BEGIN
    display_4_even <= display_4_buf_even;
END PROCESS;

PROCESS (display_4_buf_odd) -- 直接输出
BEGIN
    display_4_odd <= display_4_buf_odd;
END PROCESS;

PROCESS (display_4_buf)
BEGIN
    CASE display_4_buf IS
        WHEN "0000" => display <= "1101101"; --2
        WHEN "0001" => display <= "1111110"; --0
        WHEN "0010" => display <= "0110000"; --1
        WHEN "0011" => display <= "1110011"; --9
        WHEN "0100" => display <= "1111110"; --0
        WHEN "0101" => display <= "0110000"; --1
        WHEN "0110" => display <= "0110000"; --1
        WHEN "0111" => display <= "1111111"; --8
        WHEN "1000" => display <= "1111001"; --3
        WHEN "1001" => display <= "0110000"; --1
        WHEN "1010" => display <= "1110011"; --9
        WHEN "1011" => display <= "1101101"; --2
        WHEN "1100" => display <= "1111110"; --0
        WHEN "1101" => display <= "1111111"; --8
        WHEN OTHERS => display <= "0000000";
    END CASE;
END PROCESS;
END fire;

```

实验小结

1. 本次实验我学习了EDA软件的使用, 学会了CPLD的下载等操作, 对VHDL语言进一步熟悉, 对硬件的调试也有了更多的了解。
2. 在本次实验中遇到了M1 mac 兼容性的一些问题, 最后在同学提供的电脑上完成了实验, 非常感激。
3. 非常感谢老师和助教的帮助~