

硬件描述训练 2（第八次 1、2）实验报告

软件 1801 肖云杰 201826010113

（本次实验做完了基于 D、C 级任务的 B 级任务，报告如下）

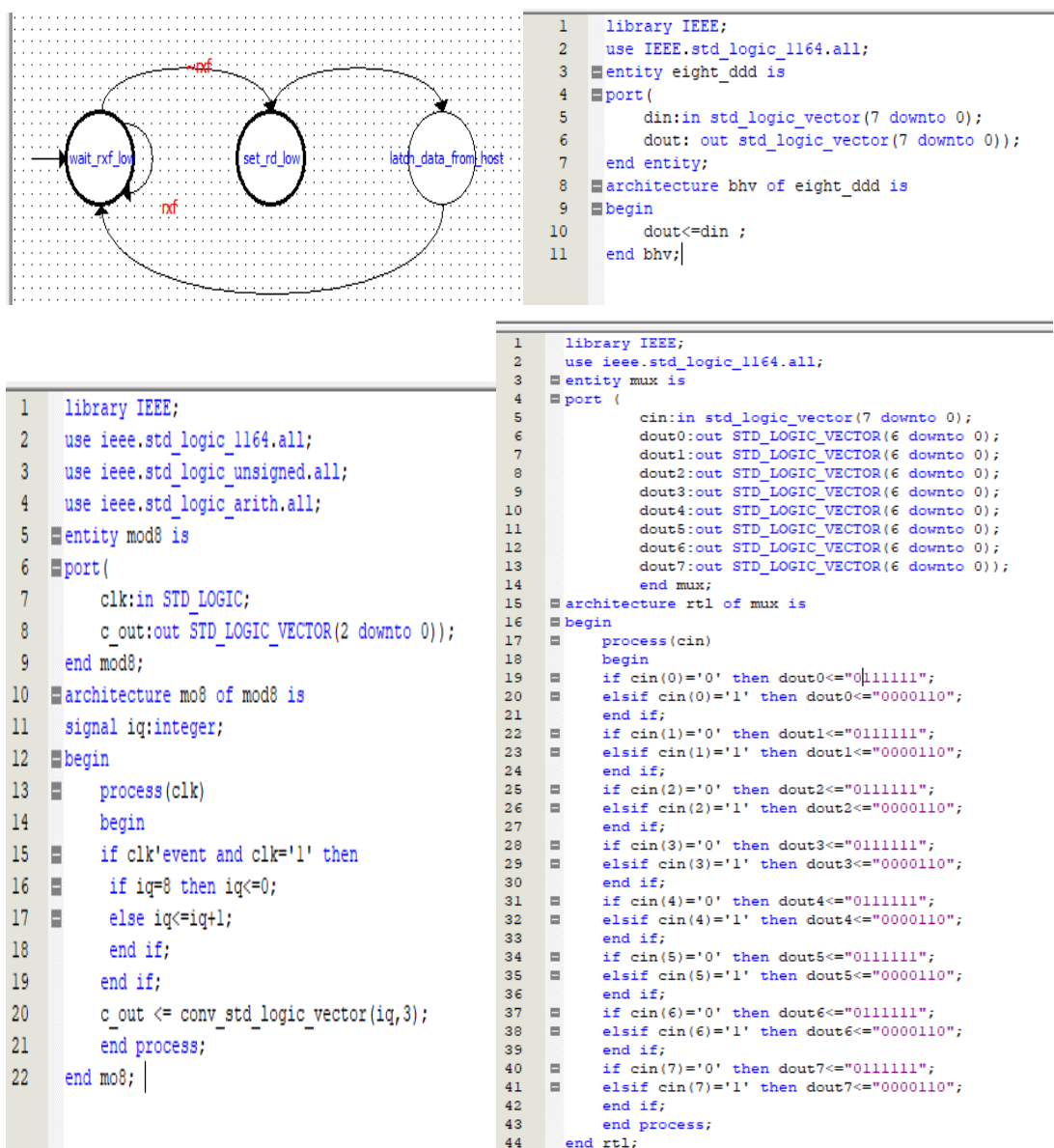
（D、C 级任务的文件部分已经根据 B 更改，图片可能会超出等级）

D 级任务、C 级任务

使用状态图生成实验 VHDL 文件的基本框架，将状态图作修改，使其不包括输入输出（变得容易处理），同时输入输出文件，单独写 VHDL 文件，分别作为模块，加入到顶层文件中。

加入时钟信号，手动重写模八计数器，使模八计数器能将时钟信号转化为 8 位的位选信号，并根据位选信号，从 8 位二进制（0，1）输入中选取相应位，选取其段选信号位，显示。

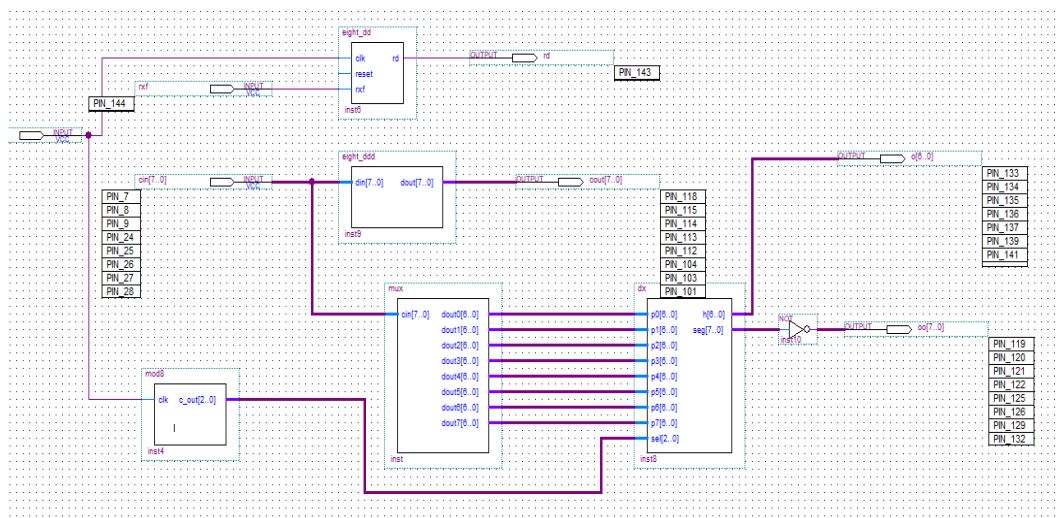
状态图和部分 VHDL 文件如下



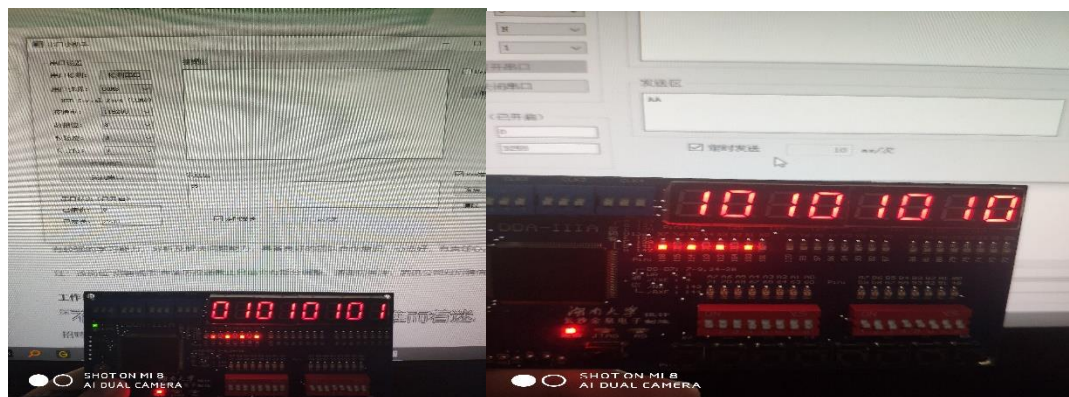
```

1  library IEEE;
2  use ieee.std_logic_1164.all;
3  use ieee.std_logic_unsigned.all;
4  use ieee.std_logic_arith.all;
5  entity dx is
6  port(
7      p0,p1,p2,p3,p4,p5,p6,p7:in STD_LOGIC_VECTOR(6 downto 0);
8      sel:in STD_LOGIC_VECTOR(2 downto 0);
9      h:out STD_LOGIC_VECTOR(6 downto 0);
10     seg:out std_logic_vector(7 downto 0));
11 end dx;
12 architecture dac of dx is
13 begin
14     process(sel)
15     begin
16         if sel="000" then h<=p0;
17         elsif sel="001" then h<=p1;
18         elsif sel="010" then h<=p2;
19         elsif sel="011" then h<=p3;
20         elsif sel="100" then h<=p4;
21         elsif sel="101" then h<=p5;
22         elsif sel="110" then h<=p6;
23         elsif sel="111" then h<=p7;
24         end if;
25     end process;
26     process(sel)
27     begin
28         if sel="000" then seg<="00000001";
29         elsif sel="001" then seg<="00000010";
30         elsif sel="010" then seg<="00000100";
31         elsif sel="011" then seg<="00001000";
32         elsif sel="100" then seg<="00010000";
33         elsif sel="101" then seg<="00100000";
34         elsif sel="110" then seg<="01000000";
35         elsif sel="111" then seg<="10000000";
36         end if;
37     end process;
38 end dac;
39

```



结果图如下



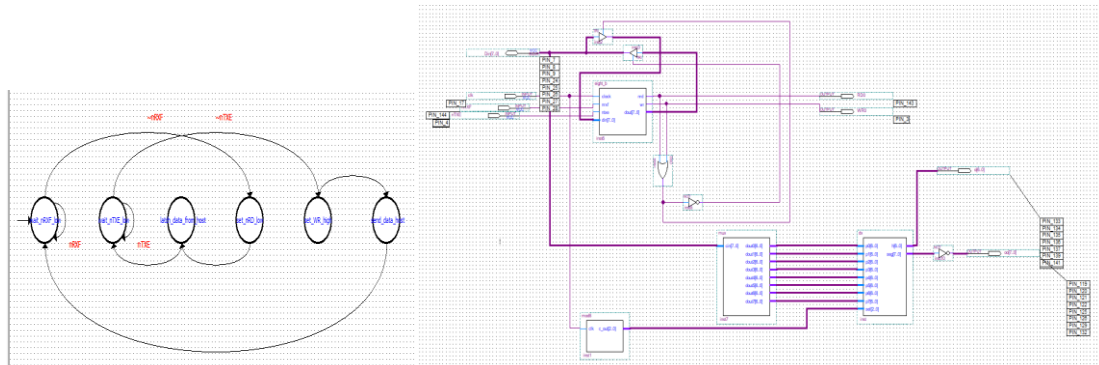
部分分析

数码管显示正确，选用 AA 和 55 是因为 AA 和 55 包括了所有的数码管，可以全方位检测问题。

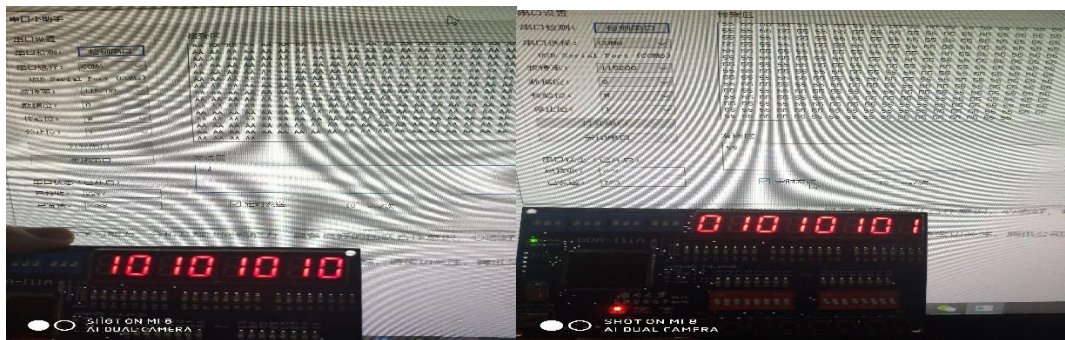
B 级任务

采用状态图，实现基本底层的 VHDL 文件，在其中修改，得到正确的 VHDL，数码管显示电路同 C 级任务，连接输入，分配管脚即可

底层图如下（其余 VHDL 相同）



验证正确图如下



总结

本次实验中，遇到问题，其中

1. 位选信号低电平有效（debug 了一个多小时才发现）-> 需要注重基础知识
2. 手写模八计数器和译码器和位选选择时，发现 VHDL 语法是不是特别熟悉，部分代码需要参考之前写过的代码来写。

了解了状态图输入法的优点，极容易生成基本底层状态图，不过细节处理比较困难（可能是因为我还不熟悉），可以在通过状态图生成的 VHDL 的文件上，修改 VHDL 的基本代码。然后再整合相关模块文件，进行不同模块的测试。

实现了上位机和实验板的相互通信，看到数据传输成功，在实验板的数码管上成功显示，有一种终于成功的自豪感（修改数码管耗时 4h）