**《数字逻辑》实验报告**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **姓名** | **魏永森** | | **年级** | | **2019级** |
| **学号** | **20192242** | | **专业、班级** | | **计算机科学与技术卓越01班** |
| **实验名称** | **项目三 密码锁设计** | | | | |
| **实验时间** | **12.4** | **实验地点** | | **DS1410** | |
| **实验成绩** |  | **实验性质** | | **□验证性 □设计性 □综合性** | |
| 教师评价：  □算法/实验过程正确； □源程序/实验内容提交 □程序结构/实验步骤合理；  □实验结果正确； □语法、语义正确； □报告规范；  评语：  评价教师签名（电子签名）： | | | | | |
| 一、实验目的  设计一个密码锁电路，可以通过拨码开关设置初始密码，通过按钮来输入密码，密码位数至少为4位，判断输入密码与设置的密码是否一致。 | | | | | |
| 二、实验项目内容  **（1）密码锁源代码编写**  **（2）密码锁管脚文件编写**  **（3）密码锁仿真文件编写**  **（4）下板测试** | | | | | |
| 三、实验设计  （1）密码设置为4位（十进制），但用16位来储存，每4位标识一位。  用拨码开关（16个）来设置密码。  五个按钮来输入密码，分别表示了up,down,cnf,back,rst，即对密码某一位的加，减，确定，回退，重置操作。其中重置是直接重置密码所有位，重新输入。  （2）状态转移图 | | | | | |
| 四、实验过程或算法  实验设计过程，首先看总体的模块设计    Clk\_div是时钟分频器，将clk(100MHz)信号转为hz(100Hz)时钟信号。  datain模块实现按钮输入密码的功能  debounce模块是按钮消抖模块  display模块是实现数码管的显示  lock是在没有写状态机的时候，直接对输入密码和设置密码的比较。  State是后来加上去的Moore状态机功能判断，相当于序列检测器。  Top是顶层模块，调用了子模块形成完整的功能。  顶层模块代码如下：  module top(input clk,  input [15:0]code,  input [4:0]button,  output [3:0]sm\_wei,  output [7:0]sm\_duan,  output pass,  output fail  );  wire [15:0]data;  wire hz;  wire up,down,cnf,back,rst;  wire signal;  wire [3:0]pw;  assign signal=cnf;    clk\_div div(clk,hz);  debounce d0(hz,button[0],up);  debounce d1(hz,button[1],down);  debounce d2(hz,button[2],cnf);  debounce d3(hz,button[3],back);  debounce d4(hz,button[4],rst);  datain din(hz,up,down,cnf,back,rst,data,pw);  display dis(clk,data,sm\_wei,sm\_duan);  //lock lk(signal,rst,code,data,pass,fail);  state st(signal,rst,code,pw,pass,fail);  endmodule  其中clk是时钟信号，code代表拨码开关输入的16位2进制密码，但是有效的是4位的十进制密码；[4:0]button是五个拨码开关（加，减，确定，回退，重置）;sm\_wei和sm\_duan是控制数码管输出显示的信号最后的pass和fail信号是最终产生的通过还是失败信号。  Wire型变量[15:0]data用来记录按钮输入的密码。  Hz记录分频之后的时钟信号  Up…..cnf….rst记录加，减，确定，回退，重置信号  Signal记录每次状态机改变状态的信号  [3:0]pw记录每次状态机中每次用来对比的四位二进制输入密码，相当于一位十进制密码  每次按下确定键的时候让signal等于cnf，这样就实现了一个signal的时钟跳变效果，此时在state模块中使得状态改变。  其中除顶层模块之外最重要的就是datain模块。代码如下  module datain(input clk,  input up,input down,input cnf,input back,input rst,  output reg [15:0]data,  output reg [3:0]pw  );  reg [3:0]x=0;  reg [1:0]cnt;    always@(posedge clk or posedge rst)begin  if(rst) begin  x<=0;  cnt<=0;  pw<=0;  data<=16'b1111111111110000;  end    else begin  if(up) begin  if(x==4'b1001)begin  x=4'b0000;  data={data[15:4],x};  end  else begin  x=x+1'b1;  data={data[15:4],x};  end  end  else if(down) begin  if(x==4'b0000)begin  x=4'b1001;  data={data[15:4],x};  end  else begin  x=x-1'b1;  data={data[15:4],x};  end  end  else if(cnf) begin  pw=x;  data<={data[11:0],4'b0000};  x<=0;  end  else if(back) begin  data<={4'b1111,data[15:4]};  x<=0;  pw<=0;  end  else ;  end  end  endmodule  模块定义变量：  Clk时钟信号，up，down，cnf，back，rst是五个按钮经过消抖之后的信号。Data[15:0]data是将五个按钮输入的密码作为输出，传到display模块中进行显示输出。[3:0]pw是每次按下确定键cnf之后当前的四位二进制密码，也就是一位十进制密码，并将其传到state中与code中的四位进行对比，并改变状态。  其中reg型矢量cnt是当使用lock模块而不是state状态机模块的时候记录输入了几位十进制密码的变量。在state模块使用中无作用。  另一个reg型矢量x是记录当前四位十进制数字中的某一位。也就是四位二进制。  接下来在always模块中主要分为两大部分，一部分是rst信号起作用，一部分是rst信号不起作用的时候。比较复杂的是后者。  在这之中需要用条件判断实现4个按钮（除rst信号）的输入功能。  当信号为up的时候，使得当前位密码加一（十进制）。因为4位可以最高存储到15，所以再加一个条件判断语句，当输入已经到9之后，下一位就是0.然后将当前输入的四位二进制密码与data数据中存储的最低四位互换，这样就实现了只在四位二进制数字中操作。  当信号为down的时候，使得当前密码位减一（十进制）。过程与up信号作用过程大同小异。  当信号为cnf的时候，使得pw矢量保存当前data信号的最低四位，也就是十进制的一位数字。然后利用data<={data[11:0],4'b0000};语句使得数码管上显示的数字向高位移动一位。这样看起来有动态效果，并且易于分辨（如果不用state模块判断而使用lock模块的话，此处会有cnt作用使得到第四次确定的时候不移动位置，这样看起来更美观，实现输入四位十进制的功能并判断）  当信号为back的时候，和cnf信号作用基本一致。语句data<={4'b1111,data[15:4]};使得四位密码右移一位，高位用1111填充（再数码管中设置，1111使得该位数码管不显示，为黑暗）然后回退之后用up，down信号就可以改变该位密码的值。 | | | | | |
| 五、实验过程中遇到的问题及解决情况  1.刚开始做的时候，没有用状态机，然后输入四位密码，数码管上会移动四次，移动到最后会产生前三位有数字，最后一位为0的后果，之后通过增加cnt矢量，记录输入次数，可以使得在第四次输入之后，密码显示不移动。但这样之后back的操作会受影响，之后通过更改cnt变量使得操作正常。  2.后来增加状态机也遇到很多问题，比如输出的时序问题，需要将clk更换为signal变量（当按下确定键时起作用）。更改datain模块使得能够输出四位二进制数字到state模块中比较。 | | | | | |
| 六、实验结果及分析和（或）源程序调试过程  （1）仿真截图  State状态机    Datain输入    （2）下板截图  拨码开关输入密码1131    输入某个数之后按中间按钮左移，发现第二位是3，此时可以执行back操作，右移到第二位修改    一次输入1，1，3，1.并按确定之后最后的led亮，代表pass  若是倒数第二个led亮，则代表fail    七、小组分工情况说明  魏永森：设计密码的输入和对应的七段数码管模块，编写顶层模块。  张鑫：设计密码锁的状态转移图并编写相应的模块，编写激励程序进行仿真。 | | | | | |