# 时序电路作业

## 何尔恒 20172333090 通信6 A班

library ieee;

use ieee.std\_logic\_1164.all;

use ieee.std\_logic\_unsigned.all;

use ieee.std\_logic\_arith.all;

entity lock is

port(key1,key2,key3,key4,key5 : in std\_logic; clk : in std\_logic;

reset : in std\_logic;

dig : out std\_logic\_vector(3 downto 0); seg : out std\_logic\_vector(7 downto 0)

);

end entity lock; architecture main of lock is

signal scan : STD\_LOGIC\_VECTOR(1 DOWNTO 0);

signal jinwei,jishuclk,clear,rst,divclk : STD\_LOGIC;

signal num1out,num2out,num4out,num,num1,num2,num4,num5 : integer range 0 to 19;

signal num3out,num3 : integer range 10 to 19;

signal state : integer range 0 to 3;

signal oneseg : std\_logic\_vector(7 downto 0); signal moshi : std\_LOGIC\_VECTOR(1 DOWNTO 0);

signal state1,state2,state3,state4,state5,key1filt,key2filt,key3filt,key4filt,key5filt : std\_logic;

signal key1cnt,key2cnt,key3cnt,key4cnt,key5cnt : integer range 0 to 50000000; -- 用于对key1 按键输入有效时间进行计数

begin process(clk)

constant clkfrq :integer := 50000000;

constant jishuclkfrq :integer := 500000000;

constant scanfrq :integer := 50;

constant jishufrq :integer := 50;

variable count:integer range 0 to 50000000;

variable jishucount:integer range 0 to 50000000;

begin

if clk'event and clk = '1' then

if count = clkfrq /(scanfrq\*4)-1 then count := 0;

divclk <= '1';

else

count := count + 1; divclk <= '0';

end if;

if jishucount = jishuclkfrq /(jishufrq\*4)-1 then jishucount := 0;

jishuclk <= '1';

else

jishucount := jishucount + 1;

jishuclk <= '0';

end if;

end if;

end process;

process(divclk,rst)

begin

if rst = '0' then

scan <= "00";

elsif divclk'event and divclk = '1' then scan <= scan + '1';

end if;

end process;

process(moshi)

begin

case moshi is

when "00" => state <= 0;

when "01" => state <= 1;

when "10" => state <= 2;

when "11" => state <= 3;

end case;

end process;

process(num)

begin

case num is

when 0 => seg <= "11000000"; -- 0

when 1 => seg <= "11111001"; -- 1

when 2 => seg <= "10100100"; -- 2

when 3 => seg <= "10110000"; -- 3

when 4 => seg <= "10011001"; -- 4

when 5 => seg <= "10010010"; -- 5

when 6 => seg <= "10000010"; -- 6

when 7 => seg <= "11111000"; -- 7

when 8 => seg <= "10000000"; -- 8

when 9 => seg <= "10010000"; -- 9

when 10 => seg <= "01000000"; -- 0.

when 11 => seg <= "01111001"; -- 1.

when 12 => seg <= "00100100"; -- 2.

when 13 => seg <= "00110000"; -- 3.

when 14 => seg <= "00011001"; -- 4.

when 15 => seg <= "00010010"; -- 5.

when 16 => seg <= "00000010"; -- 6.

when 17 => seg <= "01111000"; -- 7.

when 18 => seg <= "00000000"; -- 8.

when 19 => seg <= "00010000"; -- 9.

when others =>NULL;

end case;

end process;

process(scan)

begin

case scan is

when "00" => dig <= "1110";--选通第一个数码管

num <= num1;--同时把第一个数码管显示的数据送给总线

when "01" => dig <= "1101"; --选通第二个数码管

num <= num2; --同时把第二个数码管显示的数据送给总线

when "10" => dig <= "1011"; --选通第三个数码管

num <= num3 ; --同时把第三个数码管显示的数据送给总线

when "11" => dig <= "0111"; --选通第四个数码管

num <= num4; --同时把第四个数码管显示的数据送给总线

end case;

end process;

process(clk)

begin

if(clk'event and clk ='1')then

if(key2filt='1')then

moshi<="00"; end if;

if(key3filt='1')then

moshi<="01"; end if;

if(key4filt='1')then

moshi<="10"; end if;

end if;

end process;

process(jishuclk)

begin

if(state = 0)then

if jishuclk = '1' then num1 <= num1out;

if num1out = 9 then

num1out <= 0;

else

num1out <= num1out + 1;

end if;

if num1out = 0 then num2 <= num2out;

if num2out = 9 then

num2out <= 0;

else

num2out <= num2out + 1;

end if;

end if;

if num2out = 0 and num1out = 0 then

num3 <= num3out;

if num3out = 19 then

num3out <= 10;

else

num3out <= num3out + 1;

end if; end if;

if num3out= 10 and num2out = 0 and num1out = 0 then

num4 <= num4out;

if num4out = 9 then num4out <= 0;

else

num4out <= num4out + 1;

end if;

end if;

end if;

end if;

if(state = 2)then

num1out <= 0;

num2out <= 0;

num3out <= 10;

num4out <= 0;

num1 <= num1out;

num2 <= num2out;

num3 <= num3out;

num4 <= num4out;

end if;

end process;

xiao\_dou2:process (key2)

constant N :integer := 5000000; --消抖时间，对于 50Mhz 的基准时钟，这相当于 0.1S

begin

if clk'event and clk = '1' then

if key2 = '0' then --当 key2 输入低电平，即按键按下

if key2cnt /= N then --一直计数到 N

key2cnt <= key2cnt + 1;

end if;

if key2cnt = N-1 then --最后一个计数时输出 key1filt 脉冲

key2filt <= '1';

else

key2filt <= '0';

end if;

else --若 key2 输入高电平，表明按键被释放

key2cnt <= 0; end if;

end if; end process;

xiao\_dou3:process (key3)

constant N :integer := 5000000; --消抖时间，对于 50Mhz 的基准时钟，这相当于 0.1S

begin

if clk'event and clk = '1' then

if key3 = '0' then --当 key3 输入低电平，即按键按下

if key3cnt /= N then --一直计数到 N

key3cnt <= key3cnt + 1;

end if;

if key3cnt = N-1 then --最后一个计数时输出 key3filt 脉冲

key3filt <= '1';

else

key3filt <= '0';

end if;

else --若 key3 输入高电平，表明按键被释放

key3cnt <= 0; end if;

end if; end process;

xiao\_dou4:process (key4)

constant N :integer := 5000000; --消抖时间，对于 50Mhz 的基准时钟，这相当于 0.1S

begin

if clk'event and clk = '1' then

if key4 = '0' then --当 key4 输入低电平，即按键按下

if key4cnt /= N then --一直计数到 N

key4cnt <= key4cnt + 1;

end if;

if key4cnt = N-1 then --最后一个计数时输出 key1filt 脉冲

key4filt <= '1';

else

key4filt <= '0';

end if;

else --若 key4 输入高电平，表明按键被释放

key4cnt <= 0;

end if;

end if;

end process;

end architecture main;

图片包含 电子产品, 电路

描述已自动生成

归零

图片包含 电子产品

描述已自动生成

暂停

图片包含 电子产品

描述已自动生成

继续计时