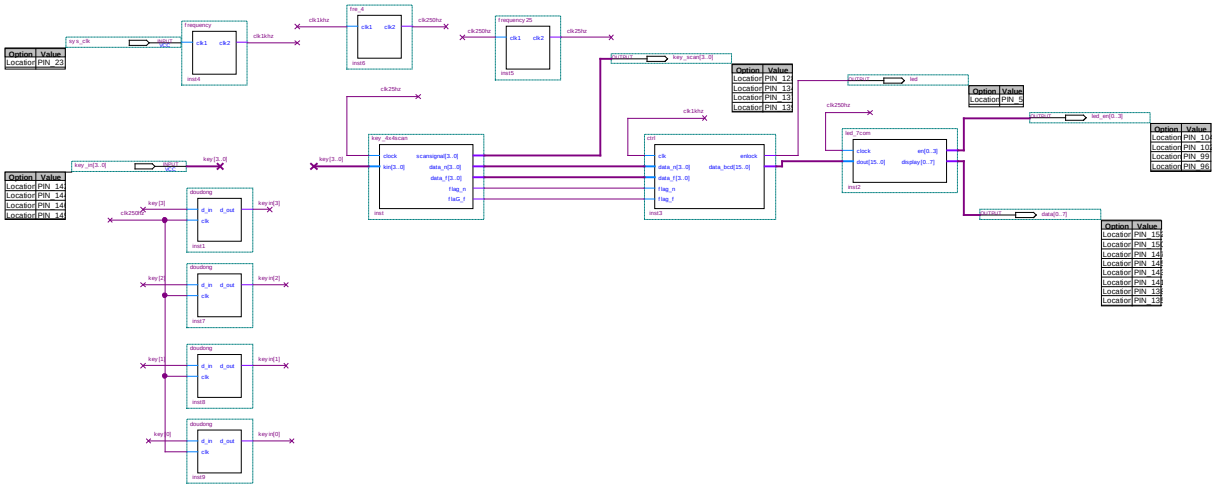

```
1  --键盘去抖动
2  library ieee;
3  use ieee.std_logic_1164.all;
4  entity doudong is
5  port(d_in,clk:in std_logic;
6  --   dd1,dd0,qq1,qq0:out std_logic;
7     d_out:out std_logic);
8  end;
9  ----------------------------------------
10 architecture bhv_doudong of doudong is
11 component dcfq is
12 PORT(CLK,clrn,prn,d:in std_logic;
13 q:out std_logic);
14 end component dcfq;
15 ------------------------------------------
16 signal vcc,inv_d:std_logic;
17 signal q0,q1:std_logic;
18 signal d1,d0:std_logic;
19 begin
20 vcc<='1';
21 inv_d<=not d_in;
22 u1:dcfq port map(clk=>clk,clrn=>inv_d,prn=>vcc,d=>vcc,q=>q0);
23 u2:dcfq port map(clk=>clk,clrn=>q0,prn=>vcc,d=>vcc,q=>q1);
24 ------------------------------------------------
25 process(clk)
26 begin
27 if clk'event and clk='1' then
28 d0<=not q1;
29 d1<=d0;
30 end if;
31 end process;
32 --dd0<=d0;
33 --dd1<=d1;
34 --qq1<=q1;
35 --qq0<=q0;
36 d_out<=not(d1 and not d0);
37 --d_out1<=not q1;
38 end ;
39
```

```
1  --十分频电路用于键盘扫描电路
2  library ieee;
3  use ieee.std_logic_1164.all;
4  use ieee.std_logic_unsigned.all;
5  entity frequency25 is
6  port
7  ( clk1:in std_logic;
8   clk2:out std_logic );
9  end ;
10 architecture count of frequency25 is
11 signal tout:integer range 0 to 4;--50mhz分频为25hz
12 signal clk:std_logic;
13 begin
14 process(clk1)
15 begin
16 if rising_edge(clk1)then
17   if tout=4 then--每计到5个，上升沿反转一次
18       tout<=0;
19       clk<=not clk;
20   else tout<=tout+1;
21   end if;
22 end if;
23 end process;
24 clk2<=clk;
25
26 end ;
```

```
--密码控制电路
library ieee;
use ieee.std_logic_1164.all;
use ieee.std_logic_arith.all;
use ieee.std_logic_unsigned.all;
entity ctrl is
port(
     clk:in std_logic;
     data_n:in std_logic_vector(3 downto 0);
     data_f:in std_logic_vector(3 downto 0);
     flag_n:in std_logic;
     flag_f:in std_logic;
     enlock:out std_logic;--1:lock,0:unlock
     data_bcd:out std_logic_vector(15 downto 0)--数码管显示
);
end;
-------------------------------------------------
architecture bhv_c of ctrl is
signal acc,reg:std_logic_vector(15 downto 0);
                    --acc 用于暂存键盘输入的信息  reg用于存储输
氲拿苏
signal nc :std_logic_vector(2 downto 0);
signal rr2,bb,qa,QB:STD_LOGIC;
SIGNAL R1,R0:STD_LOGIC;
BEGIN
-----------------------------------------------------------
--寄存器清0信号的产生过程
process(clk,r0,r1,rr2)
begin
if clk'event and clk='1' then
 r1<=r0;
 r0<=flag_f;
end if;
rr2<=r1 and not r0;
end process;
-----------------------------------------------------------
--按键输入的数据的存储,清零过程
keyin_process :block is
signal rst,d0,d1:std_logic;
begin
rst<=rr2;
process(flag_n,rst)
begin
if rst='1' then
  acc<=(others=>'0');
  nc<="000";
else
if flag_n'event and flag_n='1' then
 if nc<4 then
 acc<=acc(11 downto 0)& data_n;
nc<=nc+1;
end if;
end if;
```

```
1  --44键盘去抖
2  --采用计数值的方法来确认按键是否按下
3  library ieee;
4  use ieee.std_logic_unsigned.all;
5  use ieee.std_logic_1164.all;
6  entity key_enter is
7  port
8  ( clock:in std_logic;
9    numin:in integer range 0 to 15;
10   numout:out integer range 0 to 15
11 );
12 end;
13 architecture behavior of key_enter is
14 signal tempnum:integer range 0 to 15;
15 signal counter :integer range 0 to 31;
16 --signal start:std_logic;
17 begin
18 process(clock)
19 begin
20 if rising_edge(clock)then
21 --if start='0' then--上电后立即输出的键值赋予无效值
22  -- tempnum<=15;-----此处沿用上一程序的做法，将1作为无效值
23 --numout<=15;
24 --start<='1';
25 --else
26 if numin/=tempnum then--上一键值与此键值不同
27   tempnum<=numin;
28   counter<=0;
29 else
30 if counter=31 then--当键值保持31个时钟周期不变
31    numout<=numin;--即确认为有效值，并输出
32    counter<=0;
33 else
34 counter<=counter+1;
35 --end if;
36 end if;
37 end if;
38 end if;
39 end process;
40 end;
41
42
43
44
45
46
47
48
49
50
51
52
53
```

54
55

```
1  --十分频电路
2  library ieee;
3  use ieee.std_logic_1164.all;
4  use ieee.std_logic_unsigned.all;
5  entity frequency is
6  port
7  ( clk1:in std_logic;
8   clk2:out std_logic );
9  end ;
10 architecture count of frequency is
11 signal tout:integer range 0 to 24999;--50mhz分频为1000hz
12 signal clk:std_logic;
13 begin
14 process(clk1)
15 begin
16 if rising_edge(clk1)then
17   if tout=24999 then--每计到5个，上升沿反转一次
18       tout<=0;
19       clk<=not clk;
20   else tout<=tout+1;
21   end if;
22 end if;
23 end process;
24 clk2<=clk;
25 end ;
```

```
1  --2007。4。4
2  --串行联接的七段数码管驱动程序，用了4个数码管
3  --四位数码管动态显示。
4  --高电平选通位选端。共阴数码管
5
6  library ieee;
7  use ieee.std_logic_1164.all;
8  use ieee.std_logic_unsigned.all;
9  entity led_7com is
10 port(clock:in std_logic;--时钟频率为250hz
11      dout:IN std_logic_vector(15 downto 0);
12      en:out std_logic_vector(0 to 3);--分别接4个数码管的公共端
13      display:out std_logic_vector(0 to 7 ));--接数码管的7个控制端
     abcdefg
14     end;
15 ------------------------------------------------------------
16 architecture light of led_7com is
17 signal counter:integer range 0 to 3;
18 begin
19   process(clock)
20  variable num:std_logic_vector(3 downto 0);
21 begin
22
23 if clock'event and clock='1' then
24  if counter =3 then
25       counter<=0;
26 else
27       counter<=counter+1;
28 end if;
29
30 case counter is
31 when 0 =>
32        en<="1000";             --点亮第一个数码管，屏蔽其它5个数码
   管
33        num:=dout(3 downto 0);              --显示第一个数
34 when 1=>
35        en<="0100";
36    num:=dout(7 downto 4);
37 when 2=>
38       en<="0010";
39   num:=dout(11 downto 8);
40 when 3=>
41       en<="0001";
42    num:=dout(15 downto 12);
43 when others=>null;
44 -------------------------------------------------
45 end case;
46 case num is
47 when "0000"=> display <="11111101";--0abcdefg段
48 when "0001"=> display <="01100000";--1
49 when "0010"=> display <="11011011";
50 when "0011"=> display <="11110010";
51 when "0100"=> display <="01100111";
```