xxx

实验报告

**实验一 序列检测器**

**代码：**

library ieee;

use ieee.std\_logic\_1164.all;

entity String\_Detector is

port(z,clock,reset: in bit;

f: out bit);

end String\_Detector;

architecture arch of String\_Detector is

type state is (s0,s1,s2,s3,s4,s5,s6,s7);

signal pr\_state,nx\_state: state;

begin

process(reset,clock)

begin

if (reset = '1') then

pr\_state <= s0;

elsif (clock'event and clock = '1') then

pr\_state <= nx\_state;

end if;

end process;

process(z,pr\_state)

begin

case pr\_state is

when s0 =>

f <= '0';

if (z = '0') then

nx\_state <= s0;

else

nx\_state <= s1;

end if;

when s1 =>

f <= '0';

if (z = '0') then

nx\_state <= s0;

else

nx\_state <= s2;

end if;

when s2 =>

f <= '0';

if (z = '0') then

nx\_state <= s0;

else

nx\_state <= s3;

end if;

when s3 =>

f <= '0';

if (z = '0') then

nx\_state <= s4;

else

nx\_state <= s3;

end if;

when s4 =>

f <= '0';

if (z = '0') then

nx\_state <= s5;

else

nx\_state <= s1;

end if;

when s5 =>

f <= '0';

if (z = '0') then

nx\_state <= s0;

else

nx\_state <= s6;

end if;

when s6 =>

f <= '0';

if (z = '0') then

nx\_state <= s7;

else

nx\_state <= s2;

end if;

when s7 =>

f <= '1';

if (z = '0') then

nx\_state <= s0;

else

nx\_state <= s1;

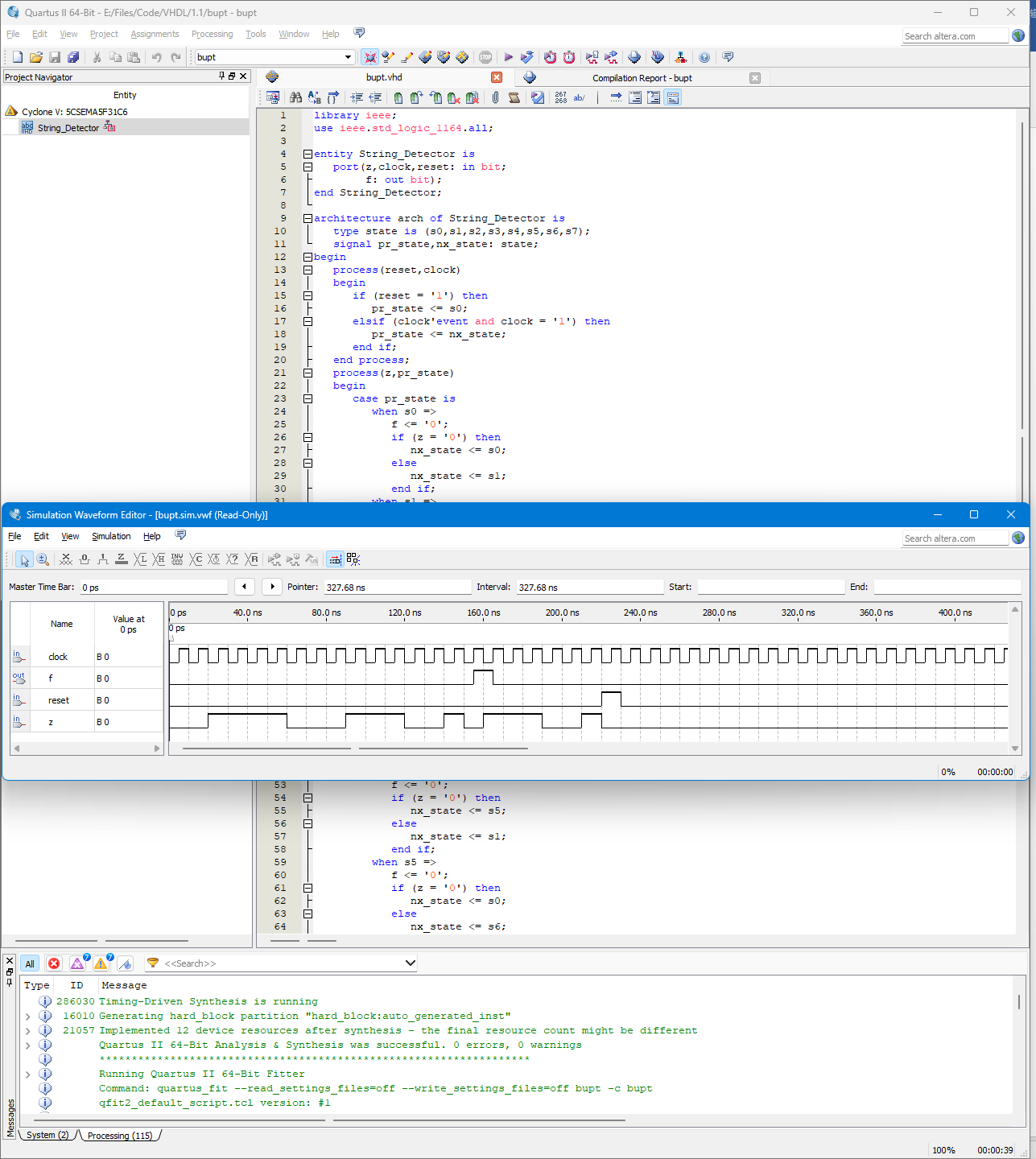
end if;

end case;

end process;

end arch;

**波形图：**



输入序列为：

z:

00111100011100101110010

reset:

00000000000000000000001

第一段匹配串处f正确输出1

第二段匹配串处reset成功，f正确输出0

**实验二 计数器**

**代码：**

library IEEE;

use IEEE.STD\_LOGIC\_1164.ALL;

use IEEE.STD\_LOGIC\_UNSIGNED.ALL;

entity Count4 IS

port(clock,reset: IN STD\_LOGIC;

f : OUT STD\_LOGIC\_VECTOR(3 downto 0));

end Count4;

architecture arch of Count4 is

signal temp:STD\_LOGIC\_VECTOR(3 downto 0);

begin

process(clock,reset)

begin

if (clock'event AND clock='1') then

if (reset='1') then

temp<=(OTHERS=>'0');

else

temp<=temp+1;

end if;

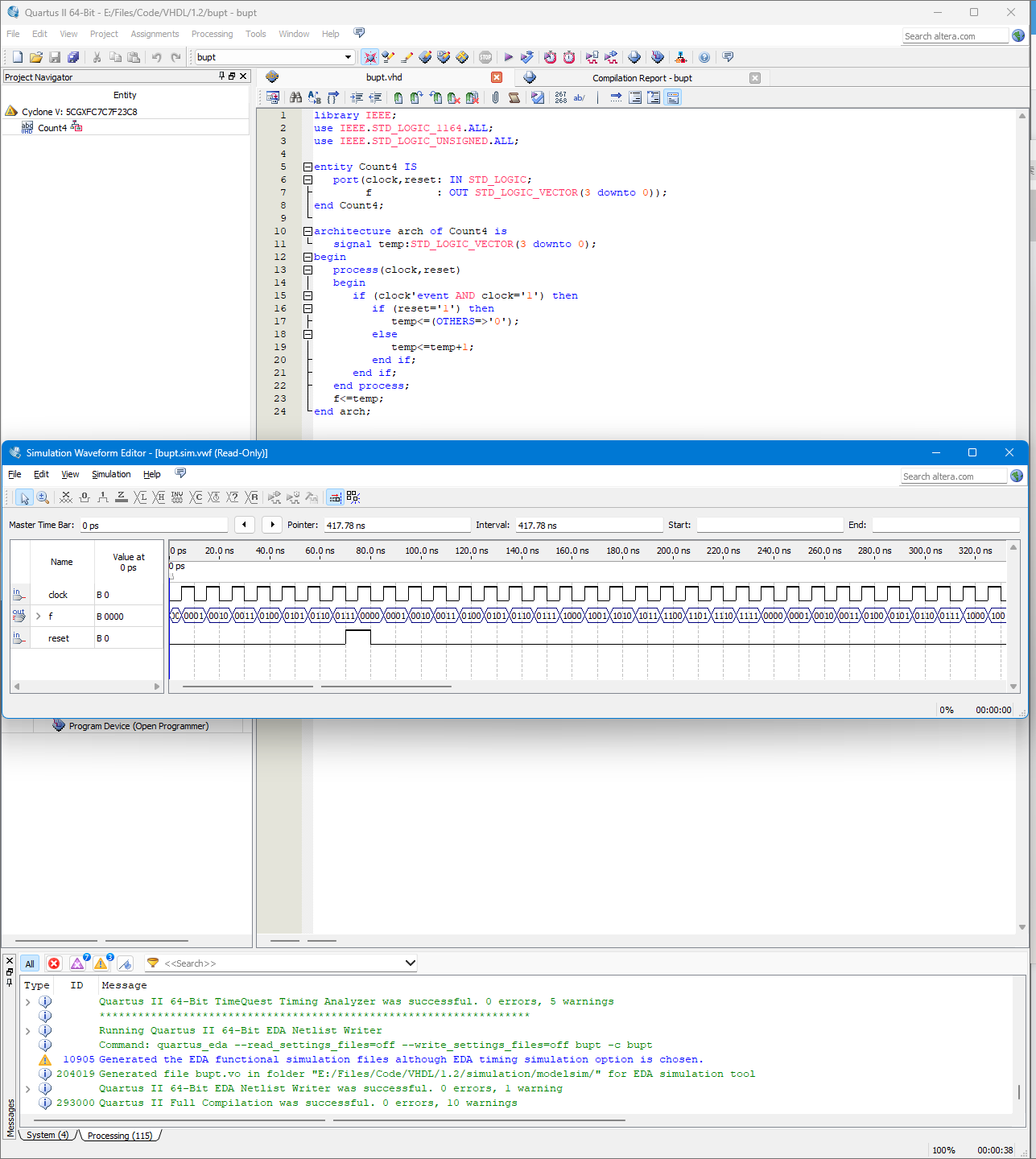
end if;

end process;

f<=temp;

end arch;

**波形图：**



clock上升沿时f状态正确改变

reset成功f状态正确重置为0000

**实验三 8位寄存器74374**

**代码：**

library IEEE;

use IEEE.STD\_LOGIC\_1164.ALL;

entity Register8 is

port (D:IN STD\_LOGIC\_VECTOR(7 downto 0);

CLK,OE:IN STD\_LOGIC;

Q:OUT STD\_LOGIC\_VECTOR(7 downto 0));

end Register8;

architecture arch of Register8 is

signal temp:STD\_LOGIC\_VECTOR(7 downto 0);

begin

process(CLK,OE)

begin

if (OE = '0') then

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

temp<=D;

end if;

else

temp<="ZZZZZZZZ";

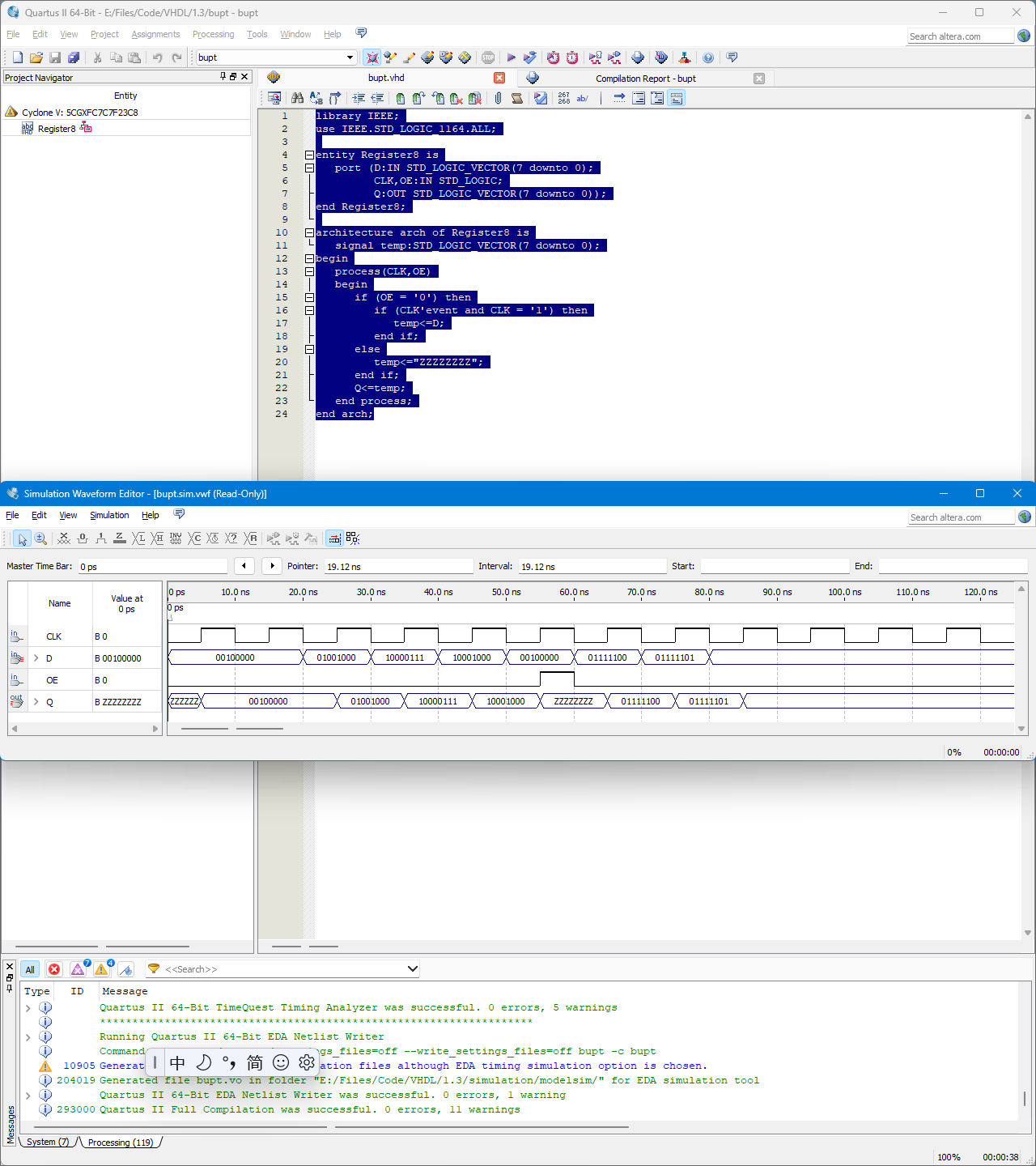
end if;

Q<=temp;

end process;

end arch;

**波形图：**



OE为0时Q正确输出D

OE为1时Q正确输出高阻Z

**实验一 8421 码和格雷码的转换**

**代码：**

library IEEE;

use IEEE.STD\_LOGIC\_1164.ALL;

entity Encoder8421toGray is

port (z:IN STD\_LOGIC\_VECTOR(3 downto 0);

f:OUT STD\_LOGIC\_VECTOR(3 downto 0));

end Encoder8421toGray;

architecture arch of Encoder8421toGray is

signal temp:STD\_LOGIC\_VECTOR(3 downto 0);

begin

process(z)

begin

temp(3)<=z(3);

for i in 2 downto 0 loop

temp(i)<=z(i+1) xor z(i);

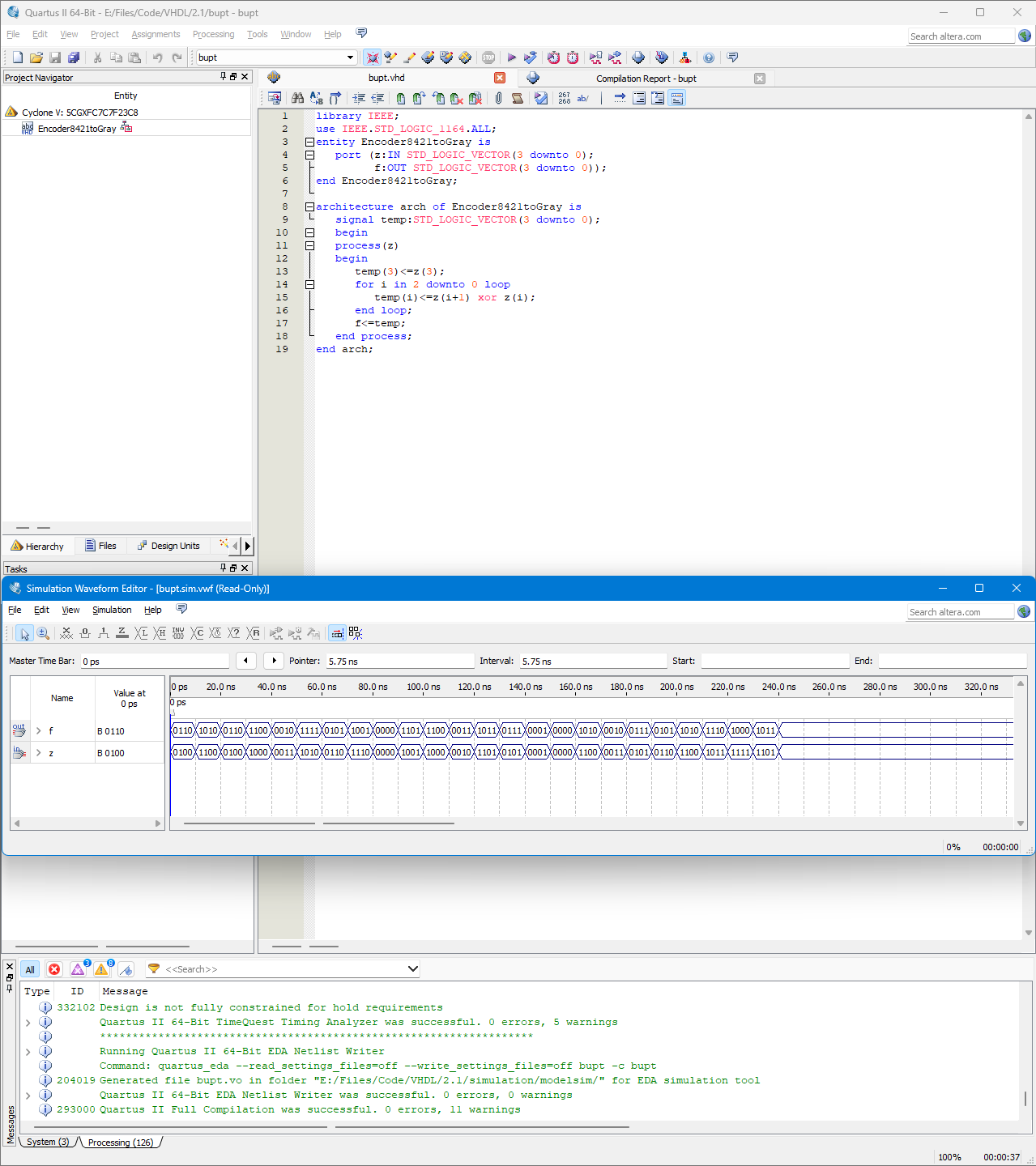
end loop;

f<=temp;

end process;

end arch;

**波形图：**



F正确输出

**实验二 数值比较器**

**代码：**

library IEEE;

use IEEE.STD\_LOGIC\_1164.ALL;

entity compare4 is

port (A,B:IN STD\_LOGIC\_VECTOR (3 downto 0);

YA,YB,YC:OUT STD\_LOGIC);

end compare4;

architecture arch of compare4 is

begin

process(A,B)

begin

if (A>B) then

YA<='1';

YB<='0';

YC<='0';

elsif (A<B) then

YA<='0';

YB<='1';

YC<='0';

else

YA<='0';

YB<='0';

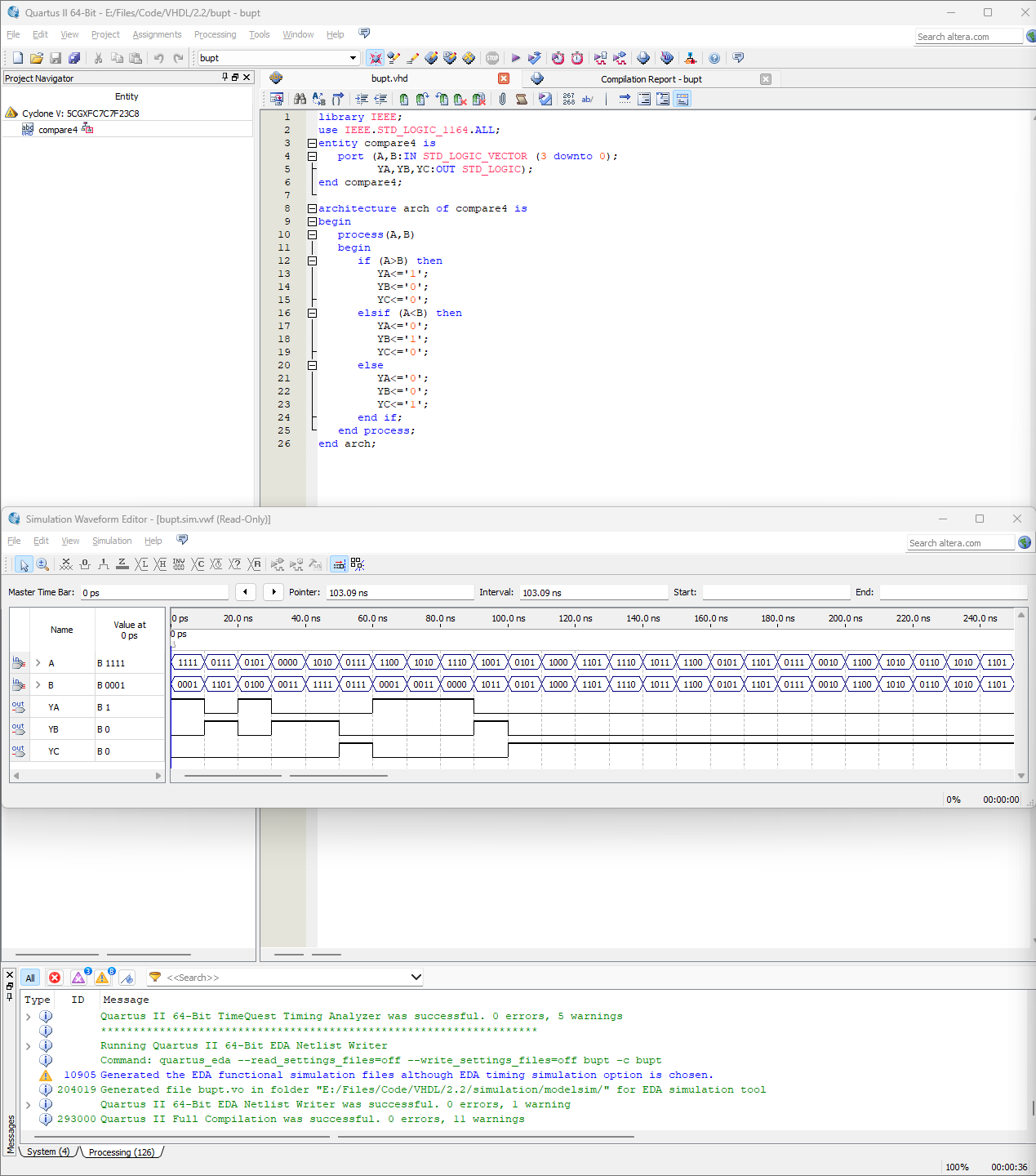
YC<='1';

end if;

end process;

end arch;

**波形图：**



YA、YB、YC正确输出

**实验三 全加器**

**代码：**

library IEEE;

use IEEE.STD\_LOGIC\_1164.ALL;

entity Fulladder is

port(A,B,Cin:IN STD\_LOGIC;

S,Cout:OUT STD\_LOGIC);

end Fulladder;

architecture arch of Fulladder is

begin

process(A,B,Cin)

begin

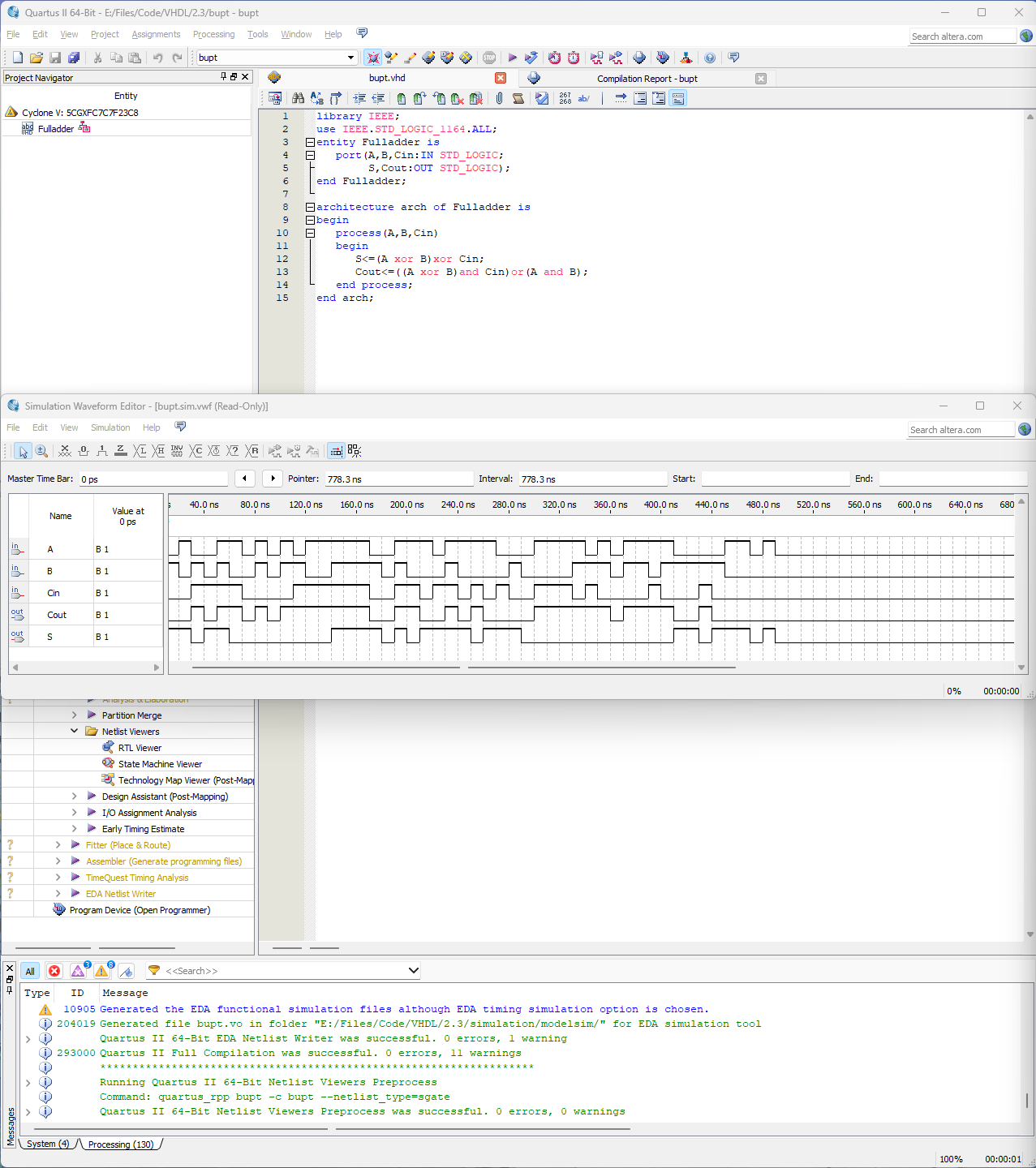
S<=(A xor B)xor Cin;

Cout<=((A xor B)and Cin)or(A and B);

end process;

end arch;

**波形图：**



S正确输出

Cout正确输出

**实验四 3 线-8 线译码器**

**代码：**

library IEEE;

use IEEE.STD\_LOGIC\_1164.ALL;

entity Decoder3to8 is

port (G1,G2A,G2B,A,B,C:IN STD\_LOGIC;

Y:OUT STD\_LOGIC\_VECTOR (7 downto 0));

end Decoder3to8;

architecture arch of Decoder3to8 is

signal temp:STD\_LOGIC\_VECTOR (5 downto 0);

begin

process(G1,G2A,G2B,A,B,C)

begin

temp<=G1 & G2A & G2B & C & B & A;

case temp is

when "100000" =>Y<="01111111";

when "100001" =>Y<="10111111";

when "100010" =>Y<="11011111";

when "100011" =>Y<="11101111";

when "100100" =>Y<="11110111";

when "100101" =>Y<="11111011";

when "100110" =>Y<="11111101";

when "100111" =>Y<="11111110";

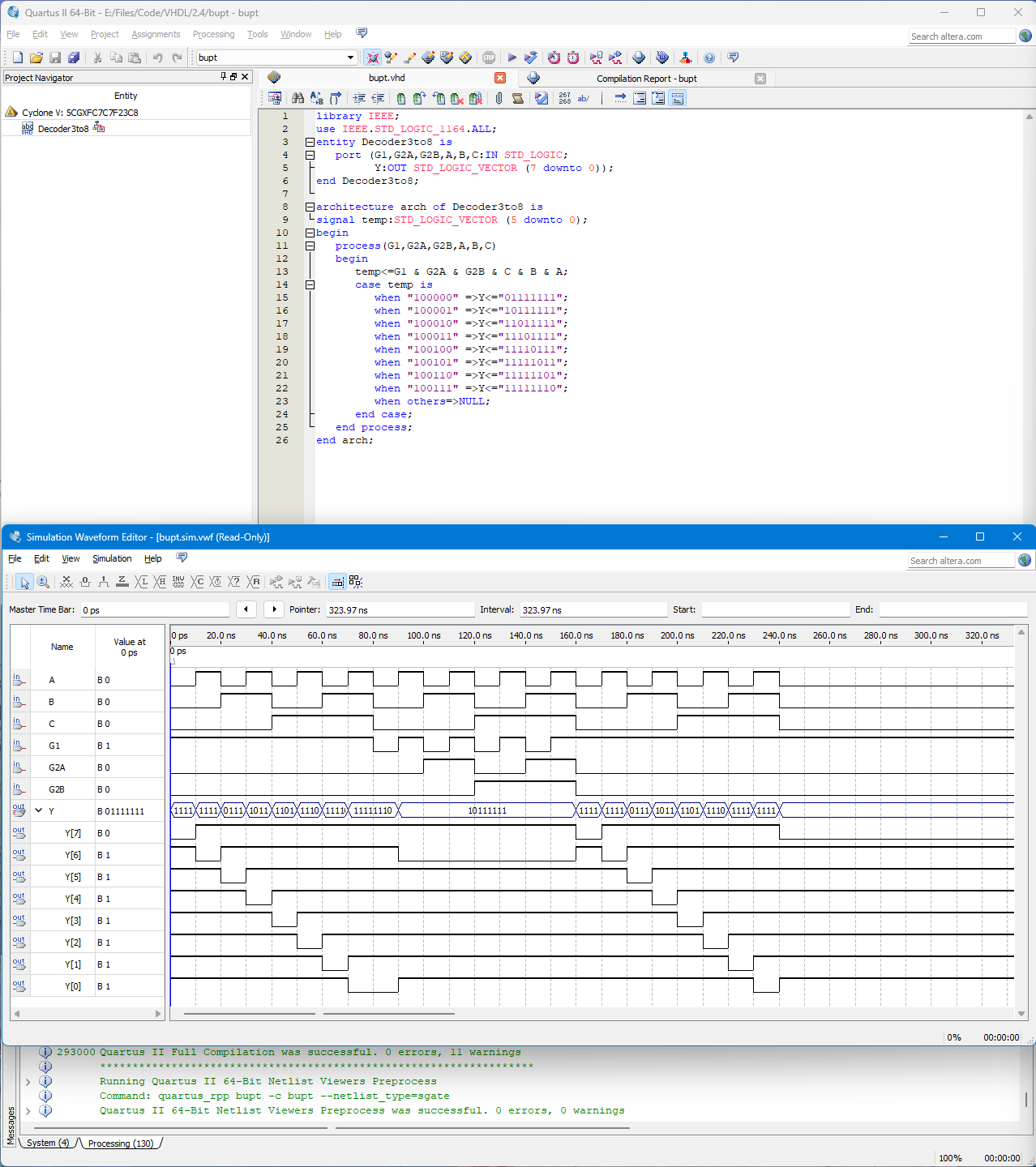
when others=>NULL;

end case;

end process;

end arch;

**波形图：**



**实验五 表决器**

**代码：**

library IEEE;

use IEEE.STD\_LOGIC\_1164.all;

entity Voter is

port(z:IN STD\_LOGIC\_VECTOR(3 downto 0);

f:OUT STD\_LOGIC);

end Voter;

architecture arc of Voter is

begin

process(z)

variable count:integer range 0 to 4;

begin

count:=0;

for n in 0 to 3 loop

if z(n)='1' then

count:=count+1;

end if;

end loop;

if(count>=3)then

f<='1';

else

f<='0';

end if;

end process;

end arc;

**波形图：**

