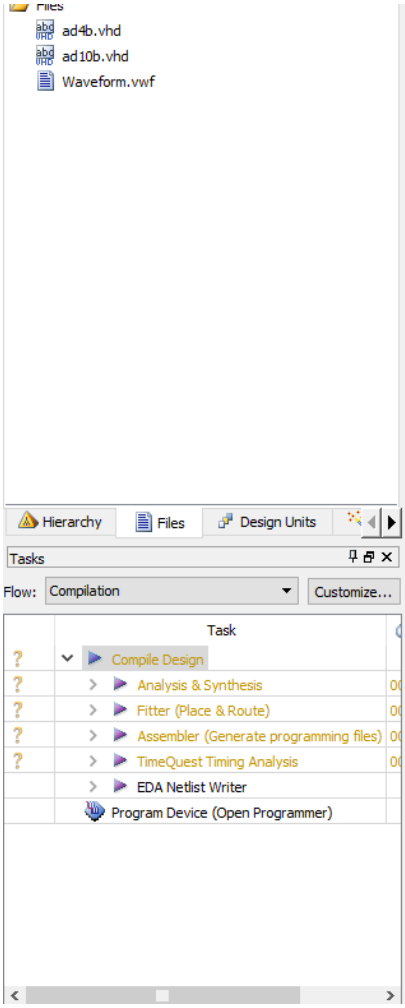
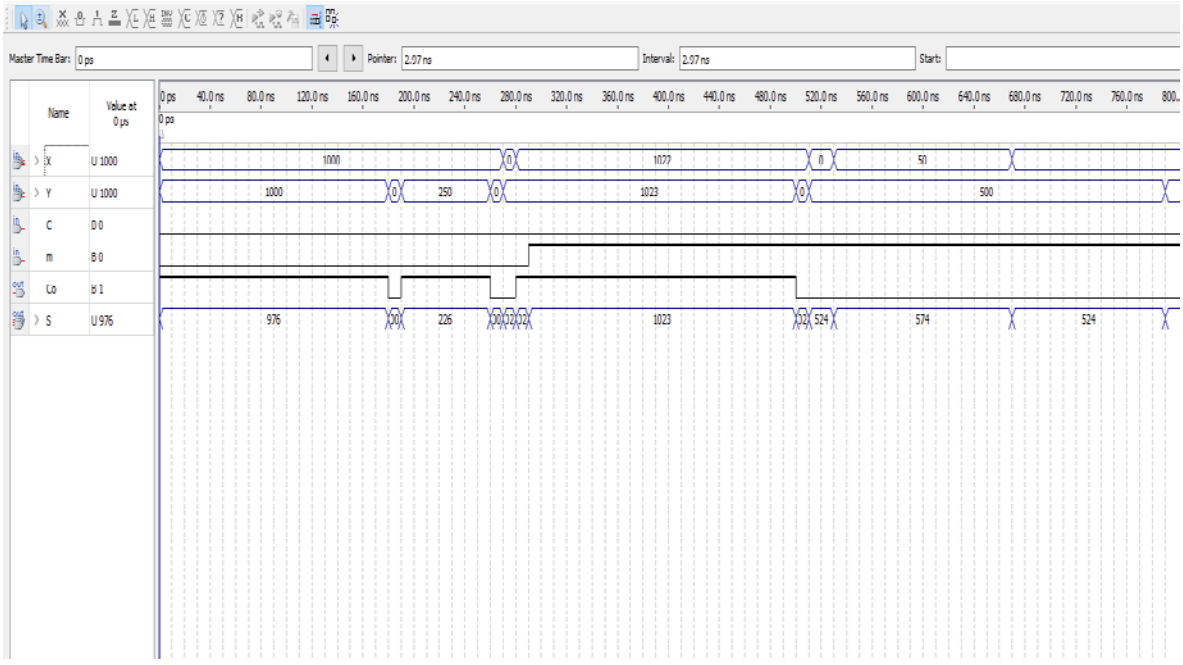
1. 以 2 個 4 位元加法器組成 10 位元加法器



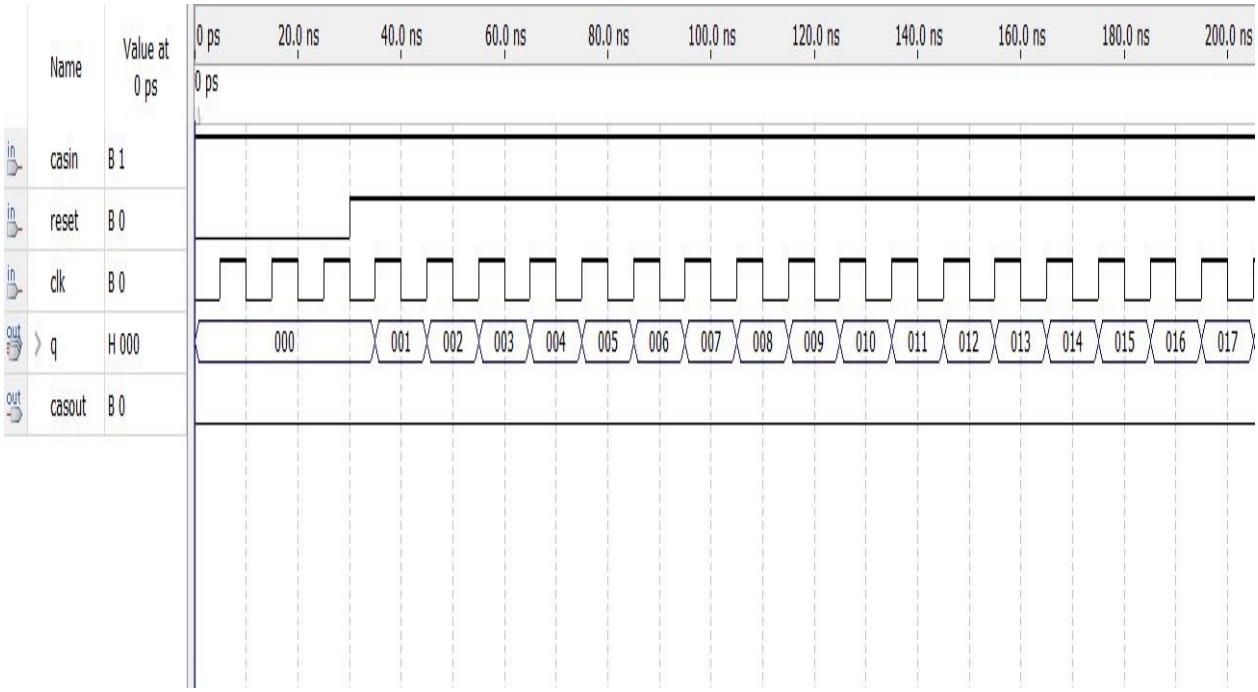

```
1  library ieee;
2  use ieee.std_logic_1164.all;
3  use ieee.std_logic_unsigned.all;
4
5  entity ad10b is
6     port(
7        X:in std_logic_vector(9 downto 0);
8        Y:in std_logic_vector(9 downto 0);
9        C:in std_logic;
10       m:in std_logic;
11       S:out std_logic_vector(9 downto 0);
12       Co:out std_logic);
13 end ad10b;
14
15 architecture ad10b of ad10b is
16 signal tmp:std_logic_vector(2 downto 0);
17 signal c1,c2:std_logic;
18
19 component ad4b is
20    port(
21       X:in std_logic_vector(3 downto 0);
22       Y:in std_logic_vector(3 downto 0);
23       C:in std_logic;
24       m:in std_logic;
25       S:out std_logic_vector(3 downto 0);
26       Co:out std_logic);
27 end component;
28
29 begin
30 U1:ad4b port map(
31                   X=>X(3 downto 0),
32                   Y=>Y(3 downto 0),
33                   C=>C,
34                   m=>m,
35                   S=>S(3 downto 0),
36                   Co=>C1);
37 U2:ad4b port map(
38                   X=>X(7 downto 4),
39                   Y=>Y(7 downto 4),
40                   C=>C1,
41                   m=>m,
42                   S=>S(7 downto 4),
43                   Co=>C2);
44       tmp<=('0' & X(9 downto 8)) + Y(9 downto 8) + C2;
45       S(9 downto 8)<=tmp(1 downto 0);
46       Co<=tmp(2);
47 end ad10b;
```

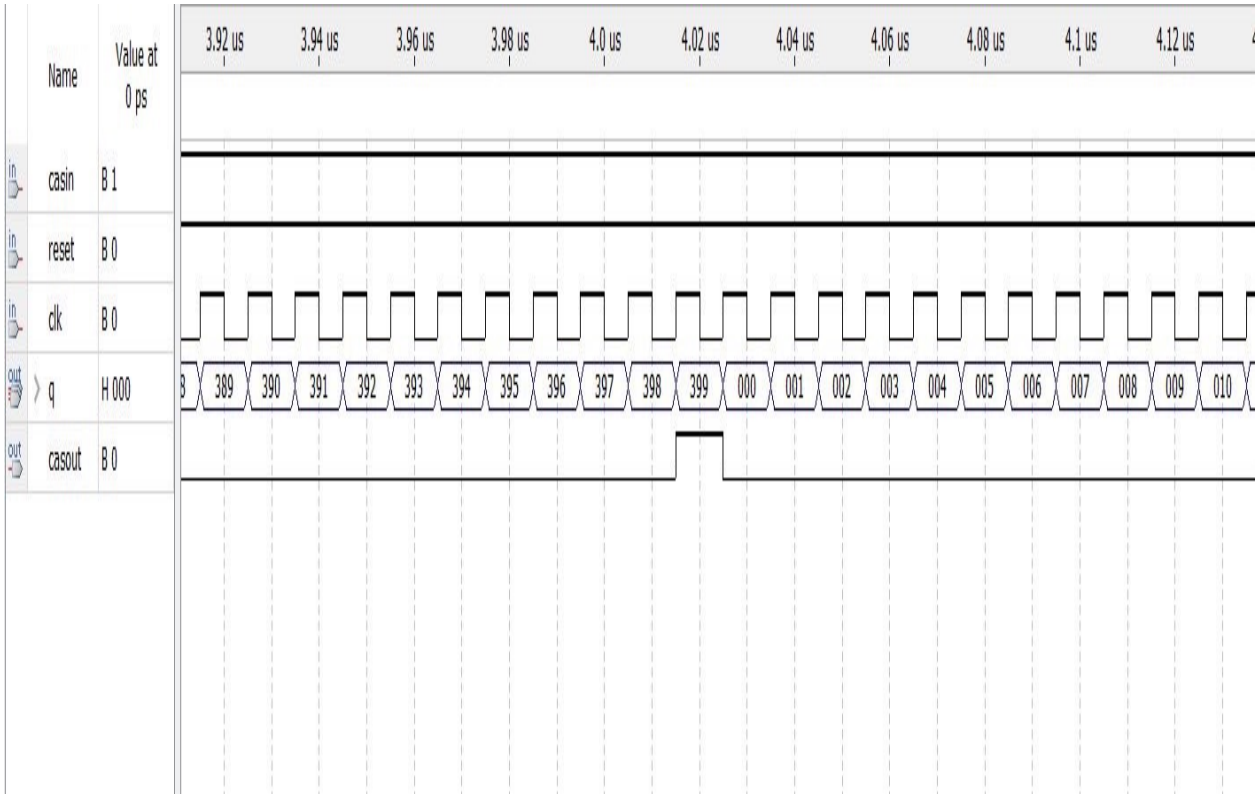


2. Cascade bcd3 counter

```
library ieee;
use ieee.std_logic_1164.all;
use ieee.std_logic_unsigned.all;

entity cscuc3 is
   port(
         clk:in std_logic;
         res:std_logic;
         csin:in std_logic;
         csot:out std_logic;
         q:out std_logic_vector(11 downto 0)
         );
end cscuc3;
architecture cscuc3 of cscuc3 is
signal cnt:std_logic_vector(3 downto 0);
signal tmp:std_logic_vector(7 downto 0);
signal c1,c2:std_logic;

component cscdc is
   port(
         clk:in std_logic;
         res:in std_logic;
         csin:in std_logic;
         csot:out std_logic;
         q:out std_logic_vector(3 downto 0)
         );
end component;

begin
   U1:cscdc port map(
                        clk,
                        res,
                        csin,
                        c1,
                        tmp(3 downto 0)
                  );
   U2:cscdc port map(
                        clk,
                        res,
                        c1,
                        c2,
                        tmp(7 downto 4)
                  );
   process(clk,res)
   begin
      if res='0' then
         cnt<=(others=>'0');
```

```
begin
   U1:cscdc port map(
                     clk,
                     res,
                     csin,
                     c1,
                     tmp(3 downto 0)
               );
   U2:cscdc port map(
                     clk,
                     res,
                     c1,
                     c2,
                     tmp(7 downto 4)
               );
   process(clk,res)
   begin
      if res='0' then
         cnt<=(others=>'0');
      elsif clk'event and clk = '1' then
         if c2 = '1' then
            if cnt<="0011" then
               cnt<="0000";
            else
               cnt<=cnt+1;
            end if;
         end if;
      end if;
   end process;
   q(11 downto 8)<=cnt;
   q(7 downto 0)<=tmp;
   csot<='1'when csin = '1' and tmp(7 downto 0)="10011001" else
         '0';
end cscuc3;
```


Name
Value at 0 ps
0 ps
20.0 ns
40.0 ns
60.0 ns
80.0 ns
100.0 ns
120.0 ns
140.0 ns
160.0 ns
180.0 ns
200.0 ns
casin
B 1
reset
B 0
clk
B 0
q
H 000
000
001
002
003
004
005
006
007
008
009
010
011
012
013
014
015
016
017
casout
B 0


Name
Value at 0 ps
3.92 us
3.94 us
3.96 us
3.98 us
4.0 us
4.02 us
4.04 us
4.06 us
4.08 us
4.1 us
4.12 us
casin
B 1
reset
B 0
clk
B 0
q
H 000
389
390
391
392
393
394
395
396
397
398
399
000
001
002
003
004
005
006
007
008
009
010
casout
B 0