7)

**What causes "accidental" transparent latches?**

A transparent latch is a storage element. It has an input, an output, and an enable or gate pin. When the enable is active, the output transparently follows the input (with some small delay). When the enable becomes inactive, the output freezes.

Transparent latches will appear if you write a combinational process or always block where an output is not assigned under all possible input conditions. In other words, it is possible for one of the inputs to change without affecting the output. In synthesis jargon, this is known as *incomplete assignment*.

**یکی از عناصر حافظه لچ است. تفاوت آن با فلیپ فلاپ در نحوه فعال شدن آن ها و انتقال ورودی داده به خروجی است.**

**فلیپ فلاپ ها با لبه ی سیگنال کلاک فعال میشوند و ورودی را در خروجی ظاهر میکنند و فورا خروجی دریافت شده را حفظ میکنند و تا لبه ی بعدی گلال آنرا تغییر نمیدهند.**

**اما لچ حساس به سطح سیگنال است و مادامی که سیگنال خاص فعال باشد، مقدار ورودی در خروجی منعکس میشود.**

**برخی اوقات در توصیف ما یک سری لچ های ناخواسته تولید میشود مانند وقتی که یک شرط بررسی نشود و حالت های اضافی بی سر و سامان باقی بمانند.(وجود شرط های ناقص در توصیف)**

process (sel, sel\_2, sel\_3, a, b)

begin

if sel = '1' then

f <= a ;

if sel\_2 = '1' then

g <= not a ;

else

g <= not b ;

if sel\_3 = '1' then

g <= a xor b ; 🡺 if without else : transparent latch

end if ;

end if ;

else 🡺 for that if!

if sel\_2 = '1' then

g <= a and b ;

else

if sel\_3 = '1' then

g <= a nand b; 🡺 if without else : transparent latch

end if ;

end if;

f <= b ;

end if ;

end process;

کد تغییر یافته:

process (sel, sel\_2, sel\_3, a, b)

begin

if sel = '1' then

f <= a ;

if sel\_2 = '1' then

g <= not a ;

else

g <= not b ;

if sel\_3 = '1' then

g <= a xor b ; 🡺 if without else : transparent latch

else

g<=’U’;

end if ;

end if ;

else 🡺 for that if!

if sel\_2 = '1' then

g <= a and b ;

else

if sel\_3 = '1' then

g <= a nand b; 🡺 if without else : transparent latch

else

g<=’U’;

end if ;

end if;

f <= b ;

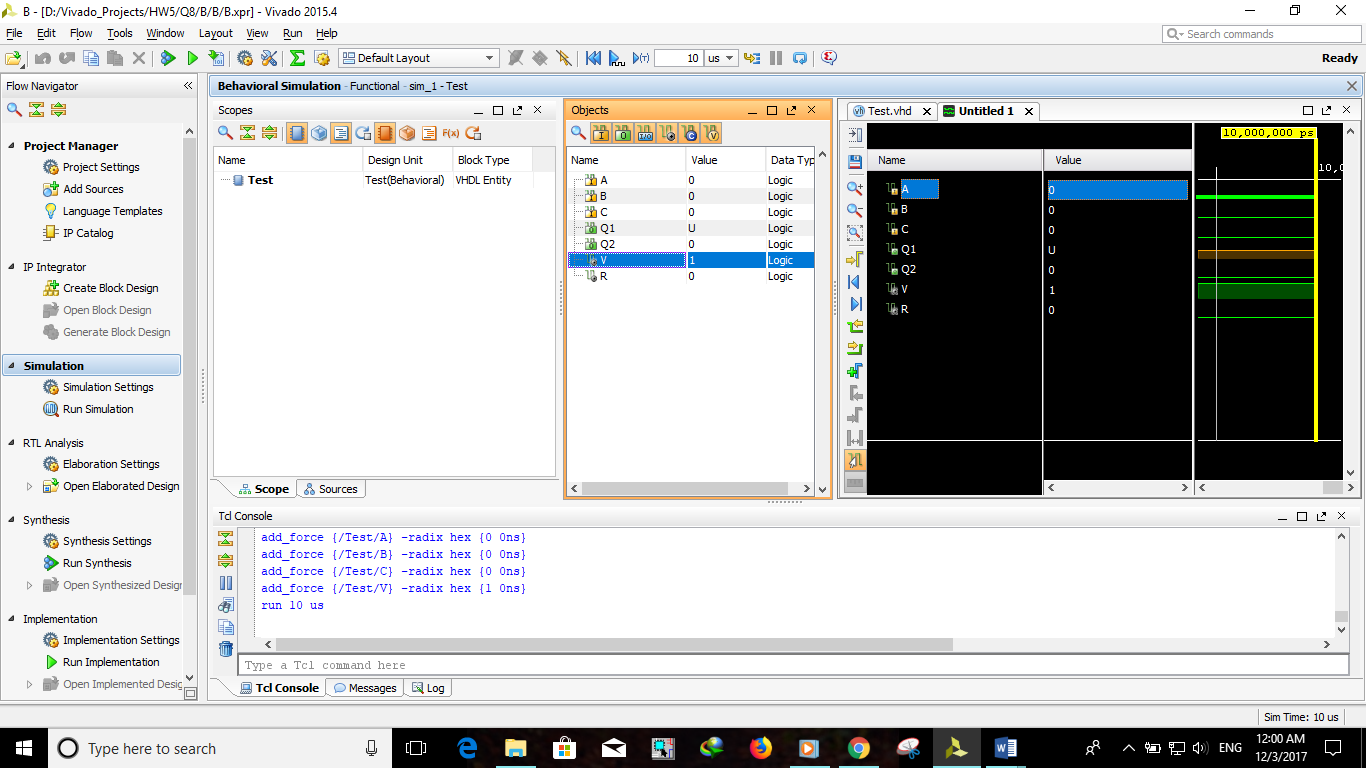
end if ;

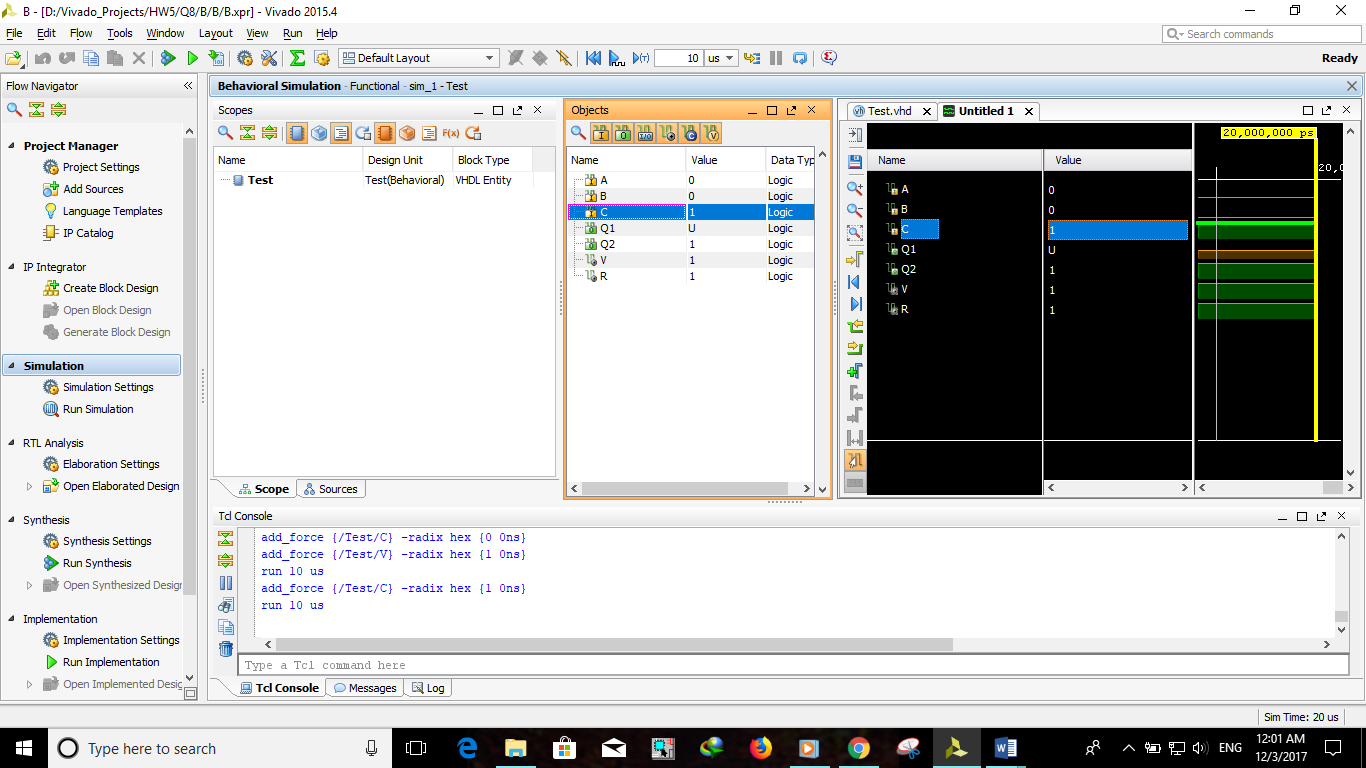
end process;

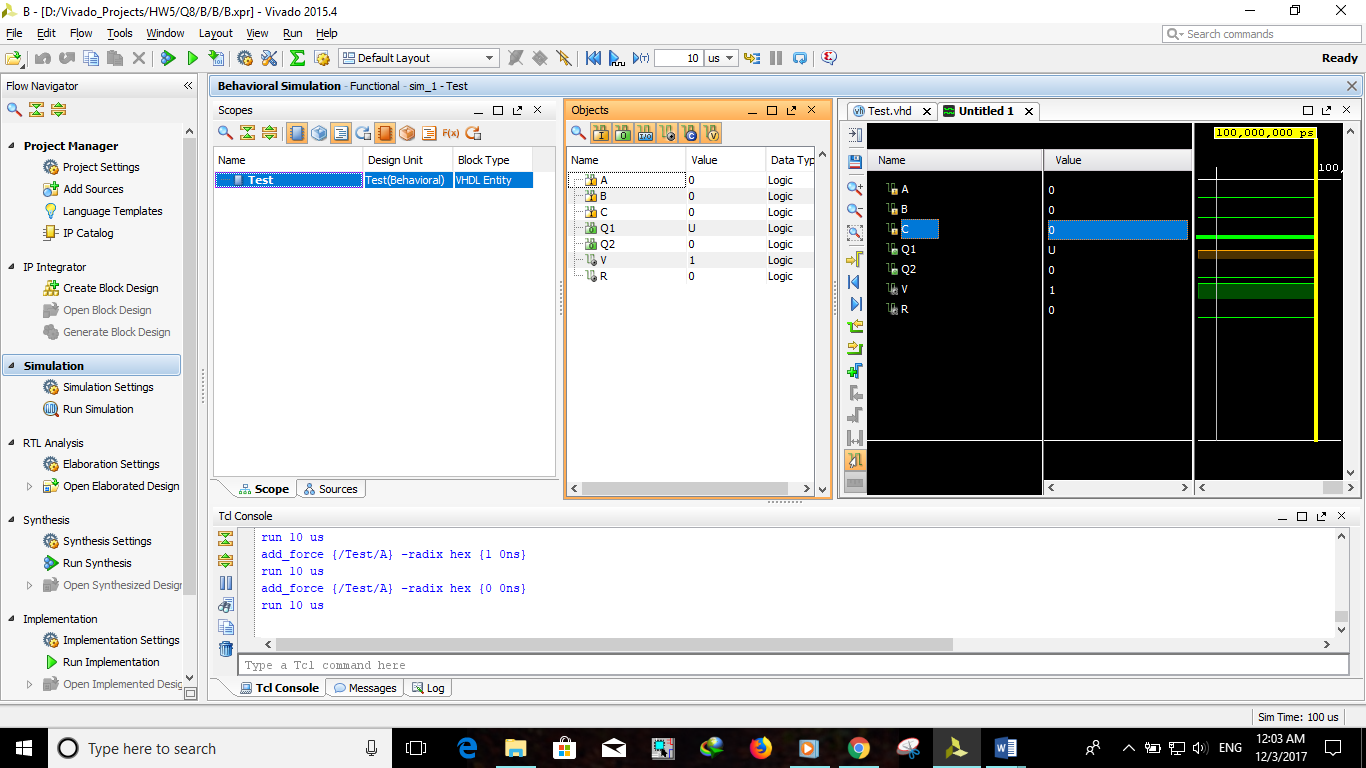
8)

a)

b)







مشکل این است که q1 مقدار نمیگیرد