ASSIGNMENT – II

Name: Soumyadip Ghosh

Stream: CSE- A

Roll Number: 1951007

Subject: Computer Architecture Lab

OR GATE

Behavioral Model:

library IEEE;

use IEEE.STD\_LOGIC\_1164.ALL;

entity or\_bv is

Port ( A : in STD\_LOGIC;

B : in STD\_LOGIC;

C : out STD\_LOGIC);

end or\_bv;

architecture Behavioral of or\_bv is

begin

process(a,b)

begin

if(a='0')and(b='0')then

c<='0';

else

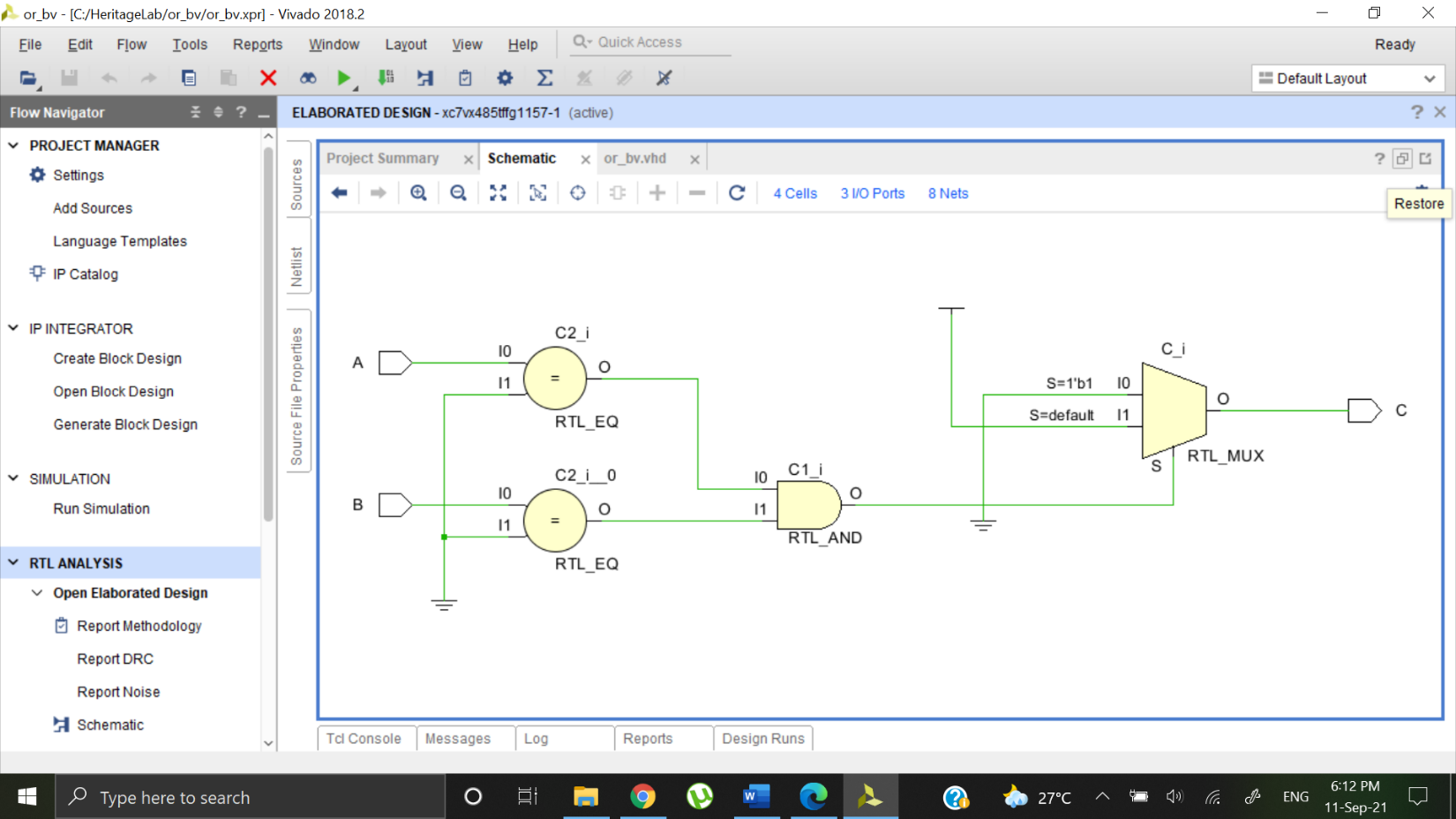
c<='1';

end if;

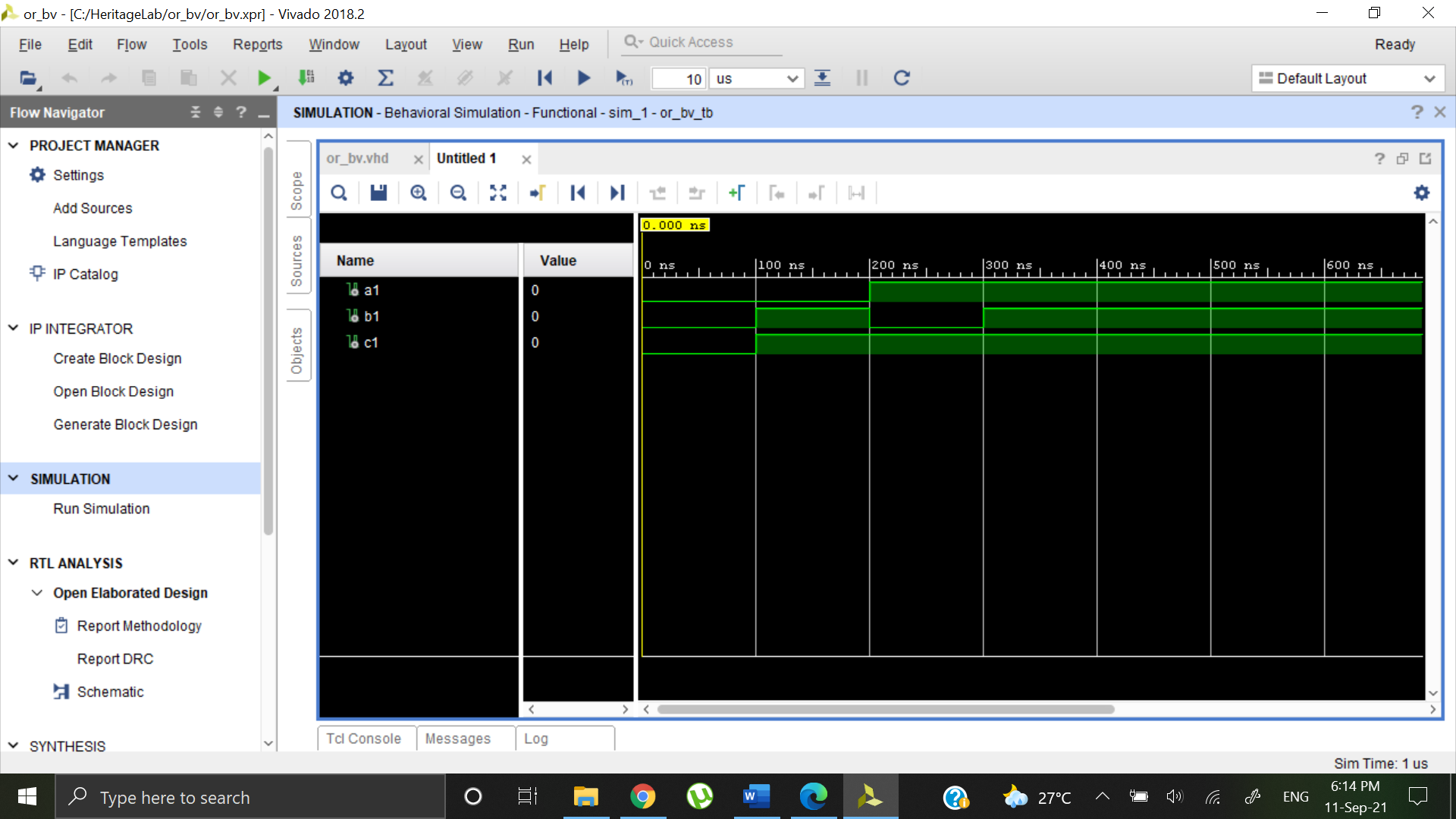
end process;

end Behavioral;

Elaborated Design:



Output Waveform:



AND GATE

Behavioral Model:

library IEEE;

use IEEE.STD\_LOGIC\_1164.ALL;

entity and\_bv is

Port ( A : in STD\_LOGIC;

B : in STD\_LOGIC;

C : out STD\_LOGIC);

end and\_bv;

architecture Behavioral of and\_bv is

begin

process(a,b)

begin

if(a='1')and(b='1')then

c<='1';

else

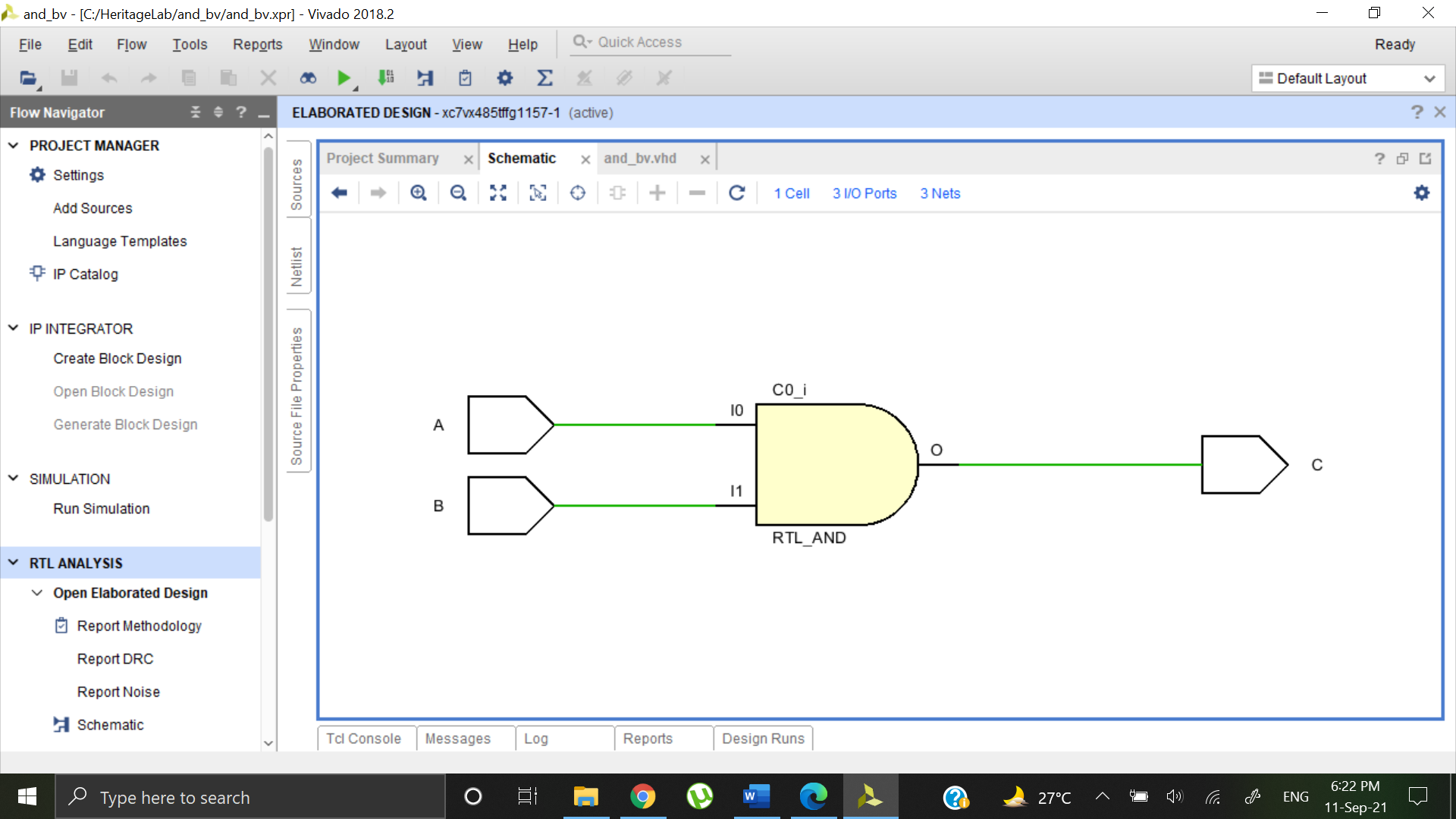
c<='0';

end if;

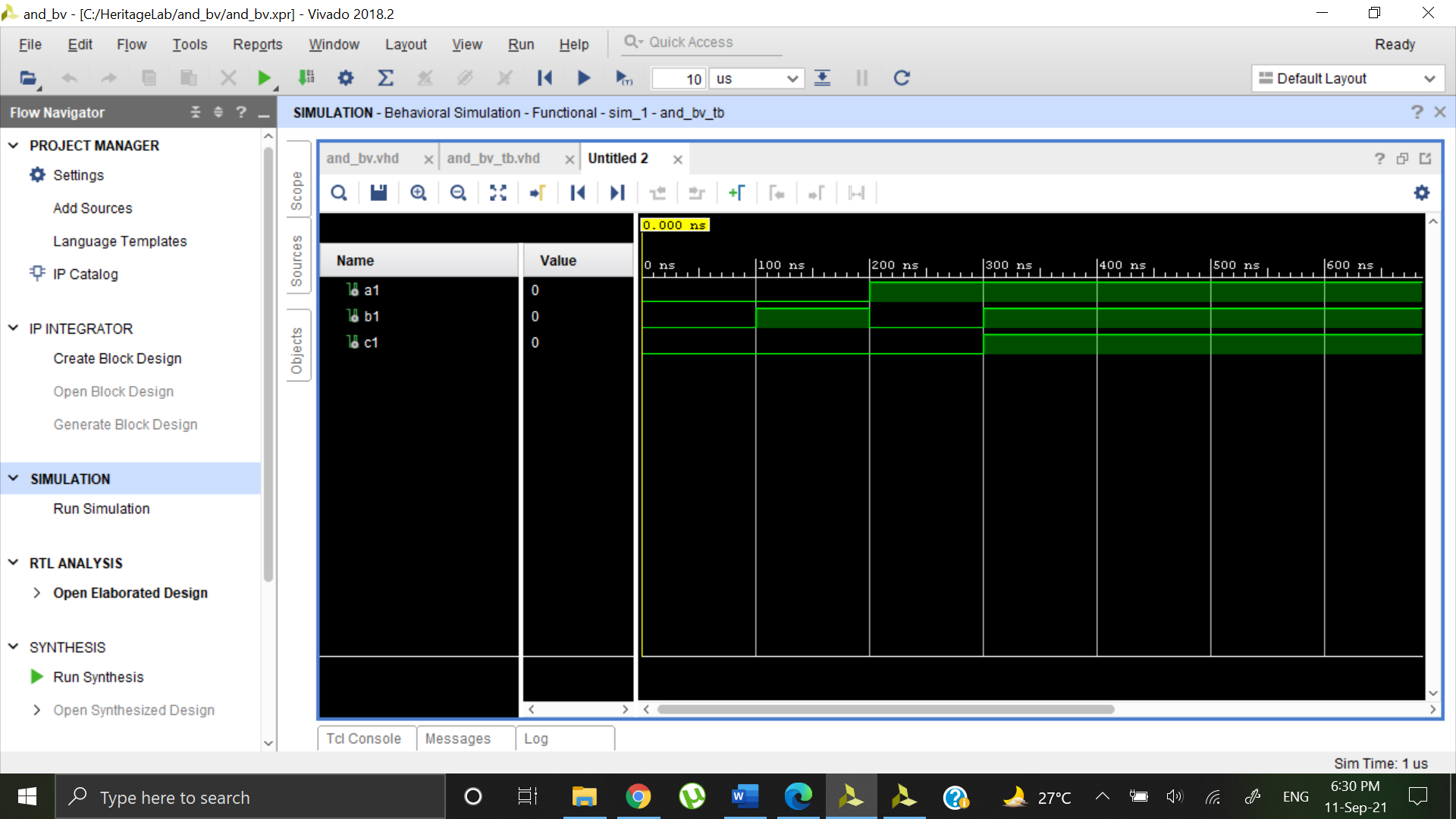
end process;

end Behavioral;

Elaborated Design:



Output Waveform:



NAND GATE

Behavioral Model:

library IEEE;

use IEEE.STD\_LOGIC\_1164.ALL;

entity nand\_bv is

Port ( A : in STD\_LOGIC;

B : in STD\_LOGIC;

C : out STD\_LOGIC);

end nand\_bv;

architecture Behavioral of nand\_bv is

begin

process(a,b)

begin

if(a='1')and(b='1')then

c<='0';

else

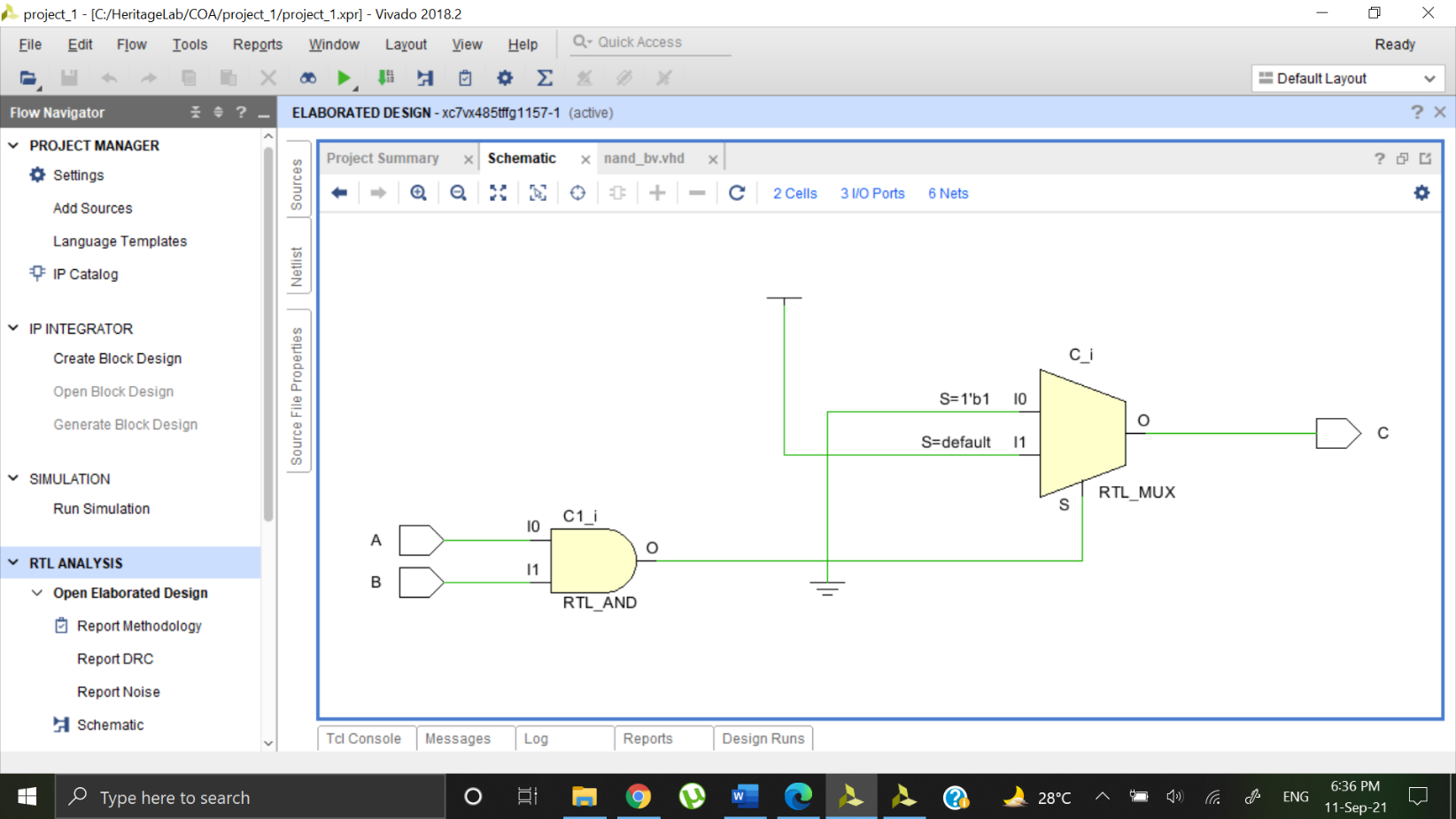
c<='1';

end if;

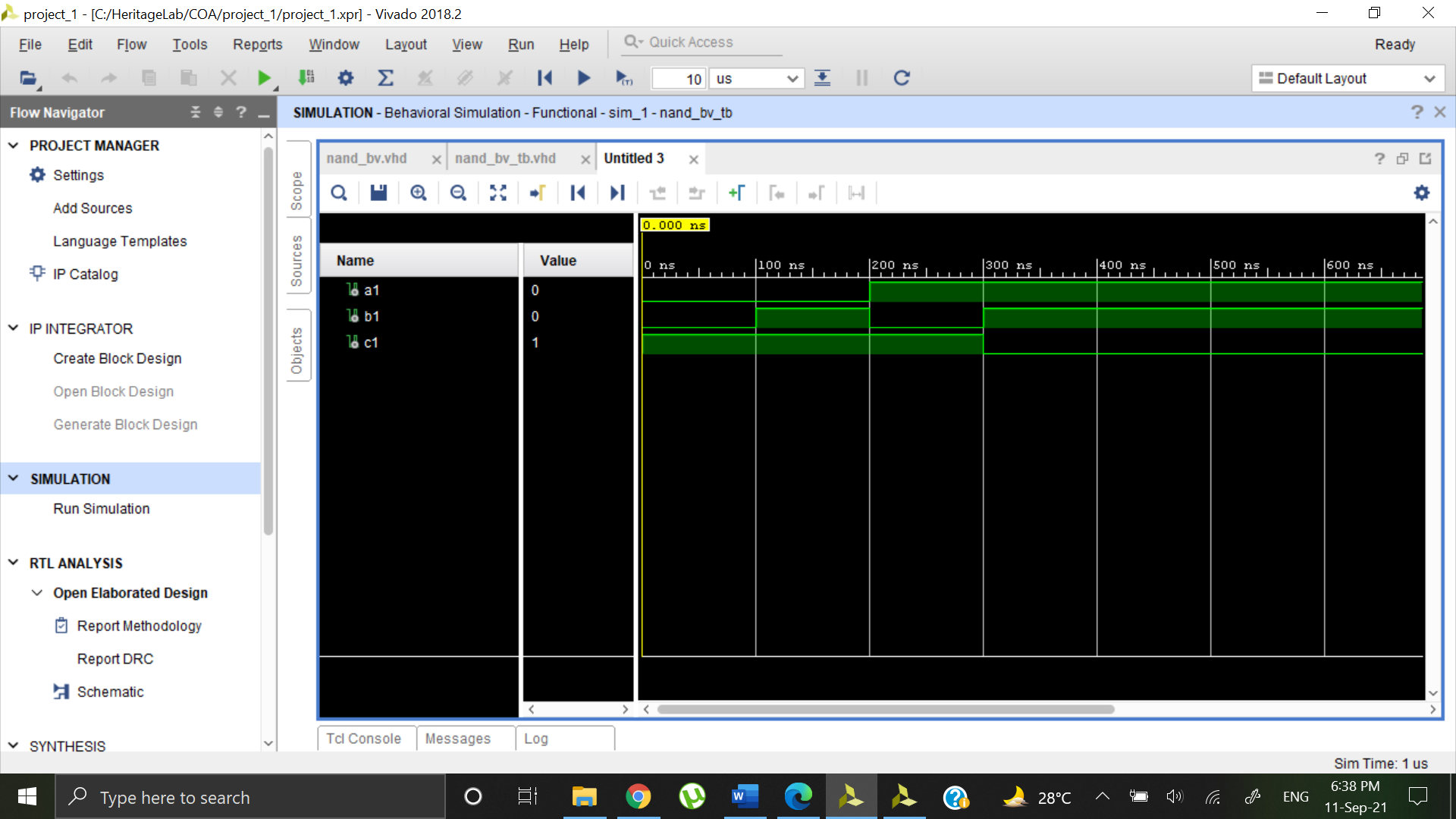
end process;

end Behavioral;

Elaborated Design:



Output Waveform:



NOR GATE

Behavioral Model:

library IEEE;

use IEEE.STD\_LOGIC\_1164.ALL;

entity nor\_bv is

Port ( A : in STD\_LOGIC;

B : in STD\_LOGIC;

C : out STD\_LOGIC);

end nor\_bv;

architecture Behavioral of nor\_bv is

begin

process(a,b)

begin

if(a='0')and(b='0')then

c<='1';

else

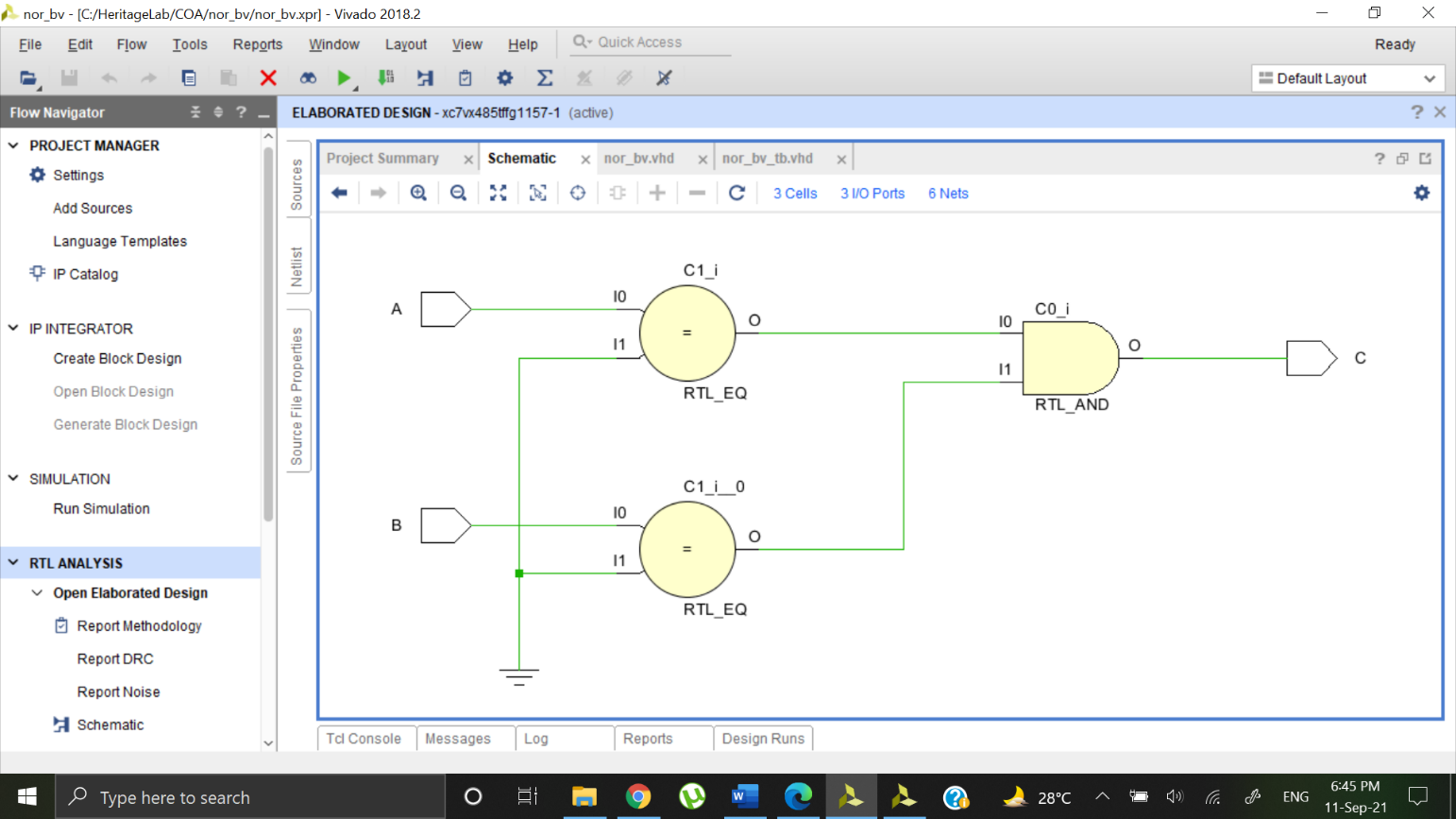
c<='0';

end if;

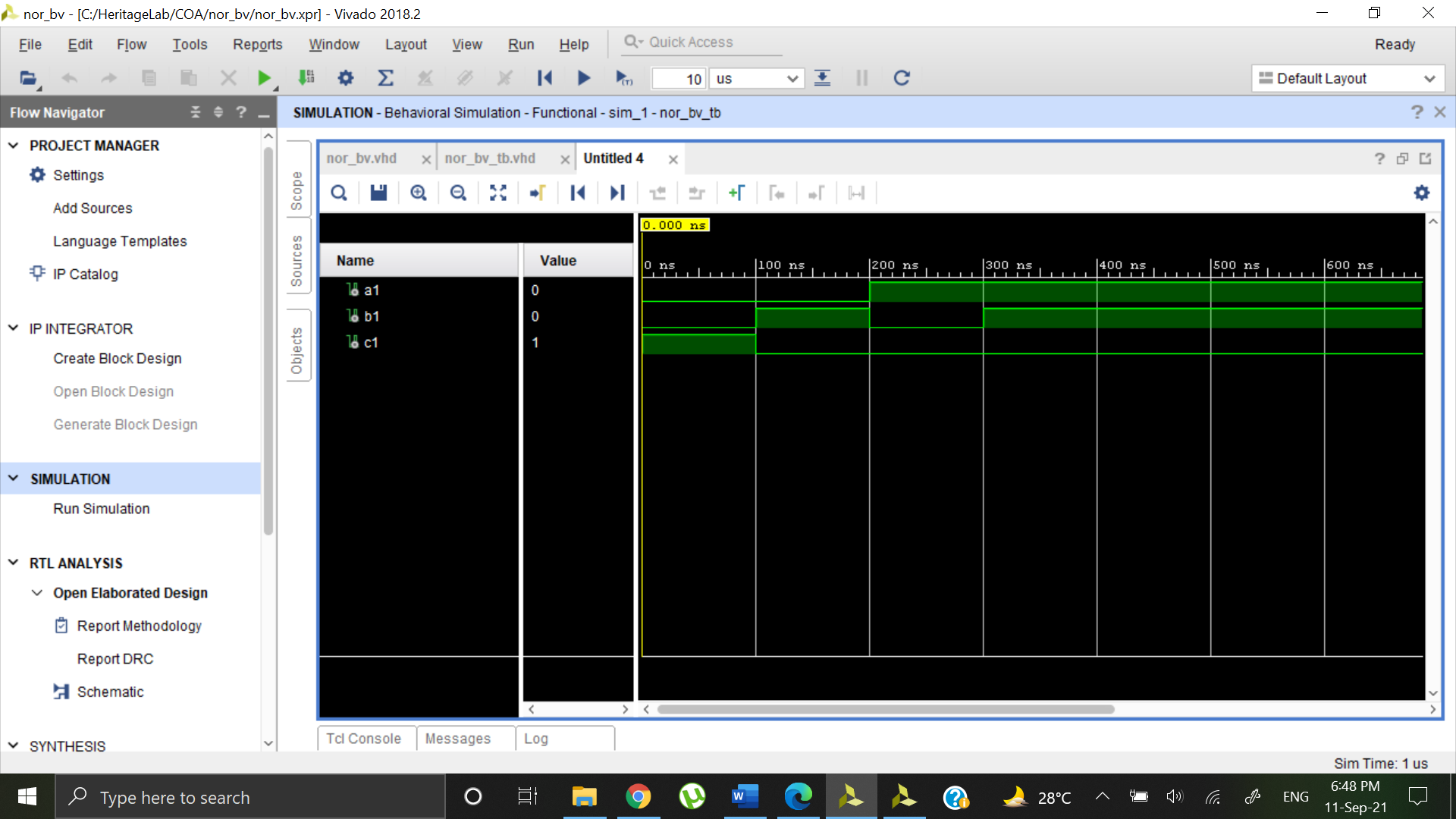
end process;

end Behavioral;

Elaborated Design:



Output Waveform:



XOR GATE

Behavioral Model:

library IEEE;

use IEEE.STD\_LOGIC\_1164.ALL;

entity xor\_bv is

Port ( A : in STD\_LOGIC;

B : in STD\_LOGIC;

C : out STD\_LOGIC);

end xor\_bv;

architecture Behavioral of xor\_bv is

begin

process(a,b)

begin

if(a/=b)then

c<='1';

else

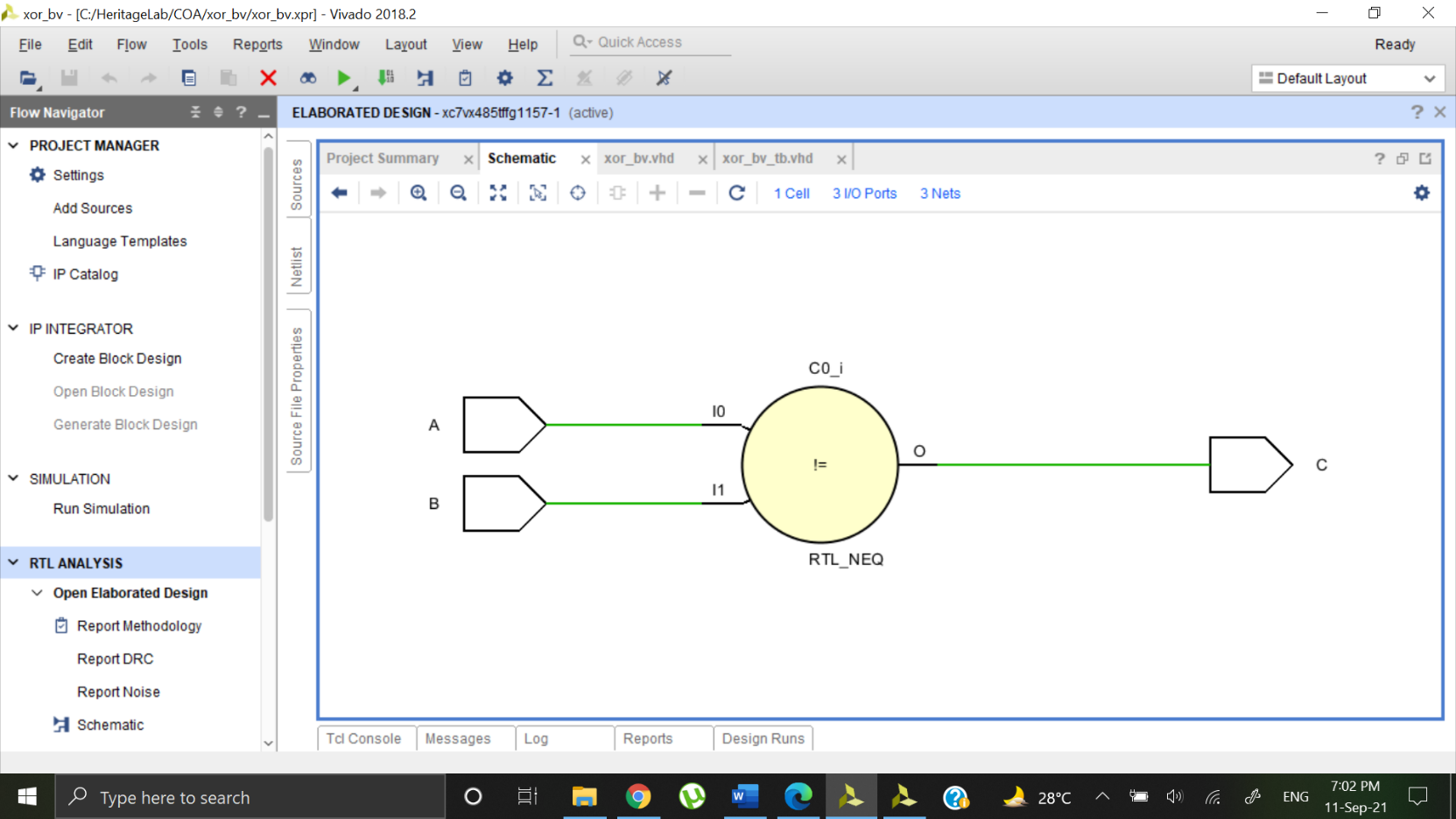
c<='0';

end if;

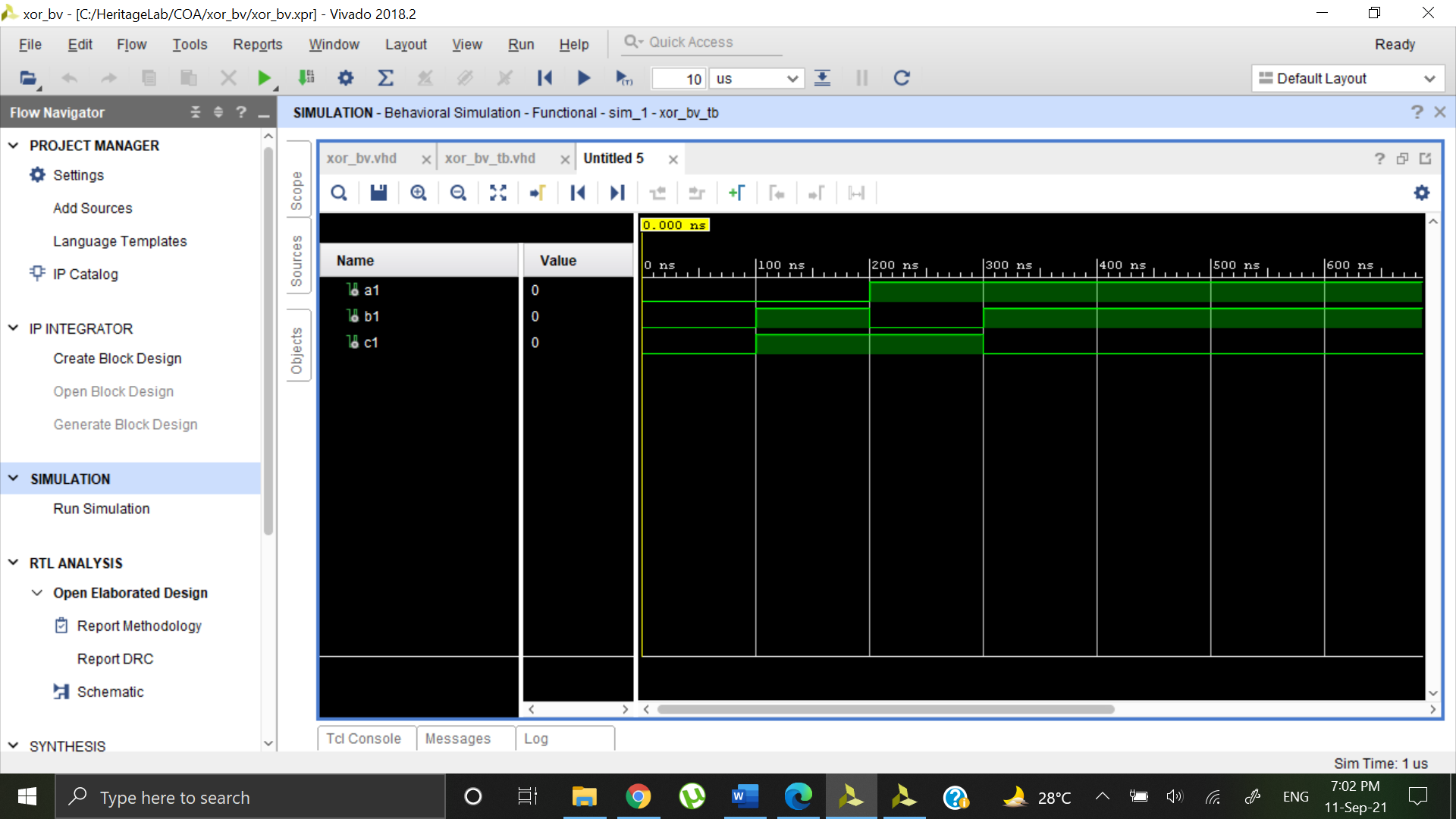
end process;

end Behavioral;

Elaborated Design:



Output Waveform



NOT GATE

Behavioral Model:

library IEEE;

use IEEE.STD\_LOGIC\_1164.ALL;

entity not\_bv is

Port ( A : in STD\_LOGIC;

B : out STD\_LOGIC);

end not\_bv;

architecture Behavioral of not\_bv is

begin

process(a)

begin

if(a='0')then

b<='1';

else

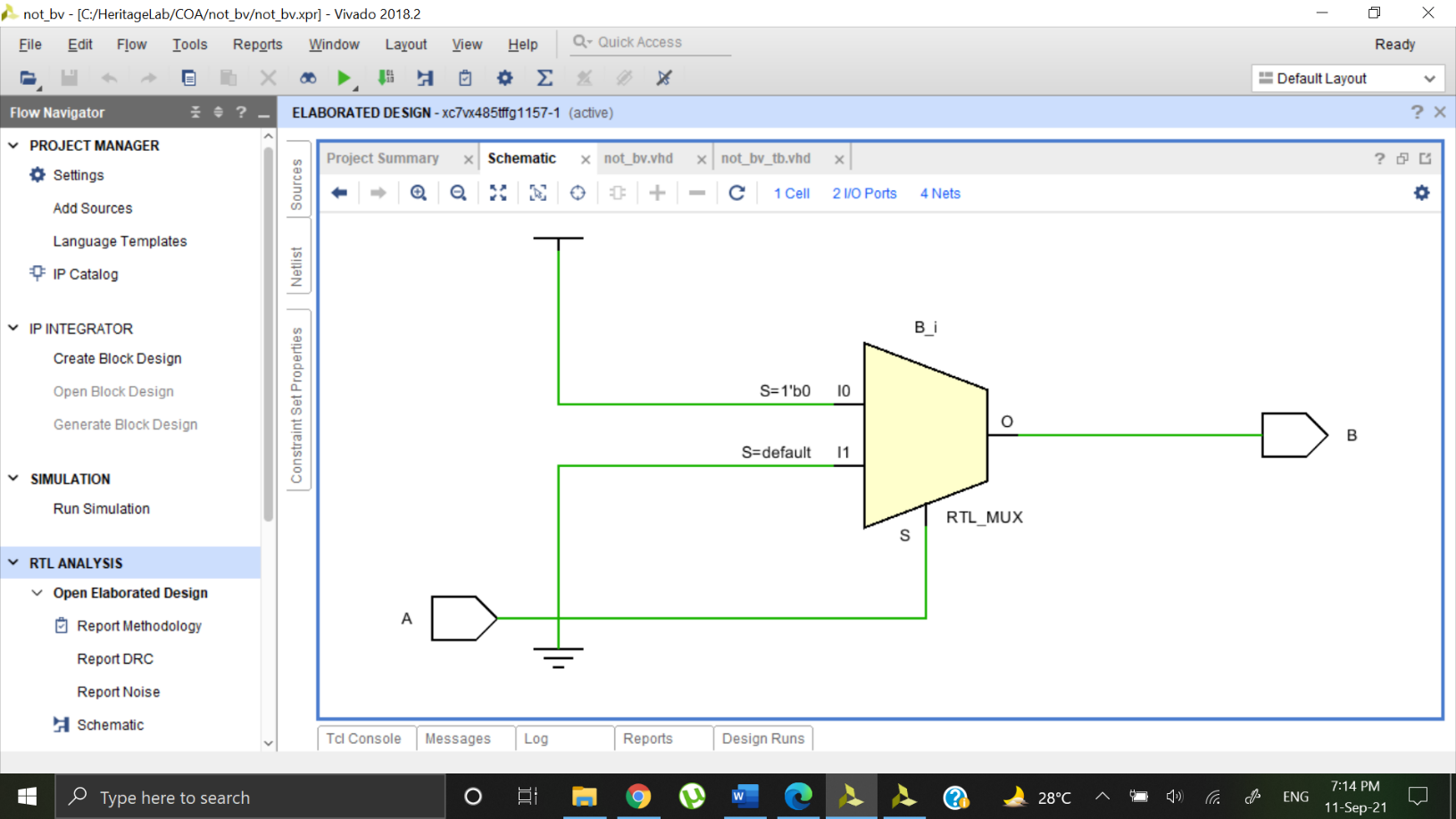
b<='0';

end if;

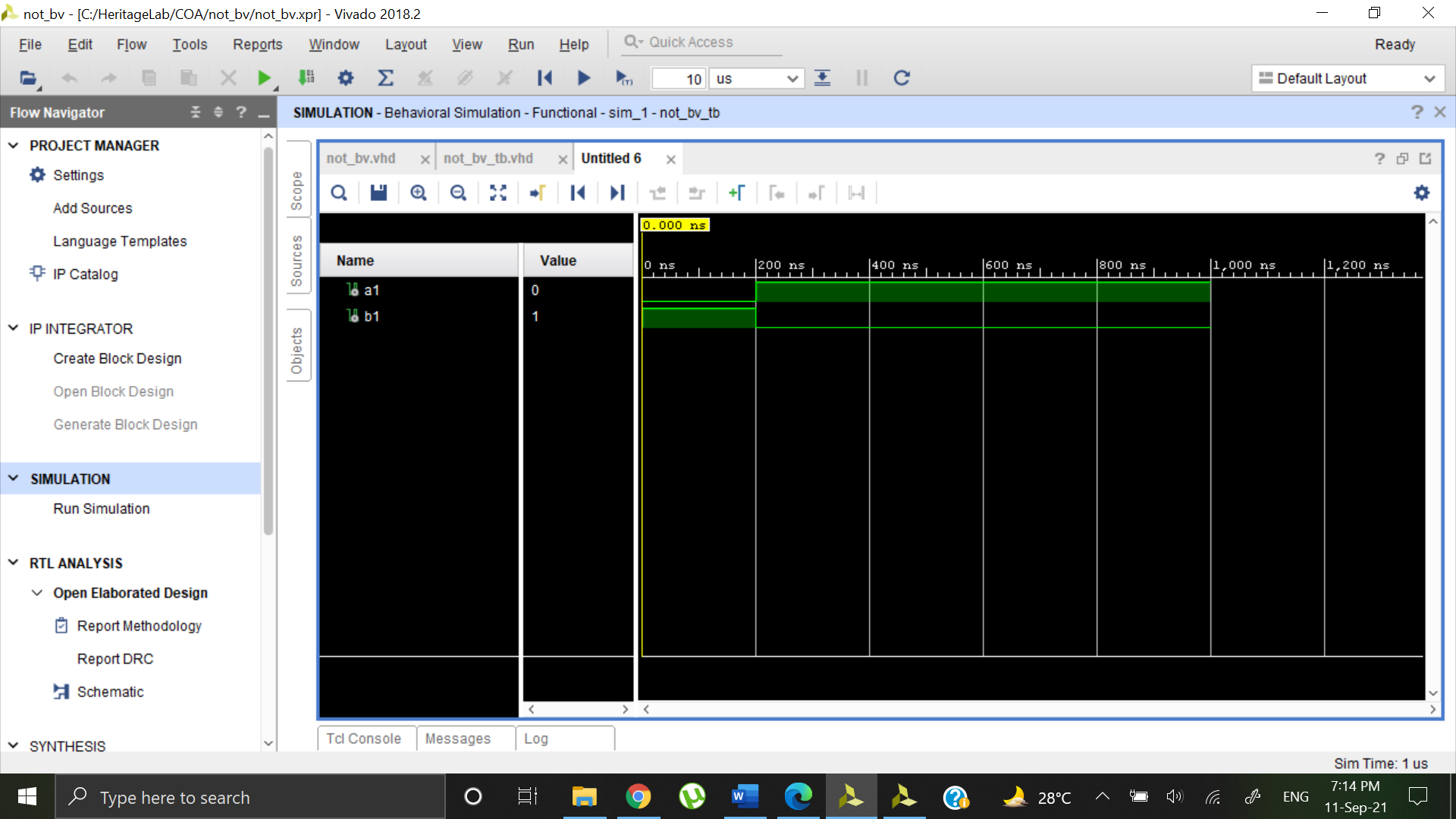
end process;

end Behavioral;

Elaborated Design:



Output Waveform:



HALF ADDER

Behavioral Model:

library IEEE;

use IEEE.STD\_LOGIC\_1164.ALL;

entity halfad\_bv is

Port ( A : in STD\_LOGIC;

B : in STD\_LOGIC;

S : out STD\_LOGIC;

C : out STD\_LOGIC);

end halfad\_bv;

architecture Behavioral of halfad\_bv is

begin

process(A,B)

begin

if(A=B)then

S<='0';

else

S<='1';

end if;

if(A='1')and(B='1')then

C<='1';

else

C<='0';

end if;

end process;

end Behavioral;

Dataflow Model:

library IEEE;

use IEEE.STD\_LOGIC\_1164.ALL;

entity halfad\_df is

Port ( a : in STD\_LOGIC;

b : in STD\_LOGIC;

s : out STD\_LOGIC;

c : out STD\_LOGIC);

end halfad\_df;

architecture Behavioral of halfad\_df is

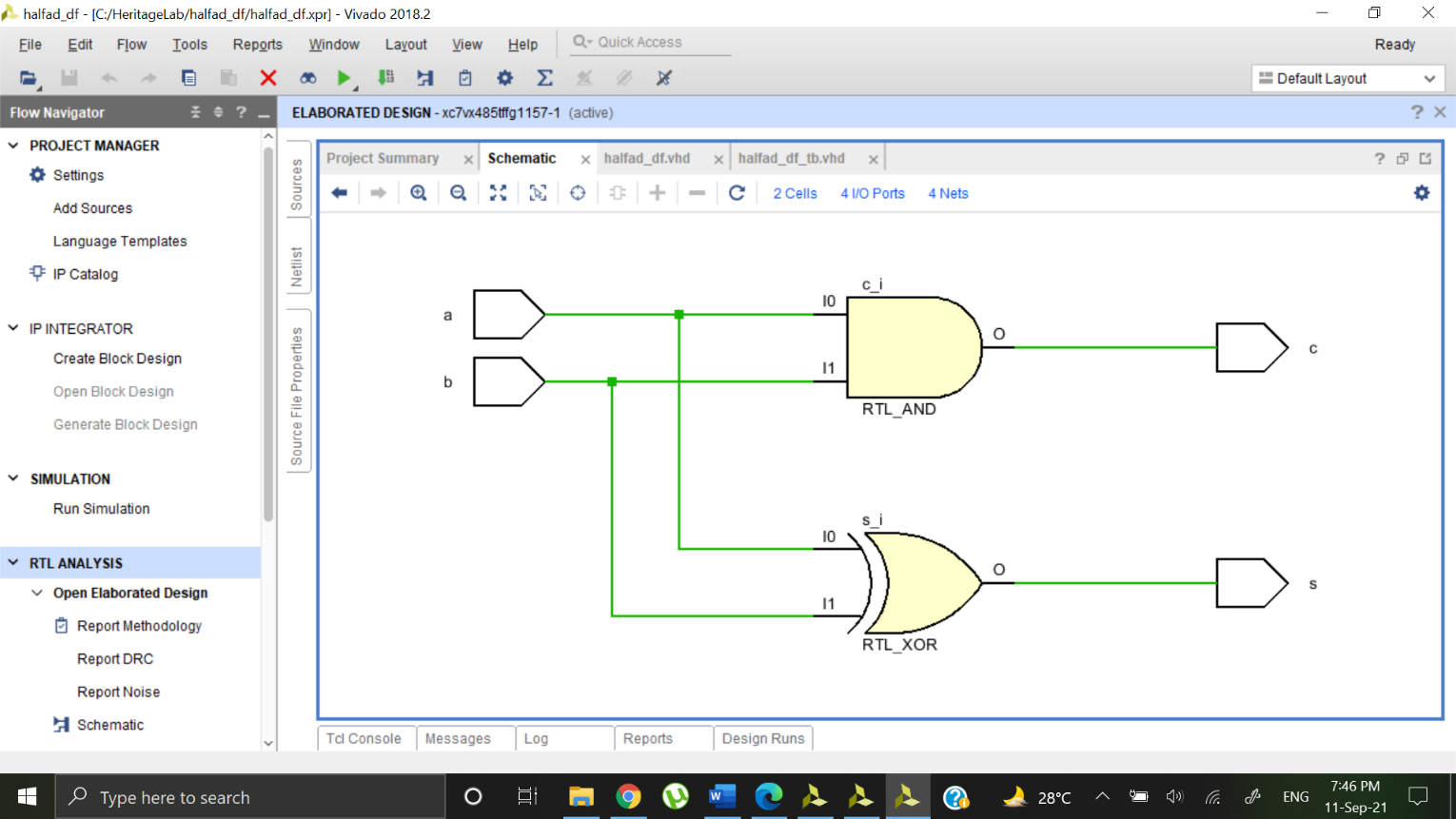
begin

s <= a xor b;

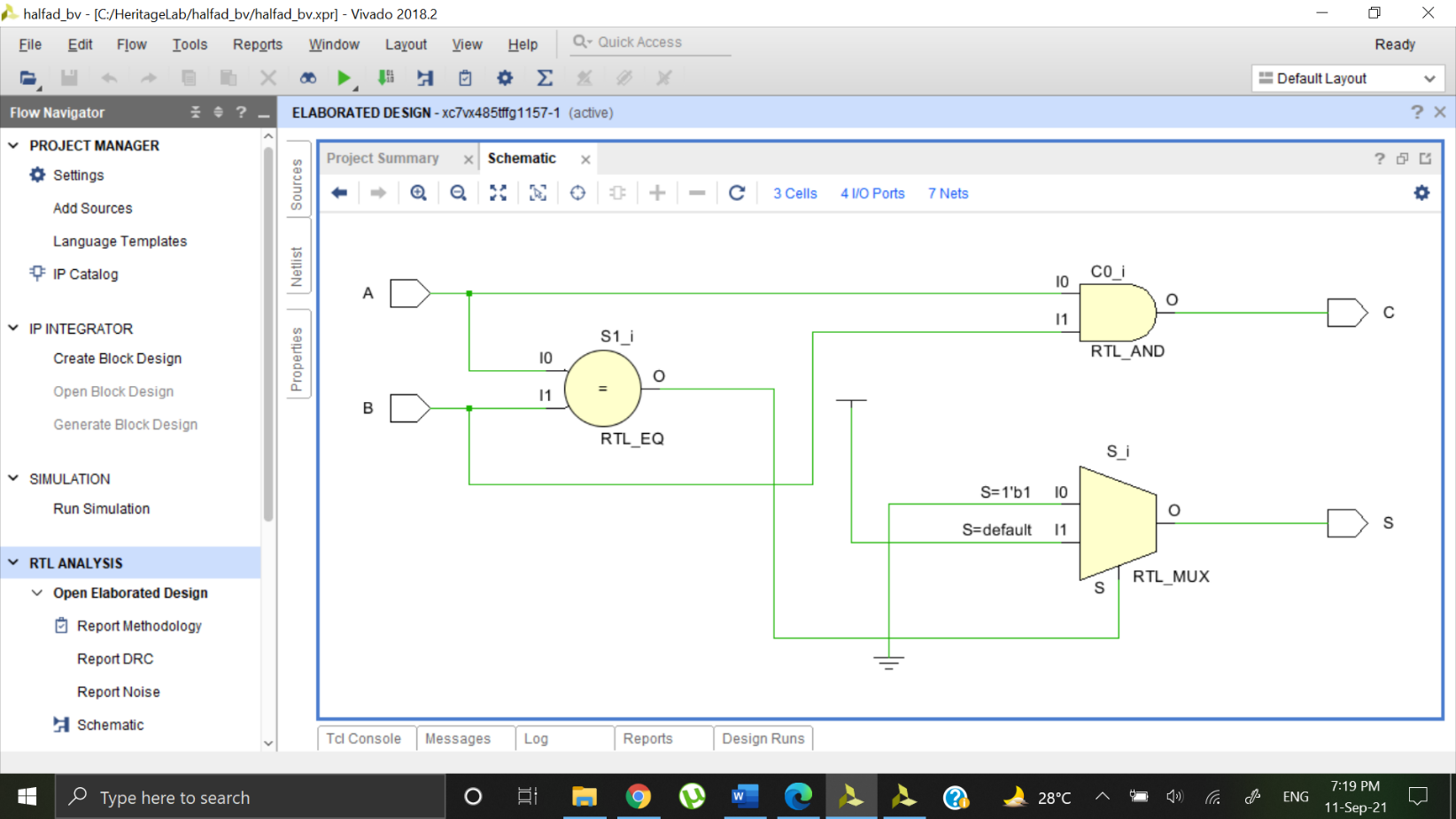
c <= a and b;

end Behavioral;

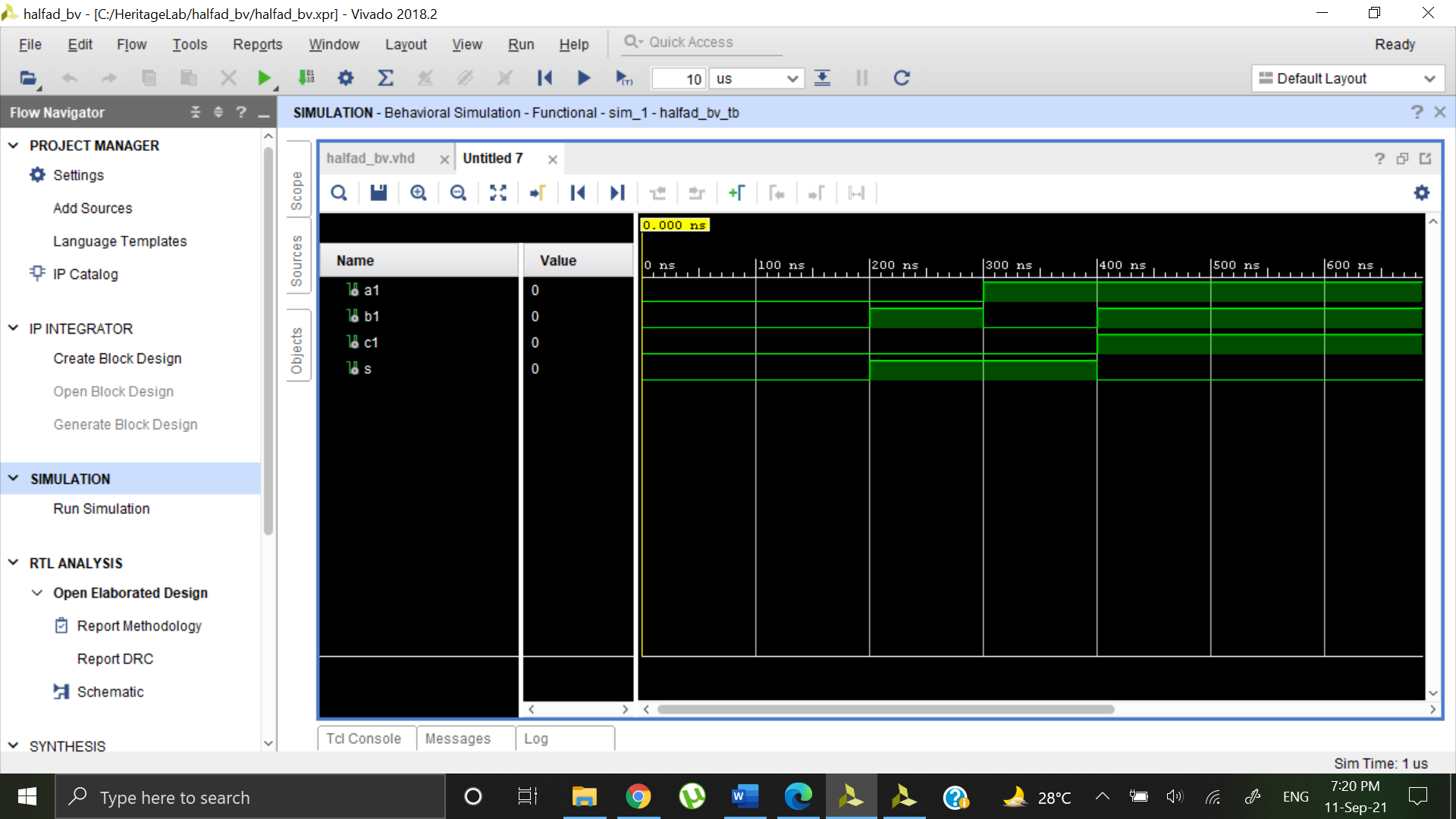
Elaborated Design (Dataflow Model):



Elaborated Design (Behavioral Model):



Output Waveform:



FULL ADDER

Behavioral Model:

library IEEE;

use IEEE.STD\_LOGIC\_1164.ALL;

entity fullad\_bv is

Port ( A : in STD\_LOGIC;

B : in STD\_LOGIC;

Cin : in STD\_LOGIC;

S : out STD\_LOGIC;

Cout : out STD\_LOGIC);

end fullad\_bv;

architecture Behavioral of fullad\_bv is

begin

process(A,B,Cin)

begin

if(A='0')then

if(B=Cin) then

S<='0';

else

S<='1';

end if;

if(B='1' and Cin='1')then

Cout<='1';

else

Cout<='0';

end if;

end if;

if(A='1')then

if(B=Cin)then

S<='1';

else

S<='0';

end if;

if(B='0' and Cin='0')then

Cout<='0';

else

Cout<='1';

end if;

end if;

end process;

end Behavioral;

Dataflow Model:

library IEEE;

use IEEE.STD\_LOGIC\_1164.ALL;

entity fullad\_bv is

Port ( a : in STD\_LOGIC;

b : in STD\_LOGIC;

cin : in STD\_LOGIC;

s : out STD\_LOGIC;

cout : out STD\_LOGIC);

end fullad\_bv;

architecture Behavioral of fullad\_bv is

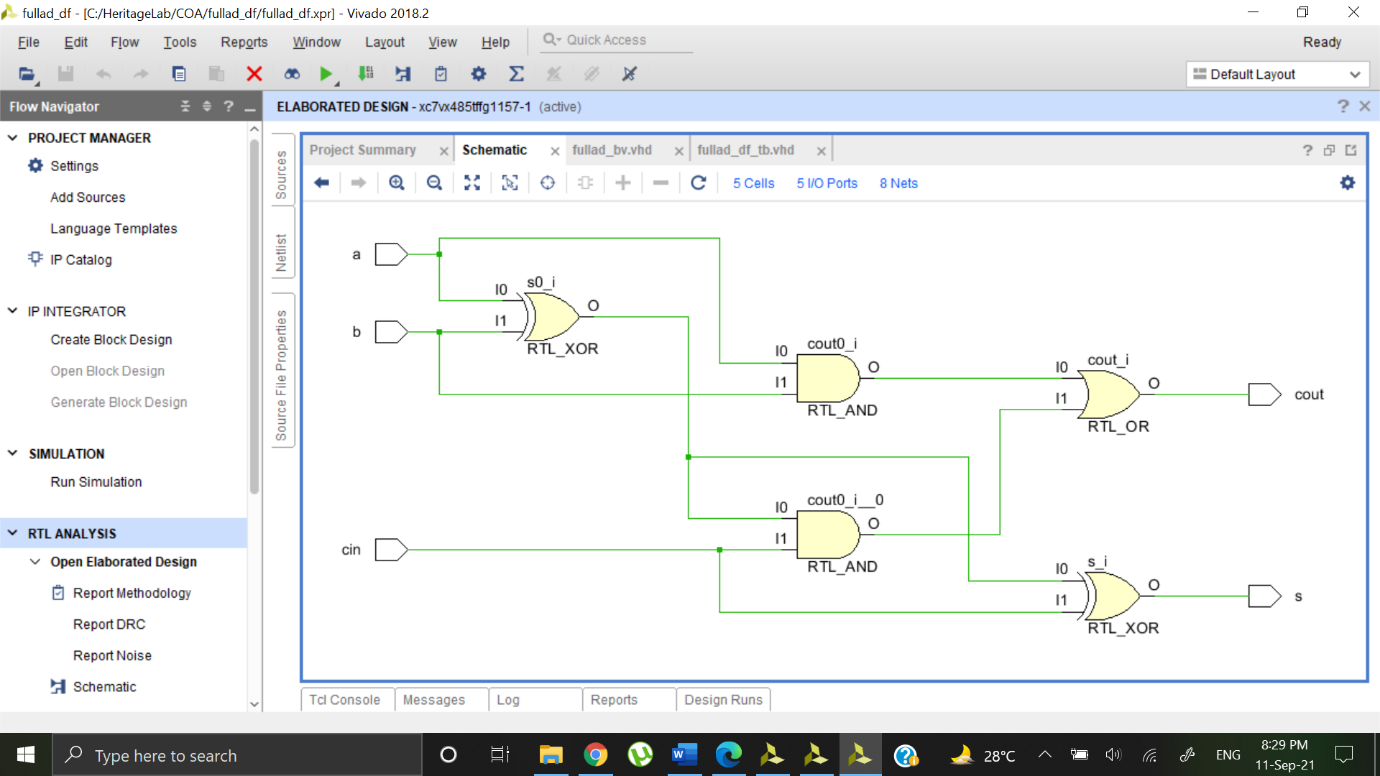
begin

s <= a xor b xor cin;

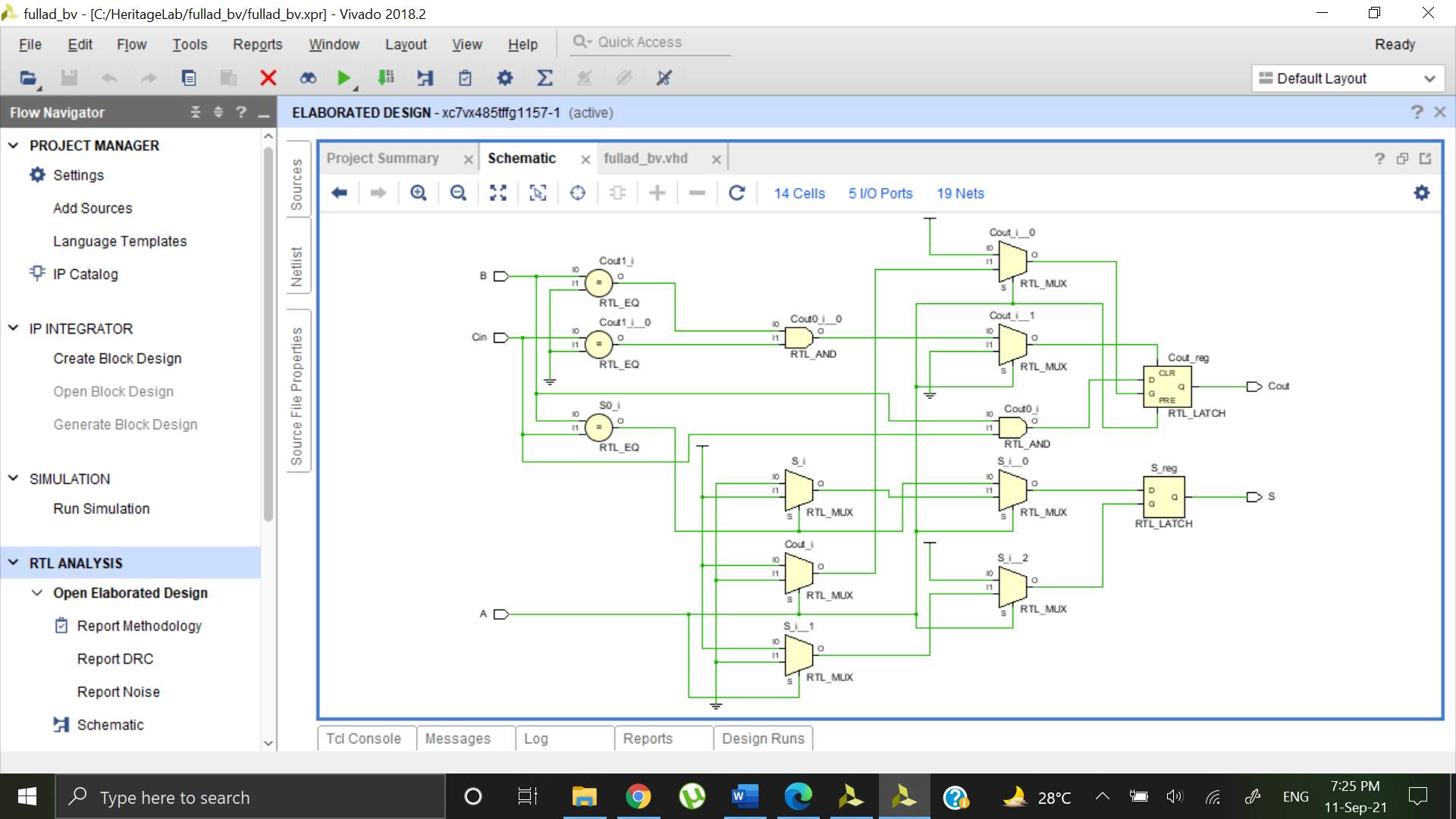
cout <= (a and b) or ((a xor b) and cin);

end Behavioral;]

Elaborated Design (Dataflow Model):



Elaborated Design (Behavioral Model):



Output Waveform:

