

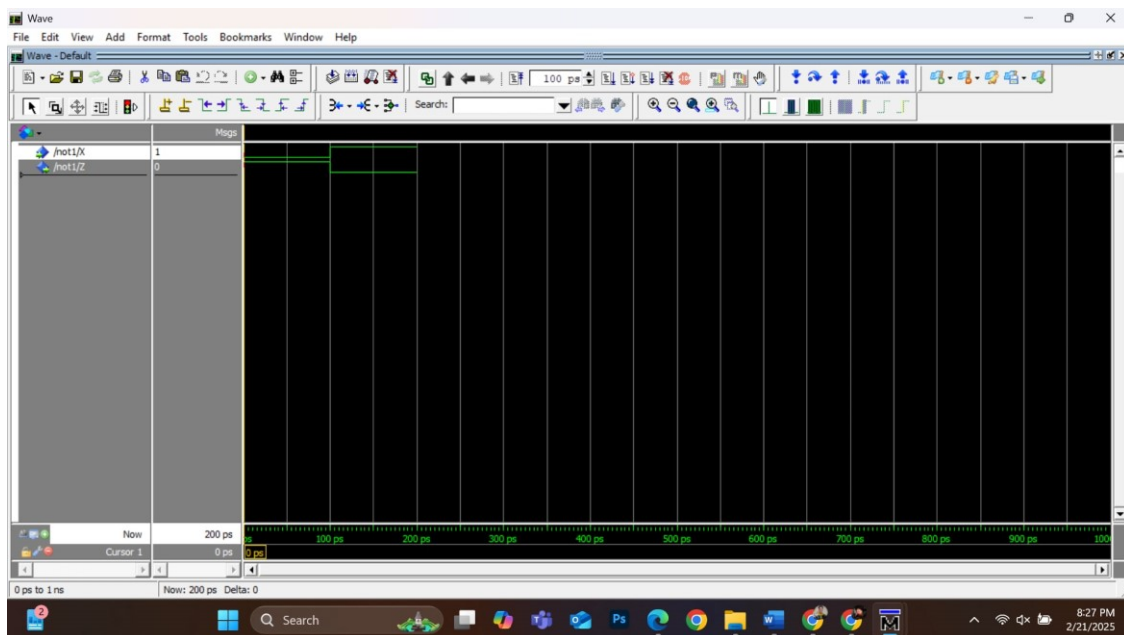
PACO, ARMIN R.  
BSCpE – 3A

## 1.LOGIC GATES

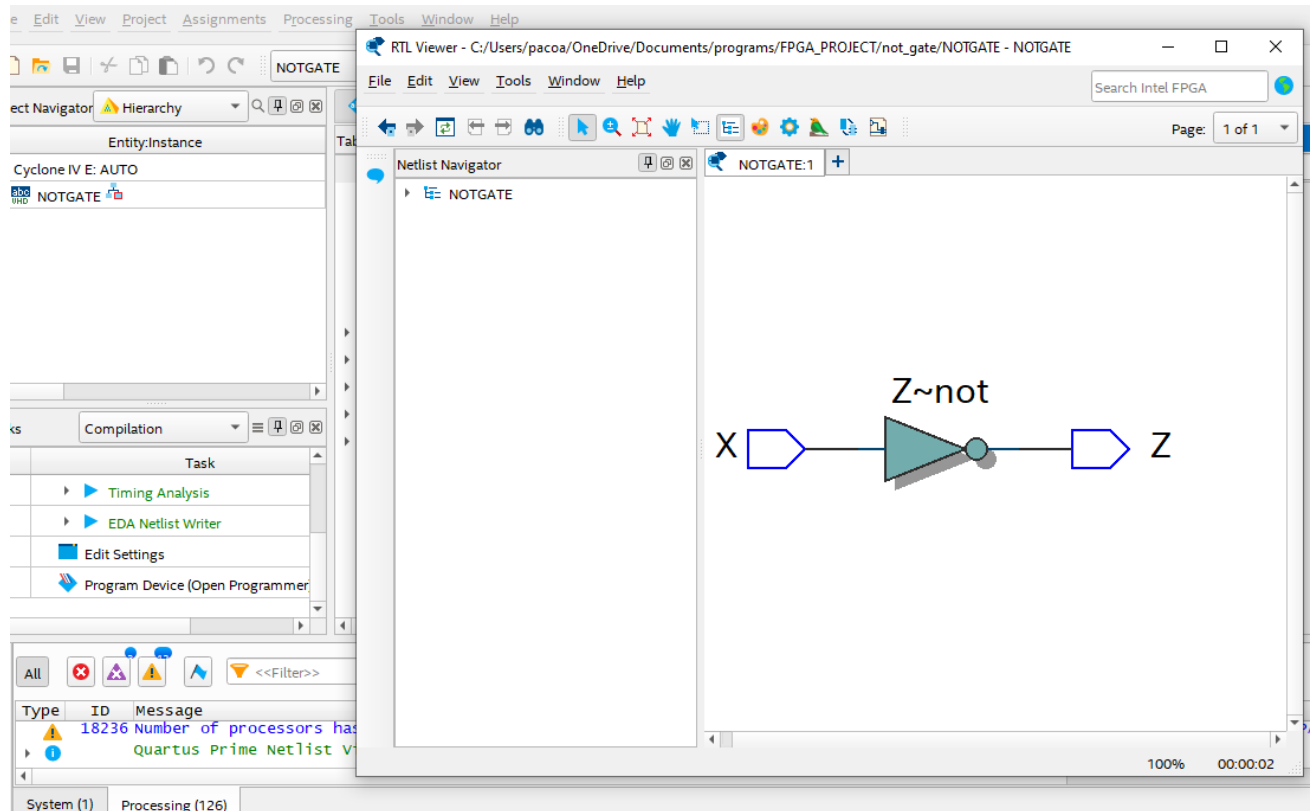
**AIM:** Write a VHDL code for all the logic gates.

#3-TITLE: NOT gate

### OUTPUT WAVE FORM:



## RTL VIEW:



## CODE:

The screenshot shows the Quartus II IDE with the "NOTGATE.vhd" file open. The code is written in VHDL and implements a NOT gate. The code is as follows:

```
1  Library IEEE;
2  use IEEE.std_logic_1164.all;
3
4  entity NOTGATE is
5      port(
6          X: in STD_LOGIC;
7          Z: out STD_LOGIC
8      );
9  end NOTGATE;
10
11  --Dataflow model
12  architecture behav1 of NOTGATE is
13      begin
14          Z <= not X; --Signal Assignment Statement
15      end behav1;
16
17  -- Behavioral model
18  architecture behav2 of NOTGATE is
19      begin
20          process (X)
21          begin
22              if (X='0') then -- Compare with truth table
23                  Z <= '1';
24              else
25                  Z <= '0';
26              end if;
27          end process;
28      end behav2;
29  end NOTGATE;
```

The code defines the NOTGATE entity with two ports: X (input) and Z (output). It includes two architectures: behav1 (Dataflow model) and behav2 (Behavioral model). The behav1 architecture uses a signal assignment statement to implement the NOT gate. The behav2 architecture uses a process block to implement the NOT gate, comparing the input X with the value '0' and assigning the output Z accordingly.