Tutorial 2

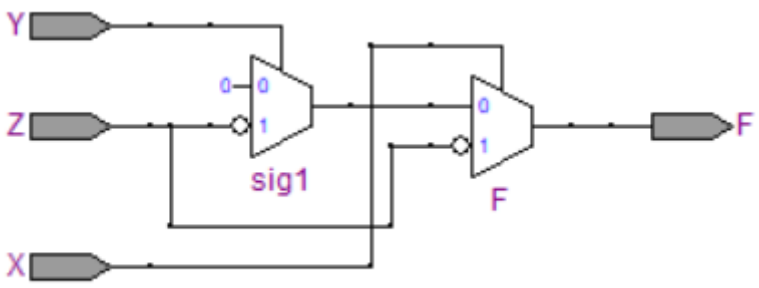
Name: Nour-aldin Ibrahim Ahmed Elbadawy
ID: 2018-13394

Sheet 2

Q2:

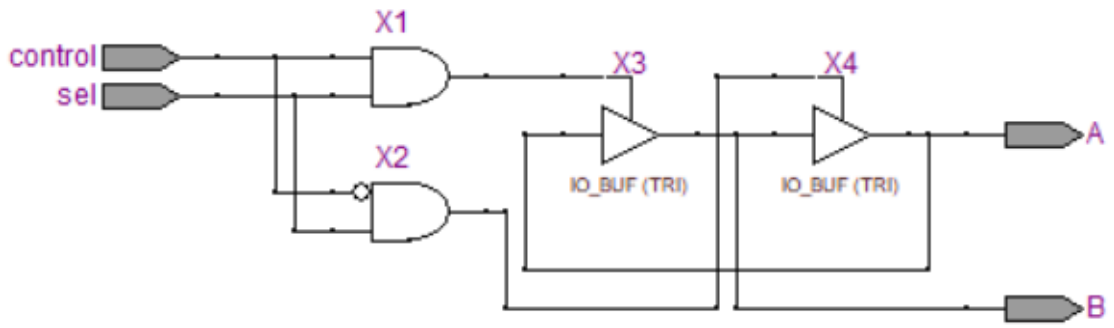
write the behavioral VHDL code

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

entity tutorial2 is
port(
          x,y,z: in std_logic;
          F: out std_logic);
end tutorial2;

architecture behave of tutorial2 is
signal sig1, sig2: std_logic;
begin
sig2 <= not(z);
sig1 <= sig2 when (y = '1') else '0';
F <= sig2 when (x='1') else sig1;
end behave;
```



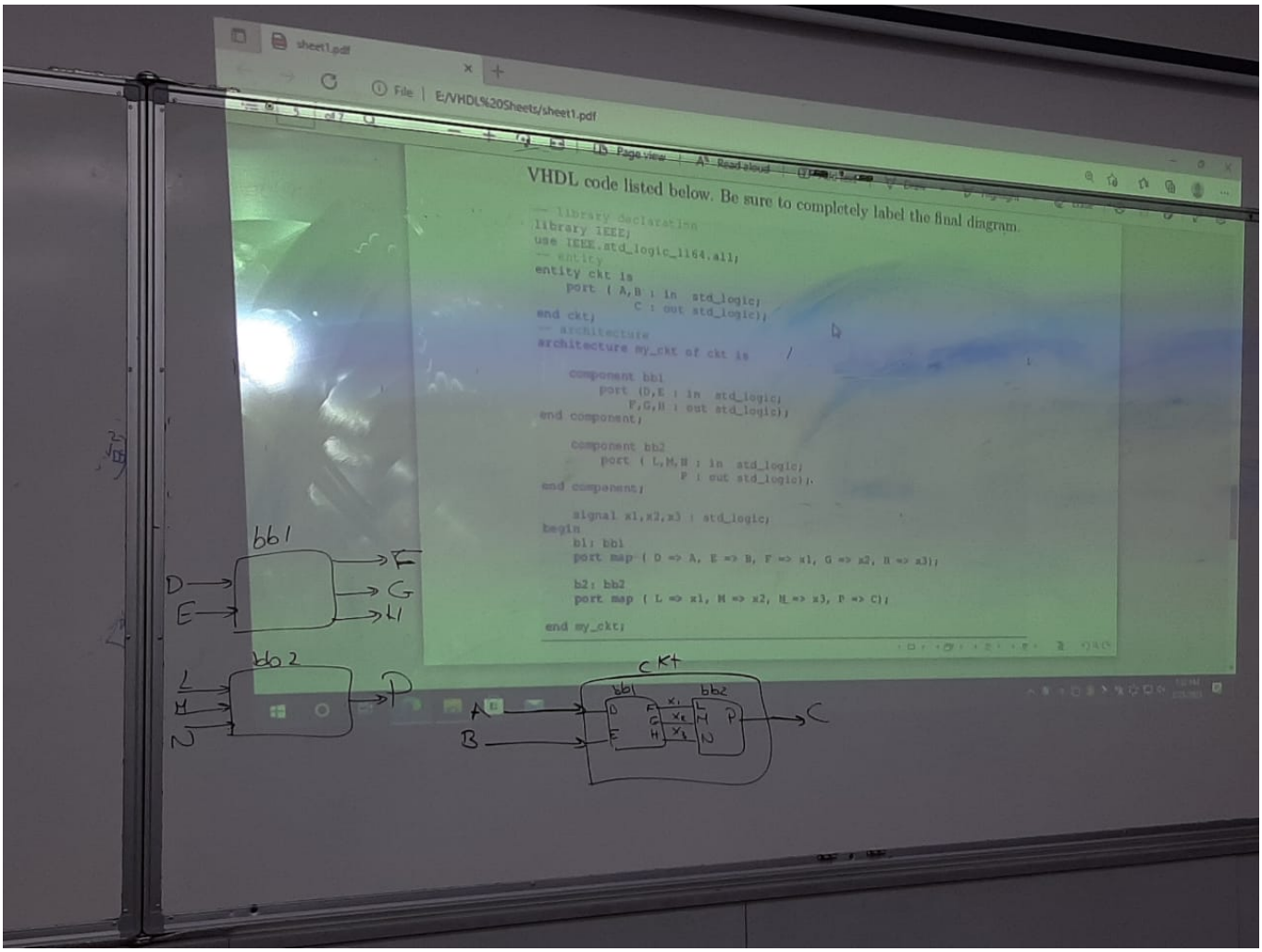
Q3:

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

entity tutorial2 is
port(
          control, sel: in std_logic;
          A,B : out std_logic);
end tutorial2;

architecture behave of tutorial2 is
signal X1, X2, X3, X4: std_logic;
begin
X1 <= control and sel;
X2 <= sel and (not(control));
X3 <= X4 when(X1='1') else 'Z';
X4 <= X3 when(X2='1') else 'Z';
A <= X4;
B <= X3;
end behave;
```



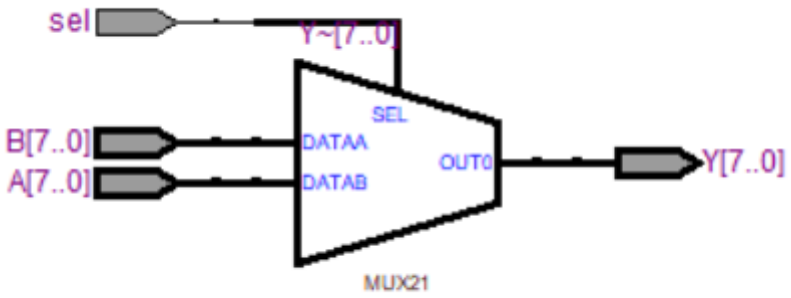
Sheet 1

Q5

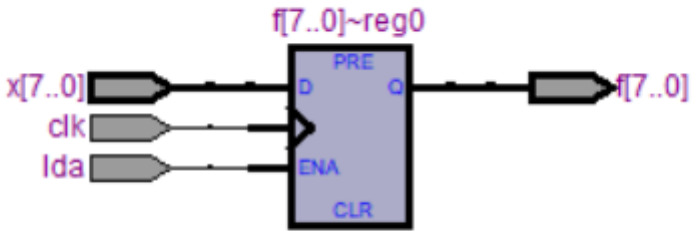


Q6

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


entity mux1 is
port(
        A,B: in std_logic_vector(7 downto 0);
        sel: in std_logic;
        Y: out std_logic_vector(7 downto 0) );
        end mux1;

architecture behave of mux1 is
begin
        Y <= A when(sel = '1') else B;
end behave;
```




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


entity reg1 is
port(
        x : in std_logic_vector(7 downto 0);
        lda,clk : in std_logic;
        f : out std_logic_vector(7 downto 0) );
        end reg1;

architecture behave of reg1 is
begin
        process (clk,lda)
        begin
                if ( rising_edge (clk) ) then
                        if(lda = '1') then
                                f <= x;
                        end if;
                end if;
        end process;
end behave;
```




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

entity tutorial2 is
port(
          lda,clk,sel: in std_logic;
          A,B: in std_logic_vector(7 downto 0);

          f: out std_logic_vector(7 downto 0)
          );

end tutorial2;


architecture behave of tutorial2 is

signal x1:  std_logic_vector(7 downto 0);

component mux1 is
port(
        A,B: in std_logic_vector(7 downto 0);
        sel: in std_logic;
        Y: out std_logic_vector(7 downto 0)
);
end component;

component reg1 is
port(
        x : in std_logic_vector(7 downto 0);
        lda,clk : in std_logic;
        f : out std_logic_vector(7 downto 0)
```

```
);
end component;

begin

u1: mux1 port map( A => A, B =>B, sel => sel, y => x1);
u2: reg1 port map( lda => lda, clk => clk, x => x1, f => f);

end behave;
```