**Matt Gambill**

**ECE 3561**

**MT1**

**MUX21**

library ieee;

use ieee.std\_logic\_1164.all;

use ieee.std\_logic\_arith.all;

ENTITY two\_one\_MUX IS

PORT(a : IN std\_logic;

b : IN std\_logic;

sel : IN std\_logic;

z : OUT std\_logic);

END two\_one\_MUX;

ARCHITECTURE BEHAVIORAL OF two\_one\_MUX IS

BEGIN

z <= ((not(sel) and a) or (sel and b));

END BEHAVIORAL;

**Test Bench**

library ieee;

use ieee.std\_logic\_1164.all;

use ieee.std\_logic\_arith.all;

ENTITY tb2\_1\_MUX IS

END tb2\_1\_MUX;

ARCHITECTURE one OF tb2\_1\_MUX IS

COMPONENT two\_one\_MUX

PORT(a,b,sel: IN std\_logic; z: OUT std\_logic);

END COMPONENT;

SIGNAL a,b,sel,z : std\_logic := '0';

SIGNAL s1 : std\_logic\_vector(3 downto 0);

signal i : integer := 0;

SIGNAL clk : BIT :='1';

BEGIN

mux1 : two\_one\_MUX PORT MAP(a,b,sel,z);

clk <= NOT clk AFTER 10 ns;

PROCESS(clk)

BEGIN

IF (i<4) THEN

s1 <= conv\_std\_logic\_vector(i, s1'length);

a <= s1(0);

b <= s1(1);

i <= i + 1;

ELSE

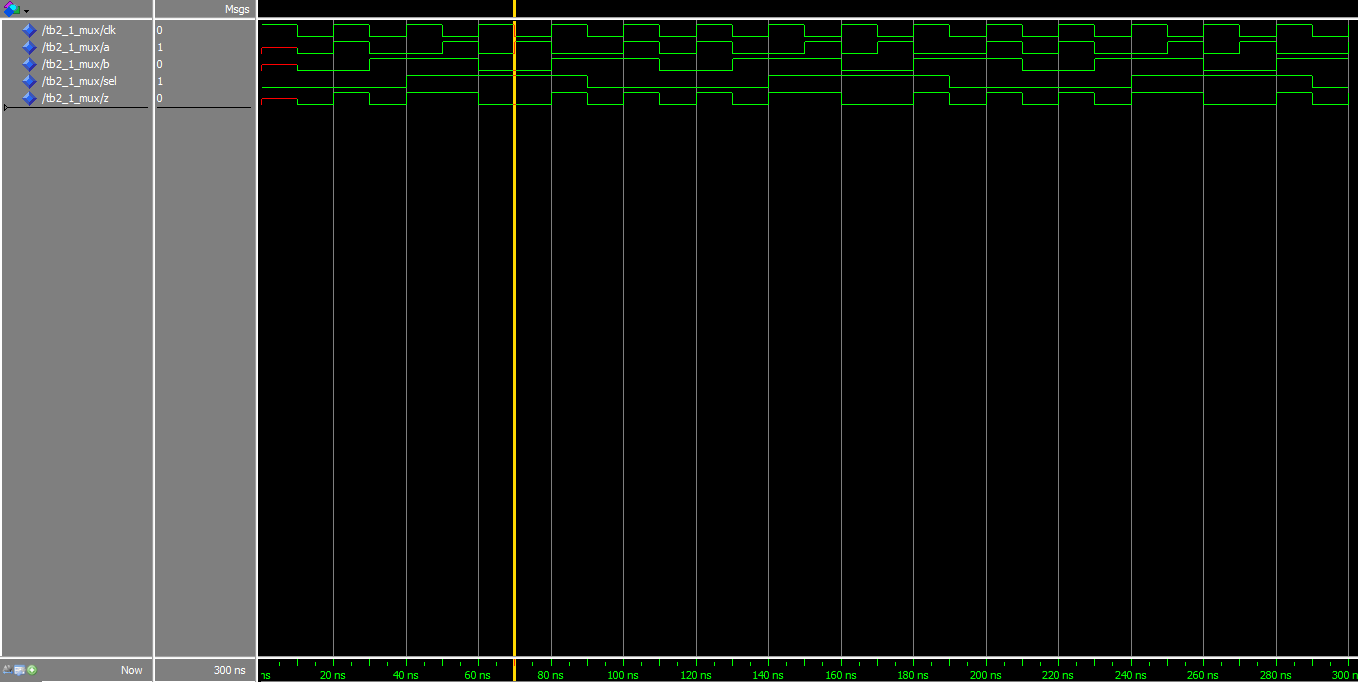
i<=0;

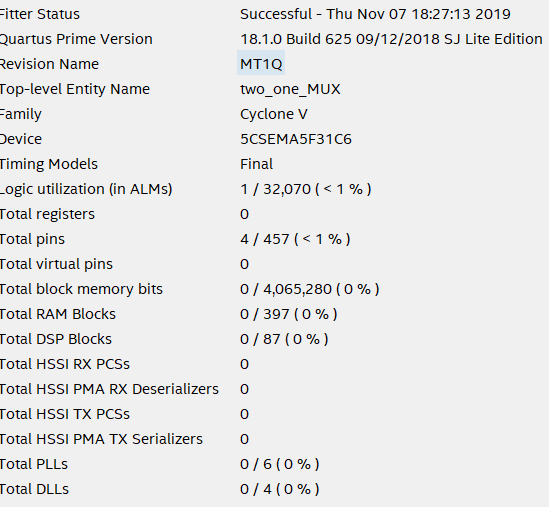
sel <= NOT sel;

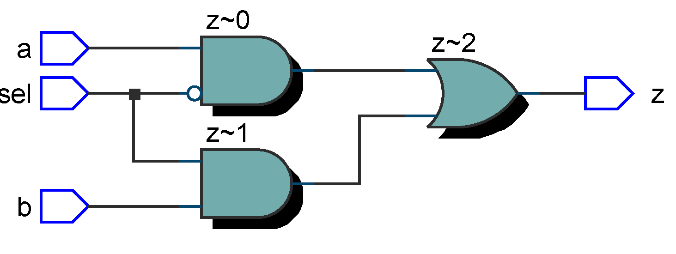
END IF;

END PROCESS;

END one;



****

****

**MUX21\_4\_Bit**

library ieee;

use ieee.std\_logic\_1164.all;

use ieee.std\_logic\_arith.all;

-- input is 4 bits, input(0) = a,input(1) = b, input(2) = sel

ENTITY MUX21\_4BIT\_BUS IS

PORT(input : IN std\_logic\_vector(3 downto 0); output : OUT std\_logic\_vector(3 downto 0) );

END MUX21\_4BIT\_BUS;

ARCHITECTURE BEHAVIORAL OF MUX21\_4BIT\_BUS IS

BEGIN

output(0) <= ((not(input(2)) and input(0)) or (input(2) and input(1)));

END BEHAVIORAL;

**TestBench**

library ieee;

use ieee.std\_logic\_1164.all;

use ieee.std\_logic\_arith.all;

ENTITY tbMUX21\_4\_BIT\_BUS IS

END tbMUX21\_4\_BIT\_BUS;

ARCHITECTURE one OF tbMUX21\_4\_BIT\_BUS IS

COMPONENT MUX21\_4BIT\_BUS

PORT(input : IN std\_logic\_vector(3 downto 0); output : OUT std\_logic\_vector(3 downto 0) );

END COMPONENT;

SIGNAL z : std\_logic\_vector(3 downto 0);

SIGNAL s1,s2 : std\_logic\_vector(3 downto 0) := "0000";

signal i : integer := 0;

SIGNAL clk : BIT :='1';

BEGIN

mux1 : MUX21\_4BIT\_BUS PORT MAP(s2,z);

clk <= NOT clk AFTER 10 ns;

PROCESS(clk)

BEGIN

IF (i<4) THEN

s1 <= conv\_std\_logic\_vector(i, s1'length);

s2(0) <= s1(0);

s2(1) <= s1(1);

i <= i + 1;

ELSE

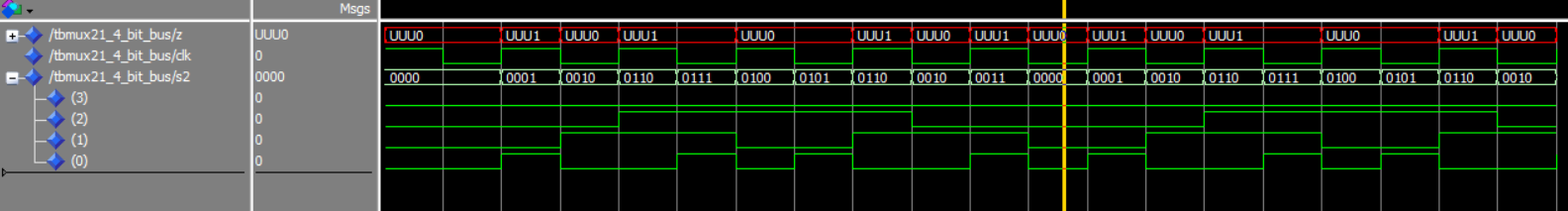
i<=0;

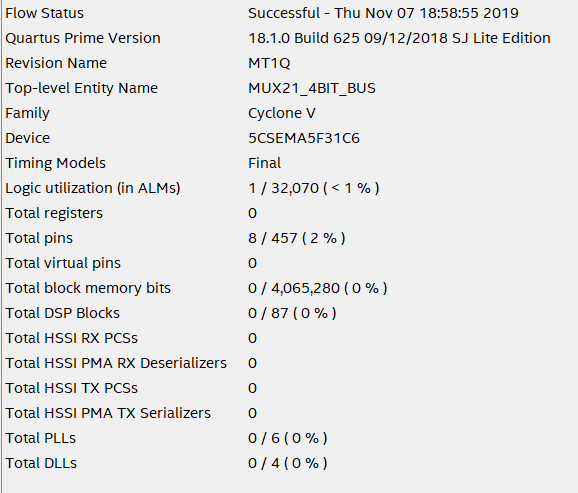
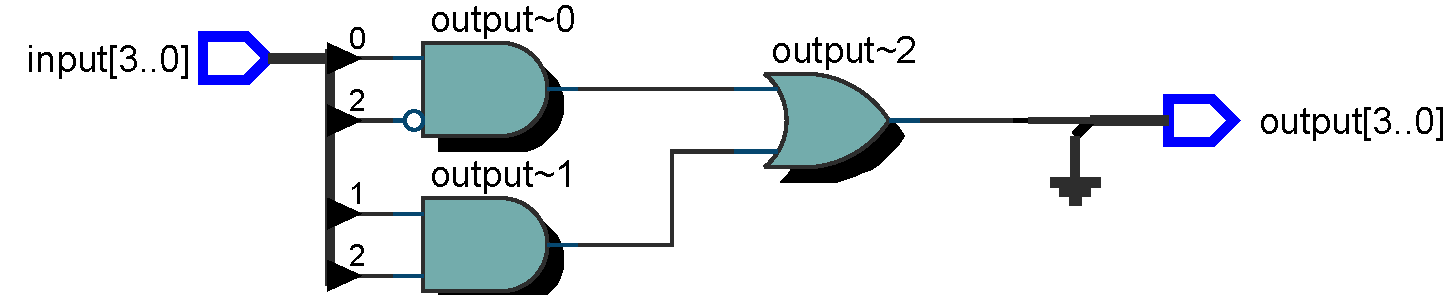
s2(2) <= NOT s2(2);

END IF;

END PROCESS;

END one;



****

**MUX21\_16\_bit**

library ieee;

use ieee.std\_logic\_1164.all;

use ieee.std\_logic\_arith.all;

-- input is 4 bits, input(0) = a,input(1) = b, input(2) = sel

ENTITY two\_one\_MUX\_16\_wide IS

PORT(input : IN std\_logic\_vector(15 downto 0); output : OUT std\_logic\_vector(15 downto 0) );

END two\_one\_MUX\_16\_wide;

ARCHITECTURE BEHAVIORAL OF two\_one\_MUX\_16\_wide IS

BEGIN

output(0) <= ((not(input(2)) and input(0)) or (input(2) and input(1)));

END BEHAVIORAL;

**Test Bench**

library ieee;

use ieee.std\_logic\_1164.all;

use ieee.std\_logic\_arith.all;

ENTITY tbMUX21\_16\_wide IS

END tbMUX21\_16\_wide;

ARCHITECTURE one OF tbMUX21\_16\_wide IS

COMPONENT MUX21\_16\_wide

PORT(input : IN std\_logic\_vector(15 downto 0); output : OUT std\_logic\_vector(15 downto 0) );

END COMPONENT;

SIGNAL z : std\_logic\_vector(15 downto 0);

SIGNAL s1,s2 : std\_logic\_vector(15 downto 0) := "0000000000000000";

signal i : integer := 0;

SIGNAL clk : BIT :='1';

BEGIN

mux1 : MUX21\_16\_wide PORT MAP(s2,z);

clk <= NOT clk AFTER 10 ns;

PROCESS(clk)

BEGIN

IF (i<4) THEN

s1 <= conv\_std\_logic\_vector(i, s1'length);

s2(0) <= s1(0);

s2(1) <= s1(1);

i <= i + 1;

ELSE

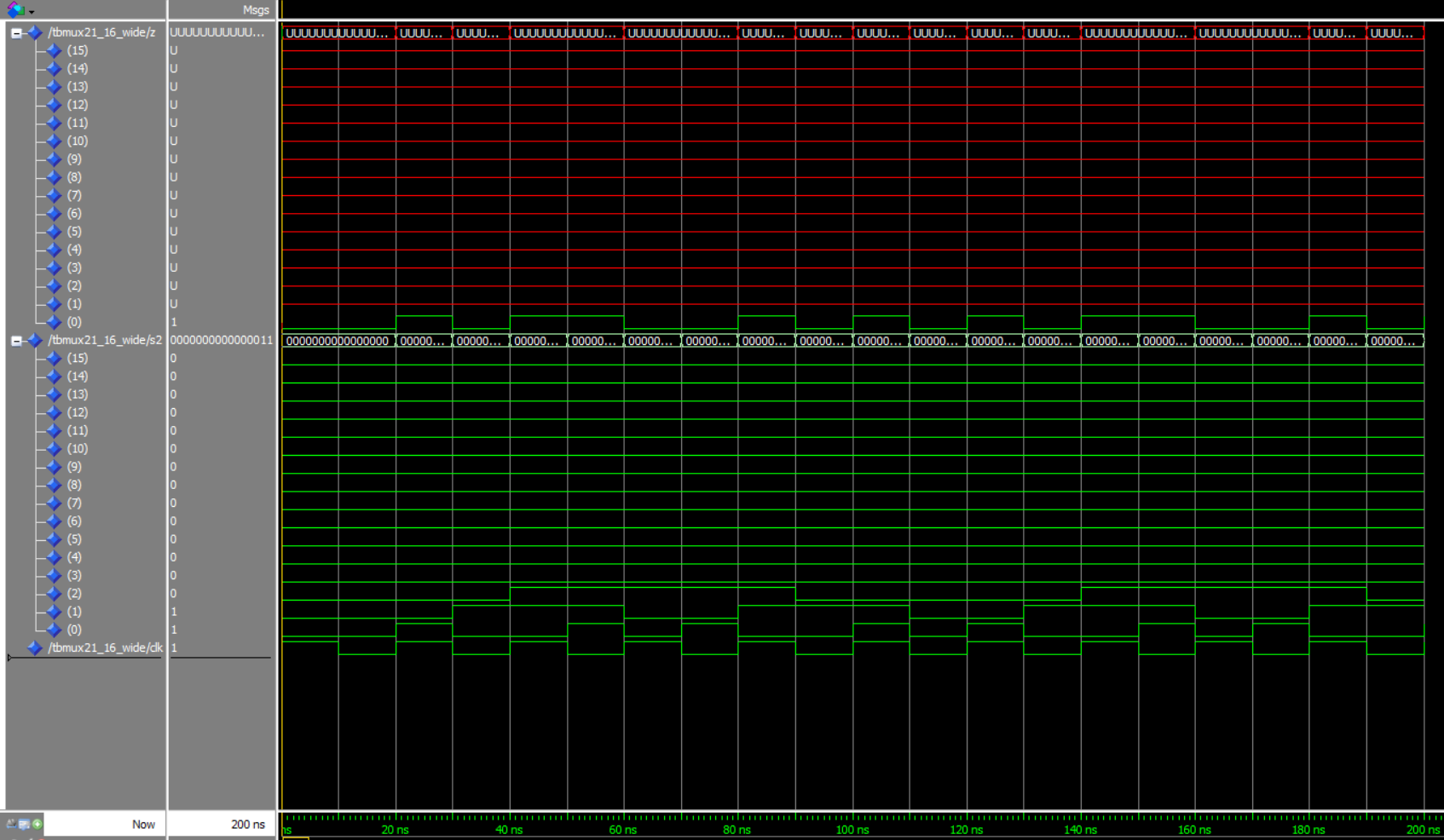
i<=0;

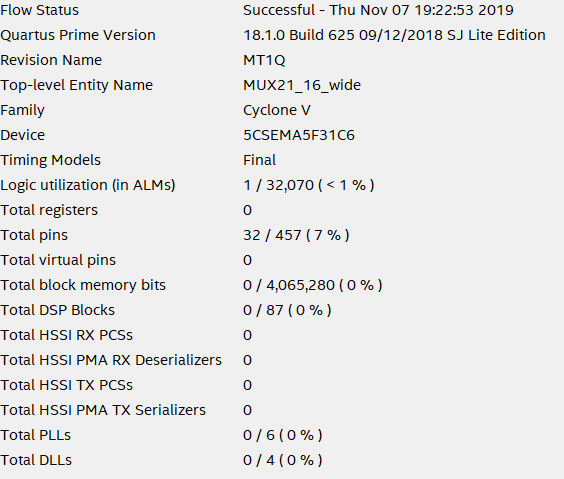
s2(2) <= NOT s2(2);

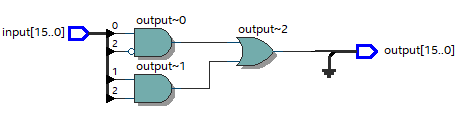
END IF;

END PROCESS;

END one;







**MUX41\_1\_Wide**

library ieee;

use ieee.std\_logic\_1164.all;

use ieee.std\_logic\_arith.all;

ENTITY MUX\_41\_1\_wide IS

port(a,b,c,d,sel\_0,sel\_1 : IN std\_logic; z : OUT std\_logic);

END MUX\_41\_1\_wide;

ARCHITECTURE BEHAVIORAL OF MUX\_41\_1\_wide IS

COMPONENT two\_one\_MUX

PORT(a,b,sel: IN std\_logic; z: OUT std\_logic);

END COMPONENT;

SIGNAL z0,z1 : std\_logic :='0';

Begin

mux0 : two\_one\_MUX PORT MAP(a,b,sel\_0,z0);

mux1 : two\_one\_MUX PORT MAP(c,d,sel\_0,z1);

mux2 : two\_one\_MUX PORT MAP(z0,z1,sel\_1,z);

END BEHAVIORAL;

**Test Bench**

LIBRARY ieee;

USE ieee.std\_logic\_1164.ALL;

USE ieee.std\_logic\_arith.ALL;

ENTITY tb4\_1\_MUX IS

END tb4\_1\_MUX;

ARCHITECTURE one OF tb4\_1\_MUX IS

COMPONENT MUX\_41\_1\_wide

PORT (

a, b, c, d, sel\_0, sel\_1 : IN std\_logic;

z : OUT std\_logic

);

END COMPONENT;

SIGNAL a, b, c, d, sel\_0, sel\_1, z : std\_logic := '0';

SIGNAL s1 : std\_logic\_vector(3 DOWNTO 0);

SIGNAL i,j : INTEGER := 0;

SIGNAL clk : BIT := '1';

BEGIN

mux1 : MUX\_41\_1\_wide PORT MAP(a, b, c, d, sel\_0, sel\_1, z);

clk <= NOT clk AFTER 10 ns;

PROCESS (clk)

BEGIN

IF (i < 16) THEN

s1 <= conv\_std\_logic\_vector(i, s1'length);

a <= s1(0);

b <= s1(1);

c <= s1(2);

d <= s1(3);

i <= i + 1;

ELSE

i <= 0;

j <= j + 1;

sel\_0 <= NOT sel\_0;

END IF;

IF(j > 1 AND j < 4) THEN

sel\_1 <= '1';

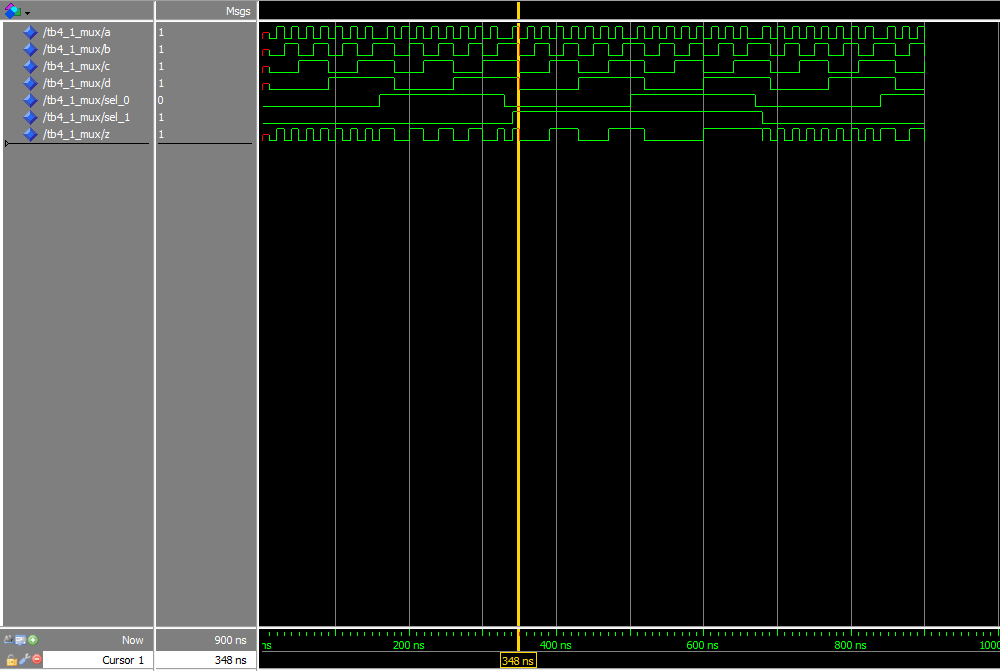
ELSE

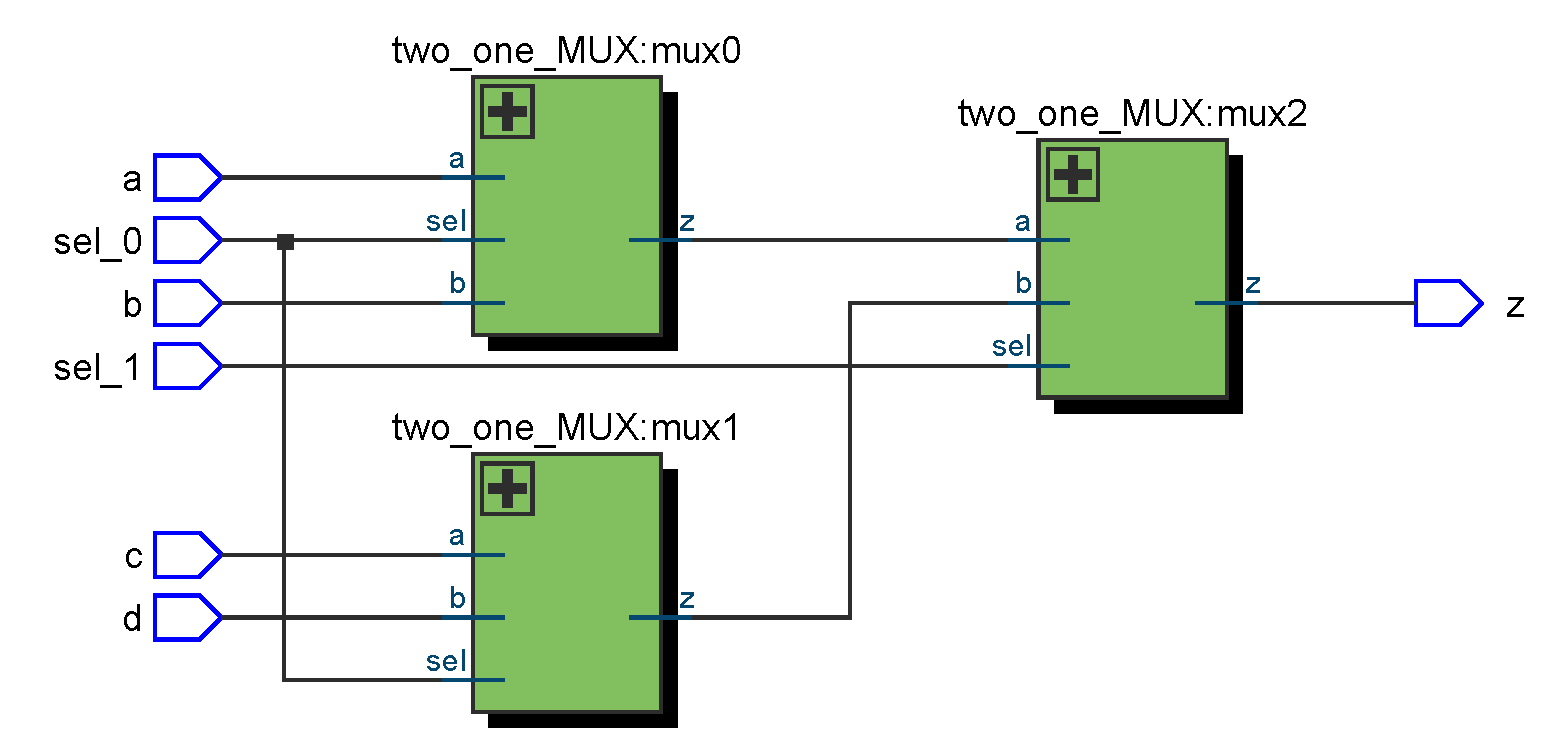
sel\_1 <='0';

END IF;

END PROCESS;

END one;



****

