

CSE 460

VLSI LAB

ASSIGNMENT 03

**Submitted by**

Md. Nasimuzzaman

Id: 19101051

CSE-460 Fall-2022

Submission Date: 1st November 2022

**Lab Assignment 03**

**Problem statement**

A sequential circuit has two inputs, **w1** and **w2**, and an output, **z**. Its function is to compare the input sequences on the two inputs. If **w1 = w2** during any **four** consecutive clock cycles, the circuit produces **z = 1**; otherwise, **z = 0**. For example:

w1 : 0 1 1 0 1 1 1 0 0 0 1 1 0

w2 : 1 1 1 0 1 0 1 0 0 0 1 1 1

   z : 0 0 0 0 1 0 0 0 0 1 1 1 0

.

**Code**

module Lab03(w1, w2, z, clk, reset);

input w1, w2, clk, reset;

output z;

reg[4:0] y;

parameter[4:0] s0 = 0, s1 = 1, s2 = 2, s3 = 3, s4 = 4;

assign x = w1 ^ w2;

always @(posedge clk , negedge reset)

begin

if (reset == 0) y <= s0;

else

begin

case(y)

s0 : if (x) y <= s0;

else y <= s1;

s1 : if (x) y <= s0;

else y <= s2;

s2 : if (x) y<=s0;

else y <= s3;

s3 : if (x) y<=s0;

else y <= s4;

default : y <= s0;

endcase

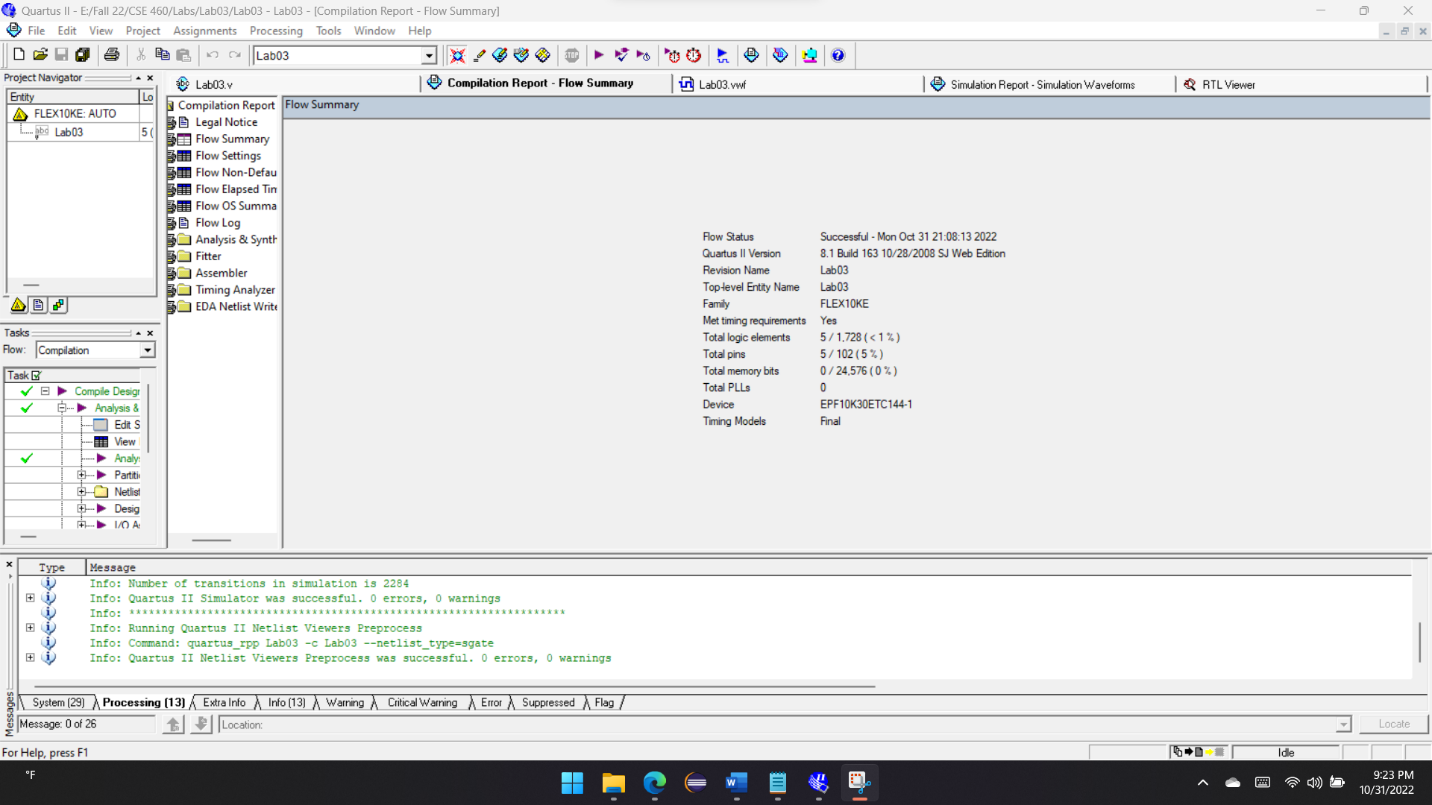
end

end

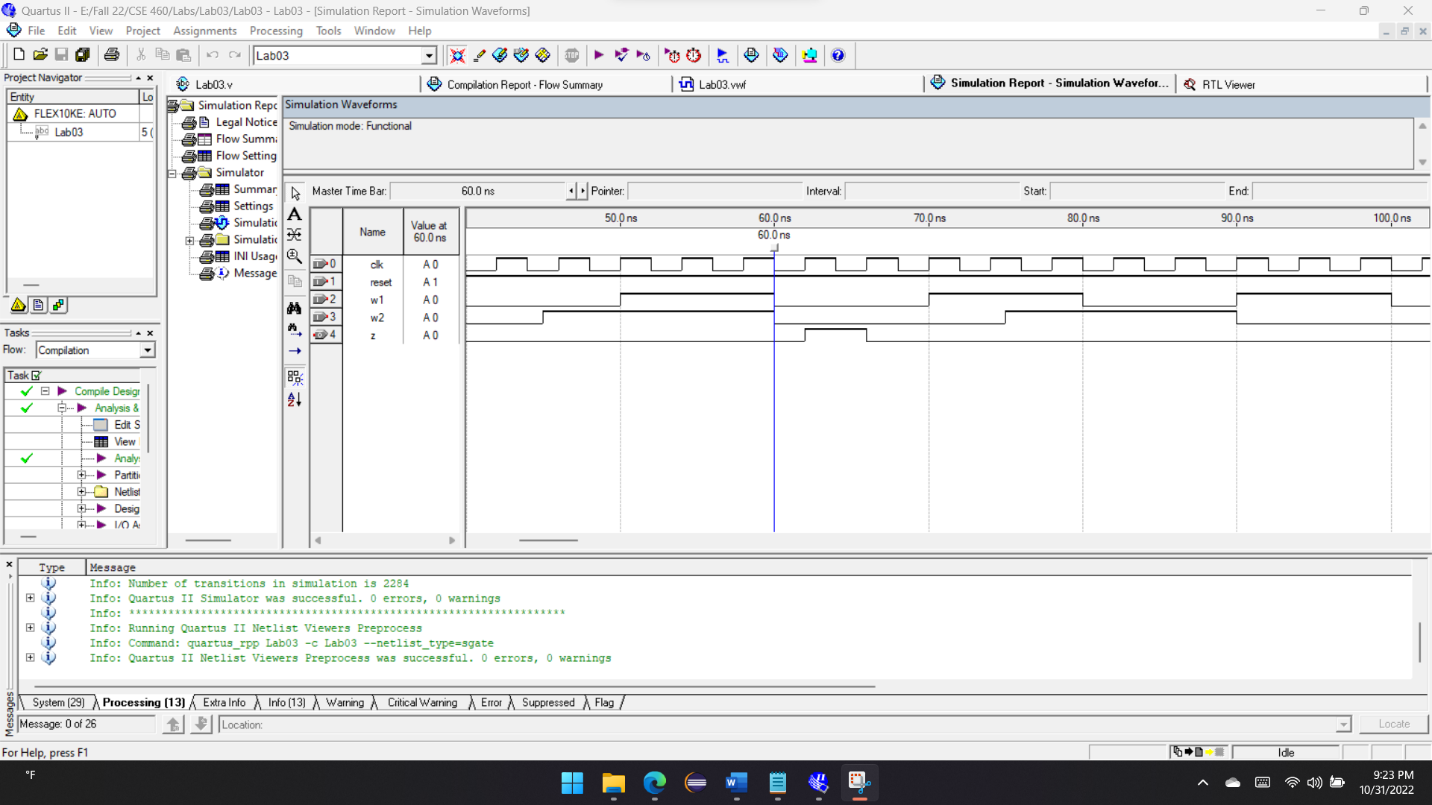
assign z = (y == s4);

endmodule

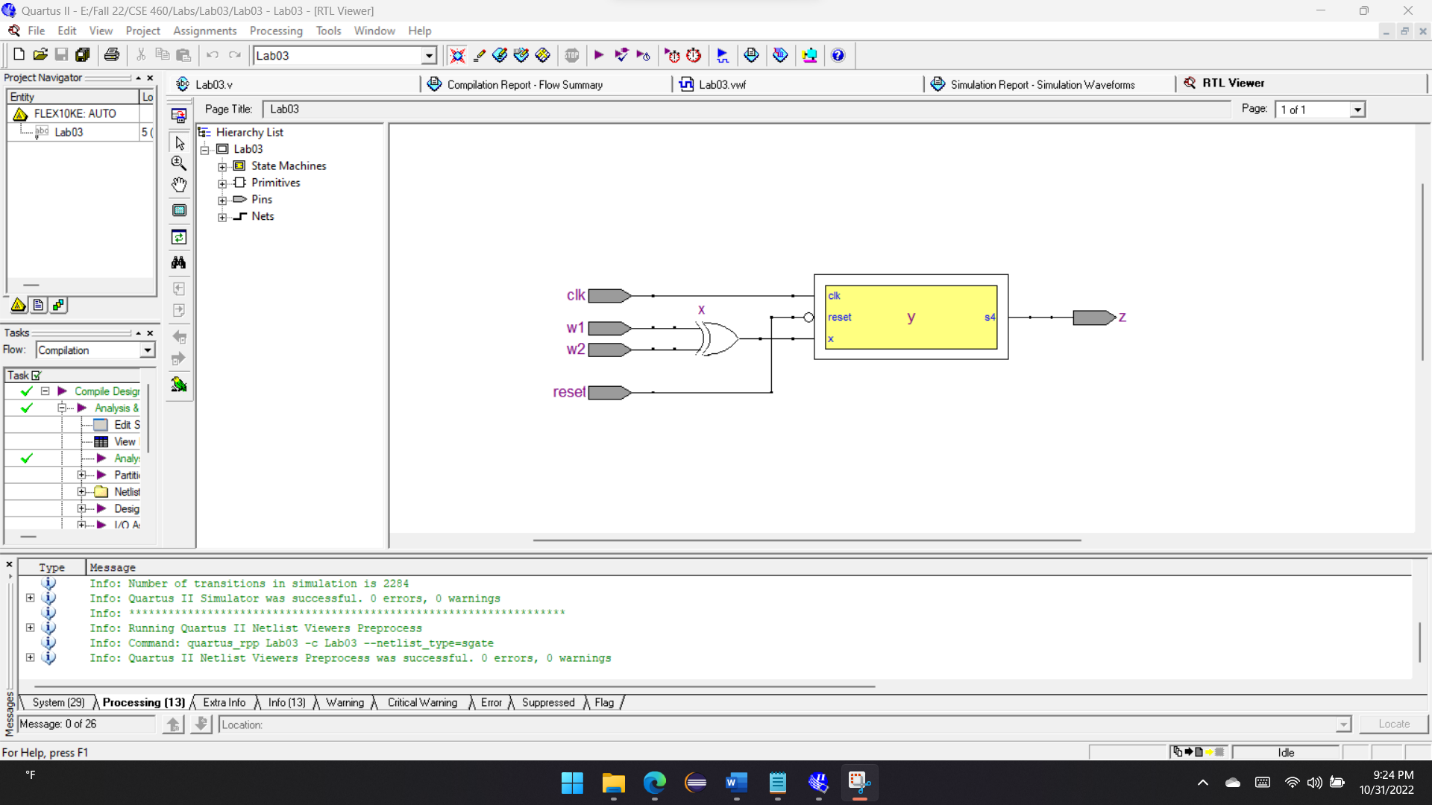
**Compilation Report**



**Simulation Report**



**RTL View**

****

**State Diagram**

**Diagram

Description automatically generated**

Figure: State Diagram

**Explanation**

In this task, I’ve simulated a sequential circuit, which is based on two inputs (w1, w2) and for 4 clock cycle if both inputs match, then it outputs high on the same clock cycle. In this case we can consider it a Mealy type of FSM. For example:

w1: 0 1 1 0 1 1 1 0 0 0 1 1 0

w2: 1 1 1 0 1 0 1 0 0 0 1 1 1

   z: 0 0 0 0 1 0 0 0 0 1 1 1 0

To execute this task, first both the inputs are XOR each other and as we know from XOR truth table, if the input match it will output a zero. So, in the positive clock cycle if the output is zero, we go to the next state and if it’s one reverts to the first state. In the ‘if/else’ block we must check this condition using case statement. If the XOR results (x) zero, we change the present state (y) to the next state. At the end if its reach to the final state which is s4, then we assign z as high (1). So, s0-s4 in the four-clock cycle if the input matches the output will be one otherwise zero.