CpE Lab

Lab 4: Introduction to VHDL

Prepared By:

Olujide Jacobs

Prepared For:

Matt Grubb

September 16, 2013

**Introduction**

The aim of this lab was to teach students how to design more complicated circuits that would normally be difficult to build using schematic design entry method, using an alternative text entry method called VHDL.

**Experiment II**

The objective of this part of the lab was to design, program and verify the operations of a 3bit adder.

**Methodology**

The group made a sketch of the 3bit adder based on the knowledge that it would consist of a half adder, two full adders, six inputs and four outputs. This was done by connecting a half adder to two other full adders, with the carry-out output from the half adder serving as the carry-in input to the first full adder, and the carry-out output from that full adder serving as the carry-in input for the second full adder. The six inputs were named and respectively, and the four outputs were named and respectively, while the two carry-in input for the two full adders were named and respectively. From the analysis of the connections made in the circuit, the group was able to pull out the Boolean equation for the circuit. For instance the Boolean equation of output was determined by observing that inputs and serve as the input to the xor gate of the half adder which yields output , hence the Boolean equation for is . This method was used to obtain the following logic expression for the circuit

( (+) );

( \* );

( (+) ) (+) ;

(( (+) ) \* ) + ( \* );

( (+) ) (+) ;

(( (+) ) \* ) + ( \* );

Next the group wrote a VHDL program for the circuit, and this is the code that was used:

LIBRARY ieee;

USE ieee.std\_logic\_1164.all;

ENTITY light IS

PORT (

X1 : IN STD\_LOGIC;

X2 : IN STD\_LOGIC;

X3 : IN STD\_LOGIC;

Y1 : IN STD\_LOGIC;

Y2 : IN STD\_LOGIC;

Y3 : IN STD\_LOGIC;

F1 : OUT STD\_LOGIC;

F2 : OUT STD\_LOGIC;

F3 : OUT STD\_LOGIC;

F4 : OUT STD\_LOGIC

);

END light;

ARCHITECTURE Behavior OF light IS

SIGNAL C1 : STD\_LOGIC;

SIGNAL C2 : STD\_LOGIC;

BEGIN

F1 <= (X1 xor Y1);

C1 <= (X1 and Y1);

F2 <= (X2 xor Y2) xor C1;

C2 <= ((X2 xor Y2) and C1) or (X2 and Y2);

F3 <= (X3 xor Y3) xor C2;

F4 <= ((X3 xor Y3) and C2) or (X3 and Y3);

END Behavior;

The inputs were assigned to pins N25, N26, P25, AE14, AF14 and AD13 while the outputs were assigned to pins AE22, AF22, W19 and V18. The final step taken was to compile the circuit.

**Result**

The circuit worked, as it should, the right output was gotten for the different input combinations tested. We tested three cases (011 + 110 = 1001, 111 + 111 = 1110, 101 + 101 = 1010) and it all worked perfect.

**Conclusion**

This was an excellent lab for so many reasons, one of them is that the desired result was obtained, and I learned how to write a VHDL program.

**Post lab questions for lab 4**

**Q1)** Input and output signals

**Q2)**

LIBRARY ieee;

USE ieee.std\_logic\_1164.all;

ENTITY light IS

PORT (

X1 : IN STD\_LOGIC;

X2 : IN STD\_LOGIC;

X3 : IN STD\_LOGIC;

Y1 : IN STD\_LOGIC;

Y2 : IN STD\_LOGIC;

Y3 : IN STD\_LOGIC;

F1 : OUT STD\_LOGIC;

F2 : OUT STD\_LOGIC;

F3 : OUT STD\_LOGIC;

F4 : OUT STD\_LOGIC

);

END light;

ARCHITECTURE Behavior OF light IS

SIGNAL C1 : STD\_LOGIC;

SIGNAL C2 : STD\_LOGIC;

BEGIN

F1 <= (X1 xor Y1);

C1 <= (X1 and Y1);

F2 <= (X2 xor Y2) xor C1;

C2 <= ((X2 xor Y2) and C1) or (X2 and Y2);

F3 <= (X3 xor Y3) xor C2;

C3 <= ((X3 xor Y3) and C2) or (X3 and Y3);

F4 <= ((X4 xor Y4) xor C3);

F5 <= ((X4 xor Y4) and C3) or (X4 and Y4);

END Behavior;

**Pre lab questions for lab 5**

**Q1)** It means to simulate the logical operation of a circuit

**Q2)** It helps to bring hardware design from concept to realization and it is used to verify the correctness of a hardware design