Julian Duran

September 13, 2018

VHDL Design with FPGAs

Lab 2 – Complex Switch Count

CPE 3020 (01)

Code

library IEEE;

use IEEE.STD\_LOGIC\_1164.ALL;

use IEEE.std\_logic\_unsigned.all;

entity lab2 is

port(

sw: in std\_logic\_vector(4 downto 0); --switches to enter input

clk: in std\_logic;

an: out std\_logic\_vector(3 downto 0); --selecting which 7seg u want

seg: out std\_logic\_vector(0 to 6);

dp: out std\_logic;

led: out std\_logic\_vector(0 to 15)

);

end lab2;

architecture lab2\_arch of lab2 is

----button states-----------------------------------------CONSTANTS

constant button\_on: std\_logic := '0'; --decimal

constant button\_off: std\_logic := '1'; --hex

----seven segment display--------------------------CONSTANTS

constant zero: std\_logic\_vector(6 downto 0) := "0000001";

constant one: std\_logic\_vector(6 downto 0) := "1001111";

constant two: std\_logic\_vector(6 downto 0) := "0010010";

constant three: std\_logic\_vector(6 downto 0) := "0000110";

constant four: std\_logic\_vector(6 downto 0) := "1001100";

constant five: std\_logic\_vector(6 downto 0) := "0100100";

constant six: std\_logic\_vector(6 downto 0) := "0100000";

constant seven: std\_logic\_vector(6 downto 0) := "0001111";

constant eight: std\_logic\_vector(6 downto 0) := "0000000";

constant nine: std\_logic\_vector(6 downto 0) := "0001100";

constant a: std\_logic\_vector(6 downto 0) := "0001000";

constant b: std\_logic\_vector(6 downto 0) := "1100000";

constant c: std\_logic\_vector(6 downto 0) := "0110001";

constant d: std\_logic\_vector(6 downto 0) := "1000010";

constant e: std\_logic\_vector(6 downto 0) := "0110000";

constant f: std\_logic\_vector(6 downto 0) := "0111000";

----Decimal Seven segment display--------------------------CONSTANTS

constant ZERO\_decimal: std\_logic\_vector(6 downto 0) := "0000001";

constant ONE\_decimal: std\_logic\_vector(6 downto 0) := "1001111";

constant TWO\_decimal: std\_logic\_vector(6 downto 0) := "0010010";

constant THREE\_decimal: std\_logic\_vector(6 downto 0) := "0000110";

constant FOUR\_decimal: std\_logic\_vector(6 downto 0) := "1001100";

constant FIVE\_decimal: std\_logic\_vector(6 downto 0) := "0100100";

constant SIX\_decimal: std\_logic\_vector(6 downto 0) := "0100000";

constant SEVEN\_decimal: std\_logic\_vector(6 downto 0) := "0001111";

constant EIGHT\_decimal: std\_logic\_vector(6 downto 0) := "0000000";

constant NINE\_decimal: std\_logic\_vector(6 downto 0) := "0001100";

constant TEN\_decimal: std\_logic\_vector(6 downto 0) := "0000001";

constant ELEVEN\_decimal: std\_logic\_vector(6 downto 0):= "1001111";

constant TWELVE\_decimal: std\_logic\_vector(6 downto 0):= "0010010";

constant THIRTEEN\_decimal: std\_logic\_vector(6 downto 0):= "0000110";

constant FOURTEEN\_decimal: std\_logic\_vector(6 downto 0):= "1001100";

constant FIFTEEN\_decimal: std\_logic\_vector(6 downto 0) := "0100100";

----internal connections------------------------------------SIGNALS

signal hex: std\_logic\_vector(0 to 6);

signal decimal: std\_logic\_vector(0 to 6);

signal clkcount:std\_logic\_vector(19 downto 0);

signal clk\_res: std\_logic := '0';

signal anode\_sel: std\_logic\_vector(1 downto 0);

begin

process(clk, clk\_res)

begin

if(clkcount="11111111111111111111") then

clk\_res <= not clk\_res;

clkcount <= "00000000000000000000";

elsif(rising\_edge(clk)) then

clkcount <= clkcount + 1;

end if;

end process;

--Hex display

with sw(3 downto 0) select

hex <= zero when "0000",

one when "0001",

two when "0010",

three when "0011",

four when "0100",

five when "0101",

six when "0110",

seven when "0111",

eight when "1000",

nine when "1001",

a when "1010",

b when "1011",

c when "1100",

d when "1101",

e when "1110",

f when others;

--Decimal display

with sw(3 downto 0) select

decimal <= ZERO\_decimal when "0000",

ONE\_decimal when "0001",

TWO\_decimal when "0010",

THREE\_decimal when "0011",

FOUR\_decimal when "0100",

FIVE\_decimal when "0101",

SIX\_decimal when "0110",

SEVEN\_decimal when "0111",

EIGHT\_decimal when "1000",

NINE\_decimal when "1001",

TEN\_decimal when "1010",

ELEVEN\_decimal when "1011",

TWELVE\_decimal when "1100",

THIRTEEN\_decimal when "1101",

FOURTEEN\_decimal when "1110",

FIFTEEN\_decimal when others;

--led magnitude

with sw(3 downto 0) select

led(0 to 14) <= "000000000000000" when "0000",

"100000000000000" when "0001",

"110000000000000" when "0010",

"111000000000000" when "0011",

"111100000000000" when "0100",

"111110000000000" when "0101",

"111111000000000" when "0110",

"111111100000000" when "0111",

"111111110000000" when "1000",

"111111111000000" when "1001",

"111111111100000" when "1010",

"111111111110000" when "1011",

"111111111111000" when "1100",

"111111111111100" when "1101",

"111111111111110" when "1110",

"111111111111111" when others;

--Mode select

with sw(4) select

seg <= decimal when button\_on,

hex when others;

--Showing point for decimal mode

with sw(4) select

dp <= '0' when '0',

'1' when others;

with sw(4 downto 0) select

led(15) <= '1' when "01010",

'1' when "01011",

'1' when "01100",

'1' when "01101",

'1' when "01110",

'1' when "01111",

'0' when others;

an(3 downto 0) <= "1110";

end lab2\_arch;

Block Diagram