**CSCE2114 Digital Logic – Lab 5**

In this lab you will write (or modify) VHDL code to convert BCD numbers to the corresponding 7-segment display characters and then download it to a FPGA.

You will write VHDL code for a circuit which has 4 inputs (4 keys) and 7 outputs (7 LEDs). The inputs are connected to 4 switches which give values of 0 or 1. The Outputs are connected to a 7 segment led display.

* If the input keys are 0000, then 7 segment display should display 0.
* If the input keys are 0001, then 7 segment display should display 1.
* ………………..
* If the input keys are 1001, then 7 segment display should display 9.
* If the input keys are have a value above 1001, then 7 segment display should be blank.

Led b

Led a

Key 0

Led g

Key 1

Key 3

Key 2

*c*

*e*

*a*

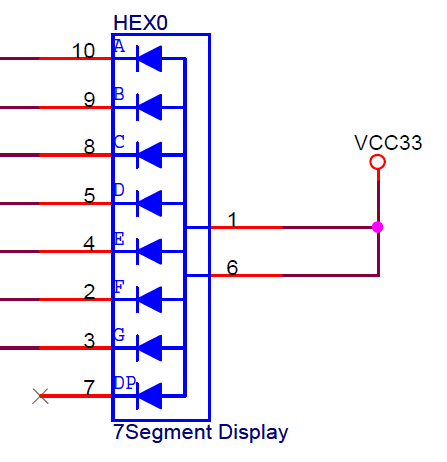
*g*

*b*

*f*

*d*

In order to make 7 segment display show ‘F’, the LED a,f,e and g should be ON and the rest of the leds should be OFF.

* For a LED to be ON, the corresponding output should be Zero.
* For a LED to be OFF, the corresponding output should be One. 

Therefore, the VHDL code looks like this:

***LIBRARY ieee ;***

***USE ieee.std\_logic\_1164.all ;***

***ENTITY seg7 IS***

***PORT ( bcd : IN STD\_LOGIC\_VECTOR(3 DOWNTO 0) ;***

***leds : OUT STD\_LOGIC\_VECTOR(1 TO 7) ) ;***

***END seg7 ;***

***ARCHITECTURE Behavior OF seg7 IS***

***BEGIN***

***PROCESS ( bcd )***

***BEGIN***

***CASE bcd IS ------ abcdefg***

***WHEN "0000" => leds <= "0000001" ;***

***WHEN "0001" => leds <= "1001111" ;***

***WHEN "0010" => leds <= "0010010" ;***

***WHEN "0011" => leds <= "0000110" ;***

***WHEN "0100" => leds <= "1001100" ;***

***WHEN "0101" => leds <= "0100100" ;***

***WHEN "0110" => leds <= "0100000" ;***

***WHEN "0111" => leds <= "0001111" ;***

***WHEN "1000" => leds <= "0000000" ;***

***WHEN "1001" => leds <= "0001100" ;***

***WHEN OTHERS => leds <= "1111111" ;***

***END CASE ;***

***END PROCESS ;***

***END Behavior ;***

*This is a binary-coded-decimal (BCD) to 7-segment display decoder. It is meant to display the numbers 0 to 9 on the display. The value displayed is the decimal equivalent of the binary number. Any binary values above 9 are not displayed.*

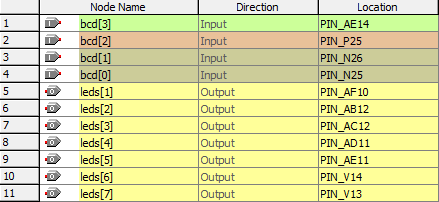
----------------------------------------------------------------------------------------

**Instructions:**

***1-Start a new project, and then add a VHDL file to your project. If you have problems with the details, refer to lab #2 instructions.***

* Click on File -> New Project Wizard -> Next
* Click on the upper left … and browse to your design directory and create a new folder named Lab4.
* For File name and top level entity: enter seg7 -> Next -> Next
* For the Family select Cyclone II, under Available devices select EP2C35F672C6 -> Next -> Next -> Finish
* Click on File -> New and select VHDL File. Add the VHDL code, then save it.

***2-Pin assignment:***

* Connect the bcd inputs to switches SW3 - SW0, and the leds outputs to the HEX0 segment display as following table suggests. Notice that the elements in the display (LED elements abcdefg) are used to create the character being displayed.
* 

***3-compile your design and the program and configure the FPGA device***.

* 1-Click on Processing -> Compiler Tool -> Start, when it is finished click on OK and close the Compiler Tool.
* Click on Tools -> Programmer you will see that the USB-Blaster is beside the Hardware Setup tab and that the File BCD.sof is listed. Click on the box below Program/Configure then click on Start.
* Notice that when the keys go from 0000 to 1001, the 7 seg display shows numbers from 0 to 9.
* Notice that when the keys go from 1010 to 1111, the 7 seg display **does not** show the correct characters from A to F. Why?

4***- Complete the VHDL code, so the 7 seg display shows the correct characters from 10 (1010) to 15 ( 1111).***

* *Demonstrate the correct function of your circuit to the TA (16 input combinations, and the display). Go over all 16 possible inputs. All characters must be displayed correctly.*

LIBRARY ieee ;

USE ieee.std\_logic\_1164.all ;

ENTITY seg7 IS

PORT ( bcd : IN STD\_LOGIC\_VECTOR(3 DOWNTO 0) ;

leds : OUT STD\_LOGIC\_VECTOR(1 TO 7) ) ;

END seg7 ;

ARCHITECTURE Behavior OF seg7 IS

BEGIN

PROCESS ( bcd )

BEGIN

CASE bcd IS ------ abcdefg

WHEN "0000" => leds <= "0000001" ;

WHEN "0001" => leds <= "1001111" ;

WHEN "0010" => leds <= "0010010" ;

WHEN "0011" => leds <= "0000110" ;

WHEN "0100" => leds <= "1001100" ;

WHEN "0101" => leds <= "0100100" ;

WHEN "0110" => leds <= "0100000" ;

WHEN "0111" => leds <= "0001111" ;

WHEN "1000" => leds <= "0000000" ;

WHEN "1001" => leds <= "0001100" ;

WHEN "1010" => leds <= "0001000" ;

WHEN "1011" => leds <= "1100000" ;

WHEN "1100" => leds <= "0110001" ;

WHEN "1101" => leds <= "1000010" ;

WHEN "1110" => leds <= "0110000" ;

WHEN "1111" => leds <= "0111000" ;

WHEN OTHERS => leds <= "1111111" ;

END CASE ;

END PROCESS ;

END Behavior ;