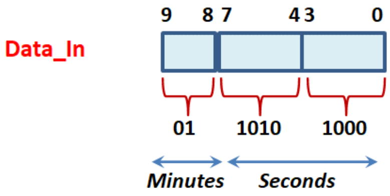
Instructions for Lab – 3

Thu. 28 March, Tue. 16 April and Thu. 18 April (12 pm - 2 pm)

[3 Marks]

(This lab can be done in group of 3 students).

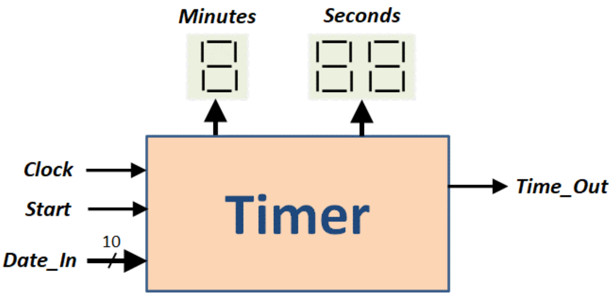
The aim of this lab is to design a simple programmable timer, which can display minutes and seconds (up to 3 minutes and 59 seconds) and implement it on the FPGA Development board. The output of this timer is connected to 7-segment displays as shown below. Output signal ***Time_Out*** will change from '0' to '1' when the specified time elapsed (or in other words when the timer times out) and it is displayed on one LED on the FPGA board. The timer is initialized through ***Data_In*** input, which is connected to ***10 switches*** on FPGA board and the 10 switches positions determine the required time. For example, if the switches are set as **"01 0010 0110"** that means the required time is ***1 minute and 26 seconds***. Note that bits **3 to 0** and bits **7 to 4** each represent a BCD number as shown below. If the related switches (for group bits 3 to 0 or 7 to 4 are set mistakenly to indicate a 4-bit binary value bigger than 1001, then it should be considered as 1001).



The timer set up value (which was initialized through ***Data_In*** input) **is registered** and the timer starts working **either from 0 up to the specified registered value or the other way from the specified value in the register down to 0** (either ways are acceptable depending on how you design your digital system) when input ***Start*** (a push button switch on the FPGA board) is pressed.

(Note that input ***Start*** is also used to register the value of ***Data_In*** input in 10-bit register to indicate the timer set up).

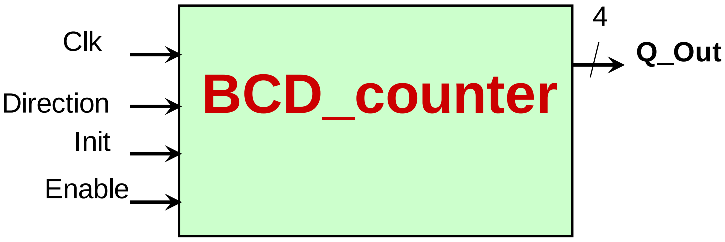
As the timer works, its current counted values are displayed on the three seven-segment displays on the FPGA board.



Hint: *Note that the input clock frequency in the given FPGA board is 50 MHz. However, you need to have clock frequency of 1 Hz for your timer. This can be done by designing a component (which will be placed between the input clock pin from the FPGA board and your timer clock input to provide the required input clock frequency).*

Note: Several versions of Quartus II software might have been installed on lab computers. You should use Quartus II 18.1 for this lab (and DE0-CV FPGA board).

(a) *** Write **synthesizable** VHDL code for a one-digit up/down BCD counter, which works when input ***Enable*** is '1' (otherwise the output is frozen at its current value). Input ***Init*** initializes the counter to 0 or 9 depending on the value of ***Direction*** input. (When ***Direction*** is '0', it is an up counter).
(***Init***, ***Enable*** and ***Direction*** are **synchronous** inputs).



(b) *** Write a test bench (or reuse your testbench from Lab-2) to test the functionality of your BCD counter.

***** Note: If you wrote a synthesizable code for BCD counter in Lab-2 then this component can be reused for this lab. Otherwise, you should write synthesizable code for the BCD counter as explained in (a).**

(c) Use this BCD counter as a component to design the timer (and add any required additional code). Also, the BCD to 7-Segement converter should be used (Refer to the given code *BCD_to_7Seg.vhd*) to display each digit.
(d) Write a testbench to test your code for the timer system and test it using ModelSim. (**You may use a clock generator with frequency of 1 Hz in your testbench for the main design component or use 50 MHz for the system**).
(e) Do the required pin assignments (with TA help in the lab) to make proper connections to switches, push button LED and seven-segment displays on the board in Quartus II EDA tool. Then synthesize your code for the target FPGA device **Cyclone V 5CEBA4F23C7** (or for DE0-CV FPGA board).
(f) Determine how many resources (registers and ALMs) are used for your design.
(g) Download the generated FPGA bit-stream to program DE0-CV board.

Note that the bits order for seven segment display is shown as follows: (bit 0 is the least significant bit).

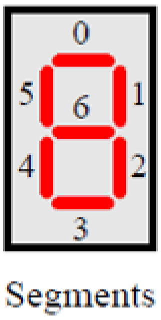


Figure 1: A seven-segment display

Refer to Chapter 3 of DE0-CV board manual (especially pages 24 to 27) for more details about the board and FPGA pin assignments used in this lab.

DE0-CV