CS M152A DIGITAL DESIGN LAB

Lab 4 Stopwatch

In this lab, you will be designing a stopwatch using Xilinx and the FPGA.

CS M152A DIGITAL DESIGN LAB

STOPWATCH

Introduction

In this assignment, you will design a basic stopwatch circuit using the Xilinx ISE Foundation software. Then you will program the circuit onto the FPGA to demonstrate its operation. You will use the push buttons and the switches to interact with the stopwatch. The digits of the stop watch will be displayed on the four 7-Segment LEDs.

Prelab

Due Next Session:

1. Sketch your initial high level design for the Stopwatch. A hand drawing is perfectly fine, but you are welcome to use a program like Logism to facilitate this process.

Functionality

The stopwatch will start as a basic clock which will count minutes and seconds. The two 7-Segment LEDs on the left will count minutes and the two on the right will count seconds. For example, without touching anything, after 1 minute and 43 seconds, the stopwatch should display: “0143”. The time stored in the stopwatch should be adjustable by the use of two inputs: SEL and ADJ. Both of these inputs are switches on the FPGA.

- SEL is a select switch which will choose minutes or seconds. The selected 7-segment LEDs should blink reasonably while in the adjusting mode.

SEL	Selected
0	Minutes
1	Seconds

- ADJ will increase either minutes or seconds at a rate of 2Hz (twice a second).

ADJ	Action
0	Counter behaves normally
1	Counter stops and ‘*Selected*’ increases at 2Hz

In addition to the two switches, your circuit will use two push buttons (on the FPGA) which will control the timer behavior.

- **RESET** will force all of your counters to the initial state 00:00
- **PAUSE** will pause the counter when the button is pressed continue the counter if it is pressed again

Implementation

The basic building block of the stopwatch is the decade (modulo 10) counter. The Xilinx library part number for this counter is CD4CLE – a 4-Bit loadable cascadable BCD counter with a clock enable bit. Feel free to use another decimal counter in the library if you are inclined to do so. The logic block of the CD4CLE is shown in Figure 1. This counter is similar to the 74LS163 binary counter except that its maximum value is 9 = [1001], as opposed to 15 = [1111]. With a small amount of additional logic, two decade counters can be combined to count with other periods, such as 60 (for seconds and minutes).

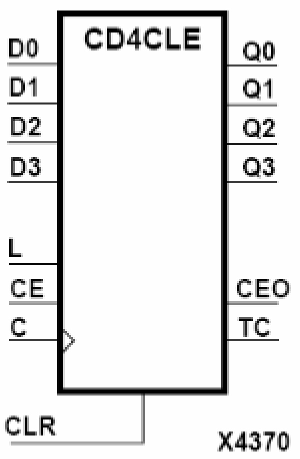


FIGURE 1- CD4CLE DECADE COUNTER LOGIC SYMBOL

The counters will be chained together. A 2 Hz clock input will be used to clock the counters, and which will be used to create a 1Hz signal (by using a T Flip Flop). The seconds counter will increment every second, and every 60 seconds, it will reach the maximum value of 59 and enable the minutes counter, which will increment on the next rising edge of the clock. Both counters are clocked by the same 2Hz signal, so this stopwatch is an example of a clocked synchronous state machine.

While creating your schematics, you will inevitably have to use bus taps. Bus taps are Xilinx's way of taking certain bits in a wire. To bus tap, first name your input using the following convention: NAME(X:0), where NAME represents the name of the input and X is the Most Significant Bit. Go ahead and draw a line from the input. The select the bus tap tool and place a tap (triangle flush with wire). Rename the tap using the following convention: NAME(Y), where Y can be any number from 0 to X.

Make sure to have a hierarchical design. The highest level should have two of the counter modules, the select module, and the clock module.

Clocks

For this assignment, you will be using four different clocks – a 2 Hz clock, a 1 Hz clock, and a much faster clock (50 – 700 Hz), and a clock for blinking in the adjust mode (>1Hz). We recommend that you create a Clock Module that takes the 100 MHz master clock (internally connected to pin V10 of the FPGA board) as input and outputs 4 different clock signals.

The purpose of the third, faster clock is to assist in the multiplexing of the 7 Segment Display. The display is designed in such a way that all four 7 Segment Displays will display the same number. For this reason, you will need to multiplex the four sets of 7 using a much faster clock so the human eye doesn't recognize this. This range is between 50-700 Hz; we'll leave it to you to find a good value.

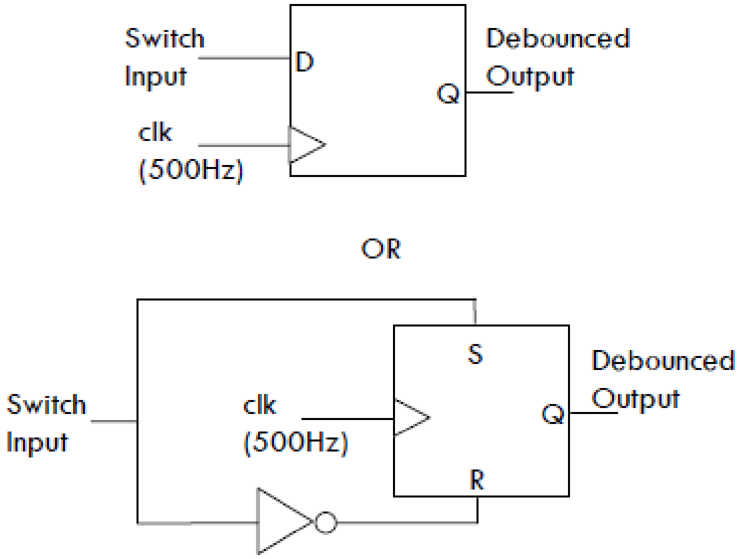
Additionally, you need to have an accurate 2 Hz clock. The 2 Hz clock signal that clocks the counters must be obtained by dividing the 100 MHz master clock (internally connected to pin V10 of the FPGA board). To obtain a 2 Hz clock, the master clock frequency should be divided by 50 million. Clock signals can generally be 'divided' by using one or more counters. Instead of dividing by 50 million directly, you can divide the master clock by 5000, again by 5000, and then by 2 (50,000,000 = 5000*5000*2).

Debouncers

In this lab, you must make both buttons and switches on the board useful. However, due to physical contact instability, the buttons and switches are very bouncy: it implies that when you are intended to press the button only once, it rather generates an oscillation of signals due to poor metal contact.

There are two pre-designed debouncer codes available in the CourseWeb: (i) debounce.vhd and (ii) DebounceSync.sch. The second file (i.e., DebounceSync.sch) is an upper level schematic file which contains modules (i.e., schematic symbols) of "debounce.vhd". Thus, you must first compile "debounce.vhd" and make it as a schematic symbol, which is later used by "DebounceSync.sch". Please open up "DebounceSync.sch" in order to see how the modules of "debounce.vhd" is used. Note that there are two clock inputs to "DebounceSync.sch:" clk1 and clk2. The period of clk1 provides the length of time in which you would debounce your input as a single input. The period of clk2 provides information about how often your output is generated from "DebounceSync.sch". Please use 50MHz for clk1 and 2Hz for clk2. However, since the physical metal conditions vary from boards to boards, the frequency for clk1 may vary from 100MHz to 500Hz. Please try different frequencies for clk1 to successfully debounce your input buttons. The frequency for clk2 may stay constant at 2Hz. Note that "DebounceSync.sch" should be ONLY used for buttons, but not for switches.

The switches on the board are also very bouncy. However, they cannot be debounced by "DebounceSync.sch" provided in the CourseWeb. Thus, you must implement a very simple debouncer circuit for switches using an SR latch. The debouncer circuit for switches looks like the following diagram. Note that 500Hz is a typical value for the clock but it may also vary from boards to boards.



Lab Requirements

You will need to demonstrate that your design works properly and meets the requirements as specified in the instructions.

Report

Please write a maximum 3 pages report describing your design of stopwatch. You must have an overall description of your design as well as detail descriptions for each sub-component of your design. Please include one or more diagram in order to support your explanation. Strictly follow the report template, otherwise you will lose points.

In this project you must submit both hard and softcopy of your project. Please do not include ANY vhdl codes or schematic diagrams in the hard copy of report that you must submit in the class. All software codes must be compressed in a "zip" file and submitted to the CourseWeb under Assignments section. Please organize your "zip" file to contain:

1. README.txt
   a. Names and student IDs for all group members
   b. Comments that you want to leave to help your TA evaluating your design
2. Project File (director)
   a. Please include all project files

Please also name your submission file as "LAB4_(Partner Name 1)_(Partner Name 2).zip".