EE 2390 Laboratory # 10

Final Project: State Machine Design for a Stopwatch

Purpose

In this lab, the student will design a multi-level synchronous finite state machine (FSM) which implements the functionality of a conventional stopwatch with a variety of modes, including lap counter and countdown. The design should be entered modularly and incrementally into the Xilinx Vivado system and simulated at various levels to verify behavior. The synthesized design will be downloaded to a Xilinx Artix-7 FPGA and a Spansion Configuration ROM. The downloaded hardware implementation will tested at the benchtop using the student's own Digilent Basys3 prototyping board.

As a final project, this lab has a progressive series of expectations (that is, there are various levels of "being done" that the student can select, based upon the difficulty, effort required, and desired project grade). The final three weeks of laboratory for this semester are dedicated to this project.

Overview of Problem

The Stopwatch External View: For the purposes of this lab, our stopwatch functionality can be summarized briefly in terms of the I/O resources available on the Digilent Basys3 prototyping board. Consider the basic user interface diagram shown below in Figure 10.1. The four seven-segment LED displays of the Basys3 will serve to provide stopwatch times 0:00.0 to 9:59.9 (that is, from time zero to time nine minutes and 59.9 seconds). A combination of logic switches and pushbuttons on the Basys3 will provide user control inputs. The hardware clock (100 MHz) on-board the Basys3 shall provide the internal synchronous clocking source `clk` for all machine components, and is the base clock from which a 0.1 second time reference must be derived.

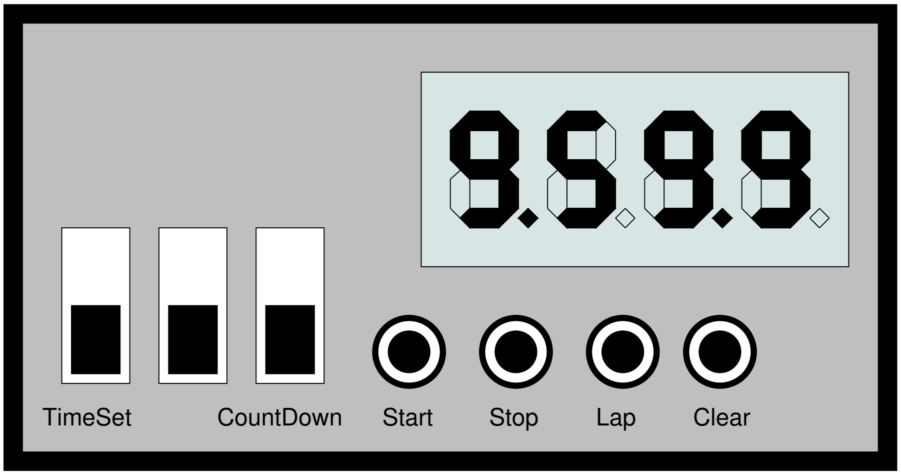

Figure 10.1 User Interface Elements of the Stopwatch Design

The User Control Inputs: Logic switches and pushbuttons should be used to provide the user inputs for control of each action of the stopwatch. Choice of logic switch or pushbutton is a design step left up to the student/designer; however, use of pushbuttons is strongly recommended for most functionality (for example, this will be most effective for the Start input). The system/stopwatch should have an "active high" synchronous *master* reset that will ensure return of the machine to a known initial state (this would typically be hidden from the user but will be indispensible for debugging). The actions of the Start, Stop, Clear, Lap, CountDown and TimeSet inputs are described below. Additional inputs may be added by the designer as they see fit.

Available LED Displays: The total time and time resolution of our stopwatch is limited by the displays available on the Digilent Basys3 prototyping board. The seven-segment displays should be used to display the minutes, seconds, and tenths of seconds, respectively, of the current stopwatch reading. **A decimal point should be displayed to convey the separation between minutes and seconds, as well as the separation between seconds and tenths of seconds.** Due to limits of our display, the more common colon symbol between minutes and seconds will be replaced with a decimal point.[1]

The Project Levels of Completion: The following levels of difficulty (and the corresponding maximum available project grade) are available for this project. Each level assumes the functionality of previous levels. The total available grade is based on the equivalent of two standard lab grades as accumulated through the semester.

Level 1: (max grade 70%) For the initial level, the machine should reset synchronously, via the input reset, to a time display of 0 minutes and 00.0 seconds. When the Start input is pressed, the machine should count up from zero input, correctly incrementing the tenths of seconds, seconds and minutes indicators through 9 minutes, 59.9 seconds, then wrapping-around and continuing to count from zero. The measure of time should be as precise as possible with the Basys3 on-board clk. For this level, the input Stop should result in the ability to "stop" the count process with display of the resulting watch time. Asserting Start while "stopped" should result in continuation of the watch count, incrementing upward from the "stopped" value. Asserting Clear while "stopped" should result in a time display of 0 minutes and 00.0 seconds, and staying in the "stopped" state. The direction of the count process should be settable by the level of the input CountDown, which changes the count direction only when the watch is "stopped." When counting down to zero, the time should wrap-around to the maximum time and continue to count down until Stop is pressed. Pressing Start should then resume counting in the direction specified by CountDown.

Level 2: (max grade 80%) When the watch count reaches 0:00.0 while either counting up or down, the stopwatch behavior should now result in a time that remains at 0:00.0 while the seven segment display flashes off and on. When at this terminal count value, pressing Start should allow counting to continue in the current direction, while pressing Stop should halt counting (and either press should stop the flashing of the display).

Level 3: (max grade 90%) Introduction of the Lap input should allow the machine to stop the display of minutes and seconds (when the watch is in conventional stop/watch mode, not counting down), for easy viewing by the user, without stopping the underlying count process. Pressing Start after a "lap" is asserted should result in return to display of the stopwatch count (with the new count being whatever the running time happens to be. That is, Lap doesn't result in stopping of the count).

Level 4: (max grade 100%) Assertion of the TimeSet input, when the watch is "stopped," should allow the watch time to be rapidly set to any valid displayable time. The setting process should

[1] The Basys3 board does not provide access to the colon symbols on the LED displays.

proceed as follows: (1) if the watch is "stopped," change to the time setting mode begins by asserting the `TimeSet` input, (2) the direction of time change is specified by the level of `CountDown`, (3) an accelerated counting process (as fast as the designer deems reasonable) should begin when `Start` is pressed, and should stop counting on an event chosen by the designer (examples include when `Start` is released or when `Stop` is pressed, the choice is up to the designer), (4) an indeterminant number of accelerated up and down count processes should be possible to reach a desired time, with immediate wrap-around at zero time, (5) the watch should leave the time setting mode by change of the `TimeSet` input.

Task A ("Prelab:" Notebook Design) You are *strongly* encouraged to begin with Level 1, proceeding through the design, debugging, implementation and hardware testing process *one level at a time.* Save the source code and Xilinx project for each working design level in its own directory, and *modify only a copy* for the next level. Design each level with flexibility for future expansion to the next level (the future might be tomorrow). **As you complete each level, it's best to have your lab TA confirm proper operation and "check you off as having completed that level!"** Sometimes the best intentions of getting the next higher level working doesn't come to fruition as the end-of-semester crunch hits.

You are responsible for the documentation of your design for each level of this lab. Tools like state diagrams, state transition and output tables, ASM charts, schematics and Verilog code are expected at each level, both to document your design as well as to provide a basis for explaining your intended design behavior to your TA or instructor. Because ad-hoc designs tend to be extremely difficult to debug, you must have complete and clear documentation if you expect any assistance during the debugging process.

While your Prelab tasks should be done before you come to lab, your TA *will not* gather your notebooks to verify a Prelab for this project. It's your responsibility to have a viable, documented design.

Task B (Design Entry and Simulation in Lab, EN 5030). Use the Xilinx Vivado system to input your state machine design and to run any simulations necessary. Keep in mind that pre-synthesis simulation may not match post-synthesis simulation if one or more bugs are in your HDL code. The hardware operation will more closely match post-synthesis simulation. The Verilog code for any and all modules of your project is entirely up to you. You should make sure to document the success of your design simulation with annotated waveforms.

As you progress through the levels of circuit complexity, you should find it helpful to "partition" your design by constructing sub-modules.

Task C (Hardware Implementation in Lab, EN 5030). Use the Xilinx Vivado system to synthesize your design from Task B, then configure and download it to the Digilent Basys3 board. You'll proceed through these steps several times. The pin configuration requirements for your project can be derived from the documentation provided in previous lab exercises.

Task D (Hardware Testing in Lab, EN 5030). Test all aspects described for the Level you're trying to complete. Your TA *must confirm* proper operation for the highest Level at which you desire to be graded for this project. It's recommended that you get checked-off at each level, beginning at Level 1, just to be safe.

NOTE: ALL CONFIRMATIONS OF YOUR FINAL PROJECT MUST BE MADE NO LATER THAN Friday, May 9, 2025 at 5:00 PM.