9/23/2024

**Lab 02 Report**

Ryan Beck

Jared Bronson

**Objectives**

Implement a stopwatch on the DE-10 Lite development board. The stopwatch should be accurate to hundredths of a second. The 6 seven-segment displays will be used to represent minutes, seconds, and hundredths of seconds in a MM.SS.HH format. The two push buttons will be used as a reset and a go button. The stopwatch will start when the go button is pushed and stop when go is released. The reset button will set the timer to zero. No clock dividers will be used, and the project will be completed entirely in VHDL.

**Procedures**

Using the system builder, create a Quartus Project File to include the on-board clock, seven-segment displays, and push buttons. Open the Verilog file created by the system builder and create a VHDL file with the same name. In the VHDL file, create a stopwatch entity with ports for the on-board clock, seven-segment displays, and push buttons using the same names given in the Verilog file. Only use the 10 MHz clock for this design. In the stopwatch entity, create a generic with an integer equal to 100,000. This integer is set to 100,000 to equal a hundredth of a second when using the 10 MHz clock. Next, create the architecture of the stopwatch. After making the necessary signals and constants, begin a process using the clock and push buttons. This process should reset the stopwatch when KEY(0) is pressed, and count when KEY(1) is pressed. Begin another process using the clock and variables that keep track of when the stopwatch ticks and is reset. When the count ticks, increment the hundredths by one and rollover to each of the larger places when needed. Display the count as it is updated. If reset is pressed, the display will go back to all zeros. Next, create a testbench to simulate and verify the stopwatch design will function properly when implemented on the DE-10 Lite board. After successful simulation, implement the design on the board for final verification of the design.

**Results**

The stopwatch file, shown in Figures, successfully implements a stopwatch accurate to hundredths of a second. The file also implements a start and a reset button successfully, using entirely VHDL. The function of the file was verified with a testbench file used to run simulations. Both the testbench and the simulations can be found in Figures.

**Figures**

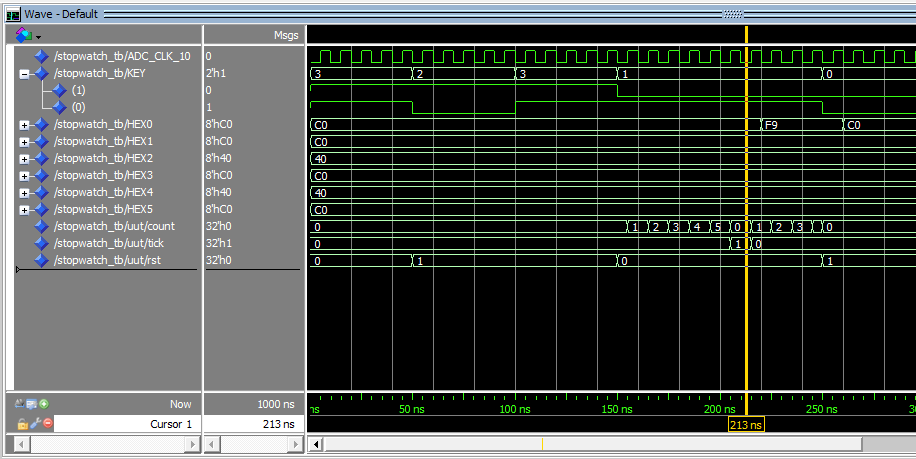
****

Figure 1: Counting sequence, incremented on 'tick' variable

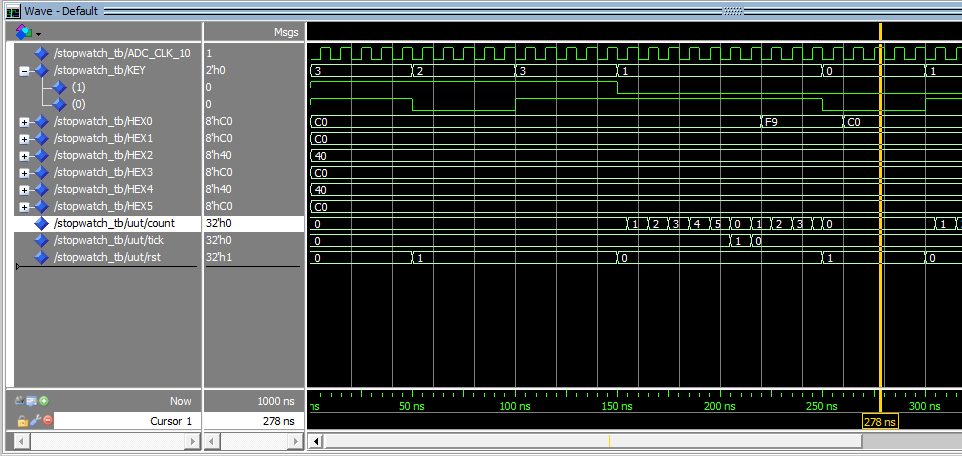


Figure 2: Reset behavior

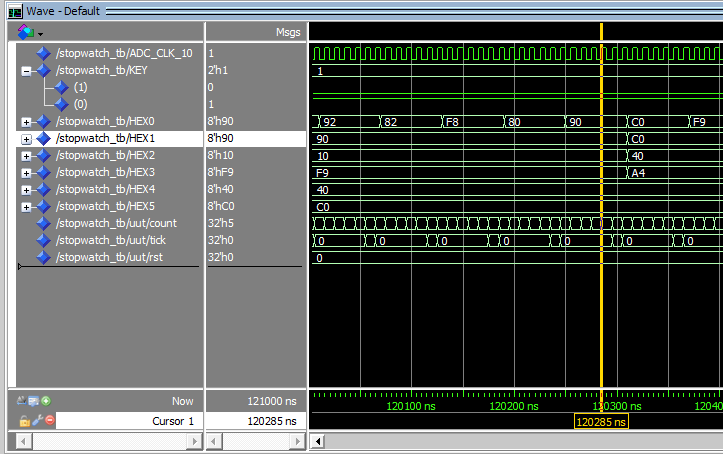


Figure 3: Timer incrementing effect

A screenshot of a computer code

Description automatically generated

Figure 4: Stopwatch.vhd file Pt. 1

A computer code with numbers and symbols

Description automatically generated

Figure 5: Stopwatch.vhd Pt. 2

A computer screen shot of a computer code

Description automatically generated

Figure 6: Stopwatch.vhd Pt. 2

A screen shot of a computer code

Description automatically generated

Figure 7: Stopwatch.vhd file Pt. 4

A screenshot of a computer program

Description automatically generated

Figure 8: Stopwatch\_TB.vhd file Pt. 1

A screenshot of a computer program

Description automatically generated

Figure 9: Stopwatch\_TB.vhd file Pt. 2

**Conclusion**

In conclusion, we were able to implement a stopwatch accurate to hundredths of a second, shown on the 6 seven-segment displays. The design implemented the two push buttons to reset and start the stopwatch. We did this by creating a counter that incremented every 100,000 clock cycles because the 10 MHz clock was in use. We used the seven-segment display by creating an array and a constant table to with hex values corresponding to 0-9. Each clock cycle, each of the seven-segment displays were assigned a hex value from the table to show the current time on the stopwatch. Simulations were performed using a testbench to show proper function of the design before final implementation and testing on the development board.