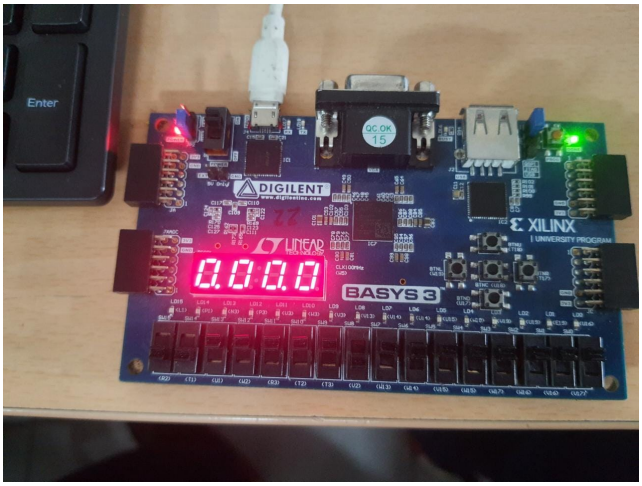
COL215P Hardware Assignment 1

Stopwatch

Gurarmaan S. Panjeta & Aryan Dua
2020CS50426 & 2020CS50475

Module descriptions, Block diagrams, Test runs and General approach



September 9, 2022

September 9, 2022

1 Module functionality

1.1 ”stopwatch.vhd”

The top level entity responsible for incorporating all other components, this entity takes as input the user inputs and the clock, and provides the anode and cathode signals for the LED display.

1.2 ”counter.vhd” (Get_all)

An essential part of the assignment was to get the correct time out of the 100MHz clock present on the board. Along with resetting, pausing and playing options, the entity outputs the Tenth-second, second, ten-second, and minutes value of time after the clock was most recently started/reset. This is done using a series of appropriate modulo n counters as mentioned in the problem statement.

1.3 ”timing-circ.vhd” (TimingCircuit)

The LED panel accepts only one set of cathode inputs for 4 LED’s, hence the corresponding anode signal needs to be activated in cyclic manner. This is achieved using a modulo n counter while making sure that the frequency remains in the 1KHz-60Hz range to avoid flickering. This circuit hence lights up the correct anode in a cyclic manner.

1.4 ”mux.vhd” (Multiplexer)

With ”counter.vhd” giving 4 outputs for each of the digits and ”timing_circ.vhd” selecting the correct anode, we need to pick one of these 4 signals (to display) as ordered by the anode signal. This is achieved using a simple 4:1 multiplexer, and this file hence selects the correct digit to display as per the anode signal.

1.5 ”debouncer.vhd” (Deb)

As mentioned in the problem statement, the metal contacts in the push buttons are erroneous, and often result in ”bounces” of signals when activated. This is handled using logic that ignores a small number of cycles (using a modulo n counter on the board clock to make a 500Hz clock) whenever one of the buttons is pressed. This ensures none of the fluctuations affect the measured time and the outputs.

1.6 ”SSegment.vhd” (Display)

The ”counter.vhd” gives it outputs as 4 numbers, each 4bits with values from 0 to 9 in the binary format. Out of the 4 numbers, ”mux.vhd” and ”timing_circ.vhd” choose the correct number to display. This number must now be sent to the 7 leds (correctly chosen by the anode signal) and the corresponding DP, which should Light up to display the number in a human readable format. This is done by implementing Karnaugh maps for each of the 7 LED’s, and keeping the 1st and 3rd DP on but 2nd off.

This entity thus lights up the numbers according to it's 4-bit input.

1.7 "basys.xdc"

This file is the constraint file that links the signals in the software to actual components on the board. Thus, the "cats" (cathode) signal was connected to the 7 bits for the seven segment display. The anode was similarly connected to the ans 4 bits, indicating which of the led turns on. The DP bit displayed whether the DP display should be on or off, and the three buttons were connected for user input. Instructions to operate the device are mentioned ahead.

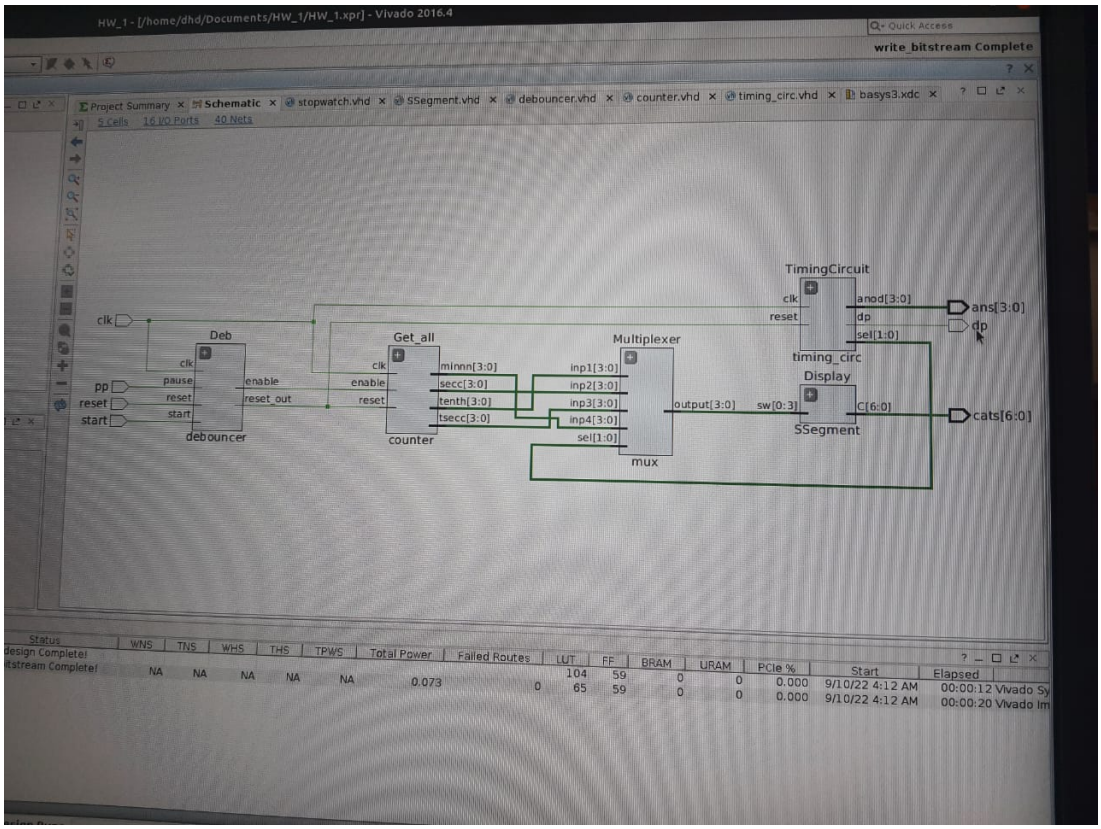
2 Operating the stopwatch

The stopwatch, once programmed by vivado, runs as per the commands mentioned in the assignment. These are-

- **Down Button**- To pause the clock at it's current state.
- **Up Button**- As the Start/Continue button. By default, the clock was designed to run when it is first programmed. After pausing and resetting it again and again, we can use it as many ties as necessary.
- **Center Button**- To Reset the clock. We have designed reset to reset the time to 0, but retain the running state. If the clock was paused when reset was pressed, the clock returns to 0 and remains paused. If the clock was running when reset was pressed, the clock resets to 0 and remains running.

3 Block Diagram

The Block diagram, displaying all the relevant connections, components (with the inputs and outputs of each) and the complete circuit, is as attached



4 Running State of the stopwatch

Paused at an arbitrary instance (3.25.6) to illustrate completeness.