COL215 Hardware Assignment 2 Report

By Disha Shiraskar & Srushti Junghare

In the assignment, we are asked to make a stopwatch that measures minutes, seconds, and one-tenth of a second.

- **APPROACH**

In this assignment, in order to implement a stopwatch, we made the following helper functions:

1. Clock_Divider:
   This takes the default clock waveform (100Mhz, time period: 10^{-8}sec) as input and returns an output clock waveform of frequency 10Hz (time period: 0.1sec). In this function, we have maintained a counter which starts from zero and keeps incrementing on detecting a rising edge of the inbuilt clock waveform, until it reaches 5000000. This means 5000000 rising edges of the inbuilt clock have been covered i.e. 10000000 cycles have been completed i.e. time equal to 10^{-8}x10^{7} = 0.1 sec has been covered every time the counter increments itself. This function helps to display the rightmost digit (corresponding to 0.1 sec) on the board.
2. Numeric:
   This function takes enable_watch, reset_watch, and clock as input and returns two outputs, out_1 corresponds to the waveform obtained by combining 10 cycles of the input clock and out_2 corresponds to the value of the counter. The counter 'count' increments every time it detects the rising edge of the input clock until it reaches the value of 10. The digit to the left of the rightmost digit measures 1 sec (increments every 1sec until it reaches 10) which is equivalent to 10x0.1sec and thus 10 cycles have the input clock have been combined. Similarly, the second digit from the left increments every 10sec which is equivalent to 10x1sec, and thus 10 cycles of the input clock (1sec) have been combined. The output clock of this function is used to display the rightmost and second rightmost digits on the display.
3. Numeric_6:
   This function works similarly to numeric except that it combines 6 cycles of the input clock instead of 10. The output clock of this function is used to display the leftmost digit on the display.

4. int_bin:
   This takes a one-digit integer as input and gives the corresponding 4 bits of its binary form as output.
5. switch:
   This function takes four inputs start, pause, continue and reset and returns the appropriate values of enable_watch and reset_watch. According to the requirements in the assignment, only the 0->1 transition of every switch should affect the working of the stopwatch and the 1->0 transition shouldn't. In order to meet this, we need to keep a track of the value of a switch just before the transition and the value of the switch just after the transition. Call the value of the switch just before the transition switch_prev and just after the transition as switch. Whenever a rising edge of the clock occurs, the if-else statements first check if switch_prev isn't equal to switch, which would mean that a transition (either 0->1 or 1->0 has happened) and if it is 0->1, it assigns values to enable_watch and reset_watch depending upon the switch considered (start/pause/continue/reset).

The main function of our assignment is Stopwatch_main.

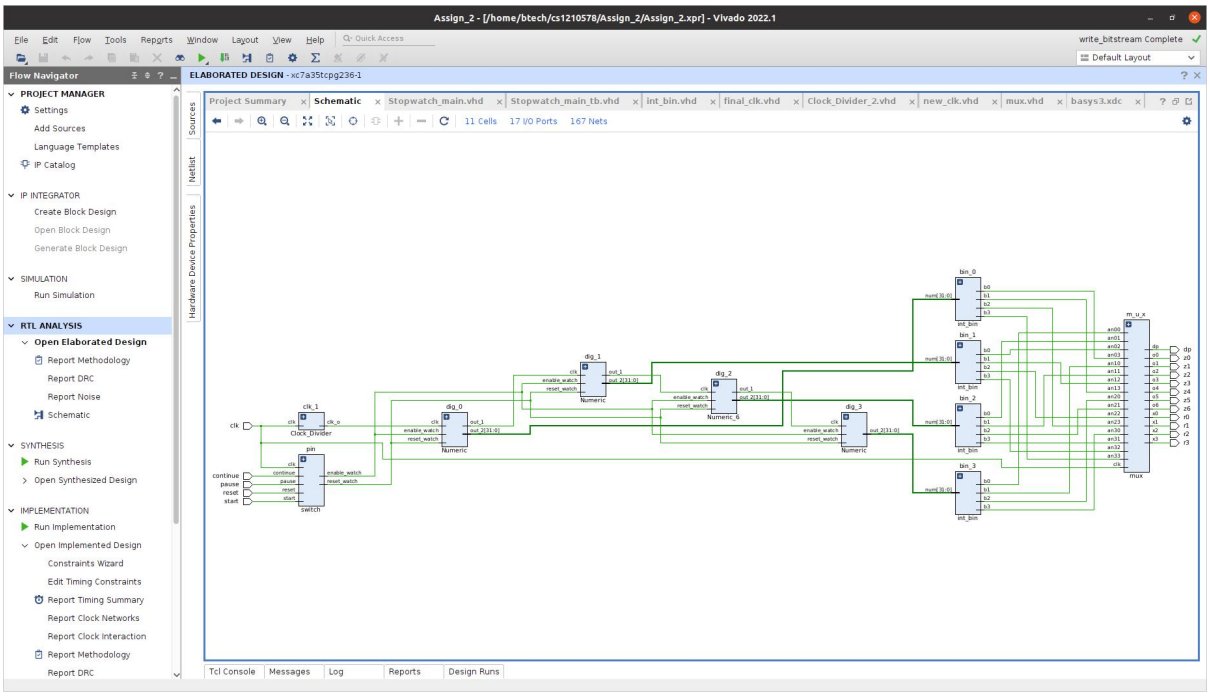
This function takes inbuilt clock, start, pause, continue and reset as inputs and returns 11 outputs, 7 corresponding to the cathodes of 7 segment display and 4 corresponding to anodes of the same. Along with the helper functions mentioned above, we have also imported the mux made in assignment 1.

First, we call the switch function which takes start, pause, reset, and continue as input and returns the corresponding values of enable_watch and reset_watch.

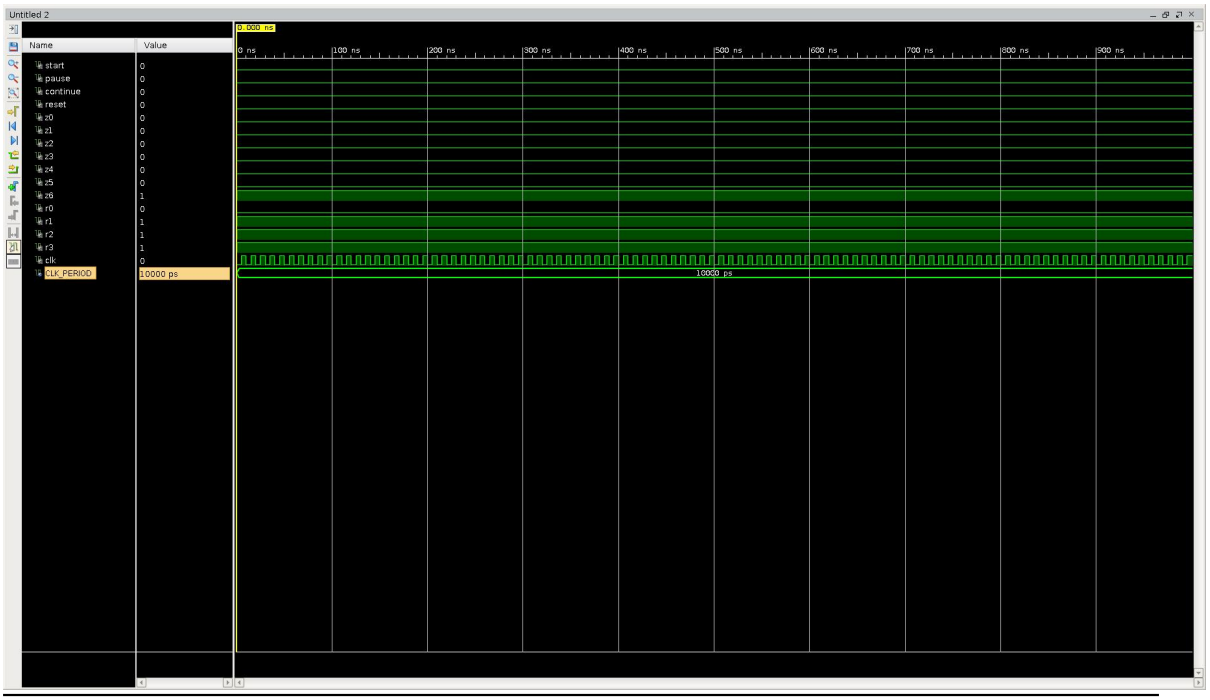
Next, we gave the inbuilt clock waveform as input to the clock_divider and it returned an output clock of 10 Hz(0.1 sec). This clock was then given as input to the Numeric which then returned the output clock of 1sec and the decimal value of the rightmost digit. The 1sec clock again goes as an input to numeric and returns an output clock of 10 sec and the decimal value of the second digit from the right. This output clock of 10 sec then goes as an input into Numeric 6 (as the second digit from the left can go from 0 to 5) and returns an output clock of 1min and the decimal value of the second digit from the left. Finally, this 1min clock goes as an input in Numeric which returns the decimal value of the leftmost digit.

The decimal values of each of these 4 digits are converted to the binary by calling the int_bin functions which return 4 bits for every digit. These 16 bits are then given as input to the mux of assignment 1 which activates the cathodes and anodes of the 7-segment display, corresponding to the digit to be displayed at a refresh rate of 2.5 seconds.

- **BLOCK DIAGRAM:**



- **SIMULATION SNAPSHOTS:**



- **SYNTHESIS REPORT:**