REPORT FOR COL215 HARDWARE ASSIGNMENT-2

Objective:

To create a stopwatch with display format (M:SS:T) where M represents minutes, S represents seconds and T represents a tenth of a second using the 7-segment displays and switches of the Basys3 FPGA board.

Procedure:

The stopwatch works by using the on-board clock to cause an update after every 0.1 seconds (10^7 clock cycles). Let us name the displays as M, S1, S2 and T from left to right. The clock updates the 'T' display mentioned which further cascades onto the 'S2', 'S1' and 'M' displays in this order. A binary number associated with each display is outputted which in turn becomes the input for the code of previous assignment to display the digits on the board. Two variables, named enable_watch and reset_watch are used to control the current state of the watch.

The stopwatch supports the below functions.

1. Start
   When switch goes from 0 to 1, counter starts
   enable_watch set to 1
   reset_watch set to 0

2. Pause
   When switch goes from 0 to 1, counter stops at current value
   enable_watch set to 0

3. Continue
   When switch goes from 0 to 1, counter starts increasing again from current value.
   enable_watch set to 1

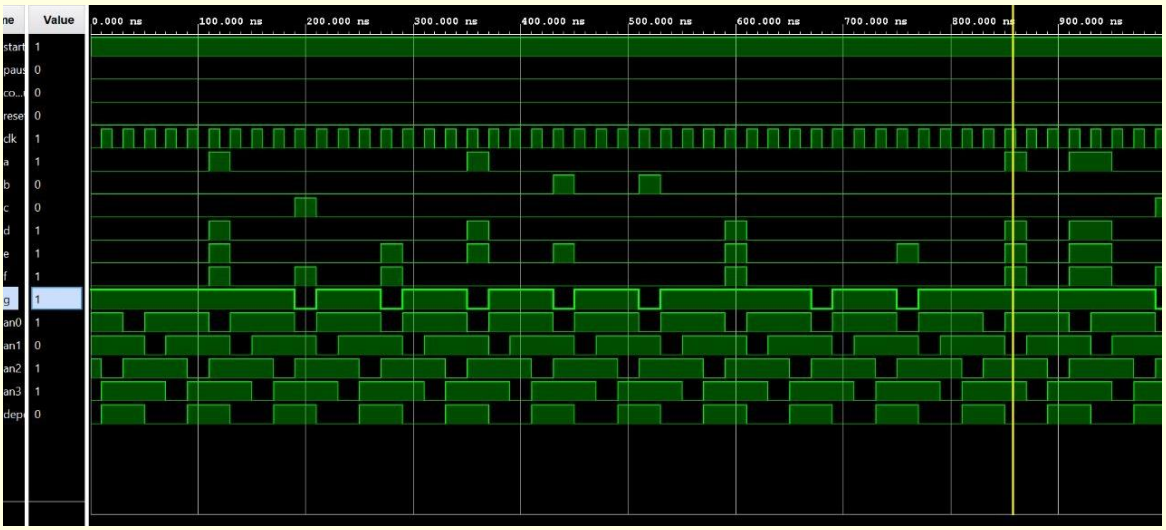
4. Reset
   When switch goes from 0 to 1, all numbers are set to 0 and counter stops there.
   enable_watch set to 0
   reset_watch set to 1

Each of the above are activated by a rising edge at their corresponding switch i.e. a change from low mode to high mode. Note that the change of a switch from high to low does not bring about any change in variables or in the state of the system.

Rising_Edge Detection:

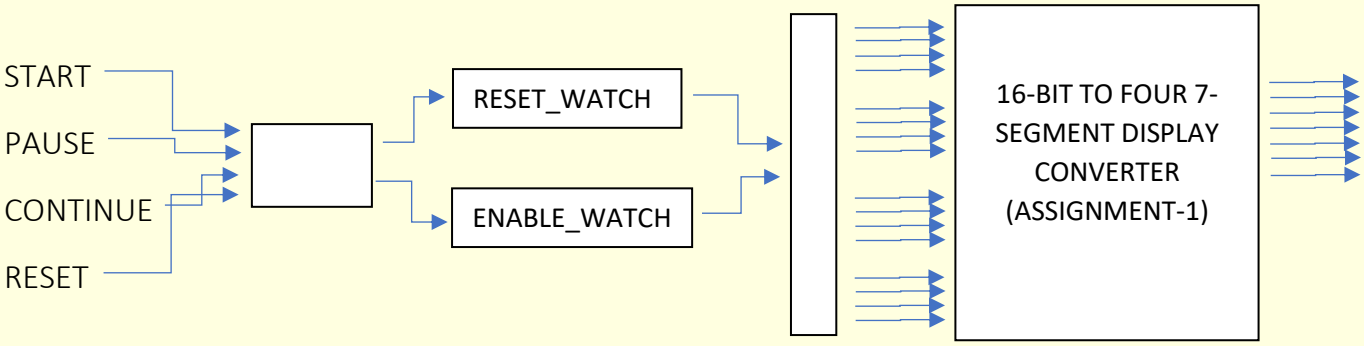
The above functions require that we are able to detect a rising edge for the signals associated with each of them. This is accomplished by storing the value of each switch in a separate signal and checking along with the current value of signal. We then create a process which checks if the stored value is 0 and the current value is 1 after every clock cycle which would confirm a rising edge and allow for the corresponding change. We then update the stored value to accommodate the change. This is done for all the 4 signals (switches).

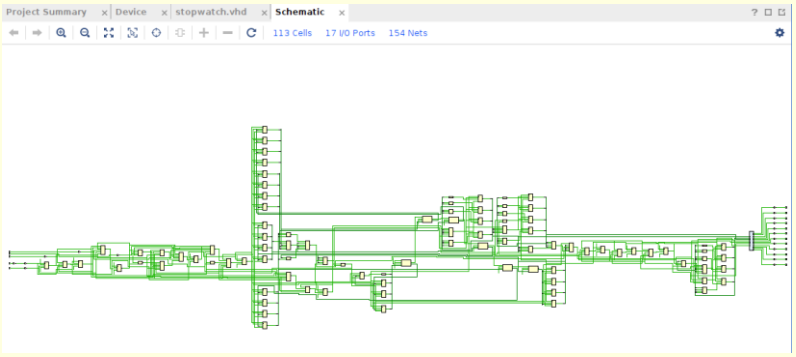
Simulation Snapshot:



We can clearly observe from the above snapshot that 4 consecutive 7-cathode formations represent a particular instant of time with each formation representing a single LED. We observe that in the initial few such configurations, 3 displays show the number 0 while 1 display ('T') goes sequentially from 0 to 9 (can be observed by looking at LED's corresponding to an0) and then the value corresponding to an1 changes (which can be seen in the figure towards the right). Note that the frequency has been changed for the simulation.

Block Diagram:





Synthesis Report:

Site Type	Used	Fixed	Prohibited	Available	Util%
Slice LUTs*	50	0	0	20800	0.24
LUT as Logic	50	0	0	20800	0.24
LUT as Memory	0	0	0	9600	0.00
Slice Registers	66	0	0	41600	0.16
Register as Flip Flop	66	0	0	41600	0.16
Register as Latch	0	0	0	41600	0.00
F7 Muxes	0	0	0	16300	0.00
F8 Muxes	0	0	0	8150	0.00

Site Type	Used	Fixed	Prohibited	Available	Util%
Block RAM Tile	0	0	0	50	0.00
RAMB36/FIFO*	0	0	0	50	0.00
RAMB18	0	0	0	100	0.00

Site Type	Used	Fixed	Prohibited	Available	Util%
DSPs	0	0	0	90	0.00

We can clearly see the number of used LUTs, BRAMs (Block RAM), DSPs and Flip-Flops. This is the synthesis utilization report. The number of LUTs further decreased to 48 post-implementation.