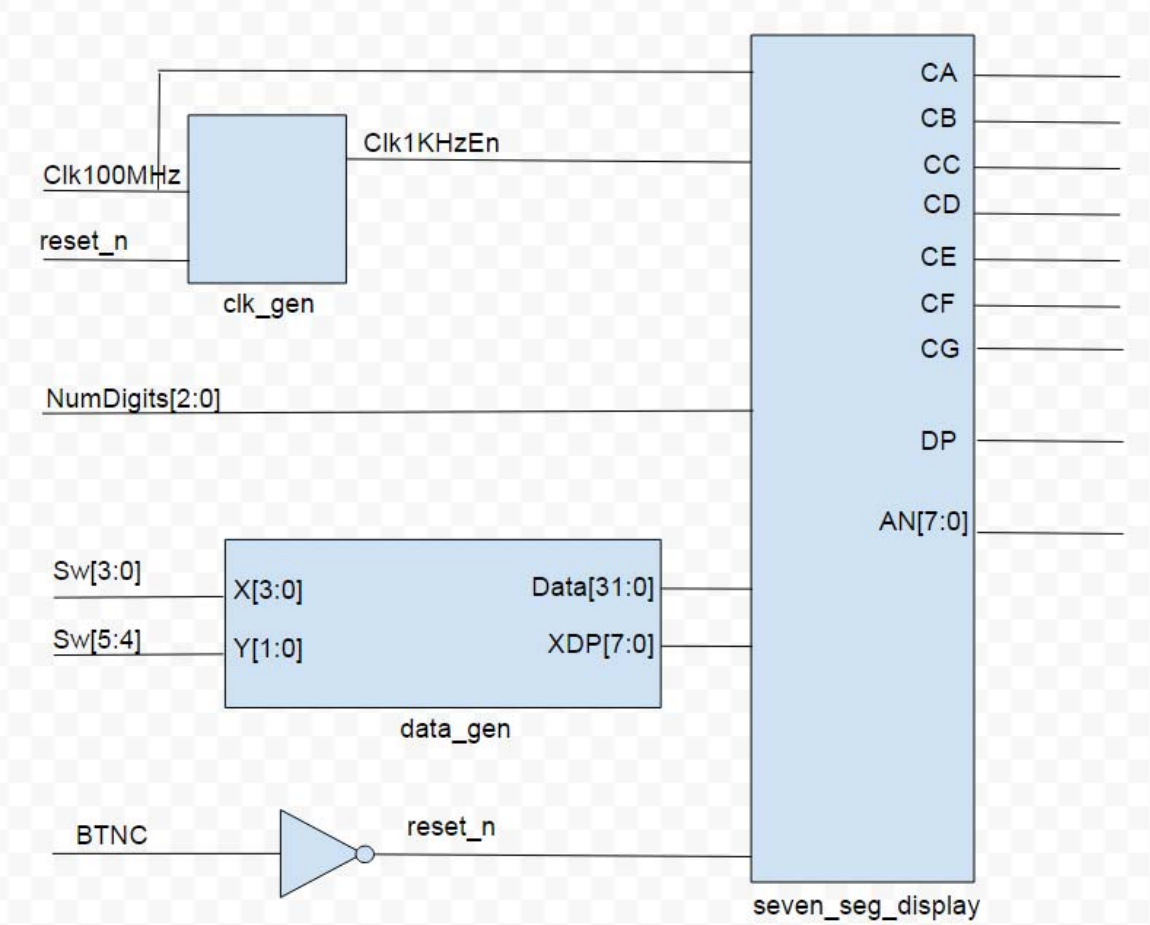
LAB #4 – Time Multiplexed Display



Note: reset_n from the bottom is connected to the top_reset_n.

Prelab:

1) The clk_gen circuit accepts a 100 MHz clock and generates a 1 KHz clock enable signal Clk1KHzEn. The period of Clk1KHz is 1 ms. The pulse width (when high) is 10 ns wide. The signal is low in the remaining time of the 1ms period. BTNC (center push button on your board) when pressed resets the counter inside clk_gen module. Write the verilog code for this module and place it in file clk_gen.v.

2) The data_gen.v module accepts 6 bits from the switches in your board. Generate Data and XDP as below:

   a. Data = {28'hECE3300, X}

   b. XDP = {6'b010101, Y}

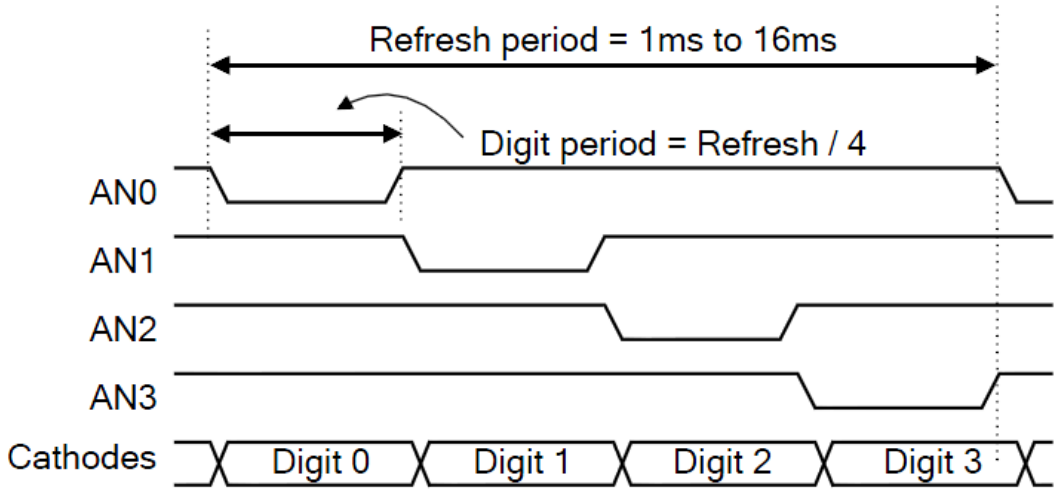
3) The seven_seg_display.v circuit is used to display data on seven segment digits. When NumDigits is 0, one 7 segment digit will be on (the rightmost segment). When NumDigits is 1, two 7 segment digits will be on. When NumDigits is 7, eight segment digits will be on.

   In this lab you will design and implement a time multiplexed circuit to control an 8-digit seven-segment display using Verilog. Each seven segment display requires 8 pins to operate. If these were all driven independently, it would require 64 pins. Multiplexing is used to save pins, and all eight displays can be driven using only 15 pins. The multiplexing is accomplished by using the common anodes as enable lines for each display.

   A property of the human eye known as persistence of vision (POV) causes the human vision system to hold an image for a short amount of time. If the displays are toggled faster than the time it takes for the image to be lost, it appears as though all the displays are on at the same time, when in reality, only one display is illuminated at a time.

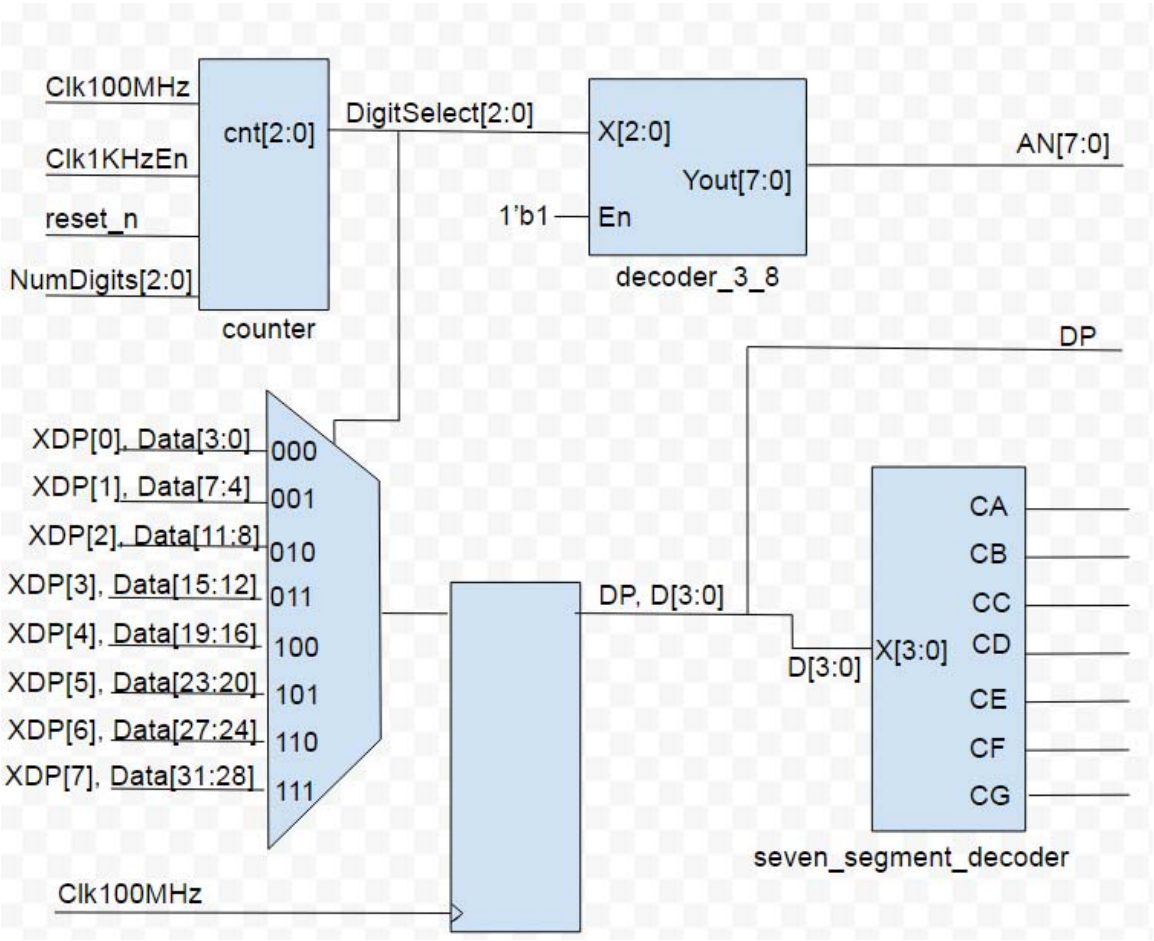
   For each of the four digits to appear bright and continuously illuminated, all four digits should be driven once every 1 to 16 ms for a refresh frequency of 1 KHz to 60 Hz. If refresh rate is slowed to around 45 Hz, a flicker can be noticed.

   An example timing diagram for a four-digit controller is shown below:



For our lab we are going to use digit period of 1 ms. For an eight digit display the refresh period is 8 ms for a refresh frequency of 125 Hz. For a four digit display the refresh period is 4 ms for a refresh frequency of 250 Hz.

Block diagram of seven segment display circuit is given below.



a. Write module counter.v to implement a counter circuit to count from 0 to NumDigits. The count changes every 1ms when NumDigits is greater 1. When NumDigits is 0 output cnt[2:0] is set to 3'b000. Signal reset_n when low resets the counter in this module.

b. Module decoder_3_8.v is the same from Lab 2.

c. DP and D[3:0] are registered version of multiplexer output from respective XDP and Data[3:0] signals. Block after the multiplexer represent 5 positive edge D flip-flops.

d. Module seven_segment_decoder.v is the same from Lab 2.

e. Write module seven_seg_display.v to describe the circuit given in the block diagram.

4) Describe the complete circuit diagram (given in page 1) in module lab4_top.v.

5) Write a simulation test bench to simulate the complete circuit (i.e. lap4_top). You should place your code in file lab4_tb.v. In you test bench include the following two lines to generate a 100 MHz clock.

```
initial Clk100MHz = 0;
always #5 Clk100MHz = ~Clk100MHz;
```

6) Submit hard copy of all your verilog design files, testbench and simulation results. Each design file, testbench file and simulation result should have your name on it. On your waveform show internal signals of clk_gen, when MAX_COUNT=9. For synthesis it should be set to 99999. For simulation, first line below is commented. For synthesis, second line below is commented.

```
localparam MAX_COUNT = 99999;   // synthesis
localparam MAX_COUNT = 9;       // simulation
```

Lab:

1. Complete the design using slide switches for your inputs. Uncomment clock signal in your constraint file to use the 100 MHz clock on your board.
2. Implement your design on the FPGA and demonstrate.
3. Describe your observations and state problems you encountered and how you solved them.