

Lab 4 Report

Design Specification

Lab 3_5:

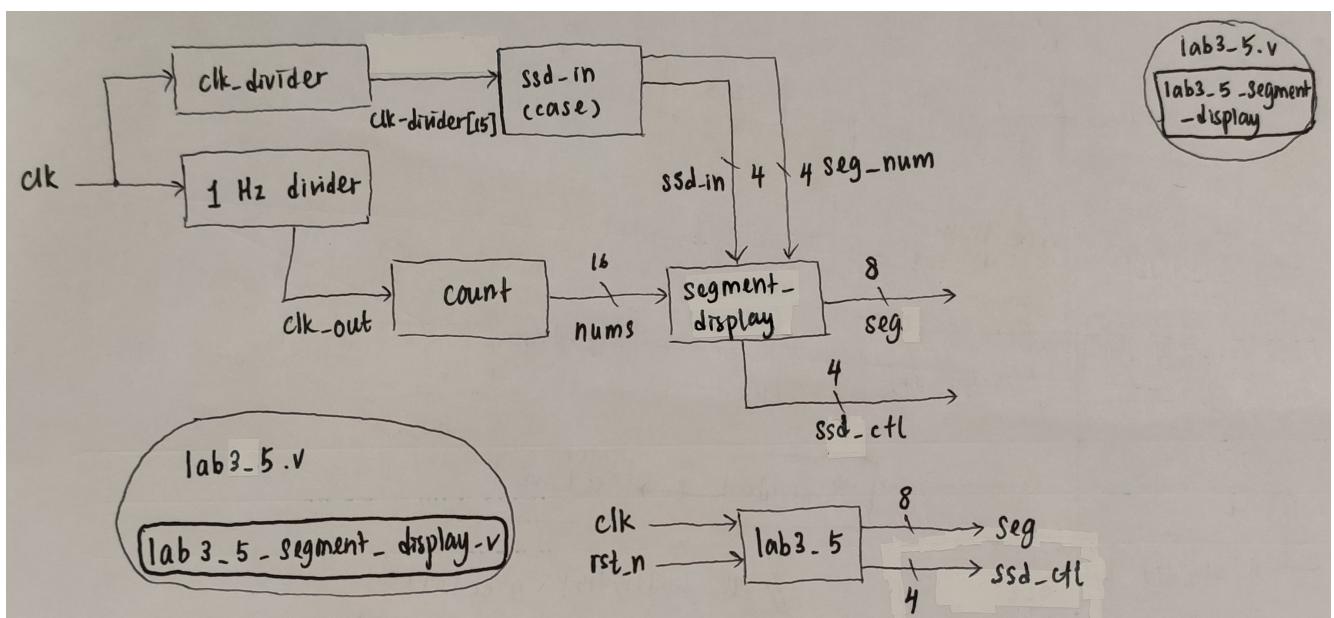
Top module :

module lab3_5:

```
    input clk,  
    input rst_n,  
    output [7:0] seg,  
    output [3:0] ssd_ctl
```

module lab3_5_segment_display :

```
    output [7:0] segs,  
    output [3:0] ssd_in,  
    input [15:0] nums,  
    input rst_n,  
    input clk
```



Lab 4 2:

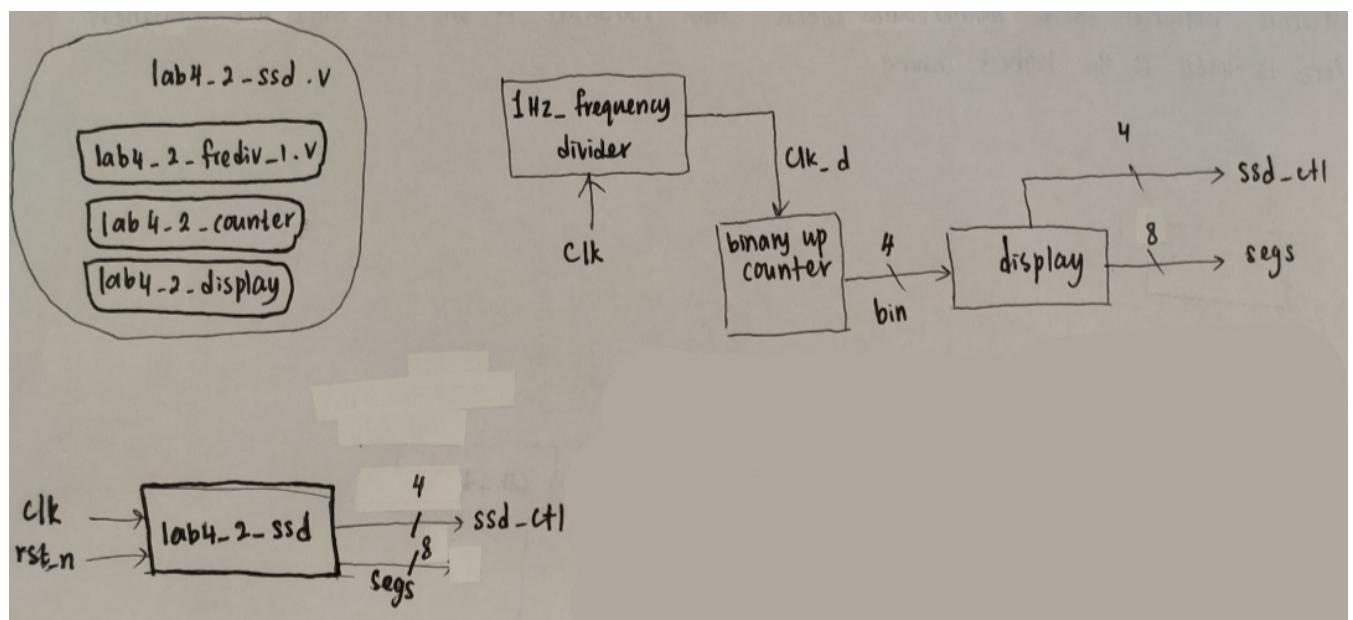
Top module:

```
module lab4_2_ssd:  
output [3:0] ssd_ctl,  
output [7:0] segs,  
input clk,  
input rst_n
```

```
module lab4_2_frediv_1:  
input clk,  
input rst_n.  
output clk_out1
```

```
module lab4_2_counter:  
output [3:0] q,  
input clk,  
input rst_n
```

```
module lab4_2_display:  
output [7:0] seg,  
input [3:0] i
```



Lab 4_3:

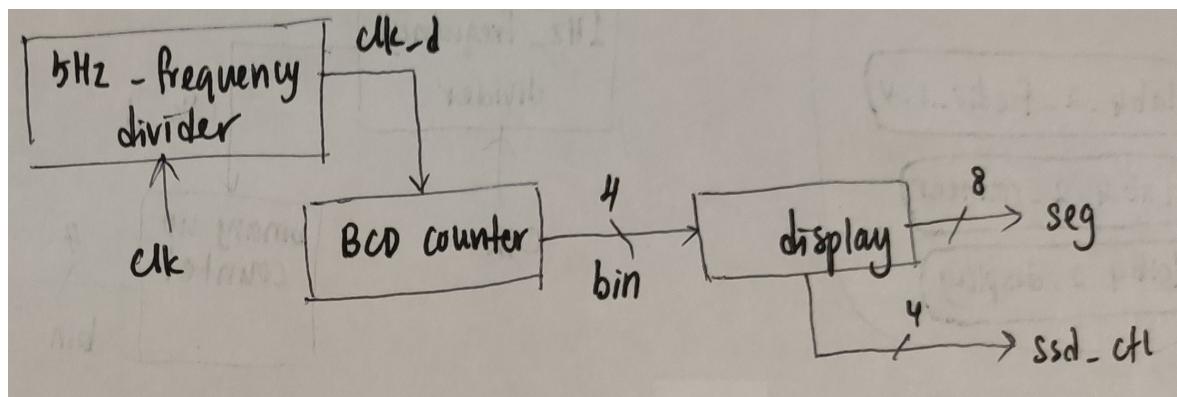
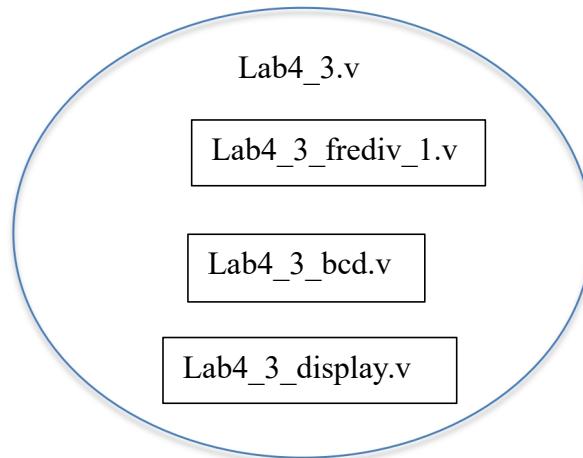
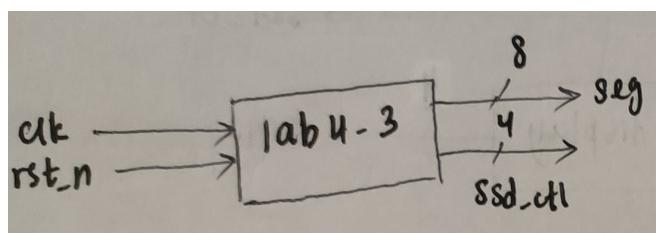
Top module:

```
module lab4_3:  
output [3:0] ssd_ctl,  
output [7:0] seg,  
input clk,  
input rst_n
```

```
module lab4_3_frediv_1:  
input clk,  
input rst_n,  
output clk_out1
```

```
module lab4_3_bcd:  
output [3:0] q,  
input clk,  
input rst_n
```

```
module lab4_3_display:  
output [7:0] seg,  
input [3:0]
```



Lab 4:

Top module:

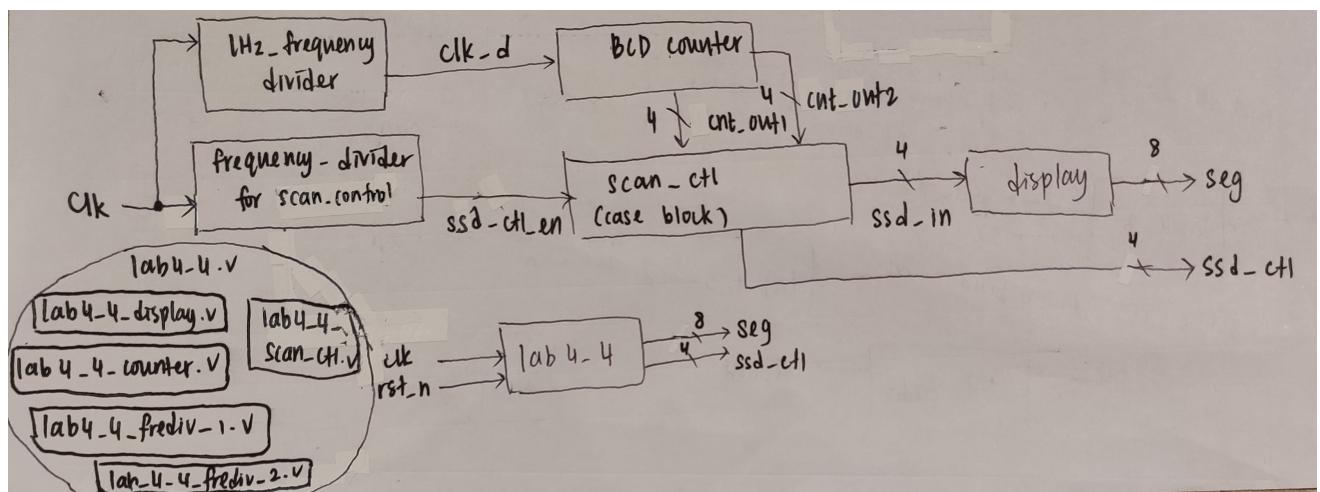
```
module lab4_4:  
output [3:0] ssd_ctl,  
output [7:0] seg,  
input clk,  
input rst_n
```

```
module lab4_4_frediv_1:  
input clk,  
input rst_n  
output clk_out1
```

```
module lab4_4_frediv_2:  
input clk,  
input rst_n,  
output clk_out2
```

```
module lab4_4_scan_ctl:  
output [3:0] ssd_in, // Binary data  
output [3:0] ssd_ctl, // scan control for 7-segment display  
input [3:0] in0,in1,in2,in3, // binary input control for the four digits  
input ssd_ctl_en // divided clock for scan control
```

```
module lab4_4_display:  
output [7:0] seg,  
input [3:0] i
```



Design Implementation

Lab 3 5:

In this lab, I had design a top module, which is the Lab3_5, and a module called lab3_5_segment_display. The top module contains the 1Hz frequency divider and a counter which count for 8, because there are 8 outputs, which are NTHUEECS. The counter will then link to the num(s), which is the alphabet will be shown on the segment in every second. There is a nums, which will combine the every alphabet's(4 alphabet will be shown at one time, and each alphabet is assigned as 4 bits, so the nums will contain 16 bits.)

In the lab3_5_segment_display, contains a clock divider, which is used to determined which segments of the alphabet will be shown on the four digit 7-segment display. Then it will keep shifting the alphabet in every second. The NTHUEECS will start shifting when the reset signal is received.

Lab 4 2:

In the top module, assign ssd_ctl equal to 4'b0000 as to let all the four digit 7-segment display bright, since it is a low active device. The lab is done by starting with creating a 1Hz frequency divider which then connect to a binary up counter. The binary up counter's result will then shown on the four digit 7-segment display, which started from 0 to F according to the counter.

Lab 4 3:

The top module the ssd_ctl is assigned 4'b1110, as to fulfill the requirement of designing a single digit BCD up counter. Because of the low active device, so the device will light up at last segment of the four digit 7-segment display, which is the part assigned as 0. The experiment was completed by starting with the 5Hz frequency divider. In order to create a 5Hz frequency divider, we replace the 1Hz divider (which count for 50M), by counting up to 10M only. The BCD counter will count from 0 up to 9 only. The display module is simply full with the segment needed to display on the four digit 7-segment display according to the counter.

Lab 4 4:

This experiment is to design a 2 digit BCD up counter, which is counting from 00 up to 99. The experiment was done by starting with the 1Hz frequency divider and use this frequency divider to create a 1Hz BCD counter. The BCD is designed with 2 outputs, this is because the requirement of the experiment is 2 digit. The counter is designed when the first digit counted up to 9, then it will start from 0 again, while the second digit will be added by 1. The counting process keep going when reached 99, both digit will be started from 00 again. The frequency divider for the scan_ctl was designed with only 1 bit(the Most Significant Bit of 16bits), this is because there are just 2 option for the four digit 7-segment display to bright, since we only need 2 digit in this experiment.

Discussion

From the question of Lab3_5, I had learned about how to let the four digit 7-segment display to keep shifting the alphabet as requested. It is kind of hard for me to understand and implement it when completing this experiment. But after that, I found that it can be implemented in a easier way, which is by the aids of the shifter from prelab 3_2.

As a conclusion of what I had learned from Lab 4 is that the scan control of the four digit 7-segment display. It help me to have a better understanding of how to implement a different frequency divider for different digit of four digit 7-segment display. For example, the question 3 and 4 of Lab 4, which required us to let only a single digit and 2 digit of four digit 7-segment display with a different frequency divider.

Conclusion

In a conclusion, from these experiments I had learned about how to code for the four digit 7-segment display to display different type of results. For example, the different alphabets and digits as assigned, and different type of frequency divider as designed, so that the results can be displayed in a different frequency (like 1 second display, almost 2 second for display from 0 to 9), and the number of digits needed to be displayed on the four digit 7-segment display (1 and 2 digits with the others are not lighted or even the 4 digits with different alphabets shifting on it).