

# lab4

## lab3\_5

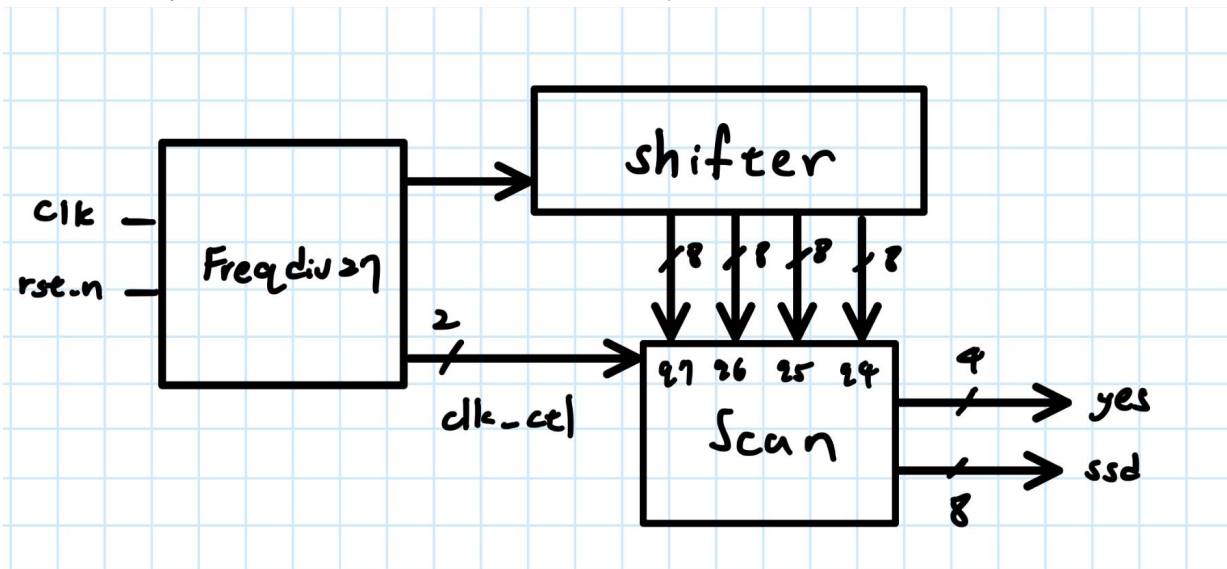
### Design Specification

Input: clk, rst\_n.

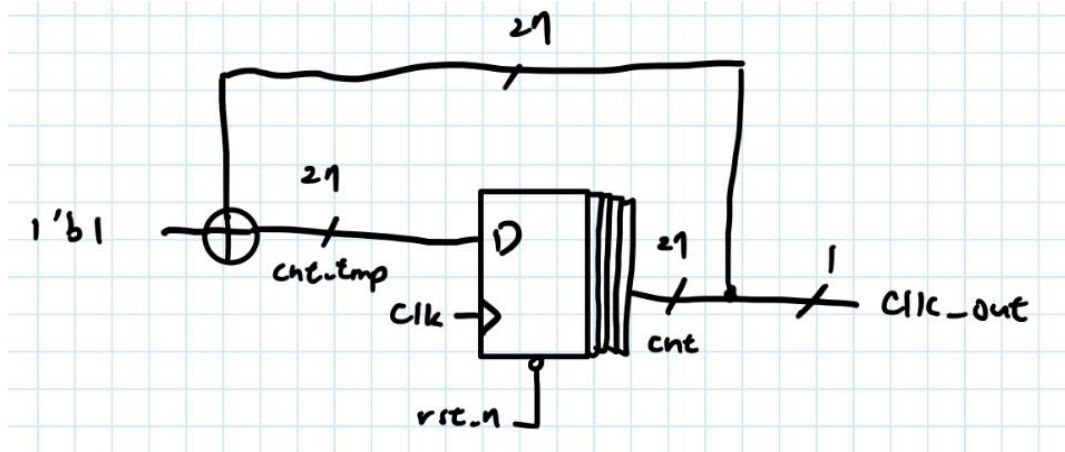
Output: [3:0] yes(anode of seven-segment display),  
[7:0] ssd(cathode of seven-segment display).

### Design Implementation

我將實驗拆成3個部分, 第一個當然就是利用除頻後的clk, 以及使四個七段顯示器顯示不同的字母的scan, 最後就是造成跑馬燈效果的shifter。



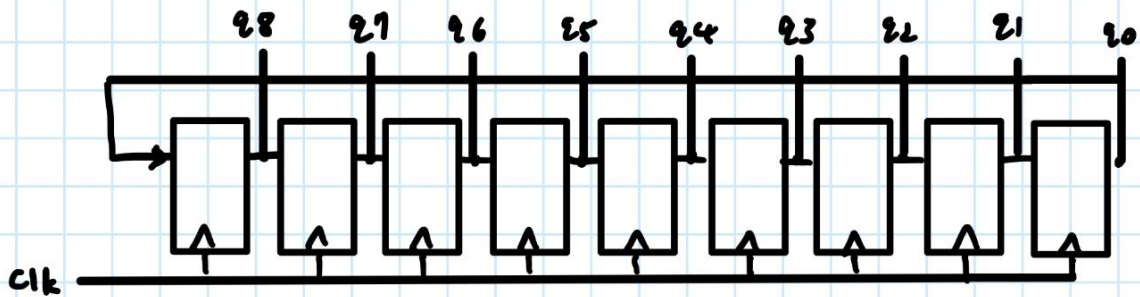
clk是利用27 bit 的binary up counter 利用最高位當成clk\_out製造一個接近一赫茲的clk



scan是利用Frequency divider的第16、17位去控制顯示的位置

Scan	clk_ctl	in_pwr	ssd	yes
	00	28 27 26 25	28	0111
	01	28 27 26 25	27	1011
	10	28 27 26 25	26	1101
	11	28 27 26 25	25	1110

shifter



- 28 = 8'b11010101 → 7
- 27 = 8'b11100001 → 7
- 26 = 8'b10010001 → H
- 25 = 8'b10000011 → U
- 24 = 8'b11111111
- 23 = 8'b01100001 → E
- 22 = 8'b01100001 → E
- 21 = 8'b01100011 → 7
- 20 = 8'b01001001 → 5

I/O	fcystal	ssd7	ssd6	ssd5	ssd4	ssd3	ssd2	ssd1	ssd0	rst_n
Site	W5	V14	U14	U15	W18	V19	U19	E19	U16	R2

I/O	yes3	yes2	yes1	yes0
Site	W4	V4	U4	U2

## Discussion

一開始看到這個題目，就想到前一個Lab 8Bit的 shifter，為了做出跑馬燈的效果，使得每一個時序後向左平移，這個時序也就是使用除頻後的clk，再利用scan使得四個七段顯示器分別顯現出前面四位數字，而因為要顯現出NTHU EECS，所以第一步是先畫出七段顯示器的對照表，並依照順序放入shifter裡面，就可以達到題目的要求。

## lab4\_1

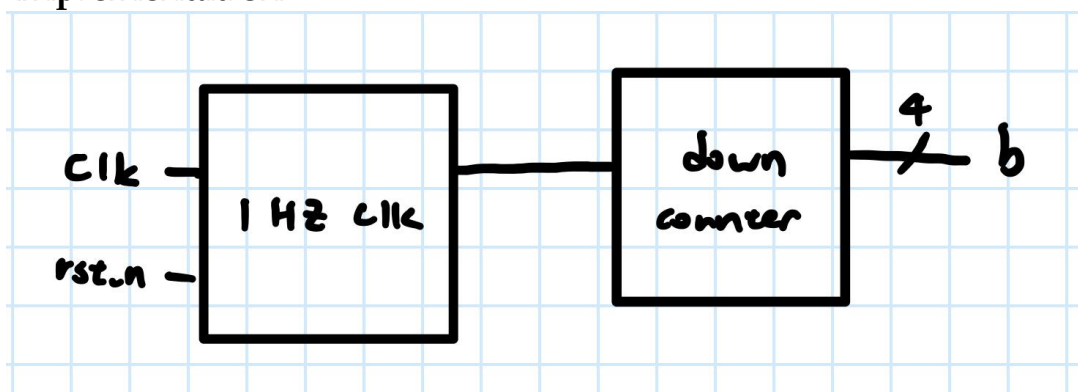
### Design Specification

For a down counter

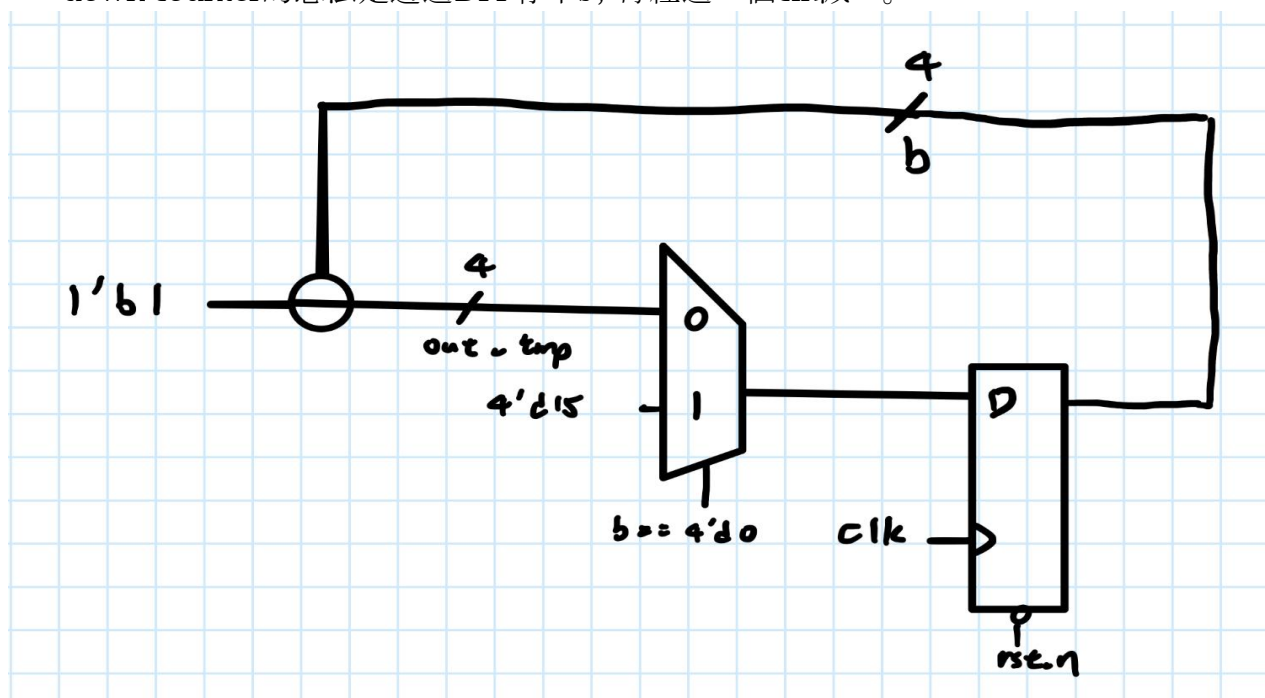
Input: clk, rst\_n

Output: [3:0] b

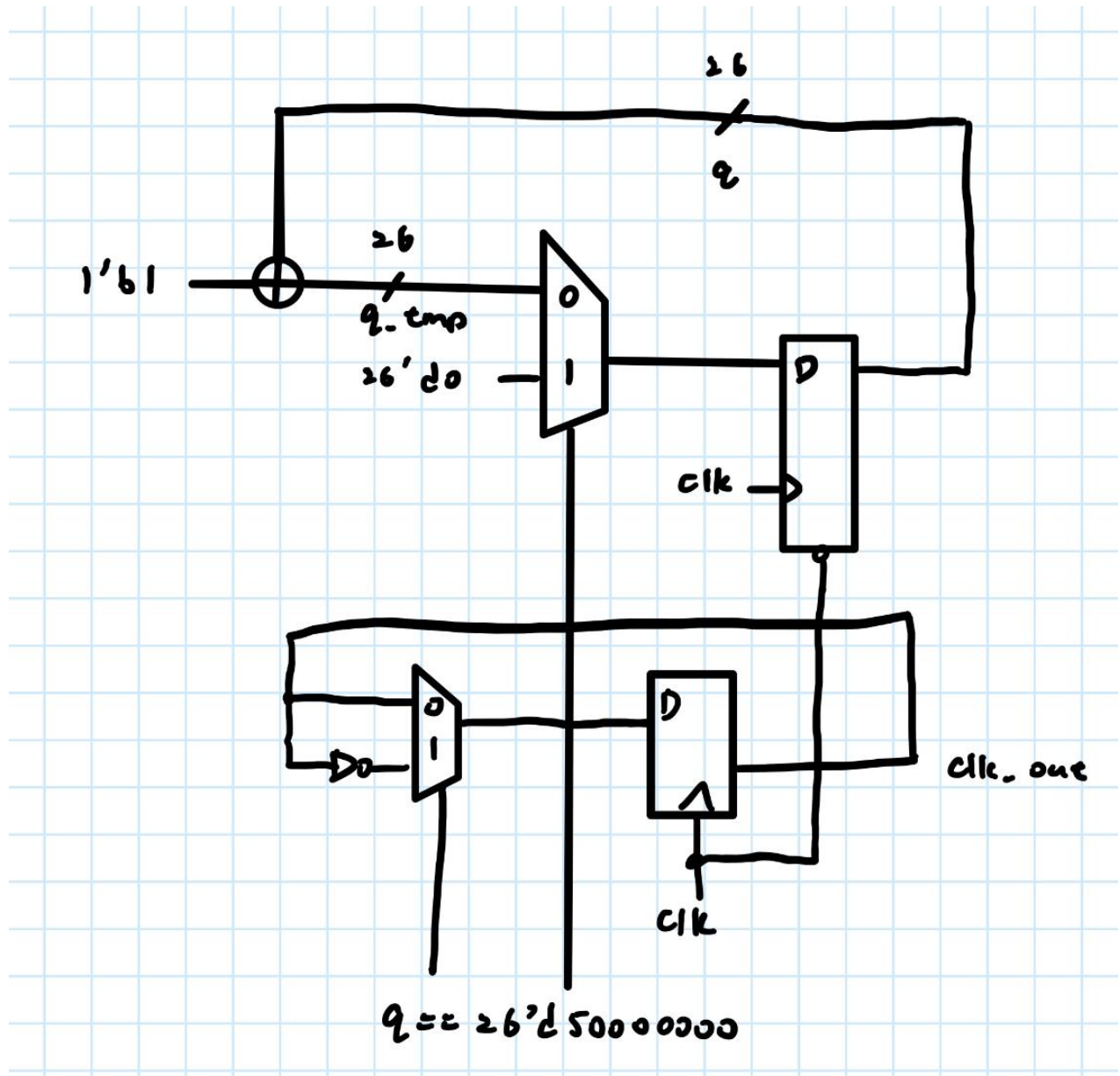
### Design Implementation



down counter的想法是透過DFF存下b，每經過一個clk減一。



clk



I/O	fcystal	b3	b2	b1	b0	rst_n
Site	W5	V19	U19	E19	U16	R2

### Discussion

本實驗是利用一赫茲的clk做出4 bit的down counter, 是個簡單的小lab

### lab4\_2

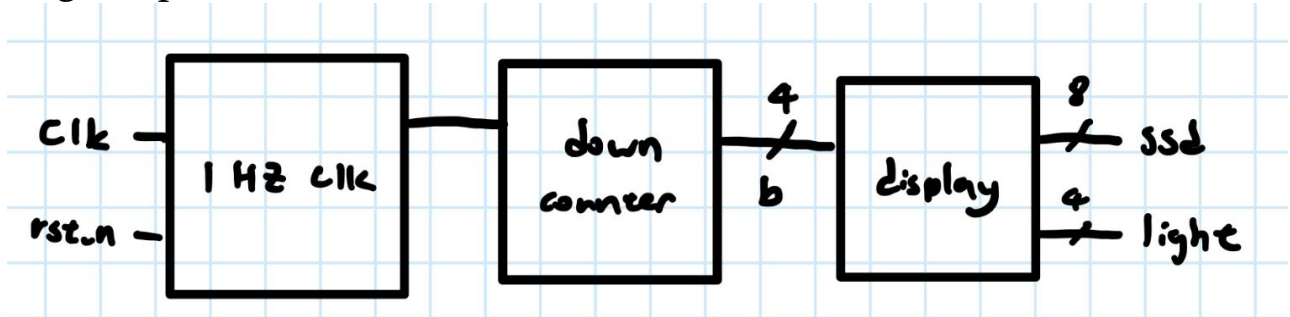
### Design Specification

For binary counting in 7-segment display

Input: clk, rst\_n

Output: light, ssd

### Design Implementation



display(查表)

b [3:0]				light[3:0]				SSD [7:0]							
i3	i2	i1	i0	d3	d2	d1	d0	A	B	C	D	E	F	G	H
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
0	0	0	1	0	0	0	1	1	0	0	1	1	1	1	1
0	0	1	0	0	0	1	0	0	0	1	0	0	1	0	1
0	0	1	1	0	0	1	1	0	0	0	0	1	1	0	1
0	1	0	0	0	1	0	0	1	0	0	1	1	0	0	1
0	1	0	1	0	1	0	1	0	1	0	0	1	0	0	1
0	1	1	0	0	1	1	0	0	1	0	0	0	0	0	1
0	1	1	1	0	1	1	1	0	0	0	1	1	1	1	1
1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1
1	0	0	1	1	0	0	1	0	0	0	0	1	0	0	1
1	0	1	0	1	0	1	0	0	0	0	1	0	0	0	1
1	0	1	1	1	0	1	1	1	1	0	0	0	0	0	1
1	1	0	0	1	1	0	0	0	1	1	0	0	0	1	1
1	1	0	1	1	1	0	1	1	0	0	0	0	1	0	1
1	1	1	0	1	1	1	0	0	1	1	0	0	0	0	1
1	1	1	1	1	1	1	1	0	1	1	1	0	0	0	1

I/O	fcystal	light 3	light 2	light 1	light 0	ssd7	ssd6	ssd5	ssd4	ssd3	ssd2	ssd1	ssd0	rst_n
Site	W5	V19	U19	E19	U16	V14	U14	U15	W18	V19	U19	E19	U16	R2

### Discussion

這個lab是結合了lab4\_1加上display，所以想法是直接利用上個lab的b當作display的輸入，然後直接轉換成七段顯示器。

### lab4\_3

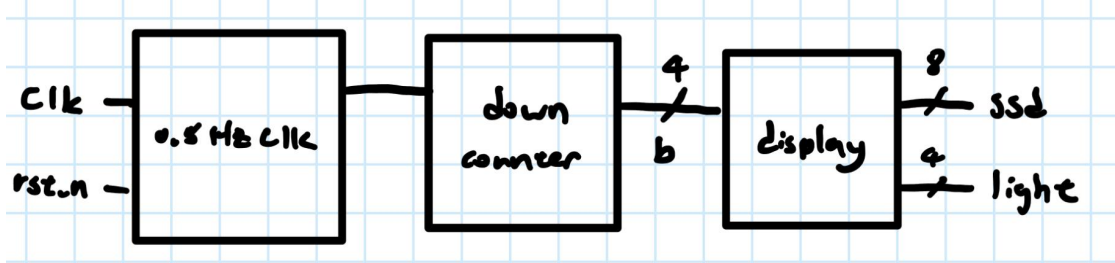
### Design Specification

For a single digit BCD up counter on the seven-segment display with 0.5-Hz

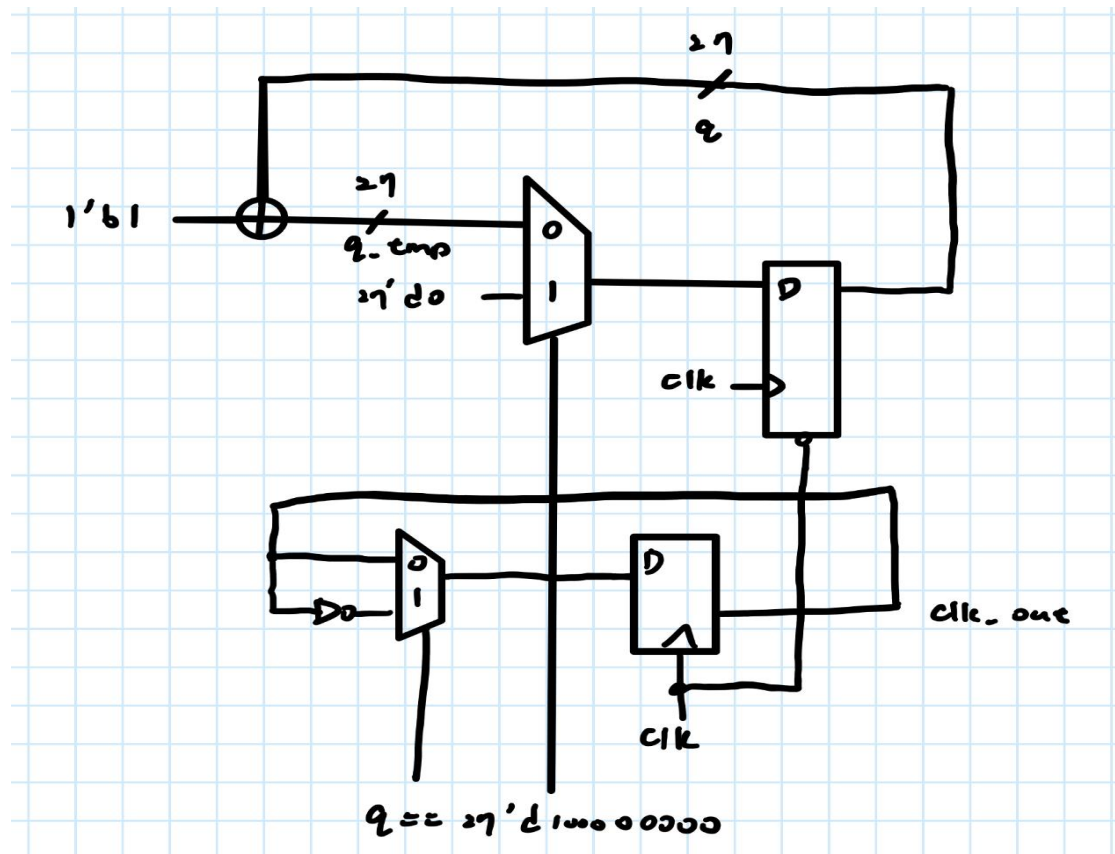
Input: clk, rst\_n

Output: light, ssd

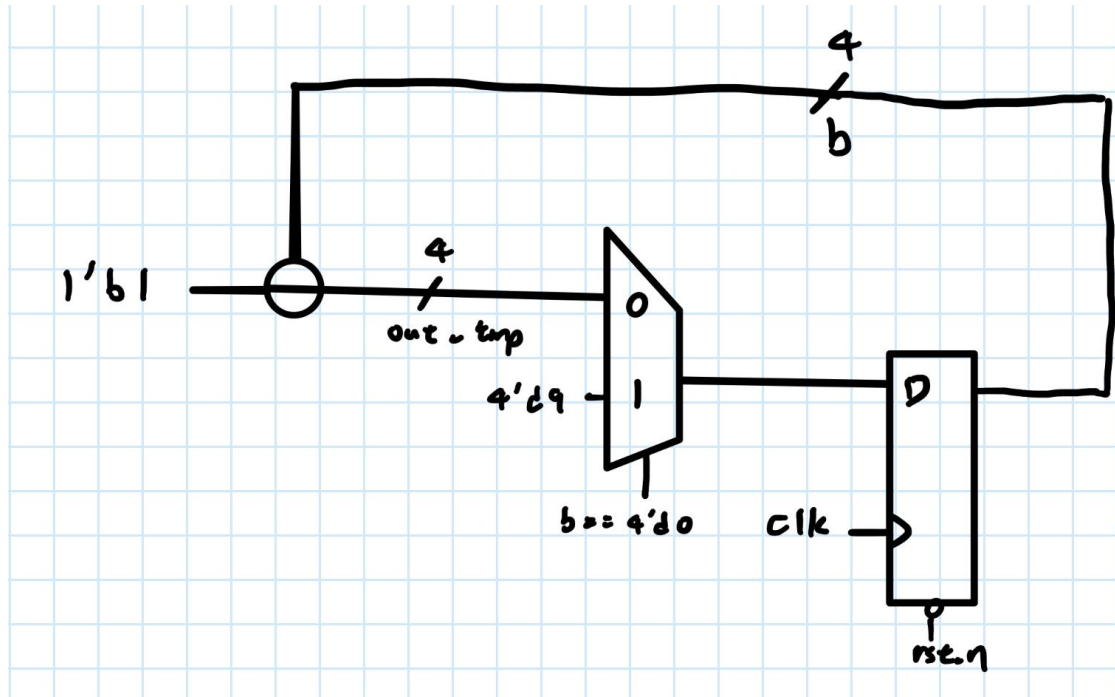
### Design Implementation



0.5Hzclk



## BCD down counter



I/O	fcystal	light 3	light 2	light 1	light 0	ssd7	ssd6	ssd5	ssd4	ssd3	ssd2	ssd1	ssd0	rst_n
Site	W5	V19	U19	E19	U16	V14	U14	U15	W18	V19	U19	E19	U16	R2

## Discussion

這個lab跟上一個lab最大的差異是數到0時，下一個會歸零到9，再從9開始數下來，還有使用的是0.5赫茲的clk，所以想法是當b等於0時，直接給定下一個值為9，而clk的部分原本是數至50M只要改為100M頻率就會變成0.5赫茲。

## Conclusion

從這次的lab中更加熟悉block之間的連接，也更加熟練地運用透過不同功能的block用top module包起來，尤其在lab3-5花了很多的時間去理解接線的方式及一些小技巧，例如把所有誇號內的變數都記得先拉出來定義等等，lab4的題目就相對平易近人，但也是個認我熟悉接線的好題目。

## References

老師的ppt