邏輯設計實驗 Lab03 結報

104060012 邱怡庭

1 Frequency Divider: Construct a 25-bit synchronous binary counter.

Use the MSB of the counter, we can get a frequency divider which provides a (1/2)^25 frequency output (fout) of the original clock (fcrystal, 40MHz). Construct a frequency divider of this kind.

1.1 Write the specification of the frequency divider.

1.2 Draw the block diagram of the frequency divider.

1.3 Implement the frequency divider with the following parameters.

I/O	fcrystal	fout
Site	R10	H5

3-1 (4/5)
Design Specification-1
block diagram of the design or Logic Diagram-1
I/O pin assignment-1
Discussion +Conclusion-1

Design Specification

output: clk_out;

output: [1:0]clk_ctl;

input: clk;

input: rst_n;

reg: clk_out;

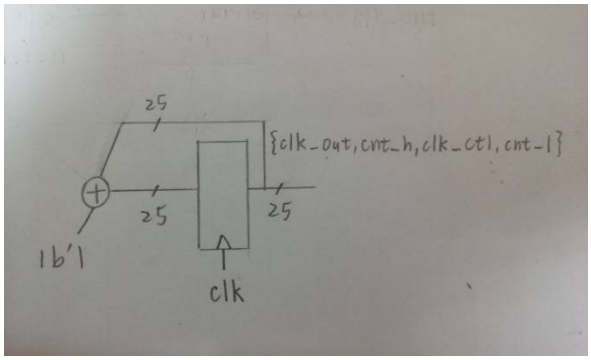
reg: [1:0]clk_ctl;

reg: [14:0]cnt_l;

reg: [6:0]cnt_h;

reg: [24:0]cnt_tmp;

block diagram:



Design Implementation

logic function:

cnt_tmp = {clk_out,cnt_h,clk_ctl,cnt_l} + 1'b1;

I/O pin assignment:

NET "clk_out" LOC =H5;

NET "clk" LOC =R10;

NET "rst_n" LOC =T1;

Discussion:

除頻器的功能為降低頻率以便觀察，當 40MHz 除以 2^25 次方後，頻率便會接近 1，因此 LED 約每秒閃爍一次。

2 Construct a single digit BCD up counter with the divided clock as the clock frequency and display on the seven-segment display.

2.1 Construct a BCD up counter.

2.2 Construct a BCD-to-seven-segment display decoder.

2.3 Combine the above two together.Design Specification

output: [14:0] display; // SSD display output

input: clk; // global clock input

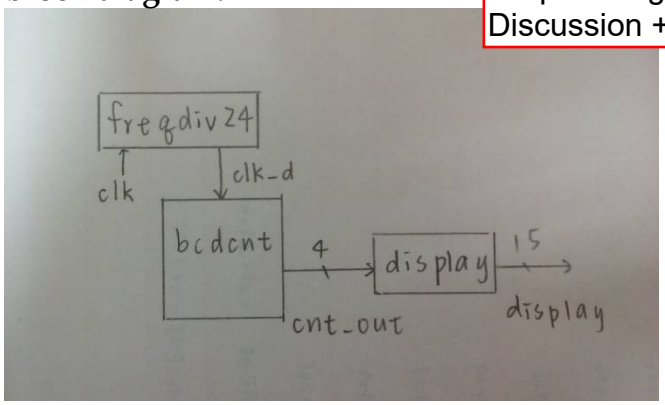
input: rst_n; // active low reset

wire: clk_d;

wire: [3:0]cnt_out;

3-2 (13/15)
Design Specification(2/2)
block diagram of the design or Logic Diagram (5/5)
I/O pin assignment (3/3)
Discussion +Conclusion(3/5)

block diagram:



Design Implementation

logic function:

display:

當 bcd=4'd0: display = 15'b0000_0011_1111_111; //0

當 bcd=4'd1: display = 15'b1111_1111_1011_011; //1

當 bcd=4'd2: display = 15'b0010_0100_1111_111; //2

當 bcd=4'd3: display = 15'b0000_1100_1111_111; //3

當 bcd=4'd4: display = 15'b1001_1000_1111_111; //4

當 bcd=4'd5: display = 15'b0100_1000_1111_111; //5

當 bcd=4'd6: display = 15'b0100_0000_1111_111; //6

當 bcd=4'd7: display = 15'b0001_1111_1111_111; //7

當 bcd=4'd8: display = 15'b0000_0000_1111_111; //8

當 bcd=4'd9: display = 15'b0000_1000_1111_111; //9

default: display = 15'b1111_1111_1111_111; //DEF

***freqdiv24**:*

cnt_tmp = {clk_out,cnt} + 1'b1;

bcdcnt:

當 out=4'd9 時 -> 歸零: tmp_cnt=4'd0;

其他狀況 -> 加一: tmp_cnt = out + 1'b1;

I/O pin assignment:

```
NET "Display[14]" LOC = P6;
NET "Display[13]" LOC = N4;
NET "Display[12]" LOC = V5;
NET "Display[11]" LOC = T5;
NET "Display[10]" LOC = U7;
NET "Display[9]" LOC = R3;
NET "Display[8]" LOC = N5;
NET "Display[7]" LOC = R5;
NET "Display[6]" LOC = T3;
NET "Display[5]" LOC = T4;
NET "Display[4]" LOC = V4;
NET "Display[3]" LOC = V7;
NET "Display[2]" LOC = R7;
NET "Display[1]" LOC = T7;
NET "Display[0]" LOC = U5;
NET "rst_n" LOC = T1;
NET "clk" LOC = R10;
```

Discussion:

因為只需設計一位數的計時器，所以我沒有加入 scan_ctl，因此四個 14 段顯示器會同時變化!

3 Construct a 2-digit BCD up counter (from 00 to 99) using exp2 as a building block. Use the divided clock as the clock frequency and display on the seven-segment display Design Specification

3-3 (14/15)
Design Specification(2/2)
block diagram of the design or Logic Diagram (5/5)圖太小
I/O pin assignment (3/3)
Discussion +Conclusion(4/5)

output: [14:0] display; // SSD display output

output: [3:0] ftsd_ctl;

input: clk; // global clock input

input: rst_n; // active low reset

wire: clk_d;

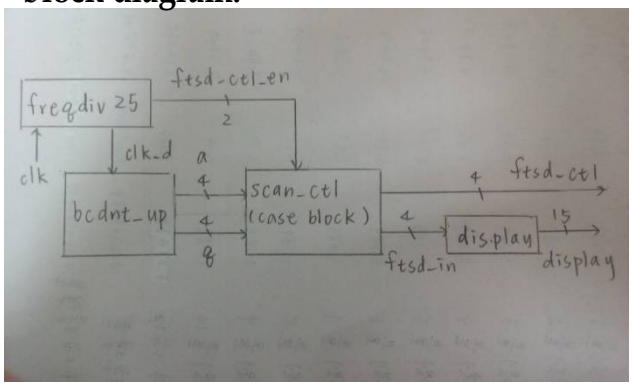
wire: [3:0]ftsd_in;

wire: [1:0]ftsd_ctl_en;

wire: [3:0]a;

wire: [3:0]q;

block diagram:



Design Implementation

logic function:

bcdcnt_up:

當個位數為 9 的時候 q_tmp=4'd0; a_tmp=a+4'd1;

當個位數與十位數皆為 9 的時候 q_tmp=4'd0; a_tmp=4'd0;

其他狀況 q_tmp=q+4'd1; a_tmp=a+4'd0;

Display:

當 bcd=4'd0: display = 15'b0000_0011_1111_111; //0

當 bcd=4'd1: display = 15'b1111_1111_1011_011; //1

當 bcd=4'd2: display = 15'b0010_0100_1111_111; //2

當 bcd=4'd3: display = 15'b0000_1100_1111_111; //3

當 bcd=4'd4: display = 15'b1001_1000_1111_111; //4

當 bcd=4'd5: display = 15'b0100_1000_1111_111; //5

當 bcd=4'd6: display = 15'b0100_0000_1111_111; //6

當 bcd=4'd7: display = 15'b0001_1111_1111_111; //7

當 bcd=4'd8: display = 15'b0000_0000_1111_111; //8

當 bcd=4'd9: display = 15'b0000_1000_1111_111; //9

default: display = 15'b1111_1111_1111_111; //DEF

freqdiv25:

cnt_tmp = {clk_out,cnt_h,clk_ctl,cnt_l} + 1'b1;

scan_ctl:

當 ftsd_ctl_en =2'b00: ftsd_ctl=4'b0111; ftsd_in=in0;
當 ftsd_ctl_en =2'b01: ftsd_ctl=4'b1011; ftsd_in=in1;
當 ftsd_ctl_en =2'b10: ftsd_ctl=4'b1101; ftsd_in=in2;
當 ftsd_ctl_en =2'b11: ftsd_ctl=4'b1110; ftsd_in=in3;
default: ftsd_ctl=4'b0000; ftsd_in=in0;

I/O pin assignment:

```
NET "Display[14]" LOC = P6;
NET "Display[13]" LOC = N4;
NET "Display[12]" LOC = V5;
NET "Display[11]" LOC = T5;
NET "Display[10]" LOC = U7;
NET "Display[9]" LOC = R3;
NET "Display[8]" LOC = N5;
NET "Display[7]" LOC = R5;
NET "Display[6]" LOC = T3;
NET "Display[5]" LOC = T4;
NET "Display[4]" LOC = V4;
NET "Display[3]" LOC = V7;
NET "Display[2]" LOC = R7;
NET "Display[1]" LOC = T7;
NET "Display[0]" LOC = U5;
NET "ftsd_ctl[3]" LOC= T6;
NET "ftsd_ctl[2]" LOC= V6;
NET "ftsd_ctl[1]" LOC= U8;
NET "ftsd_ctl[0]" LOC= V8;
NET "rst_n" LOC = T1;
NET "clk" LOC = R10;
```

Discussion:

1. 一開始燒錄時 14 段顯示器只能顯示 1 和 0，結果是 top module 中的 wire[3:0]a 和 wire[3:0]q 忘了打[3:0]，也就是忘了設成 4bits !
2. 3.3 的實驗是我第一次嘗試在老師講解前先自己思考如何設計，果然和老師教的方法差很多而且也遇到很多困難，花了很多時間 debug 但最後仍然順利完成了，此實驗中我的想法是同時考慮個位數和十位數，但缺點是只適用在兩位數，如果分別使用 bcd counter，則可在之後需要更多位數時直接引用，較

為方便，但第一次不是懵懵懂懂按部就班造著講義進行覺得相當有成就感!!

Conclusion:

這次的 Lab3 讓我比較了解了如何把多個不同功能的模組，藉由一個 top module 結合在一起，並且學會如何宣告和呼叫，及了解計數器和除頻器的邏輯，基本上目前的部分都還不會太複雜，只要把邏輯寫好，並注意細節，應該可以順利完成。