**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** |

**Design Specification**

**output:** clk\_out;

**output:** [1:0]clk\_ctl;

**input:** clk;

**input:** rst\_n;

**reg:** clk\_out;

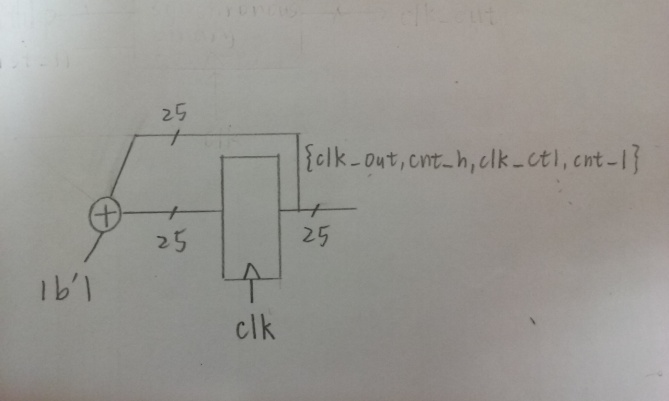
**reg:** [1:0]clk\_ctl;

**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**

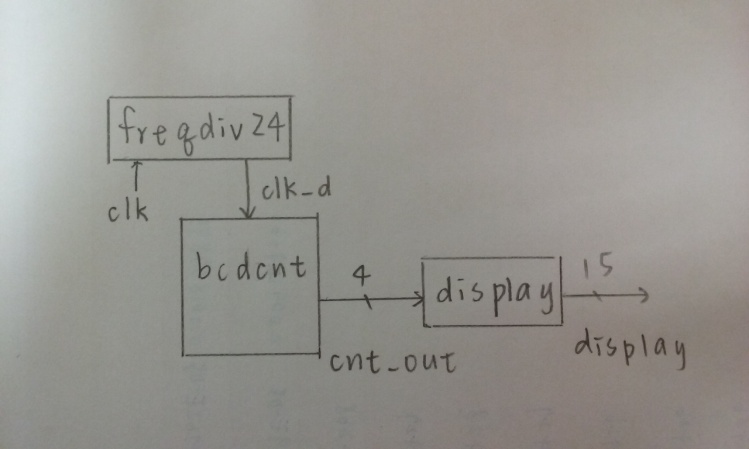
**output:** [14:0] display; // SSD display output

**input:** clk; // global clock input

**input:** rst\_n; // active low reset

**wire:** clk\_d;

**wire:** [3:0]cnt\_out;

**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**

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

**output:** [3:0] ftsd\_ctl;

**input:** clk; // global clock input

**input:** rst\_n; // active low reset

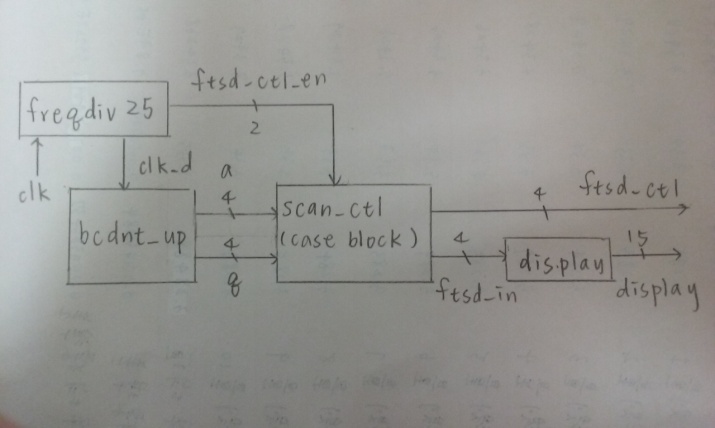
**wire**: clk\_d;

**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。