

(1) Frequency Divider: Construct a 27-bit synchronous binary counter. Use the MSB of the counter, we can get a frequency divider which provides a $1/2^{27}$ frequency output (f_{out}) of the original clock ($f_{crystal}$, 100MHz). Construct a frequency divider of this kind.

1.1 Write the specification of the frequency divider.

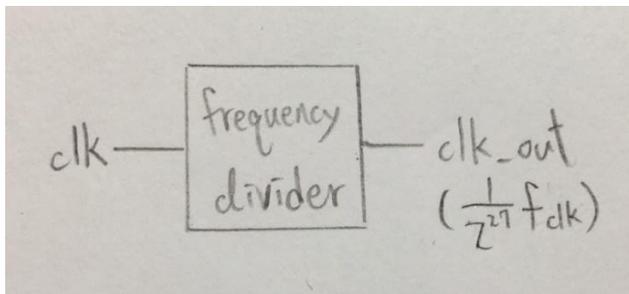
1.2 Draw the block diagram of the frequency divider.

IO:

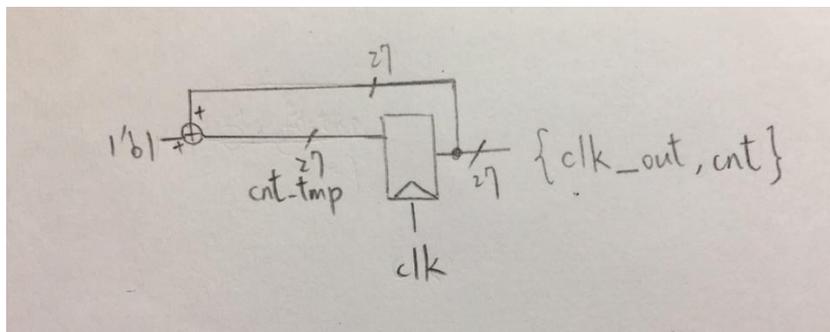
Input: clk, rst_n

Output: clk_out

Block diagram:



Logic diagram:



I/O pin assignment:

clk→W5

clk_out→U16

rst_n→R2

Discussion:

本實驗設計的是一個將頻率除以 2^{27} 的除頻器，由於原本在 FPGA 板上的頻率為 100MHz，因此透過一個 27bit 的 binary up counter，並將此 27bit 的最高位數(clk_out)當作輸出，便可將頻率除到接近 1Hz ($100\text{MHz} / 2^{27} \approx 0.745\text{Hz}$)。

(2) Frequency Divider: Use a count-for-50M counter and some glue logics to construct a 1 Hz clock frequency. Construct a frequency divider of this kind.

2.1 Write the specification of the frequency divider.

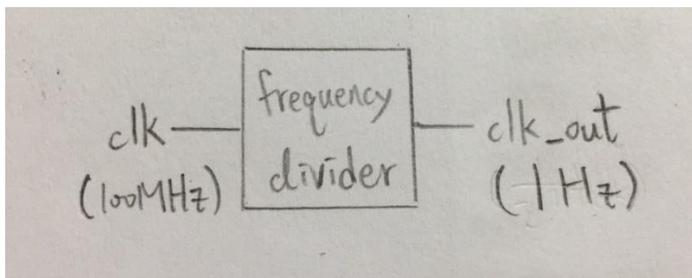
2.2 Draw the block diagram of the frequency divider.

IO:

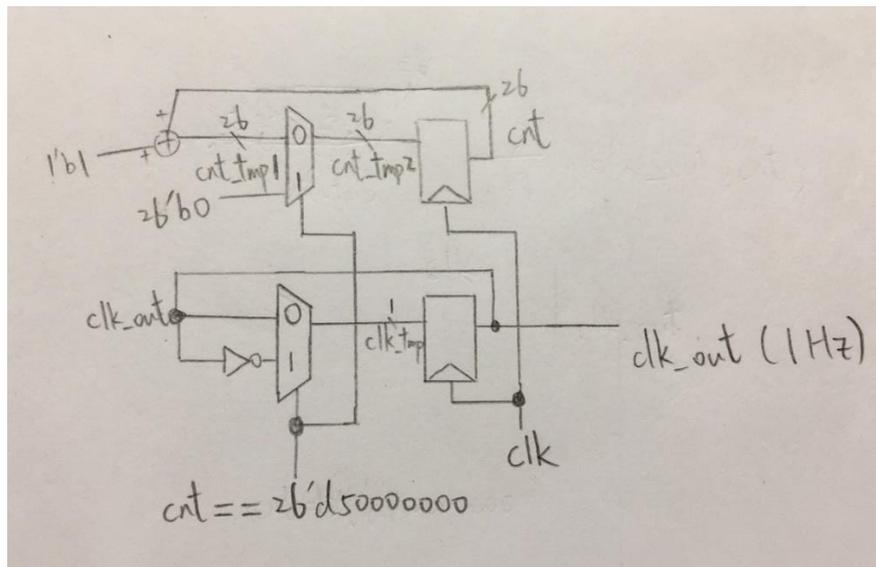
Input: clk, rst_n

Output: clk_out

Block diagram:



Logic diagram:



I/O pin assignment:

clk → W5

clk_out → U16

rst_n → R2

Discussion:

本實驗建造的是一個將 FPGA 板上 100MHz 的頻率轉換成 1Hz 頻率輸出的除

頻器，建構理念是設計一個上限為 50M 的 binary up counter，而每當 counter 數到 50M 時，便將輸出值(`clk_out`) toggle 一次，如此下來，輸出值(`clk_out`)在等於 0 和等於 1 的時間各自都是 50M 次的 `clk`，也就是說，`clk_out` 的值會在 0.5 時 toggle 一次，因此輸出值的頻率將會等於 $100\text{MHz}/(50\text{M} * 2) = 1\text{Hz}$ 。

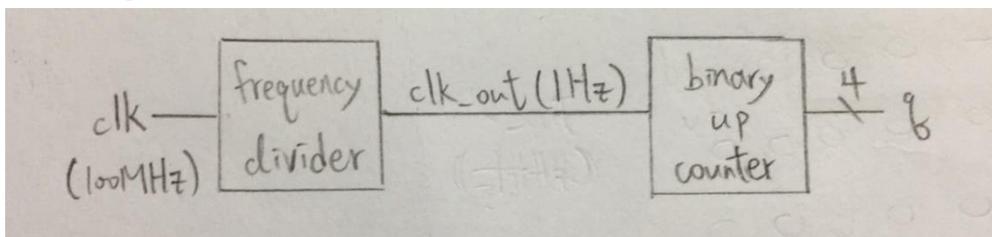
(3) Implement pre-lab1 with clock frequency of 1 Hz.

IO:

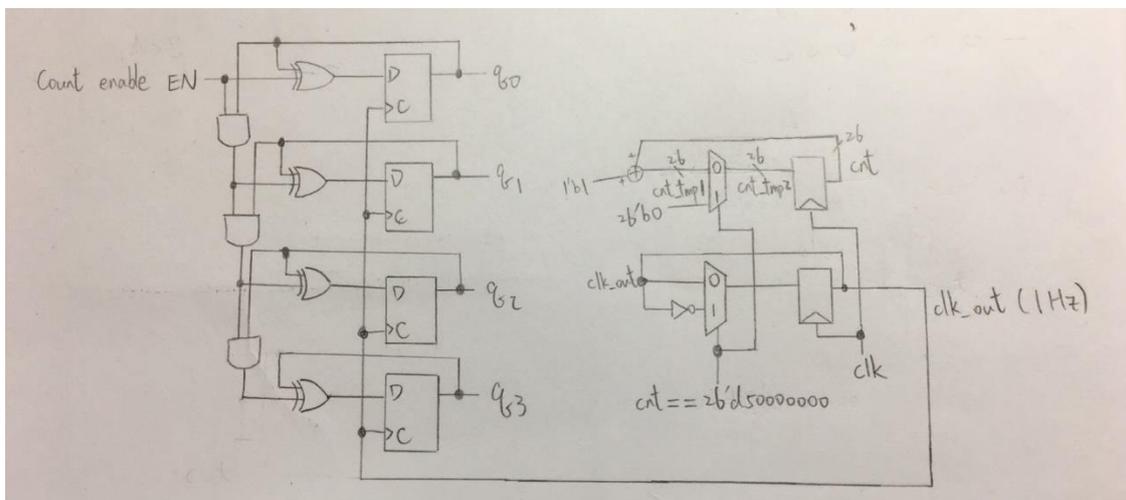
Input: `clk, rst_n`

Output: `[3:0]q`

Block diagram:



Logic diagram:



I/O pin assignment:

`q[3]` → V19

`q[2]` → U19

`q[1]` → E19

`q[0]` → U16

`clk` → W5

`rst_n` → R2

Discussion:

本實驗設計一個 clock 為 1Hz 的 binary up counter，做法為將上個實驗所做好的除頻器之輸出值(`clk_out`)接上 pre_lab 第一題的 counter 的輸入(`clk`)，便可達成本題的要求。

4-bit binary counter：透過四個 T-flip-flop，可以達到每當一個 clock 的 positive edge 來時就將輸出(`q`)的最小位數 toggle 一次的效果，當進位時，可以牽動更高位數也 toggle 一次，以此類推，進而完成一個 4bit 的 binary counter。

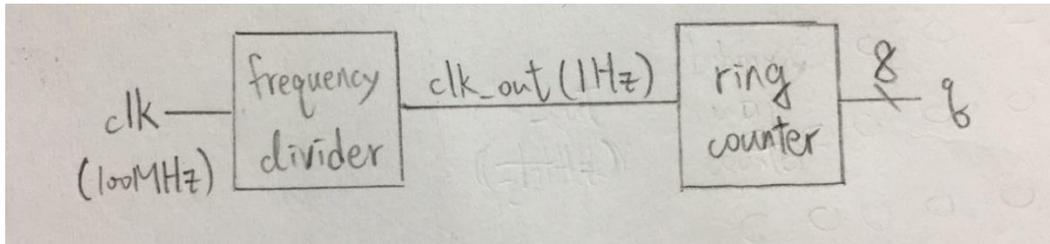
(4) Implement pre-lab2 with clock frequency of 1 Hz.

IO:

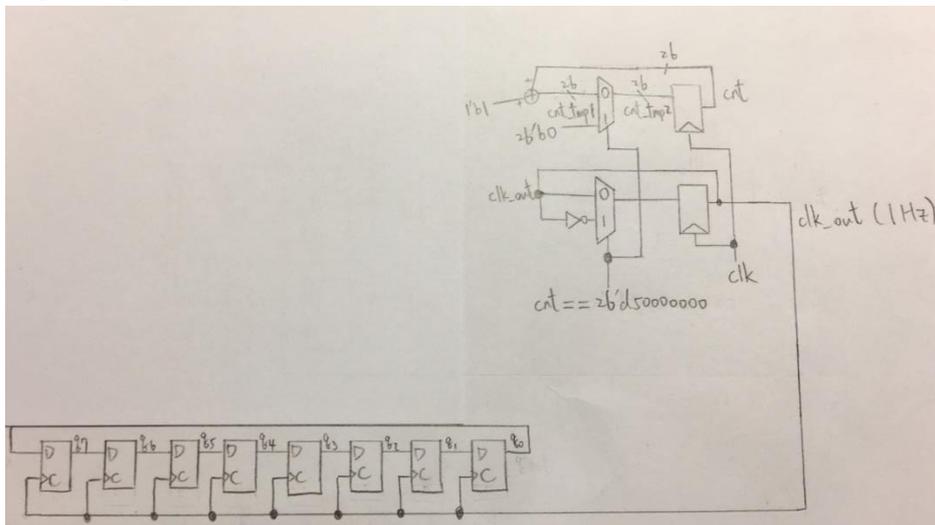
Input `clk`, `rst_n`

Output `[7:0]q`

Block diagram:



Logic diagram:



I/O pin assignment:

q[7]→V14

q[6]→U14

q[5]→U15

q[4]→W18

q[3]→V19

q[2]→U19

q[1]→E19

q[0]→U16

clk→W5

rst_n→R2

Discussion:

本實驗設計的是一個 clock 為 1Hz 的 ring counter，做法為將第二題除頻器 1Hz 的輸出(clk_out)接上 pre_lab 第二題的輸入(clk)之後，便可達成題目要求的結果。

8-bit ring counter：建構理念是透過同步的 clock 可以同時在 clock 的 positive edge 時將上一位的值傳給下一位，如此重複，而 reset 的時候輸出(q)為 01010101，如此即可完成一個 8bit 的 ring counter。

Conclusion:

這次實驗最新接觸到的是除頻器的設計，而第三、四小題就已經要接上 1Hz 的頻率，只要把做好的除頻器 output 接上其他功能 module 的 input 就可以了，也感受到除頻器好用的地方，而講義上也有介紹把除頻器 27bit 中的第 16、17bit 抓出來當作 scan control 的 input 的用法，感覺相當驚奇。經過這次實驗，我發現只要把一些作為基礎的 module(像是本次實驗的除頻器)寫好後，便可以使用在很多不同的實驗裡，這也是我覺得相當方便的一個地方。