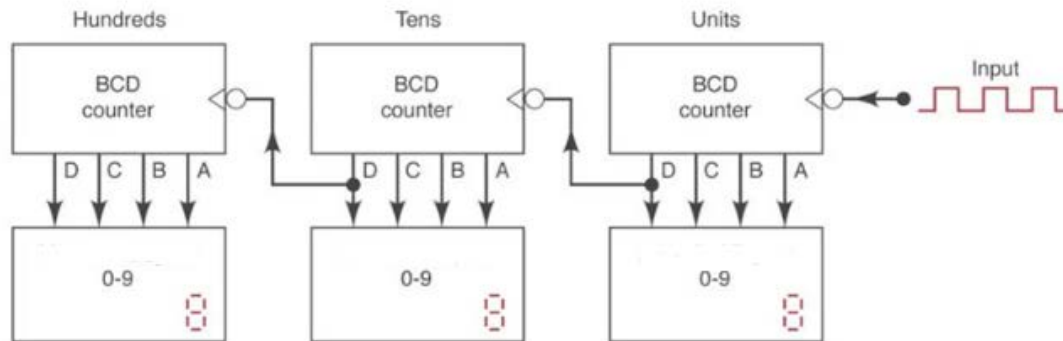


Practice Problem (1) (20 pts.)

BCD counters, also known as decade or MOD-10 counters, are often used whenever pulses are to be counted and the results displayed in decimal. A single BCD counter counts from 0 to 9 and then recycles to 0. To count a larger number than 9, we should cascade a multiple of BCD counters, one for each decade. For example, to construct a BCD counter operation that counts from 000 to 999 we need a three-BCD counter and should proceed with the following design:



1. Initially all counters are reset to 0.
2. Each input pulse advances the first counter once.
3. The 10th input pulse causes the first counter to recycle, which advances the second counter.
4. This continues until the second counter recycles, which advances the third counter.
5. The cycle repeats until 999 is reached and all three counters start again at zero.

In this problem you are required to design a **three_BCD counter** using as building block the BCD counter given below.

```
module BCD (Q, CLOCK, CLR);  
  
    output [3:0] Q; //BCD output  
    input CLOCK, CLR; //Clock and Reset  
  
    reg [3:0] Q;  
  
    always @(negedge CLOCK)  
        if (CLR | Q==4'd9)  
            Q = 4'd0;  
        else  
            Q = Q + 1;  
  
endmodule
```

Use the following module template for your **three_BCD**:

```
module three_BCD (BCDu, BCDt, BCDh, CLOCK, CLR);  
  
    output [3:0] BCDu, BCDt, BCDh; //u=1, t=10, h=100  
    input CLOCK, CLR;  
    //WRITE YOUR CODE HERE  
  
endmodule
```