```verilog
 1   `timescale 1ns / 1ps
 2   //******************************************************************//
 3   //  Class: CECS 361                                                //
 4   //  Project: Project1-Cecs361                                      //
 5   //                                                                 //
 6   //  File name: <ledClk.v>                                          //
 7   //  Abstract: Int count determines frequency of clkOut. So count   //
 8   //            =(Incoming Freq/(Outgoing Freq))/2. ClkIn (100Mhz)   //
 9   //            alternates 'count' in which clkOut will also         //
10   //            alternates. We want an output of 480hz frequency     //
11   //            to have a rate of 60Hz refresh. Determined by        //
12   //            Outgoing Freq = (Refresh Rate * Number of Pixels).   //
13   //  Created by      <Alina Suon> on <09-18-18>.                    //
14   //  Copyright © 2018 <Alina Suon>. All rights reserved.            //
15   //                                                                 //
     //                                                                     //
16   //  In submitting this file for class work at CSULB                //
17   //  I am confirming that this is my work and the work              //
18   //  of no one else. In submitting this code I acknowledge that     //
19   //  plagiarism in student project work is subject to dismissal.    //
20   //  from the class                                                 //
21   //******************************************************************//
22   module ledClk(clk, rst, ledClk);
23      input  clk, rst;
24      output ledClk;
25      reg    ledClk;
26      integer clkTick;
27
28      always @(posedge clk, posedge rst)
29         begin
30            if(rst == 1'b1)
31               begin
32                  clkTick = 0;
33                  ledClk  = 0;
34               end
35   //see clock, incr counter, check if half period past
36            else
37               begin
38                  clkTick = clkTick + 1;
39   //Produce 480Hz output clock by using 104166
40                  if (clkTick >= 104166)
41                     begin
42                        ledClk = ~ledClk;
43                        clkTick = 0;
44                     end
45               end
46         end
47   endmodule
48
```