基于FPGA的流水灯电路设计与仿真

一、项目任务和要求

1、每次仅有一个LED灯亮，每隔1 秒钟，亮点移动一次，不断循环。

2、实现自动可逆计数器，让亮点能够自动地双向移动。

3、设计分频器。对50MHz信号分频，得到1Hz脉冲，作为计数器时钟输入。

4、修改上述电路，达到每次有两个（或者3个等）LED灯亮，并自动地移动的效果。

5、试将移位寄存器和计数器实现的自动双向流水灯电路集成在一起，用一个开关选择 其中的一个电路，实现 8 个LED的流水灯。 

6、设计与实现16个LED的流水灯。要求：

 （1）设计一个带参数的通用移位寄存器模块，在顶层实例引用该模块时，将参 数值（16）传到子模块中。 

（2）或者，实例引用两次8位带预置数据端的双向移位寄存器；

二、电路的设计过程

1、组成框图

模块之间的调用关系：



2、工作原理

组合流水灯模块：完成在计数器和移位寄存器模式之间的切换

输入：50MHz的时钟信号、复位信号、暂停信号、选择信号

输出：16线流水灯信号

计数器实现流水灯模块：控制方向信号，将输出连接到灯上

输入：50MHz时钟信号、复位信号、暂停信号

输出：16线流水灯信号

移位寄存器实现流水灯模块：给串行输入，控制方向信号，将输出连接到灯上

输入：50MHz时钟信号、复位信号、暂停信号

输出：16线流水灯信号

分频器模块：完成50MHz到1Hz时钟的分频

输入：50MHz时钟信号、复位信号

输出：1Hz时钟信号

4位双向计数器模块：实现0~15和15~0的双向计数

输入：1Hz时钟信号、复位信号、保持信号、计数方向信号

输出：4位计数器当前计数值

4-16译码器模块：实现从0~15到16根线网的译码输出

输入：4位地址信号输入、使能信号

输出：16线译码输出

16位双向移位寄存器模块：实现0~15个比特位之间的双向移位

输入：1Hz时钟信号、使能信号、复位信号、保持信号、移位方向信号、1位串行 输入信号

输出：16位并行输出信号

3-8译码器模块：实现从0~7到8根线网的译码输出

输入：3位地址信号输入、使能信号

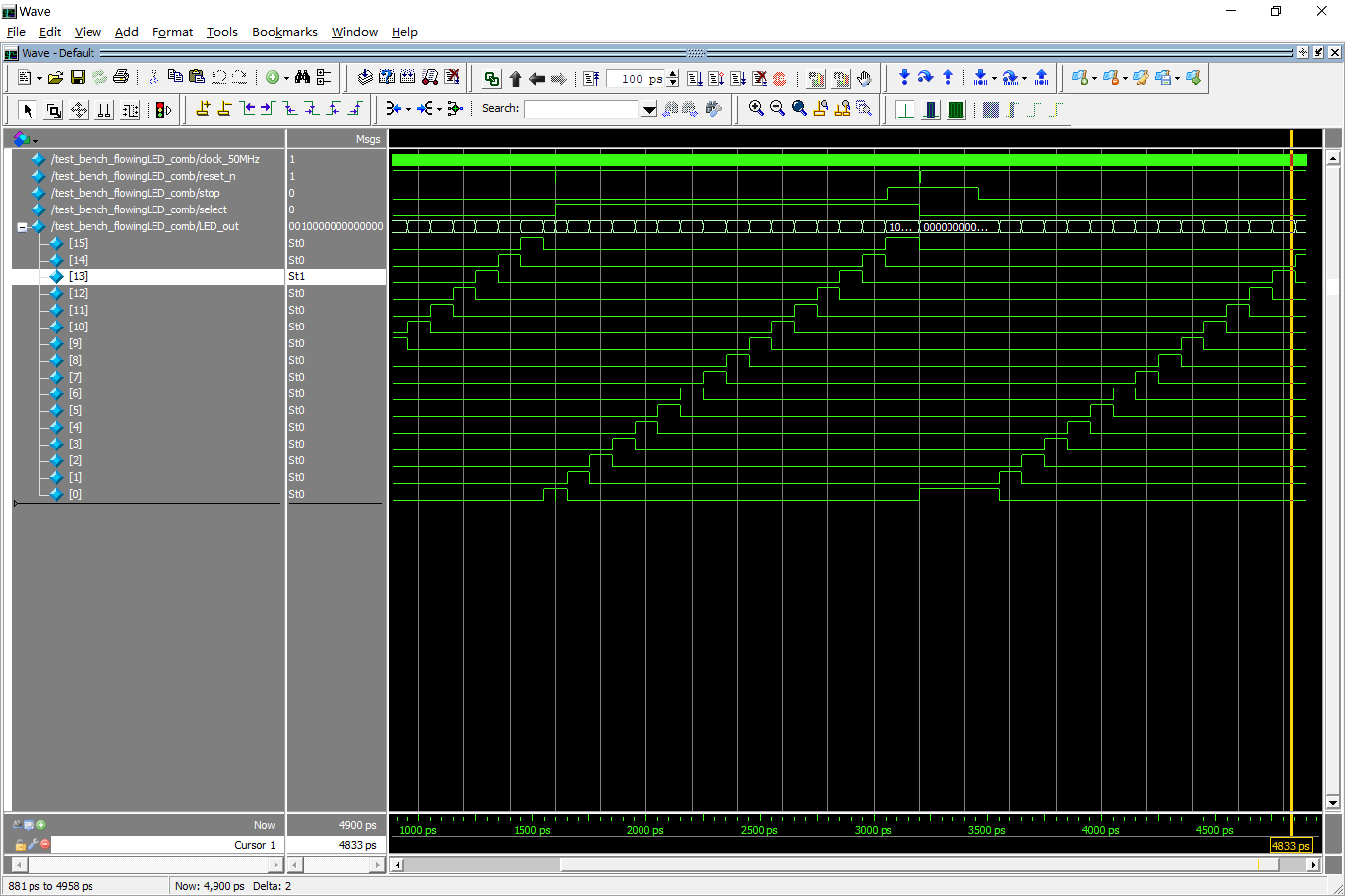
输出：8线译码输出

8位双向移位寄存器模块：实现0~7个比特位之间的双向移位

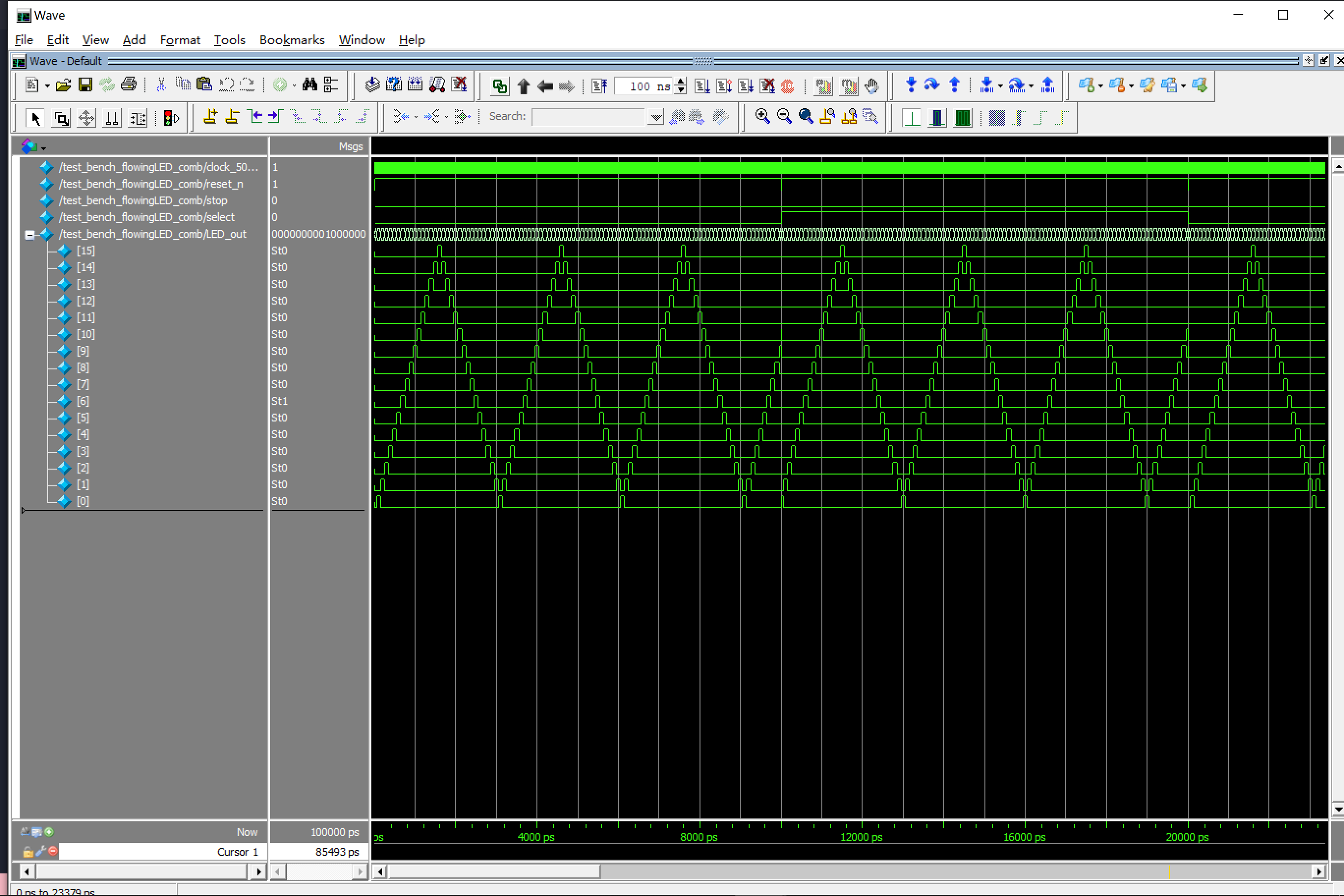
输入：1Hz时钟信号、使能信号、复位信号、保持信号、移位方向信号、1位串行 输入信号

输出：8位并行输出信号

3、仿真波形及解释：



这是单向流水灯与保持信号相结合的调试波形，在保持信号有效后，过一个时钟周期灯信号保持不变，直到保持信号解除的后一个时钟周期。在选择切换时，同时执行复位，会发生一次短暂的灯闪烁。



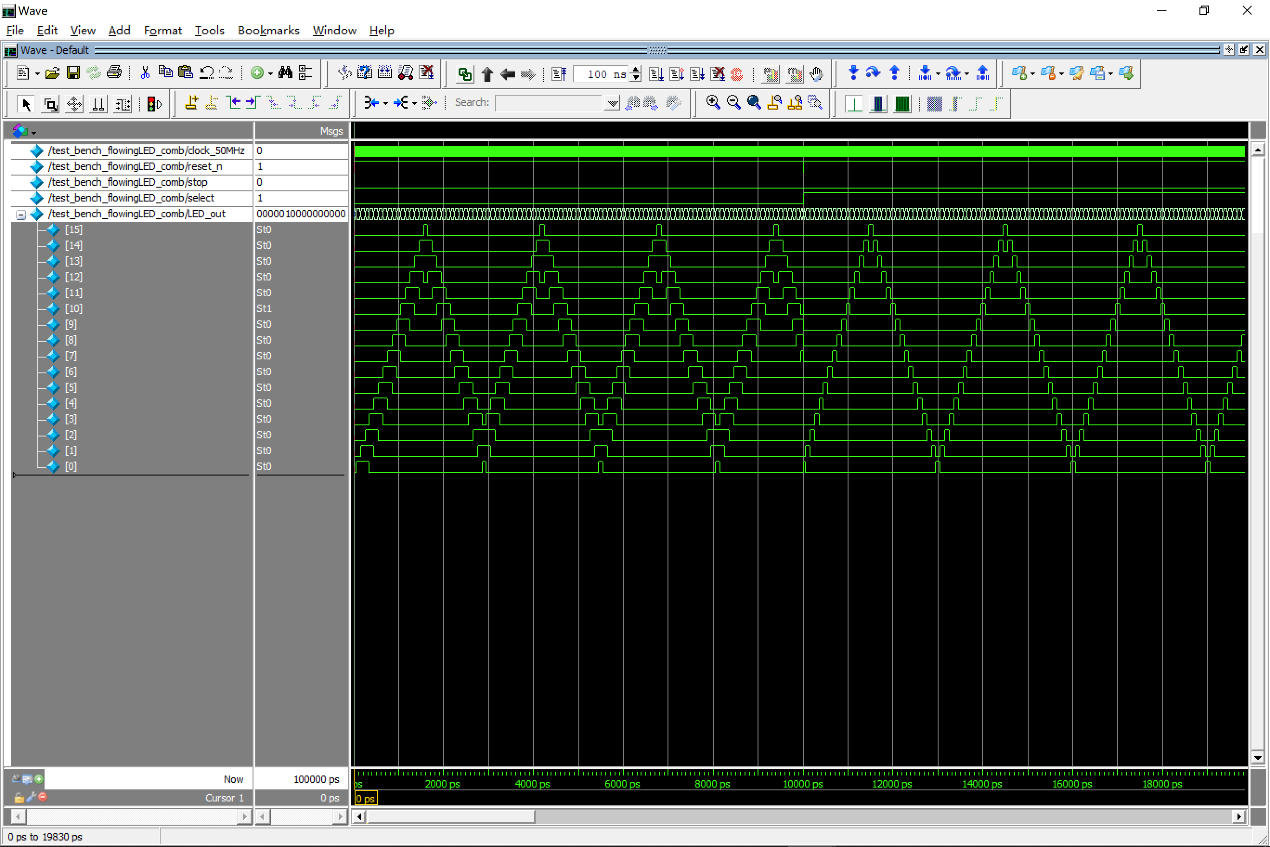
第一行最密的是原始的MHz时钟信号；

第二行是下降沿有效的同步复位信号，在刚上电时、切换工作模式时 会执行复位；

第三行为高电平有效的同步保持信号；

第四行为工作模式的选择信号，为1时工作在计数器模式，为0时工作在移位寄存器模式；

第五行及以下均为16线输出的灯信号。

实现了两个模式的切换和双向16位的流水灯功能。切换模式会导致重新从第一盏灯开始亮。

这个波形实现了多盏灯的移位流水灯功能。select为0时为移位寄存器模式，为1时是计数器模式。

三、调试过程

1、调试步骤：

激励模块：

module test\_bench\_flowingLED\_comb;

    reg clock\_50MHz = 1'b0;  //时钟信号

    reg reset\_n = 1'b1;      //复位信号

    reg stop = 1'b0;         //停止信号

    reg select = 1'b0;       //选择信号

    wire [15:0] LED\_out;     //输出的LED通道

    always #1 clock\_50MHz = ~clock\_50MHz;  //时钟翻转

    initial begin//复位 上电初始化

        #1 reset\_n = ~reset\_n;

        #1 reset\_n = ~reset\_n;

    end

    always  begin//换通道（计数器/移位寄存器）

        #10000 select = ~select;

        #1  reset\_n = ~reset\_n;

        #1  reset\_n = ~reset\_n;

    end

    flowingLED\_comb flowingLED\_comb\_inst(clock\_50MHz,reset\_n,stop,select,LED\_out);

endmodule

调用激励模块test\_bench\_flowingLED\_comb来仿真，自动地生成原始时钟、调用组合流水灯模块并执行一次复位初始化，此后自动地在计数器和移位寄存器两个工作模式间切换，观察仿真波形。

2、遇到的问题及解决方法：

 （1）用8位双向移位寄存器实现16位的时候，出现了一些奇怪的逻辑错误，之后通过给8位移位寄存器单独写激励模块来调试才得以解决（下图是单独激励的模块代码）。

（2）仿真时，分频到1Hz的话，波形会非常的卡（电脑几度无响应），所以最终在单独实现的分频器模块里面，修改了计数值的量级，使仿真时的工作时钟在纳秒量级，如果要实际烧录到开发板上的话，要记得把计数值改回到秒量级。

