

实验三 时序电路设计实验报告

班级：07111904

姓名：周一鸣

学号：1120192000

1. 实验题目

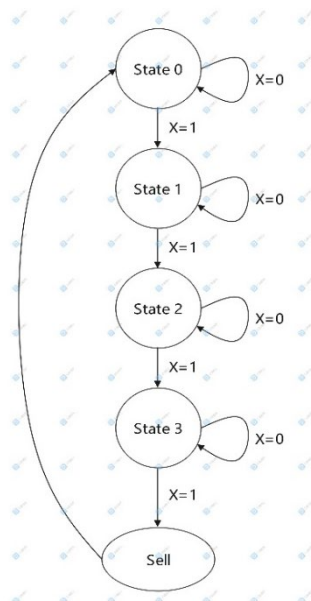
某自动售货机出售四元一瓶的饮料，但是每次只能投入一元硬币。当连续投入四个硬币后，售货机才会给出一瓶饮料。如果投入的硬币不足四元，售货机不会退回之前所投入的硬币。假设输入 $X=1$ 表示投入一元硬币， $X=0$ 表示未投入一元硬币；输出 $Z=1$ 表示售货机给出一瓶饮料， $Z=0$ 表示未给出饮料。

2. 电路设计

a) 规范化

电路设计为两输入一输出。其中，两个输入信号为时钟信号和投币信号，输出信号为表示出售与否的信号。电路的逻辑部分和时序部分均在时钟上升沿出发，通过当前状态和投币信号决定下一状态和输出信号。当电路当前状态为 S_0 且输入信号为 $X=1$ 时，输出信号 $Z=1$ ，即出售饮料。

b) 形式化



初始状态为 State 0，即未投币状态。每次投币后，状态转移至下一状态。状态

序号表示已投币数量。

c) 状态分配

State 0: S0 = 2' b00

State 1: S1 = 2' b01

State 2: S2 = 2' b10

State 3: S3 = 2' b11

3. 电路实现

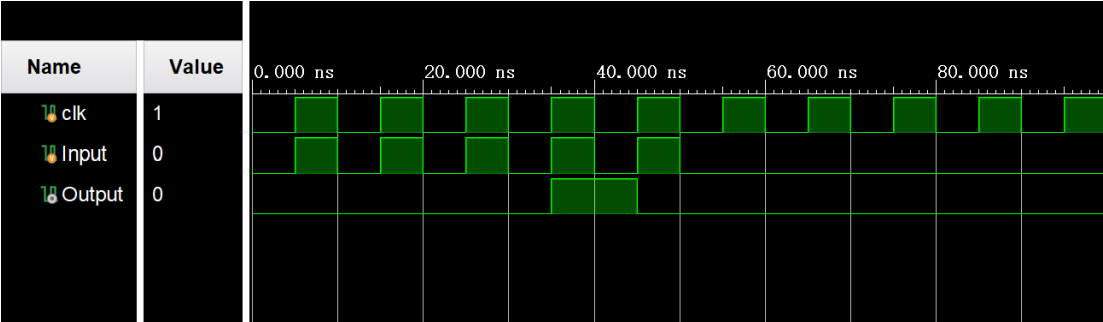
```
23 module SC(input clk, X, output reg Z);
24
25     reg [0:1] state;
26     parameter S0 = 2'b00, S1 = 2'b01, S2 = 2'b10, S3 = 2'b11;
27
28     initial
29     begin
30         state = 0;
31         Z = 0;
32     end
33
34     always @(posedge clk)
35     begin
36         if (X)
37             case (state)
38                 S0: state = S1;
39                 S1: state = S2;
40                 S2: state = S3;
41                 S3: state = S0;
42             endcase
43     end
44
45     always @(posedge clk)
46     begin
47         if (state == S0 && X == 1'b1)
48             Z <= 1'b1;
49         else
50             Z <= 1'b0;
51     end
52
53 endmodule
```

4. 电路验证

a) TestBench

```
23 module testbench();
24
25     reg clk, Input;
26     wire Output;
27
28     SC Test(.clk(clk), .X(Input), .Z(Output));
29
30     initial
31     begin
32         clk = 1'b0;
33         Input = 1'b0;
34         #5 Input = 1'b1;#5 Input = 1'b0; //每5ns投币信号一次，每次持续5ns
35         #5 Input = 1'b1;#5 Input = 1'b0;
36         #5 Input = 1'b1;#5 Input = 1'b0;
37         #5 Input = 1'b1;#5 Input = 1'b0;
38         #5 Input = 1'b1;#5 Input = 1'b0;
39     end
40
41     always #5 clk = ~clk; //时钟周期为10ns，每5ns跳变一次
42
43     always
44     begin
45         #100;
46         if ($time >= 100) $finish ; //限制模拟时间小于100ns
47     end
48
49 endmodule
```

b) 仿真结果



35ns 时，投币达到四枚，输出信号变为 1，持续一个时钟周期，在下一次时钟正边沿时回到 0。第五次投币后没有再投币，此时状态停留在 State1，知道模拟结束，输出信号一直为 0。

5. 实验心得

通过时序电路仿真，对时序电路的理解进一步加深了。

在代码实现的过程中，由于对 Verilog 语法和仿真模拟过程的了解不足，在代码实现时屡次出现无输出、输出高电平提前了一个时钟周期等问题。之后通过不断对代码进行调试和优化，通过以 initial 块代替一个 always 块并简化其他 always 块中的代码解决了问题。