**实验三 时序电路设计实验报告**

**姓名**：丘绎楦  **学号**：1820221050

**班级**：07812201 **手机**：15010901562

注：黑色字体内容不能改动，蓝色字体内容（为示例或说明）需删除和修改。

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。

1. **电路设计**
   1. **规范化**

**电路输入为信号X，信号CLK，信号RESET，输出为信号Z：**

输入信号X表示是否投入一元硬币，CLK表示时钟信号，RESET为复位信号，输出信号Z=1表示给出一瓶饮料，Z=0表示未给出饮料。

**电路的时序行为如下：**

电路的状态在每次时钟信号的上升沿和RESET信号的上升沿会发生改变，其中如果RESET信号为1，电路则直接恢复为初始状态A；如果RESET信号不为1，则电路的当前状态直接向下一状态转变。

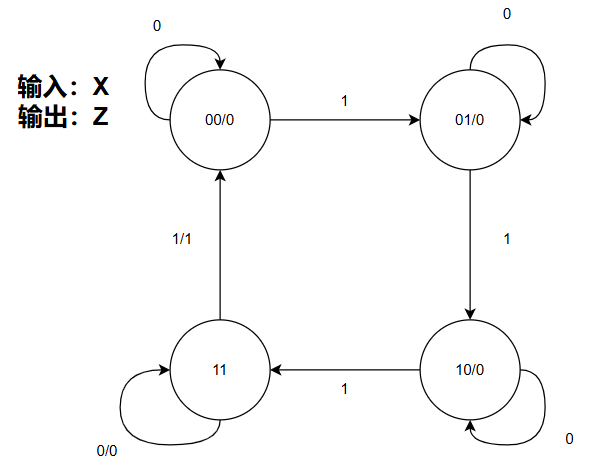
当电路为状态A，输入X=1，电路下一状态转变成B；电路为状态B，输入X=1 , 电路下一状态转变成C；电路为状态C，输入X=1 , 电路下一状态转变成D；电路为状态D，输入X=1 , 电路下一状态转变成A , 输出Z=1（其余条件下都输出Z=0）。电路处于ABCD中某一个状态时，输入X=0，电路仍然保持为当前状态

* 1. **形式化**

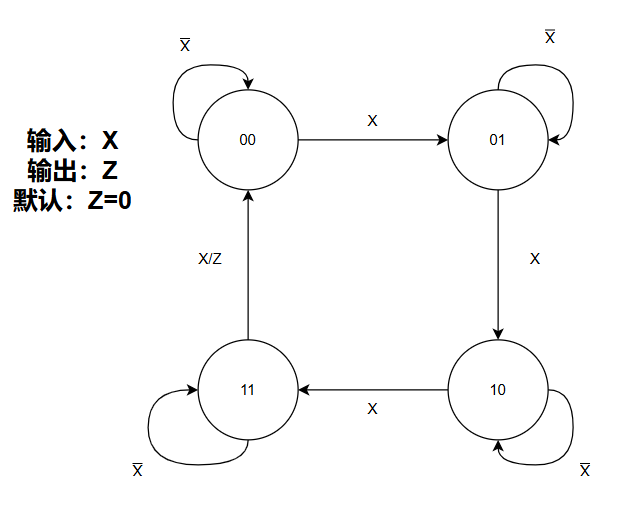
电路的状态表为：

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 当前状态 | | 下一状态 | | | | 输出 | |
| X=0 | | X=1 | | X=0 | X=1 |
| A | B | A | B | A | B | Z | Z |
| 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| 0 | 1 | 0 | 1 | 1 | 0 | 0 | 0 |
| 1 | 0 | 1 | 0 | 1 | 1 | 0 | 0 |
| 1 | 1 | 1 | 1 | 0 | 0 | 0 | 1 |

电路状态图：



电路状态机图：



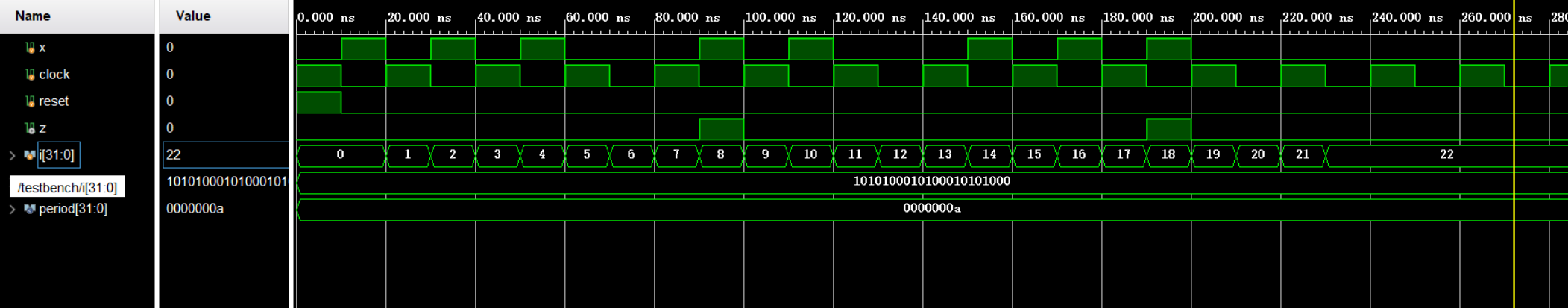
* 1. **状态分配**

对状态分配二进制码，对每个状态分配二位二进制码：

A=2'b00 , B=2'b01 , C=2'b10 , D=2'b11

1. **电路实现**
2. `timescale 1ns/1ps
4. module ex3(X,Z,CLK,RESET);
5. input X;
6. input CLK;
7. input RESET;
8. output Z;
9. reg[1:0] state, next\_state;
10. parameter A=2'b00, B=2'b01, C=2'b10, D=2'b11;  // 每一个状态对应的二进制码
11. reg Z;
13. // 只有时钟输入或复位输入才触发
14. always @(posedge CLK or posedge RESET)
15. begin
16. **if**(RESET)
17. state <= A;
18. **else**
19. state <= next\_state;
20. end
22. always @(X or state)
23. begin
24. **case**(state)
25. A: Z = 1'b0;
26. B: Z = 1'b0;
27. C: Z = 1'b0;
28. D: Z = X ? 1'b1 : 1'b0;  // 只有当第四枚硬币放入时才有饮料
29. endcase
30. end
32. // 调整下一个状态
33. always @(X or state)
34. begin
35. **case**(state)
36. A: next\_state <= X ? B : A;
37. B: next\_state <= X ? C : B;
38. C: next\_state <= X ? D : C;
39. D: next\_state <= X ? A : D;
40. endcase
41. end
42. endmodule
43. **电路验证**
    1. **TestBench**
44. `timescale 1ns / 1ps
45. //////////////////////////////////////////////////////////////////////////////////
46. // Company:
47. // Engineer:
48. //
49. // Create Date: 2024/06/10 20:18:14
50. // Design Name: 丘绎楦
51. // Module Name: testbench
52. // Project Name:
53. // Target Devices:
54. // Tool Versions:
55. // Description:
56. //
57. // Dependencies:
58. //
59. // Revision:
60. // Revision 0.01 - File Created
61. // Additional Comments:
62. //
63. //////////////////////////////////////////////////////////////////////////////////

66. module testbench(
67. );
68. reg x,clock,reset;
69. wire z;
71. integer i=0;
72. reg[0:21] test\_coin=22'b1010100010100010101000;
73. //这是输入的硬币序列中 1 表示投入一枚硬币 0表示没有投入硬币
74. //要求的输出是投入四枚硬币后Z=1
75. //输入序列1后面必定跟上一个0，因为1硬币投入的时间不可能是连续不断的投入
76. parameter period = 10;
78. initial begin
79. reset=1'b1;
80. x =1'b0;
81. //保证初始状态
82. #(period);
83. reset=1'b0;
84. **for**(i=0;i<22;i=i+1)
85. begin
86. x = test\_coin[i];
87. //输入x从当前的硬币序列号转换成下一硬币序列号
88. #period;
89. end
90. end
91. //clock表示testbench中的时钟信号
92. always
93. begin
94. clock=1'b1;
95. #(period);
96. clock=1'b0;
97. #(period);
98. end
100. ex3 u\_ex3(
101. .X(x),
102. .CLK(clock),
103. .RESET(reset),
104. .Z(z)
105. );
106. endmodule
     1. **仿真结果**



1. **实验心得**

这已经是我第三次参与数字逻辑实验，每一次都让我获得了宝贵的经验和深刻的认识。在实验过程中，我通过自己的思考和实践，不仅巩固了理论知识，还形成了自己对问题的独特见解。实验中，我学会了带着问题去探索，这不仅锻炼了我的动手能力，也在不知不觉中提升了我的解决问题的能力。

通过这一系列的实验，我更加明白了在数字逻辑设计中，每一个参数的设置都需要精确无误。同时，我也认识到了在面对问题时，耐心分析和细致调整的重要性。这些经历无疑将对我的未来学习和工作产生积极的影响。