

实验四 自动贩售机的设计与实现

一． 实验目的

1. 掌握有限状态机的设计方法。;
2. 能够使用 SystemVerilog 进行三段式状态机的建模。

二． 实验环境

1. 操作系统：Windows 10 或 Ubuntu 16.04
2. 开发环境：Xilinx Vivado 2018.2
3. 硬件平台：远程 FPGA 云平台

三． 实验原理

有限状态机分为 Moore 型状态机和 Mealy 型状态机。前者，状态机的输出仅由当前状态决定，如图 4-1 所示，在状态转换图的绘制中，输出信息标在状态（圆圈）中。后者，状态机的输出由当前时刻状态和输入共同决定，如图 4-2 所示，在状态转换图的绘制中，输出信息标在状态转换箭头之上。

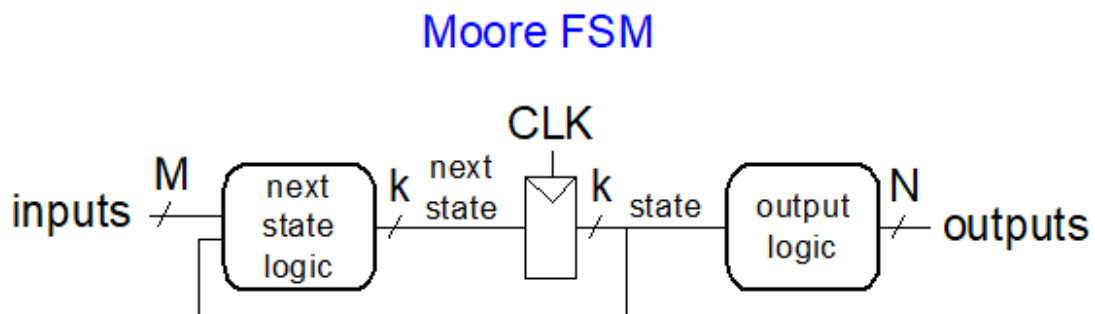


图 4-1 Moore 型状态机

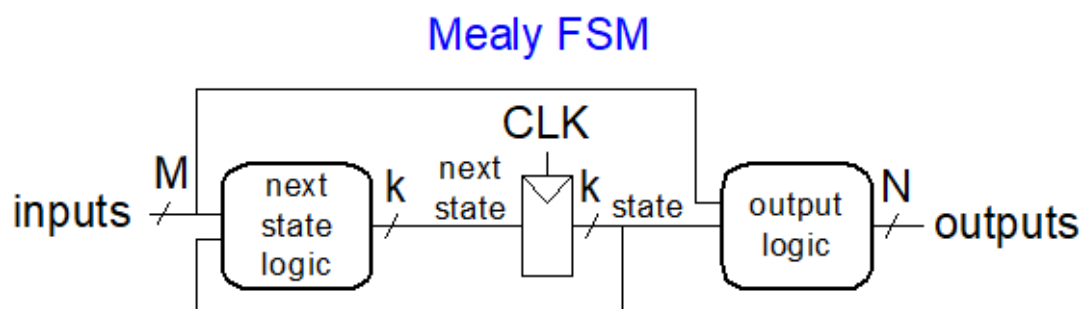


图 4-2 Mealy 型状态机

采用硬件描述语言进行状态机建模时，建议使用 3 段式。第一段描述状态的转换（即对状态机中的寄存器进行建模），采用时序逻辑实现；第二段描述状态转换条件和规律（即对状态机中的次态逻辑进行建模），采用组合逻辑实现；第三段描述输出逻辑，根据实际设计需要可采用组合逻辑或时序逻辑实现。三段式状态机建模的模板如下所示。

//第一段，同步时序 always 模块，描述状态的转换

```
always_ff (posedge clk) begin //同步复位
```

```
    if(!rst_n)
```

```
        current_state <= S0;
```

```
    else
```

```
        current_state <= next_state; //注意，使用的是非阻塞赋值
```

```
end
```

//第二段，组合逻辑 always 模块，描述状态转移条件判断

```
always_comb begin
```

```
    case(current_state)
```

```
        S1: if(...)
```

```
            next_state = S2; //阻塞赋值
```

```

...
endcase

end

//第三段，同步时序 always 模块（组合逻辑也可以），描述状态机的输出

always @ (posedge clk) begin

    ...//初始化

    case(next_state)

        S1: out1 <= 1'b1; //注意是非阻塞逻辑

        S2: out2 <= 1'b1;

        default:... //default 的作用是免除综合工具综合出锁存器

    endcase

end

```

四 . 实验内容

采用有限状态机, 基于 SystemVerilog HDL 设计并实现一个报纸自动贩售机。

整个工程的顶层模块如图 4-3 所示，输入/输出端口如表 4-1 所示。使用 **4 个七段数码管** 实时显示已付款和找零情况。其中，两个数码管对应“已付款”，另两个数码管对应“找零”，单位为分。通过 **1 个拨动开关** 对数字钟进行复位控制。使用 **两个按键** 模拟投币，其中一个按键对应 5 分，另一个按键对应 1 角。使用 **1 个 LED 灯** 标识出售是否成功, 灯亮表示出售成功, 否则表示已付款不够, 出售失败。

假设报纸价格为 15 分，合法的投币组合包括：

- 1 个 5 分的硬币和一个 1 角的硬币，不找零
- 3 个五分的硬币，不找零

- 1 个 1 角的硬币和一个 5 分的硬币，不找零
- 两个 1 角的硬币是合法的，找零 5 分。

当投入硬币的组合为上面 4 种之一时，则购买成功，LED 灯亮。**购买成功后，LED 灯持续亮 10 秒，然后自动熄灭，同时 4 个数码管也恢复为 0。**

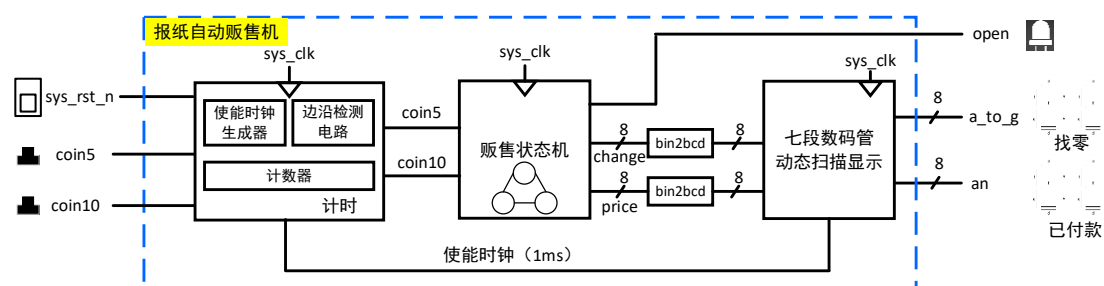


图 4-3 报纸自动贩售机顶层模块

报纸自动贩售机由 4 部分构成。

- 第一部分是**计时器模块**，该模块又由 3 个子模块构成，分别是计数器电路、使能时钟生成电路和边沿检测电路。
- 第二部分是整个自动贩售机电路的核心——**贩售机状态机**。状态机根据投币情况产生“已付款”和“找零”输出。此外，如果已付款超过 15 分，则将 LED 灯点亮，表示出售成功。
- 第三部分是两个 **8 位二进制转 BCD 模块**，分别将二进制的“已付款”和“找零”值转化为 BCD 编码，即 10 进制数。**本实验中，该模块不需要实现，由教师直接提供 IP 使用。**
- 第四部分是 **7 段数码管动态扫描显示模块**，它实现“已付款”和“找零”值的最终显示。

表 4-1 输入/输出端口

端口名	方向	宽度 (位)	作用
sys_clk	输入	1	系统基准时钟，主频 25MHz。
sys_rst_n	输入	1	连接拨动开关，对数字中进行复位。低电平有效。
coin5	输入	1	连接按键，每按下一次，表示投入 1 个 5 分硬币，按键按下为高电平。
coin10	输入	1	连接按键，每按下一次，表示投入 1 个 1 角硬币，按键按下为高电平。
open	输出	1	连接 LED 灯，灯亮表示购买成功；灯灭表示支付费用不够，购买失败。购买成功后，灯持续亮 10s，然后熄灭。
a_to_g	输出	8	连接七段数码管的数据输入端 CA~CG 和 DP，用于实时显示“已支付费用”和“找零”。采用共阳极控制，数码管低电平点亮。DP 段永远不点亮。
an	输出	4	连接 4 个七段数码管的使能端 AN0~AN3，其中 AN0 和 AN1 显示“已支付费用”，AN3 和 AN4 显示“找零”。使能信号高电平有效。

完成上述分秒数字钟的设计，需要有以下几点需要注意：

1. 7 段数码管动态扫描必须采用使能时钟实现，扫描频率为 1KHz (1ms)。
2. 必须通过边沿检测电路识别“5 分”和“1 角”按键按下产生的上升沿，以用于后续处理。
3. 用于计时的时钟频率为 25MHz (40ns)。
4. 由于 7 段数码管扫描周期是 1ms，购买成功后需要等待 10s，从而造成仿真时间过长。为了加快仿真速度，可以在仿真的时候使用较大的计时单位和扫描速度。

五．实验步骤（建议）

1. 解压缩 vend_machine_stu.rar，打开教师提供的工程文件 vend_machine.xpr，目前工程中已经提供了顶层文件 vend.sv 和约束文件 pin.xdc，并根据表 4-1 定义了输入/输出接口。同学对顶层模块的端口定义和约束文件不要进行任何

修改，只需要添加必要模块，并在顶层文件中通过结构化建模完成自动贩售机的最终设计。

2. 添加二进制转 BCD 码的 IP——bin2bcd。参考实验三完成 bin2bcd IP 核的添加，bin2bcd IP 核位于“**工程目录/bin2bcd**”目录下。
3. 同学们接着根据图 4-3 及设计要求，完成其它子模块的设计，并在顶层文件 vend.sv 中例化这些子模块，进行连接，完成报纸自动贩售机的最终设计。
4. 添加测试文件 **vend_tb.sv**，对所实现的自动贩售机进行行为仿真。注意，为了提高仿真速度，应对使能时钟频率和计时频率进行适当放大。
5. 如果行为仿真通过，则对工程进行综合、实现、生成 bin 文件。**注意，约束文件已在工程中提供，不需要再添加。**
6. 登录远程 FPGA 硬件云平台，直接导入教师提供的验证平台文件（**自动贩售机的设计与实现.epl**），无需进行绘制，验证平台如图 4-4 所示。烧写生成的 bin 文件，然后运行实验，通过按键和拨动开关验证所设计贩售机的正确性。

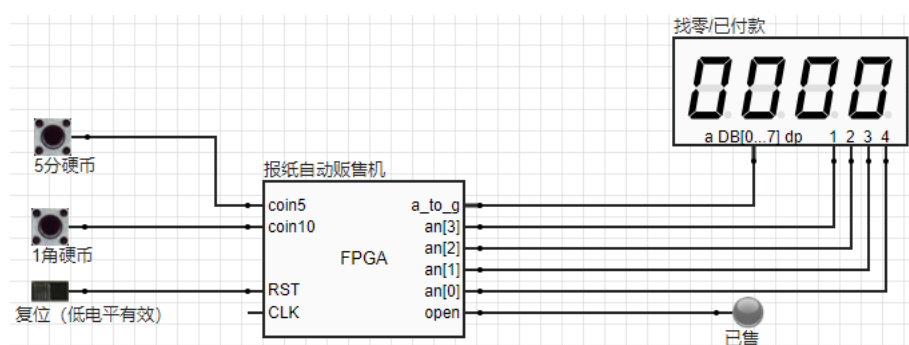


图 4-4 验证平台

7. 贩售机的操作流程如下图所示（以投入两个 1 角为例）。

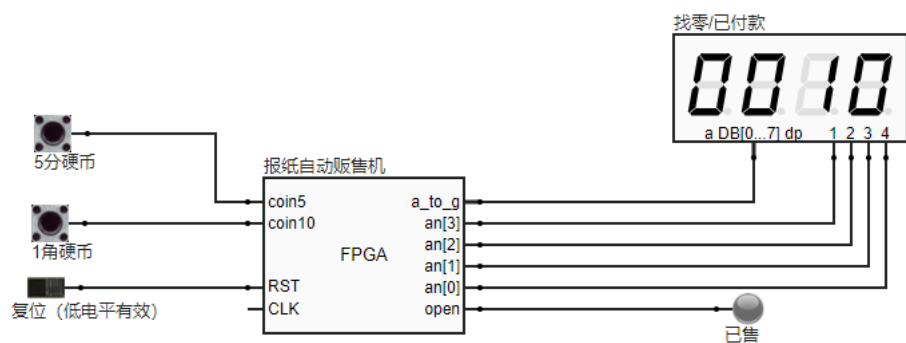


图 4-5 投入 1 角硬币

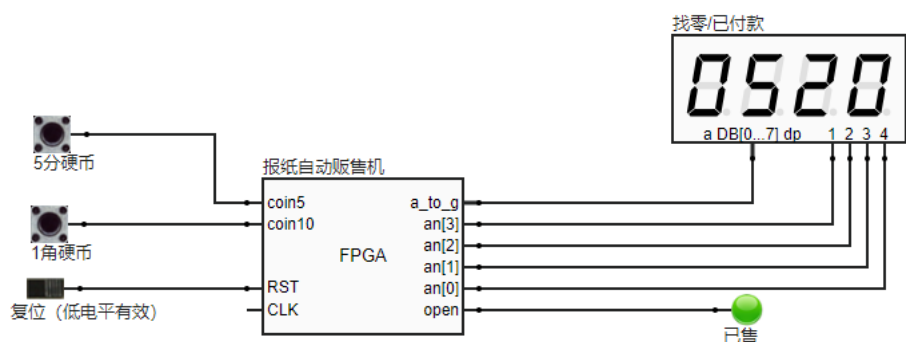


图 4-6 再投入 1 角硬币（已付款 20 分，找零 5 分，LED 灯亮表示已售）

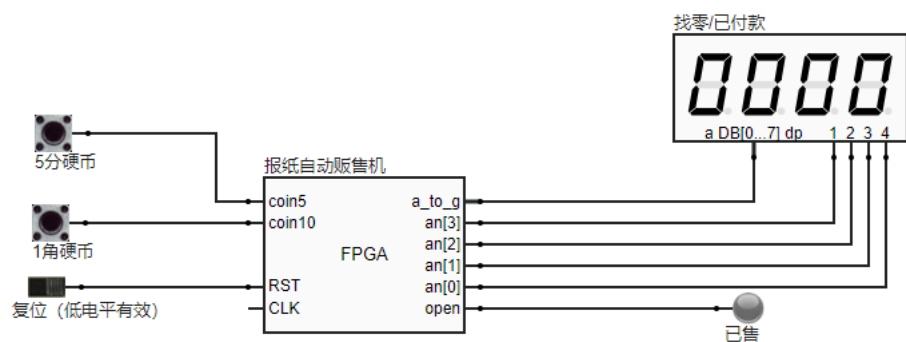


图 4-7 售出成功 10 秒后（LED 灯灭，数码管均恢复为 0）

六． 实验方式

每位同学独立上机编程实验，实验指导教师现场指导。

七． 参考内容

1. 教材内容和课件

八． 实验报告

1. 画出自动贩售机的状态转换图。

2. 画出自动贩售机电路的原理图（模块级别即可，如使能时钟模块、边沿检测模块等）。
3. 报纸自动贩售机的 SystemVerilog 代码。
4. 给出仿真的波形图和远程 FPGA 平台验证的截图。

九. 附加题

无