

## 2014 级数字逻辑课程设计题目

每组要求完成两个课程设计题目，即：第 1 个题目：题目 1、2 任选其中 1 个；第 2 个题目：题目 3、4、5、6 任选其中 1 个。

Nexys4 开发板提供了 16 个 Led 指示灯（可用于“输出信号”，例如：状态信号）、16 个拨动开关（可用于提供 0、1 输入信号）、5 个按键式开关（可用于单脉冲输入或者开关输入信号）、8 个 7 段数码显示管（可用于输出 0-9 或其它字符，例如：楼层、时间等）、还提供了一个 100Mhz 系统时钟信号（E3 引脚）。

设计者可以根据设计需要适当选择这些输入、输出信号，根据这些信号变化验证设计的正确性。

**注意 1：**如果开发板提供的输入输出资源不够实现某控制器的输入输出要求时，在不影响功能的情况下，自行取舍某些输入输出信号。

**注意 2：**如果设计者发现设计内容有缺陷或不妥之处，可与指导老师讨论，在指导老师认可的情况下，可以修改设计内容或者添加某些新功能；

**注意 3：**要考虑模块的复用。

### 1、题目一：电子钟设计（2 选 1）

#### 1.1、实验内容

- (1) 设计并利用 Nexys4 开发板来实现一个电子钟，它具有下述功能：
    - (a) 开发板上的 6 个数码显示管分别显示“时、分、秒”；
    - (b) 电子钟有复位（清零）功能；
    - (c) 电子钟有启动或者停止运行功能，在电子钟停止运行状态下，能够修改时、分、秒的值。
    - (d) 具有整点提示功能。
  - (2) 要求整个设计分为若干模块。**顶层模块用原理图设计（具体参见报告模版）**，底层模块用 Verilog HDL 语言设计。
  - (3) 在开发板上验证设计的正确性。
- 电子钟的系统框图，参见图 1-1 所示。

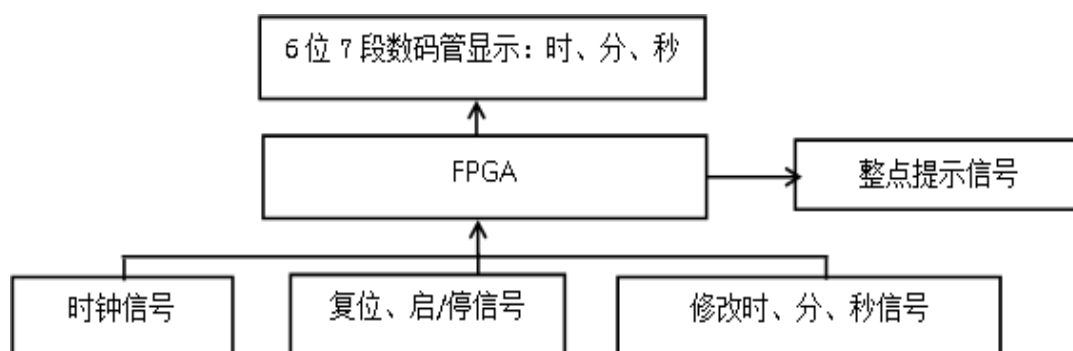


图 1-1 电子钟的系统框图

## 1.2、设计思路

本设计要求顶层采用原理图设计，各底层均采用 Verilog HDL 设计，参见图 2-2 所示。

- (1) 时钟用一个模 24 的计数器来实现；
- (2) 分钟和秒钟需要分别用一个模 60 的计数器来实现；
- (3) 要实现自动计时，则需要利用开发板提供的 100Mhz 提供脉冲，还需要一个分频的模块 dclk，使其吻合时钟速度；
- (4) 整点提示功能则用一个 ring 的模块实现，分钟进位时开始计数从而实现提示 5 秒。

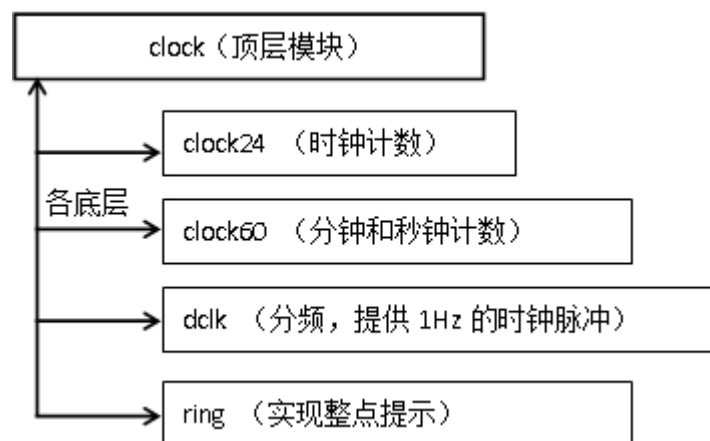


图 1-2 电子钟模块图

。。。。  
补充完善

## 2、题目二：自动销售机控制器设计（2 选 1）

采用 FSM 设计并利用 Nexys4 开发板来实现一个饮料自动投币售卖机的控制器。

### 2.1、设计内容

（1）电源开关同时作为**电路总清零信号（Reset）**，当 **Reset = Off** 不工作、为 **On** 时电路进入初始状态，且电源指示灯亮（又称售卖机运行指示灯），控制器初始状态为：“S<sub>0</sub>”；

（2）假设饮料只有一种价格为 2.5 元，硬币有 0.5 元和 1.0 元两种，考虑找零；

饮料自动投币售卖机控制器系统框图参见图 2-1 所示。

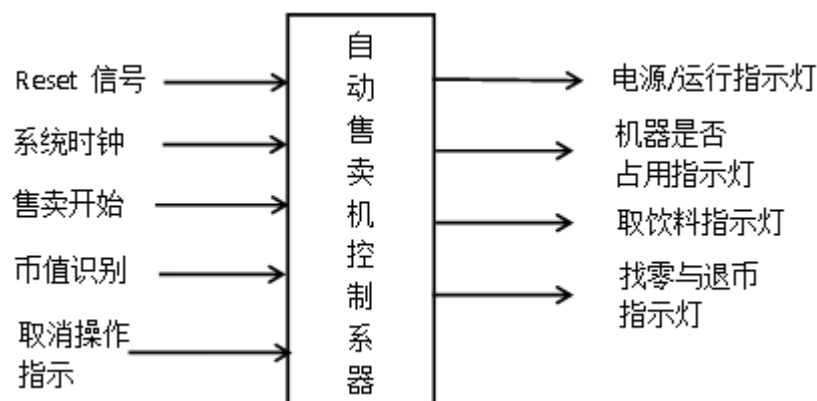


图 2-1 自动投币售卖机控制器系统框图

（a）系统时钟（clk）作为系统的同步信号；

（b）Reset 作为电路总清零信号；

（c）售卖开始信号（op\_start）

Op\_start = “1” 表示客户开始操作时的启动信号；

（d）币值识别信号（coin\_val）

由人民币识别器给出具体的币值（具体定义：coin\_val = “b01” 表示 0.5 元，coin\_val = “b10” 表示 1 元）；

（e）取消操作信号（cancel\_flag）

cancel\_flag = “1” 表示客户取消操作；

（f）机器是否占用指示灯（hold\_ind）

hold\_ind = “0” 表示该机是没被占用；

（g）取饮料指示灯（drinktk\_ind）

drinktk\_ind = “1” 表示客户可以取走饮料；

（h）找零与退币标志指示灯（charge\_ind）

charge\_ind = “1” 表示找零提示；

(i) 找零与退币币值 (charge\_val)

退给客户的币值 (charge\_val = b001 表示找 0.5 元, charge\_val = b010 表示找 1 元, charge\_val = b011 表示找 1.5 元, charge\_val = b100 表示找 2.0 元);

(3) 自动投币售卖机的状态确定

S0 : 初始态;

S1 : 已投币 0.5 元;

S2 : 已投币 1.0 元;

S3 : 已投币 1.5 元;

S4 : 已投币 2.0 元;

S5 : 已投币 2.5 元;

S6 : 已投币 3.0 元。

(4) 自动投币售卖机的状态转移图, 参见图 2-2 所示;

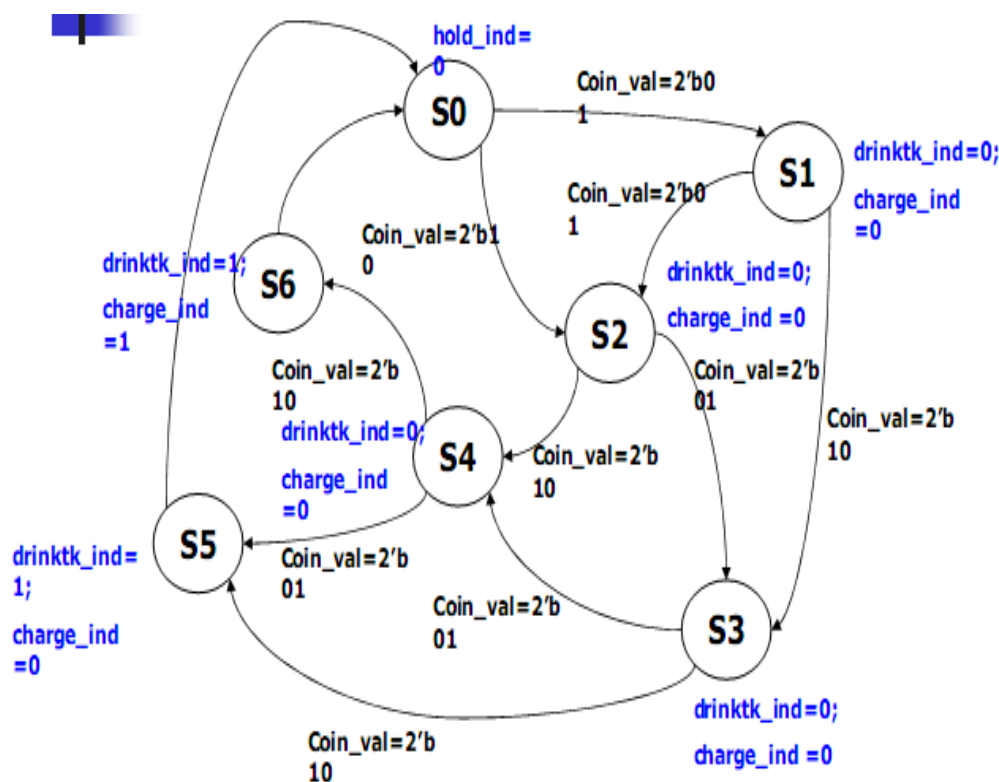


图 2-2 状态转移图

(a) 在 S0 状态下, 如果检测到 op\_start=1, 开始检测是否有投币, 如果有, 一次新的售货操作开始;

(b) 在状态 S1/S2/S3/S4 下, 如果检测到 cancel\_flag=1, 则取消操作, 状态回 S0, 并退回相应的币值;

(c) 在状态 S5 下, 卖出饮料不找零; 在状态 S6 下, 卖出饮料并找零;

(d) 在状态 S5 和 S6 操作完后, 都返回状态 S0, 等待下一轮新的操作开始;

(e) 只有在 S0 状态下, 且 hold\_ind=0 时, 可以可以进行新一轮操作, 其它状态下都为 hold\_ind= 1。

(5) 在开发板上验证设计的正确性。

---

## 2.2、设计思路

综合分析设计内容，系统的输入输出信号如下：

输入信号： clk, rst;

输入信号： 操作开始： op\_start; 定义 1 开始操作

输入信号： 投币币值： coin\_val; 定义 2' b01 表示 0.5 元; 2' b10 表示 1 元

输入信号： 取消操作指示 : cancel\_flag; 定义 1 为取消操作

输出信号： 机器是否占用： hold\_ind; 定义 0 为不占用，可以使用

输出信号： 取饮料信号： drinktk\_ind; 定义 1 为取饮料

输出信号： 找零与退币标志信号： charge\_ind; 定义 1 为找零

输出信号： 找零与退币币值： charge\_val; 定义 3' b001 表示找 0.5 元; 3' b010 表示找 1 元; 3' b011 表示找 1.5 元; 3' b100 表示找 2.0 元。

。。。。补充完善

例如：可以考虑添加 7 段数码显示管提示输入或退出币值。

### 3、题目三：电梯控制系统设计

设计并利用 Nexys4 开发板来实现一个 8 层楼的电梯控制系统，它的人机界面示意图，参见图 3-1 所示。

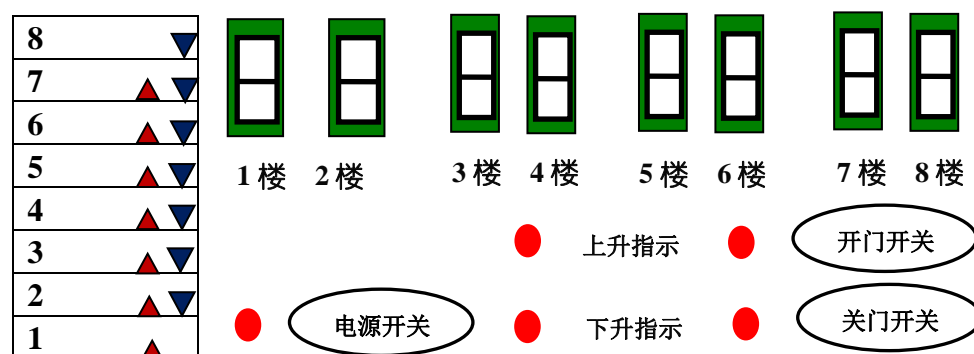


图 3-1 电梯控制系统人机界面示意图

#### 3.1、设计内容

(1) 电源开关同时作为**电路总清零信号 (Reset)**，当 **Reset = Off** 不工作、为 **On** 时电路进入初始状态，且电源指示灯亮（又称电梯运行指示灯），电梯初态可在任意楼层；

(2) 电梯可上行、下降，它到达某楼层开门后有 5s 的自动关门设置，若在这 5s 期间，按下关门键，则立刻关门；

(3) 电梯处于某层时，可以选择要到达的楼层，但是选择本楼层没有任何反应；

(4) 能显示电梯当前所在楼层；

(5) 电梯上行时，上行指示灯亮，下行指示灯灭；

(6) 电梯下行时，下行指示灯亮，上行指示灯灭；

(7) 电梯门打开时，开门指示灯亮，否则，灭；

(8) 电梯门关闭时，关门指示灯亮，否则，灭；

#### **(9) 电梯运行原则**

原则 1：每个被选择的楼层都要停，然后继续运行；

原则 2：相邻两层运行时间为 10s，采用时间倒计时实现；

原则 3：目前运行方向优先，仅当电梯已到达当前设定的最高或最低层后，方能改变设定的运行方向和楼层；

原则 4：可自定义某些原则，例如：如何减少等待时间等。

。。。。

结合上面的设计内容，参见图 3-2，采用 Verilog HDL 设计，在开发板上验证设计的正确性。

(1) 利用开发板上的 2 个 7 段数码显示管作为电梯当前所在楼层的显示；

(2) 利用开发板上的 2 个 7 段数码显示管作为相邻两层运行时间 (10s,) 采用倒计时的显示;

(3) 设计者可以灵活利用开发板上的输入、输出信号来实现所需。

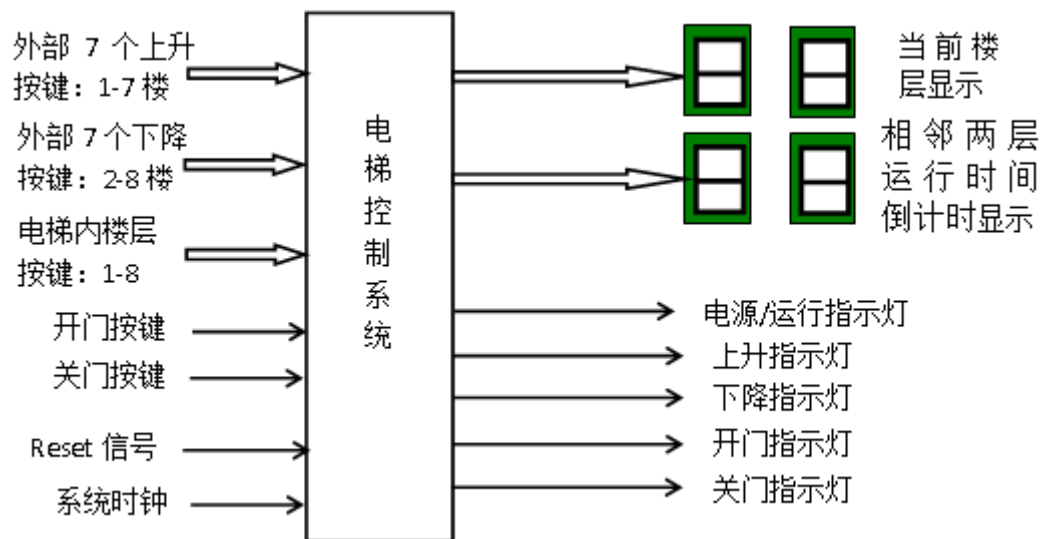


图 3-2 电梯控制系统框图

### 3.2、设计思路

自己完成。。。。

#### 4、题目四：交通灯控制系统设计（4 选 1）

设计并利用 Nexys4 开发板来实现一个交通灯控制系统。

在主干道与次干道公路十字交叉路口，为确保人员、车辆安全、迅速地通过，在交叉路口的每个入口处设置了红、绿、黄三色信号灯。红灯禁止通行；绿灯允许通行；黄灯亮提醒行驶中的车辆减速不要越过禁行线。

交通灯控制系统示意图参见图 4-1 所示。

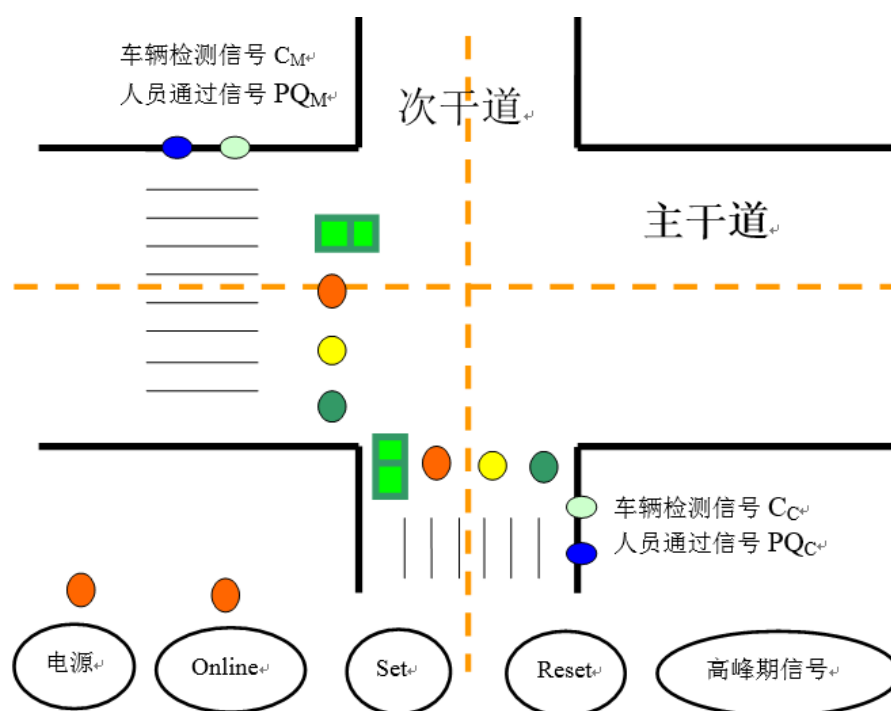


图 4-1 交通灯控制系统示意图

#### 4.1、设计内容

（1）电源开关同时作为**电路总清零信号（Reset）**，当 **Reset = Off** 不工作、为 On 时电路进入初始状态，且电源指示灯亮（又称交通灯控制器运行指示灯），控制器初始状态为：主干道通行，次干道禁止通行；

（2）主干道红灯时间等于次干道绿灯时间，主干道绿灯时间等于次干道红灯时间，主干道黄灯时间等于次干道黄灯时间；

（3）路口指示灯规则为：“红--绿--黄”循环；

（4）控制器运行/暂停控制信号  $Set = 0$  时，控制器暂停运行， $Set=1$  开始运行（接着原来的状态进行运行）；

（5）由交通控制中心发出的总控制台控制信号（Online），当  $Online=1$ ，本地交通灯控制器控制权“失效”，且主干道放行，次干道禁止通行、当  $Online=0$  本地交通灯控制器恢复控制权（接着原来的状态进行运行）；

（6）控制参数假设：红灯 16 秒，黄灯 3 秒，绿灯 16 秒；



(7) 通行规则如下：

规则 1：高峰时期，主干道和次干道均有车，车辆较多一方绿灯放行时间可以加倍（可由操控人员设定，此时，行人通过请求信号： $PQ_M=1$ 、 $PQ_C=1$  无效）如果双边车辆基本相同，则双边通行时间不变；

规则 2：非高峰时期，无论主、次干道在绿灯放行时，如果有行人通过请求信号： $PQ_M=1$ 、 $PQ_C=1$ ，则在当前红灯或绿灯倒计时为 0 时，要考虑给予行人通行；

规则 3：主干道和次干道均无车，主干道长期通行，主干道绿灯亮，它的倒计时时间为 16s，归 0 后重新开始倒计时；

规则 4：主、次干道一边有车辆，一边无车辆，则有车辆一边长期通行，有车辆一边绿灯亮，它的倒计时时间为 16s，归 0 后重新开始倒计时。

规则 5：可自定义，例如：添加相邻路口之间红、绿灯的联动，以加快车辆的通行等。

。。。。。

结合上面的设计内容，参见图 4-2，采用 Verilog HDL 设计，在开发板上验证设计的正确性。

完善交通灯控制系统框图 4-2。

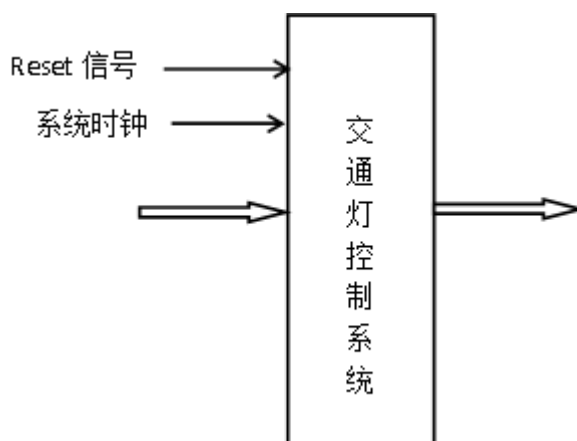


图 4-2 交通灯控制系统框图

## 4.2、设计思路

自己完成。。。。

## 5、题目五：洗衣机控制系统设计（4 选 1）

设计并利用 Nexys4 开发板来实现一个全自动洗衣机控制系统的人机界面示意图，参见图 5-1 所示。

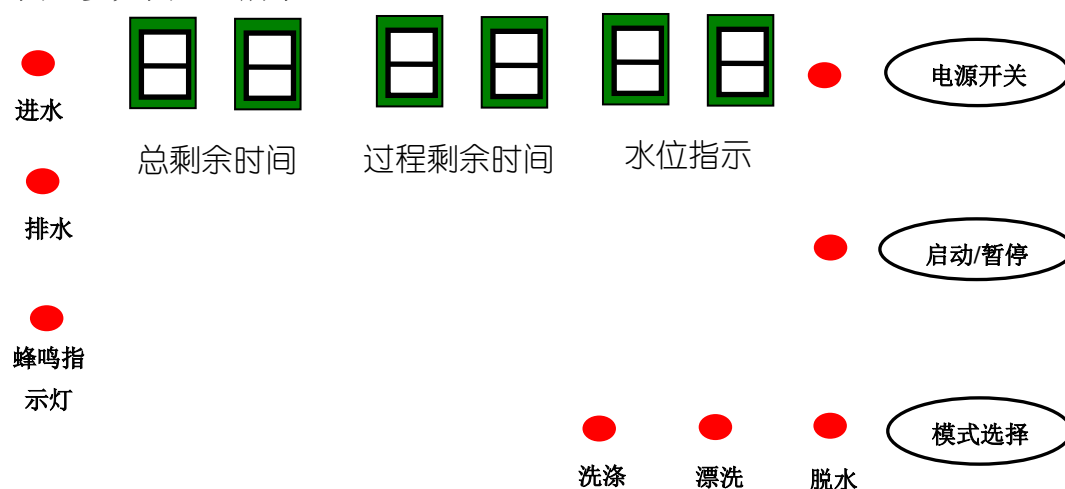


图 5-1 洗衣机控制器示意图

### 5.1、设计内容

（1）电源开关同时作为**电路总清零信号（Reset）**，当 **Reset = Off** 不工作、为 On 时电路进入初始状态，且电源指示灯亮（又称洗衣机控制器运行指示灯），控制器初始状态为：“洗漂脱”；

#### （2）启动/暂停

启动：在电源打开，“启动”指示灯不亮的情况下，按一下“启动/暂停”，“启动”指示灯亮，洗衣机开始工作或恢复暂停时的工作状态并开始工作；

暂停：在电源打开，“启动”指示灯亮的情况下，按一下“启动/暂停”，“启动”指示灯灭，此时洗衣机处于“暂停”状态，等待重新“启动”。

**暂停后可以重新选择洗衣程序；**

#### （4）洗衣机洗衣程序

##### （a）时间设置

**进水为 3 分钟**（为了方便描述假定衣物重量为 3kg）；洗衣 9 分钟；排水 3 分钟；**脱水/甩干 3 分钟**，漂洗 6 分钟；

**说明：**进水时间随衣物重量改变，这里假定衣物重量分为 5kg、4kg、3kg、2kg 4 种情况，控制器的“水位”控制分别对应为进水 5、4、3、2 分钟。

##### （b）洗涤程序

进水---洗衣，共 12 分钟；正在进行洗涤时，洗涤灯闪烁，正在进水时，进水灯亮；

##### （c）漂洗程序

排水---甩干---进水---漂洗，共 15 分钟，正在进行漂洗时，漂洗灯闪烁，正在进水时，进水灯亮，正在进行排水和脱水/甩干时，对应指示灯亮；

##### （d）脱水程序

---

排水---甩干，共 6 分钟。正在进行排水时，排水灯闪烁，正在进行脱水/甩干时，脱水灯亮；

### **(5) 洗衣模式**

目前考虑有以下 6 种洗衣模式可供选择。

为了方便提示洗衣过程，假定未完成的洗衣过程其对应的指示灯亮，正在进行的洗衣过程其对应的指示灯闪烁，已完成的洗衣过程前对应指示灯熄灭。

#### **(a) 洗漂脱**

它含有“洗涤”、“漂洗”、“脱水”3 个过程，整个程序流程包括：洗涤 12 分钟（进水---洗衣）---漂洗 15 分钟（排水---甩干---进水---漂洗）---脱水 6 分钟（排水---甩干），共 33 分钟。

#### **(b) 单洗（仅洗涤）**

它含有洗涤 1 个过程，整个程序流程包括：洗涤 12 分钟（进水---洗衣）。

#### **(c) 洗漂**

它含有“洗涤”、“漂洗”2 个过程。整个程序流程包括：洗涤 12 分钟（进水---洗衣）---漂洗 15 分钟（排水---甩干---进水---漂衣），共 27 分钟；

#### **(d) 单漂（仅漂洗）**

它含有“漂洗”1 个过程。整个程序流程包含：漂洗 15 分钟（排水---甩干---进水---漂衣）。

#### **(e) 漂脱**

它含有“漂洗”、“脱水”2 个过程。整个程序流程包含：漂洗 15 分钟（排水---甩干---进水---漂衣）---脱水 6 分钟（排水---甩干），共 21 分钟。

#### **(f) 单脱（仅脱水）**

它含有“脱水”1 个过程。整个程序流程包含：脱水 6 分钟（排水---甩干）。

### **(6) 模式选择**

系统共有 6 种洗衣模式供选择，初始状态为“洗漂脱”洗衣模式，每按一次模式选择，洗衣模式改变一次，改变顺序为：“洗漂脱”——“单洗”——“洗漂”——“单漂”——“漂脱”——“单脱”——“洗漂脱”循环。

在选择洗衣模式时，总剩余时间 7 段数码管会显示新的洗衣模式的总时间，当前模式时间 7 段数码显示管会显示当前所选洗衣模式中第一个洗衣程序的所需时间。同时“洗涤”、“漂洗”和“脱水”指示灯也随着新的洗衣模式的变化而变化。某个指示灯亮，表示在本洗衣模式中包含对应的工作流程；反之则不包含。

洗衣模式选择完毕，按启动按钮，该洗衣模式启动，当该洗衣模式结束时要发出报警声，并回到洗漂脱模式；

### **(7) 水位**

控制器会根据衣物重量（假定衣物重量分为 5kg、4kg、3kg、2kg 4 种情况）自动选择“水位”，水位由 2 个 7 段数码显示管指示；

### **(8) 蜂鸣**

在按每个按钮时，都要有“嘟”的一声提示，在每个程序结束时，应有“嘟嘟”“嘟嘟嘟”，“嘟嘟嘟”三声提示，同时“启动”指示灯熄灭。（由于器件短缺，用蜂鸣指示灯的闪烁表示报警效果）

### **(9) 关闭**

每次所选的洗衣模式结束全部工作后，10秒内无操作，自动执行“电源”开关关闭操作。

### **(10) 自定义规则**

。。。。  
结合上面的设计内容，参见图 5-2，采用 Verilog HDL 设计，在开发板上验证设计的正确性。

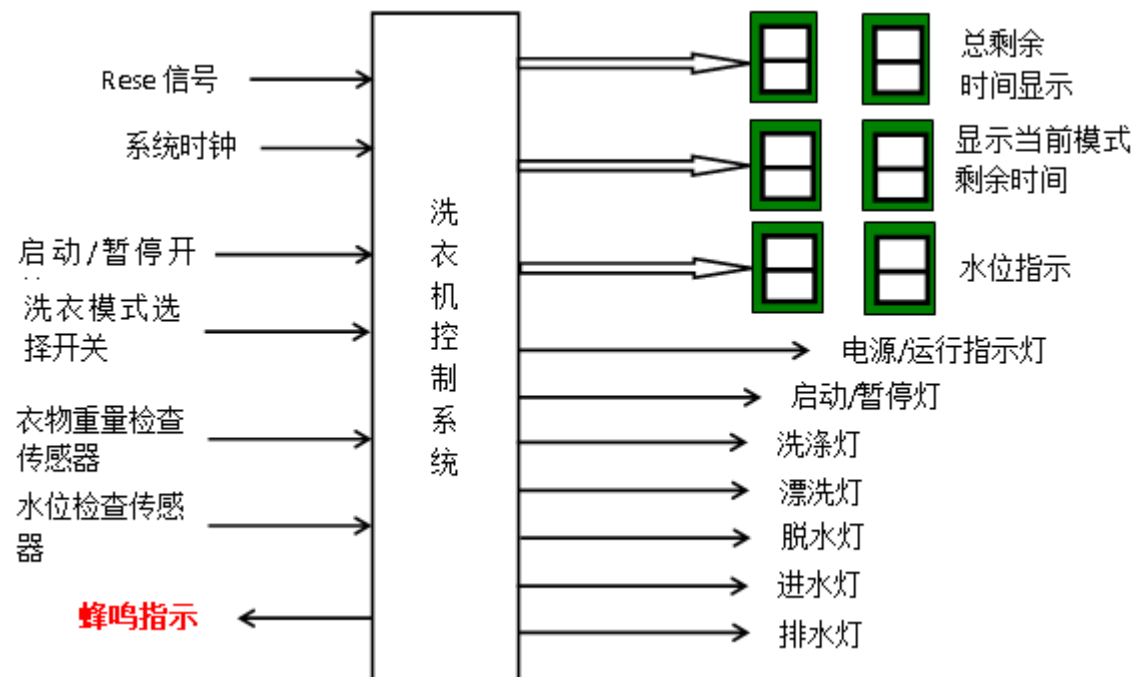


图 5-2 洗衣机控制系统框图

## 5.2、设计思路

自己完成。。。。

## 6、题目六：药片装瓶系统设计（4 选 1）

### 6.1、设计内容

如图 6-1 左面所示，药片由输送管送入漏斗装置中，后者颈部每次只允许一粒药片掉进传送带上的瓶子里。漏斗的颈部有一个光传感器，它探测到每一粒药片后产生一个电脉冲信号。这个脉冲传送到计数器中，使其计数加 1，这样在药片装入瓶子过程的任一时刻，计数器都保存着瓶子中药片数量的二进制数。这个二进制数以计数器通过并行导线传送到比较器的输入端 B。

另一方面，每个瓶子中要装入的固定药片数量（例如 50 片）通过键盘手动设置。按键信号经过编码器编码后送到寄存器 A 保存，而代码转换器 A 将寄存器 A 中的 BCD 数转换成二进制数送到比较器输入端 A。

假设每个瓶子要装 50 粒药片，当计数器的数值达到 50 后，比较器  $A=B$  出现高电平，指示瓶子已装满，立即关闭漏斗颈上的阀门使药片停止下落，与此同时它使传送带移动下一个瓶子到漏斗的下面。当瓶子到达漏斗颈的正下方时，传送带的控制电路产生一个脉冲信号使计数器清 0，比较器  $A=B$  变成低电平，打开漏斗阀门，重新开始药片滴落。

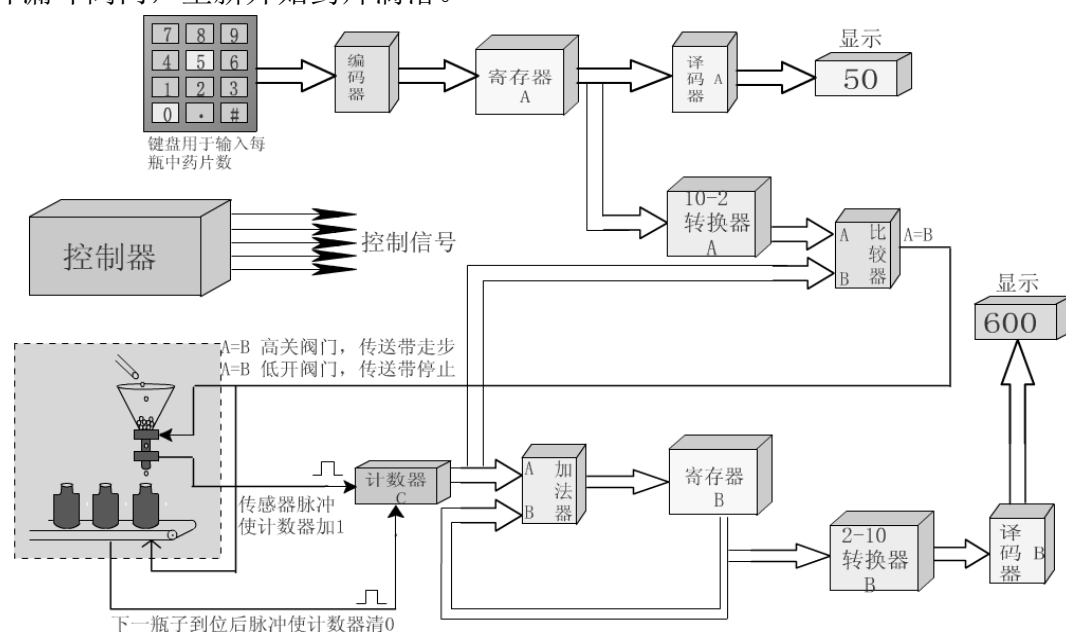


图 6-1 药片装瓶计数显示系统框图

结合上面的药片装瓶系统设计实例，参见图 6-2，采用 Verilog HDL 设计，在开发板上验证设计的正确性。

（1）开发板上的 5 个 7 段数码显示管作为显示系统，显示每瓶药片及总药片的数量；

（2）用开发板上的红绿发光二极管来模拟对机电装置系统的输出，绿色灯亮表示启动机电装置，装瓶进行中；红色灯亮表示装瓶完成，机电装置关闭；

（3）输入子系统为包括 BCD 码，它表示每瓶装药数输入与装瓶开始脉冲输

- 入，设计要求每瓶最大药片数 50 粒，最多装 18 瓶；
- (4) 启动装瓶开始脉冲后，如果输入数量超出最大装瓶数或者为零，要求显示系统出现告警提示；
- (5) 漏斗感应器送来的药片装瓶信号用 2s 信号模拟，可以用实验台提供的 100MHz 的时钟分频产生；
- (6) 在开发板上验证设计的正确性。

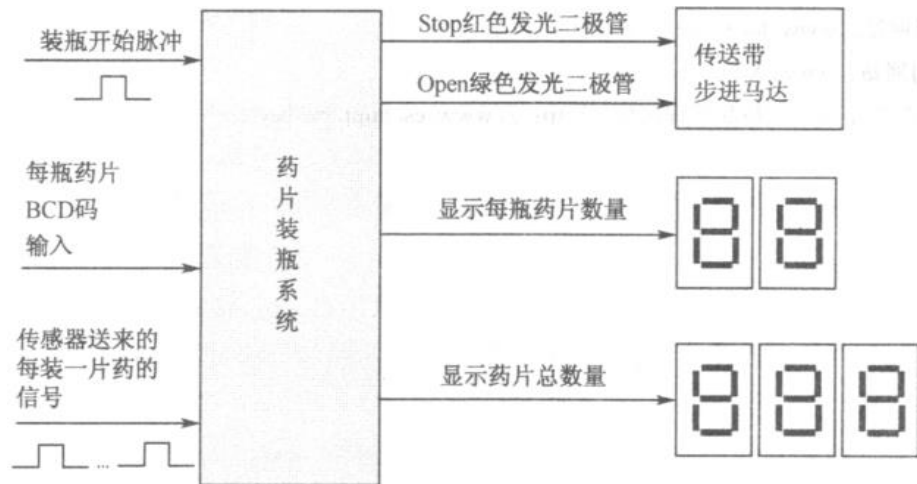


图 6-2 药片装瓶系统

### 6.2、设计思路

图 6-3 示出了药片装瓶控制与显示系统的组成总框图，它可以划分为如下子系统：

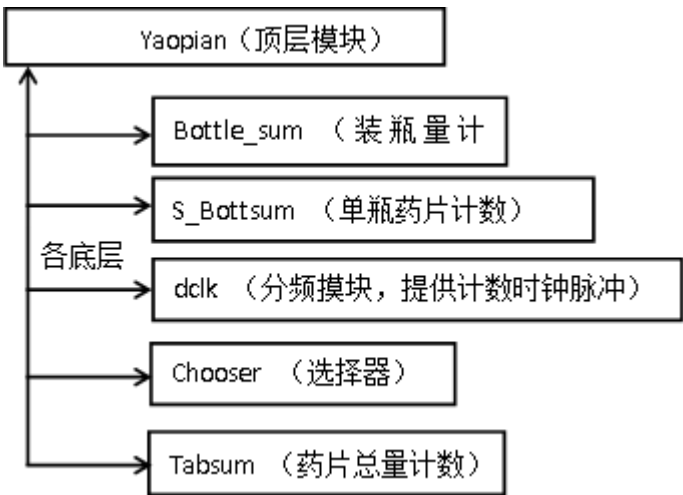


图 6-3 药片装瓶控制模块图

- (1) 装瓶量计数模块：用“进位”对总瓶数进行计数；
- (2) 单瓶药片计数模块：用时钟脉冲对药片数进行计数；
- (3) 分频模块：用系统的 100MHz 脉冲进行分频，获得所需计数脉冲；
- (4) 药片总量计数模块：可采用时钟脉冲来对药片数进行计数；
- (5) 选择器模块：对开始/暂停信号进行判断, 决定输出瓶数还是当前药片数。

。。。。补充完善