# 实验五 状态机综合实验

## 实验目的

本实验的目的是运用状态机原理实现一个具有实用功能的应用，将状态机原理运用到项目开发中。该实验需要设计者在分析阶段对电路各个模块之间的接口有较为清晰的认识，熟练状态机的设计，培养自顶向下的设计思路，进一步理解状态机原理在数字电路设计过程中的便捷之处。为后续复杂数字系统设计打好基础。

## 实验内容

**实验5.1——按照以下要求设计一个电子秤：**

1. 功能

* 单次计价：输入物品的重量、单价，显示物品的总价（=重量\*单价）。
* 累计计价：
* 第一次按下累计按键，记住当前物品的总价（当前物品记为物品1），数码管依次显示:

AC 次数 应付总价

* 继续输入物品2的重量、单价，显示物品2的总价。按下累计按键，将本次物品2的总价累加进之前的用户应付总价中，数码管显示：

AC 次数 总价

* 依次购买物品3、4……，每个物品后都通过累计按键将本次物品总价累计到应付总价中
* 退出累计状态：按下清除累计按键，恢复普通状态。

1. 显示

重量：2个数码管（十进制显示）

单价：2个数码管（十进制显示）

总价：4个数码管（十进制显示）

AC：显示这两个字母

1. 接口

* 输入

重量：拨码开关，4位（二进制输入）

单价：拨码开关，4位（二进制输入）

累计：按键，1个

复位：按键，1个

* 输出

数码管：8个

效果演示：

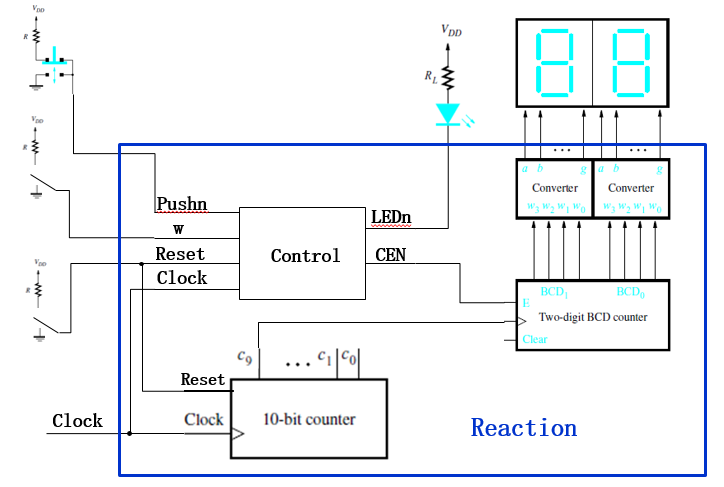
**见演示视频。**

**鼓励在基本要求的基础上进行功能扩展。**

## 实验要求

1. 描述状态机设计思路，画出状态机的**状态转移图**
2. 在实验报告中提交系统级设计模块图、设计代码、激励程序（不必须包含所有模块的）、仿真波形结果截图（与激励配套）、板级实测验证结果照片。

其中，系统级设计模块图要求给出整个系统的数据输出信号，系统内各个子模块的输入输出信号和模块间的连接关系（图1仅做示意，不需要画出数码管/LED灯/按键等，给出其信号名及位宽即可）。



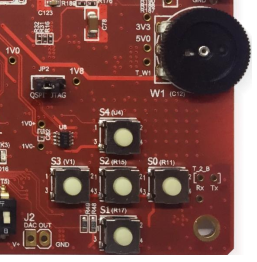
**图1 系统级设计模块图（示意图）**

1. 提交实验报告和所有源程序文件的压缩包。

## 实验参考

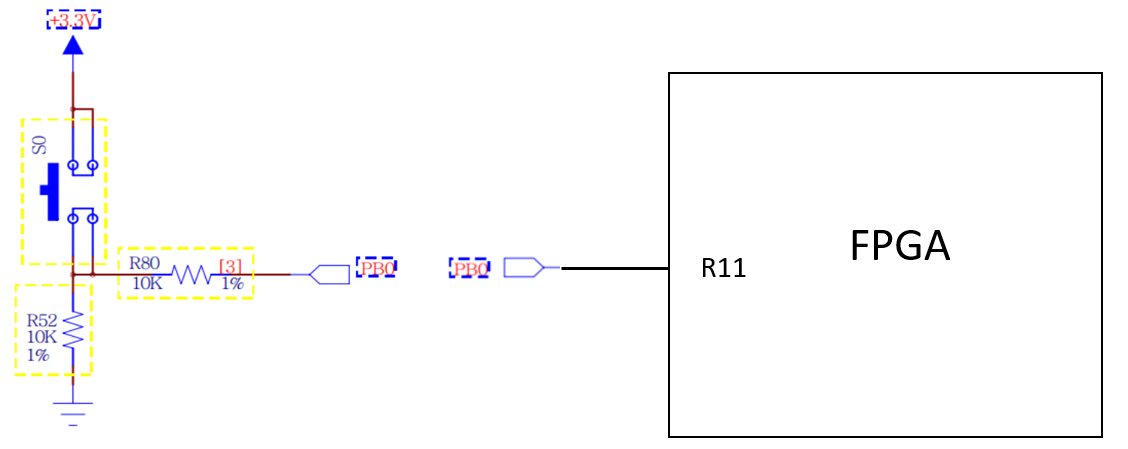
1）EG01按键说明

EGO1具有2个专用按键和5个通用按键。本实验中主要用到5个通用按键，在PCB板上的名称是S0，S1，S2，S3，S4。



下图以S0为例显示了通用按键的电路原理图，图中按键输出信号PB0连接到了PFGA的R11管脚（提示：电路原理图中，通常以同一个名称来命名多个信号线，表示这几个信号线都是同一个根信号线，这样使得复杂的原理图变简洁）。从该原理图可知：

* 当S0按键没有按下时，PB0低电平；
* 当S0被按下时，PB0为高电平；

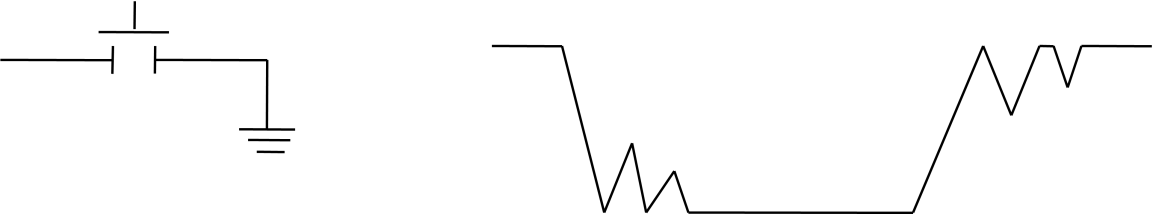


S0，S1，S2，S3，S4的管脚约束如下：

|  |  |
| --- | --- |
| 按键名称 | FPGA管脚 |
| S0 | R11 |
| S1 | R17 |
| S2 | R15 |
| S3 | V1 |
| S4 | U4 |

2）按键防抖说明

为了保证键每闭合一次 FPGA 仅作一次处理，必须去除键按下时和释放时的抖动。开发板使用的按键是触点式的，由于按键是机械触点，当机械触点断开、闭合时，会有抖动。



按键抖动示意图

按键抖动示意图中的这种抖动对于人来说是感觉不到的，但对于FPGA或者处理处理器来说，其运行速度的是在微秒级甚至纳秒级，而机械抖动的时间至少是毫秒级，因此这种抖动是一个“漫长”的时间。

对于按键存在的抖动，如果在抖动过程中高低电平的状态没发生变化，则这个抖动我们是不需要考虑的。但是，如果在抖动过程中高低电平状态发生了变化甚至是频繁变化，则这个抖动我们就消除，下面所提到的抖动就是这种类型的抖动。 为使 FPGA 能正确的读出按键的状态，对每一次按键只作一次响应，就必须考虑如何去除抖动，常用的去抖动的方法有两种：硬件方法和软件方法。

FPGA 设计中，常用软件法去抖，因此对于硬件方法我们在此不作介绍。 软件法去抖其实很简单，按键初始状态为低电平，当 FPGA 获得键值为 1 的信息后，不是立即认定按键已被按下，而是延时 5ms 或更长一些时间后再次检测按键，如果仍为低，说明按键的确按下了，这实际上是避开了按键按下时的抖动时间。而在检测到按键释放后再延时 5ms，消除后沿的抖动，然后再对键值处理。当然，实际应用中，按键的质量也是千差万别，要根据按键的不同，来设定这个延时时间，通常这个延时时间不会太短，一般设为 5~20ms。具体做法是，将按键信息延时 3 次取样 3 次，每次延迟取样间隔约为 5ms，当这 3 个取样值都一样时，说明抖动已消失，如果 3 个取样值不一样，说明抖动存在，直至这 3 个取样值一样时，才认为按键稳定。将这 3 个取样值与运算后得到的信号作为按键的状态，这个按键状态可作为稳定的按键输入，参与到后续对按键的处理操作。

3）拨码开关说明

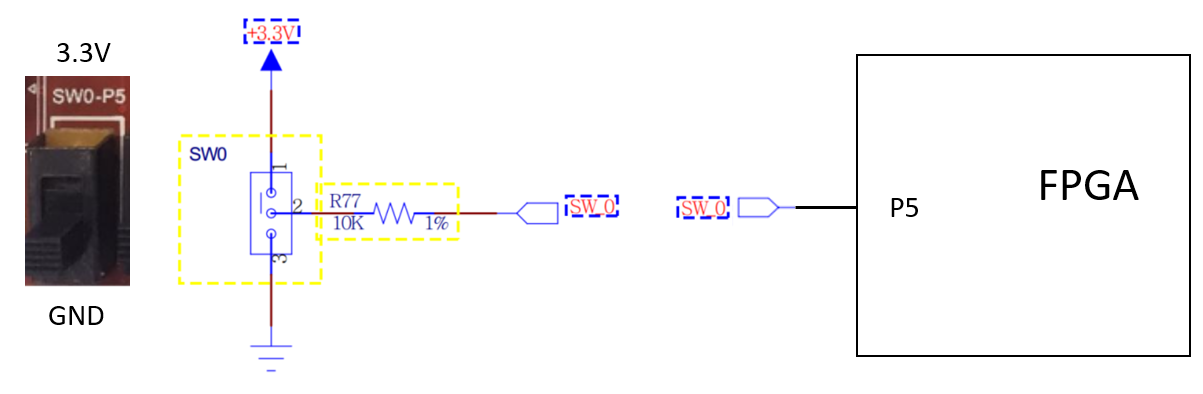
EGO1上有8个拨码开关（SW0~SW7）



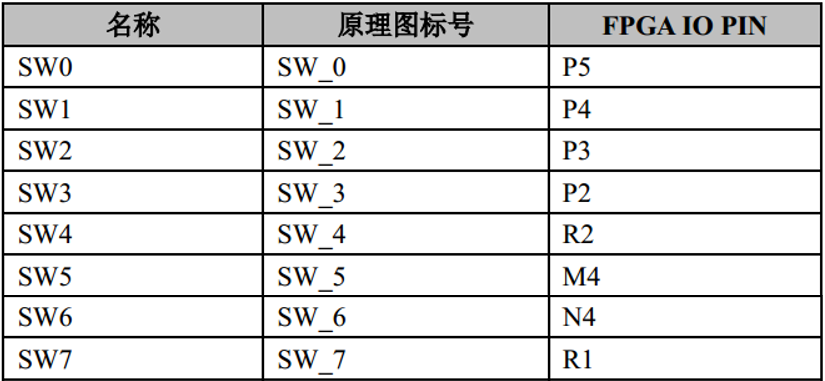
SW0~ SW7

以SW0为例，拨码开关的原理图如下图所示，开关输出管脚SW\_0与FPGA的P5管脚相连。开关具有三个状态：

* 开关拨到3.3V侧，SW\_0为高电平；
* 开关拨到GND侧，SW\_0为低电平；
* 开关拨到中间，SW\_0位悬空，建议不要用悬空。



拨码开关管脚约束



4）七段数码管说明

EGO1上共有8个七段数码管，DK1~DK8。

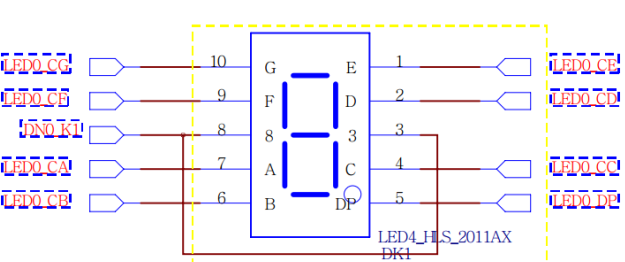


如下图以DK1为例说明**单个数码管**的控制，每个数码管有10个管脚（A~G,DP,8，3）。

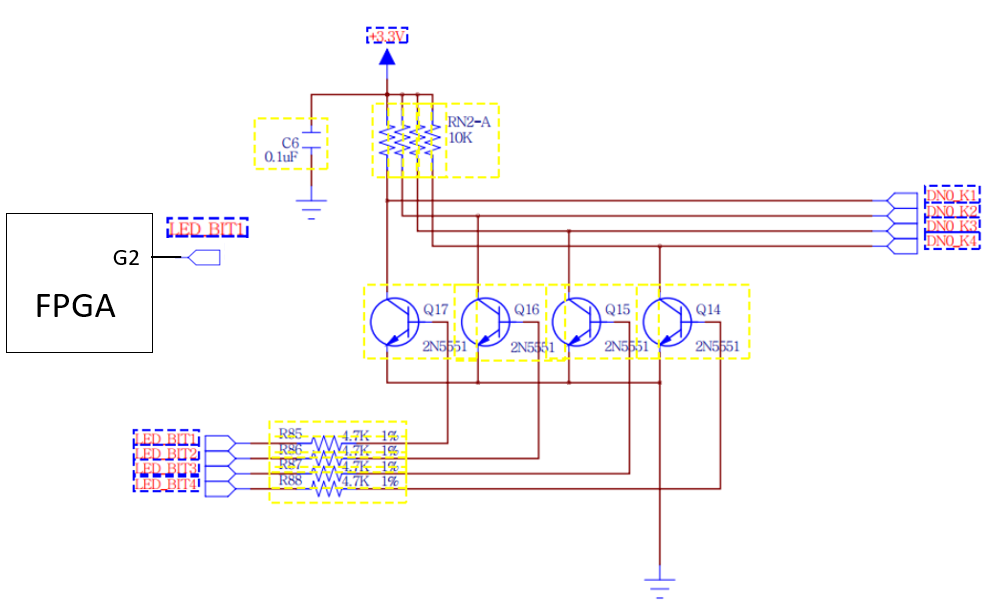
段选信号（高电平有效）：A~G控制七段显示和DP控制小数点显示；

控制信号：8和3位驱动管脚（信号线名称为DN0\_K1）。

其中，DK1 的 A~G和 DP分别通过信号线LED0\_CA~LED0\_CG, LED0\_DP，连接到FPGA的B4，A4，A3，B1，A1，B3，B2和D5管脚。



**数码管的使能**：数码管为共阴极数码管，即控制信号（DN0\_K1）输入低电平使能数码管。如下图所示，控制信号由三极管驱动，通过信号线LED\_BIT1连接到FPGA的G2管脚。因此，**G2高电平使能DK1**。



数码管使能控制

**数码管的分组管脚复用**

如下面原理图所示，8个数码管分为两组：

第一组：DK1~DK4，如图可知所有DK1~DK4的段选信号（A~G,DP）管脚复用一组信号线（LED0\_CA~LED0\_CG,LED0\_DP）。例如：DK1~DK4的A管脚都连到了LED0\_CA。 而 DK1~DK4的使能信号是单独控制的。如上图所示，DK1~DK4的使能管脚分别连到了DN0\_K1~DN0\_K4，链接到FPGA的使能控制信号分别是LED\_BIT1~LED\_BIT4。

第二组：DK5~DK8，也是段选信号复用（LED1\_CA~LED1\_CG,LED1\_DP），使能信号单独控制LED\_BIT5~LED\_BIT8。

更详细的原理图参见EGO1电路原理图。