

实验 4 矩阵键盘扫描控制电路设计

Advanced Electronics System Lab, 202309

一、实验目的

- 1、进一步学习并掌握 Quartus II 设计的方法及步骤；
- 2、熟悉 Verilog 语言电路设计方法；
- 3、熟悉 Verilog Test Bench（测试平台）的设计；
- 4、学习并掌握利用 Verilog 描述并设计电路的方法及步骤；
- 5、学习并掌握矩阵键盘的扫描输入方法及实现过程。

二、实验原理

矩阵键盘是一个由 4×4 的按键开关组成的阵列，可实现 16 种状态的输入。 4×4 矩阵键盘的硬件结构如图 1 所示，其中“+3V3”为 3.3V 的正电源，BUTTON 按键，R 为电阻。SW_R0~SW_R3（简称为 R0~R3）为按键阵列连接到 FPGA 的行控制信号，SW_C0~SW_C3（简称为 C0~C3）为按键阵列连接到 FPGA 的列控制信号。图 2 是矩阵键盘的实物图及与 FPGA 的连接信号。

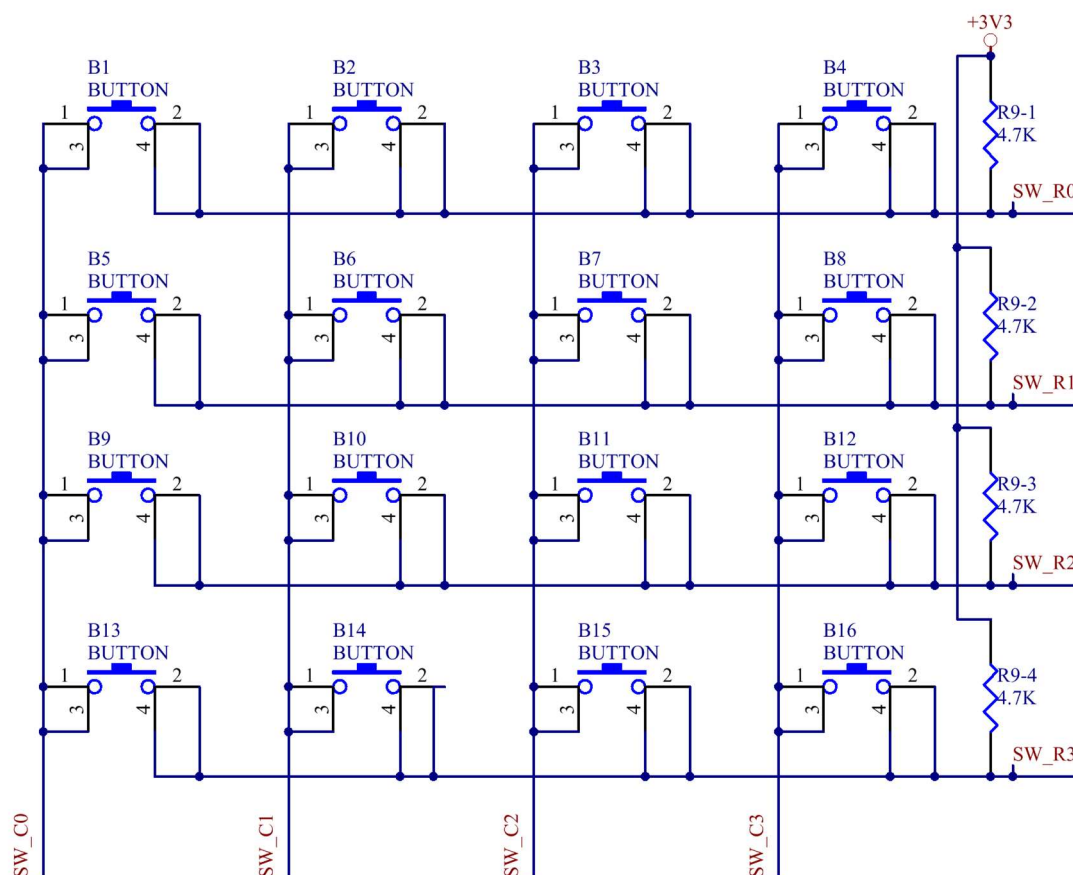


图 1 4×4 矩阵键盘硬件连接

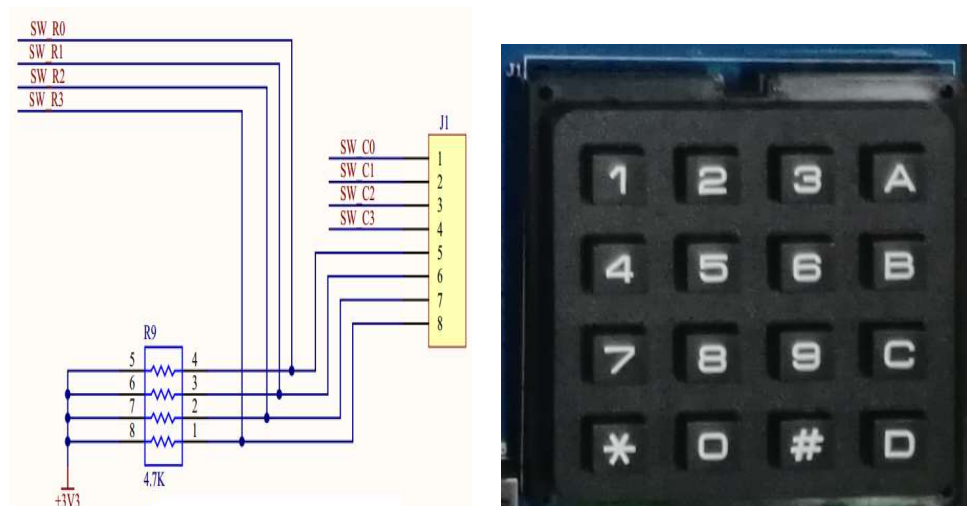


图2 矩阵键盘实物图及与FPGA连接的信号

另外，连接到FPGA的行、列信号均为3.3V的LVTTL电平标准，即电压小于0.8V为低电平，高于2.0V为高电平。

通过图1的硬件连接图可以看出，行信号通过4.7K的电阻上拉到3.3V的电源，也就是说如果FPGA通过对应的I/O引脚来读取4×4按键阵列的行信号时，应为高电平“1”，没有按键按下时读取列信号为无输入状态，即列信号是悬空的。

那么如何在FPGA中判断矩阵键盘中的哪个键被按下了呢？要实现按键识别功能，就不能同时读取行和列的信号，而应将列信号作为输出，行信号作为输入，让列信号不断输出一个扫描序列，再通过读取行信号来判断哪个按键被按下。

具体的做法是：四个列控制信号C0~C3循环输出“1110、1101、1011、0111”来驱动键盘阵列，每输出一个列序列，紧接着就读取相应的4个行信号。通过读取的数据或状态来判断16个按键中有没有按键被按下、哪个键被按下，并对其状态做编码输出。此电路不停的工作，以便实时准确地获取键盘的输入状态，以供其它电路使用，从而实现了键盘阵列的扫描输入。例如，当列驱动信号输出“1110”（信号排列顺序：C3、C2、C1、C0）时，下一时钟周期读取行信号为“0111”（信号排列顺序：R3、R2、R1、R0），则可判断按键B13（*）按下。

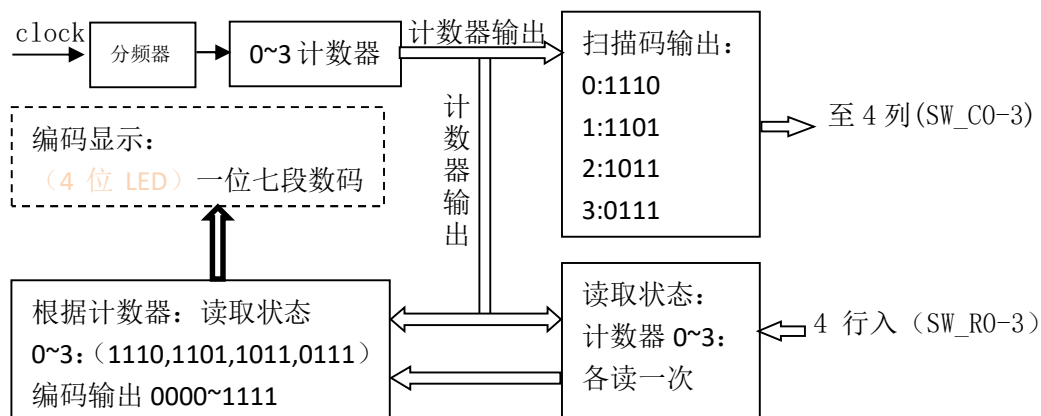


图3 矩阵键盘扫描控制电路框图

根据以上分析，我们可以得到 4×4 矩阵键盘动态扫描控制电路的框图如图 3 所示。FPGA 中的控制电路要不断向列信号线发送驱动信号，同时读取行信号线的状态进行判断。如果有按键按下，还要将按键的号码显示在一个数码管上。

三、编写仿真测试平台（Test Bench）

针对以上设计的矩阵键盘动态扫描控制电路，编写 Verilog Test Bench 进行仿真。需要注意的是，在编写包含时钟端口的 Test Bench 程序时，时钟信号要单独产生，其它的输入信号另外产生。比如：

```
initial clk = 0;
always #1 clk = ~clk; //产生时钟信号
```

```
.....
initial begin //产生其他信号
#0 a = 0;
#2 a = 1;
#2 a = 0;
#1 $stop;
end
```

若时间单位为 10ns，以上语句产生了周期为 20ns 的时钟信号，先 10ns 低电平，后 10ns 高电平，然后不断地反复。

在编写时序设计 Test Bench 时，注意不要在时钟脉冲有效沿（本实验为上升沿）改变地址、数据或控制信号的高低电平。同时还要注意满足建立时间和保持时间。

在仿真结束后，通过“Tools”菜单”Netlist Viewers”子菜单下的“RTL Viewer”命令查看设计电路的 RTL 级的电路结构，并进行截图、保存。

四、管脚分配

通过查阅“实验指导书 altera_cyclone5.pdf”文档中的“附件一 FPGA 管脚约束”，分别为设计中的输入和输出选定管脚并添加到 Quartus II 工程。

五、结束

至此，我们完成了 Altera FPGA 设计的基本流程和步骤。如果你结束了今天的实验，请备份实验数据和实验记录，关闭所有打开的软件并关闭计算机；关闭实验箱的电源，并拆除所有相关的连线，恢复到实验之前的整齐样子，别留下任何物品。O(∩_∩)O 谢谢！

