

Laboratory Exercise 2

Numbers and Displays

这是设计能够执行二进制到十进制数转换和二进制编码 - 十进制 (BCD) 加法的组合电路的练习。

Part I

我们希望在7段显示器上显示HEX3至HEX0由开关SW₁₅₋₀设置的值。令SW₁₅₋₁₂, SW₁₁₋₈, SW₇₋₄和SW₃₋₀表示的值分别显示在HEX3, HEX2, HEX1和HEX0上。您的电路应该能够显示从0到9的数字, 并且应该将估值1010到1111视为无关紧要。

1. 创建一个新项目, 用于在Altera DE2板上实现所需的电路。本练习的目的是手动派生7段显示所需的逻辑功能。您应该在代码中仅使用简单的VHDL赋值语句, 并将每个逻辑函数指定为布尔表达式。
2. 编写提供必要功能的VHDL文件。在您的项目中包含此文件, 并在FPGA上分配引脚以连接到交换机和7段显示器, 如DE2板的用户手册中所示。使用VHDL设计的Quartus II简介中介绍了进行引脚分配的过程, 该教程可在DE2系统CD和Altera网站的大学计划部分获得。
3. 编译项目并将编译好的电路下载到FPGA芯片中。
4. 通过切换开关并观察显示来测试设计的功能。

Part II

您将设计一个电路, 将四位二进制数 $V = v_3v_2v_1v_0$ 转换为两位十进制数 $D = d_1d_0$ 。表1显示了所需的输出值。该电路的部分设计如图1所示。它包括一个比较器, 用于检查V的值何时大于9, 并使用该比较器的输出来控制7段显示器。您将通过创建VHDL实体来完成该电路的设计, 该实体包括比较器, 多路复用器和电路A (此时不包括电路B或7段解码器)。您的VHDL实体应具有四位输入V, 四位输出M和输出z。本练习的目的是使用简单的VHDL赋值语句, 使用布尔表达式指定所需的逻辑函数。您的VHDL代码不应包含任何IF-ELSE, CASE或类似语句。

Binary value	Decimal digits	
0000	0	0
0001	0	1
0010	0	2
...
1001	0	9
1010	1	0
1011	1	1
1100	1	2
1101	1	3
1110	1	4
1111	1	5

Table 1. Binary-to-decimal conversion values.

执行以下步骤：

1. 为您的VHDL实体制作Quartus II项目。
2. 编译电路并使用功能仿真来验证比较器，多路复用器和电路A的正确操作。
3. 增加您的VHDL代码以包括图1中的电路B以及7段解码器。更改代码的输入和输出以使用DE2板上的开关SW₃₋₀表示二进制数V，并显示HEX1和HEX0以显示十进制数字d₁和d₀的值。确保在项目中包含DE2板所需的引脚分配。
4. 重新编译项目，然后将电路下载到FPGA芯片中。
5. 通过尝试所有可能的V值并观察输出显示来测试电路。

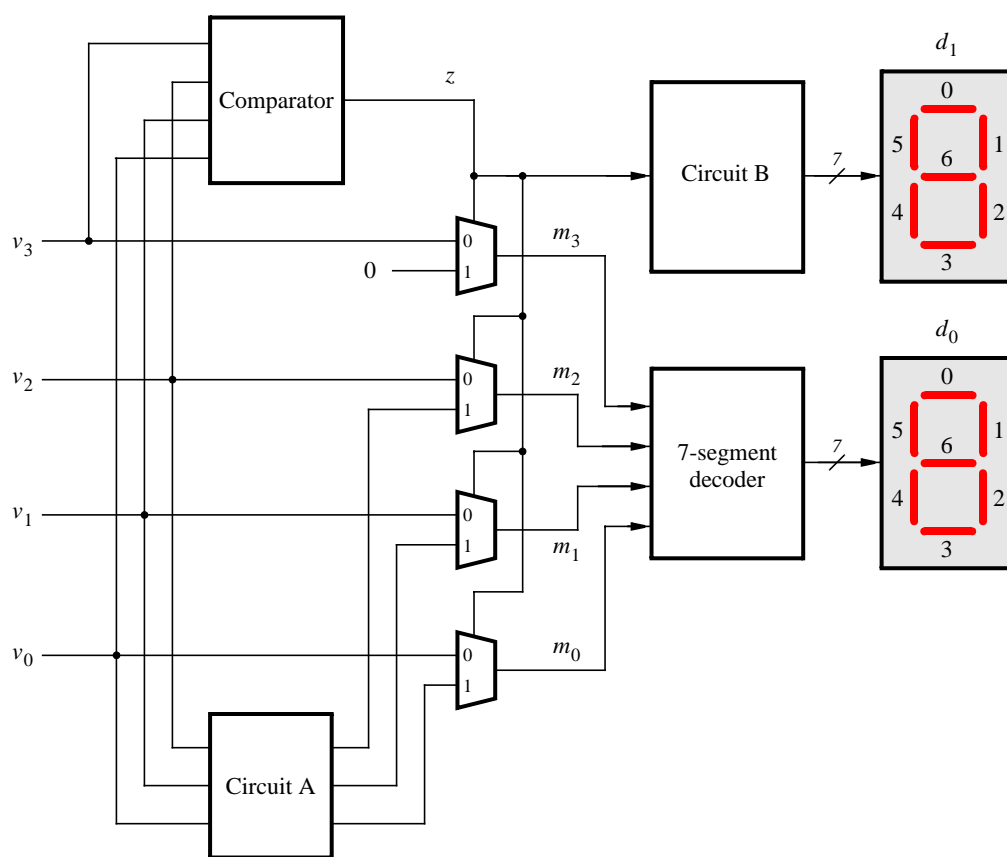


Figure 1. Partial design of the binary-to-decimal conversion circuit.

Part III

图2a示出了全加器的电路，其具有输入a，b和c_i，并产生输出s和c_o。图中的b和c部分显示了全加器的电路符号和真值表，它产生两位二进制和c_os = a + b + c_i。图2d示出了如何使用该全加器实体的四个实例来设计添加两个四位数的电路。这种类型的电路通常称为纹波进位加法器，因为进位信号从一个全加器传递到下一个加法器的方式。编写实现此电路的VHDL代码，如下所述。

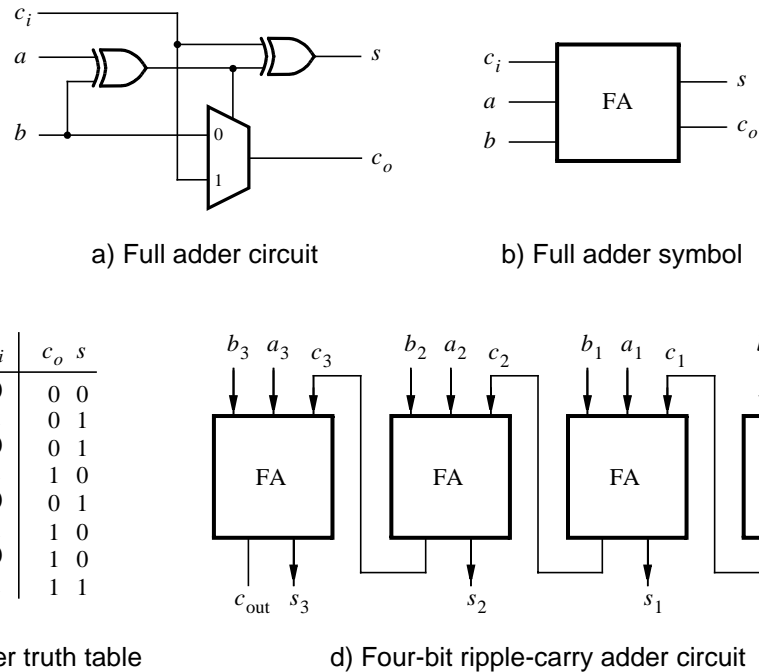


Figure 2. A ripple-carry adder circuit.

1. 为加法器电路创建一个新的Quartus II项目。为完整加法器子电路编写VHDL实体，并编写一个顶层VHDL实体，实例化该全加器的四个实例。
2. 使用开关SW_{7,4}和SW_{3,0}分别表示输入A和B。使用SW8作为加法器的随身数据。将SW开关连接到相应的红灯LEDR，并将加法器c_{out}和S的输出连接到绿灯LEDG。
3. 包括DE2板的必要引脚分配，编译电路，并将其下载到FPGA芯片中。
4. 通过尝试数字A，B和c_{in}的不同值来测试您的电路。

Part IV

在第二部分中，我们讨论了将二进制数转换为十进制数。有时建立使用这种表示十进制数的方法的电路是有用的，其中每个十进制数用四位表示。该方案称为二进制编码的十进制（BCD）表示。作为示例，十进制值59以BCD形式编码为0101 1001。

您将设计一个添加两个BCD数字的电路。电路的输入是BCD编号A和B，以及进位，c_{in}。输出应为两位数BCD和S₁ S₀。注意，该电路需要处理的最大总和是S₁ S₀ = 9 + 9 + 1 = 19。执行以下步骤。

1. 为您的BCD加法器创建一个新的Quartus II项目。您应该使用第III部分中的4位加法器电路为操作A + B产生4位和和进位。将这个最大值为19的结果转换为两个BCD的电路 数字S₁ S₀的设计方式与第二部分的二进制到十进制转换器非常相似。使用简单赋值语句编写VHDL代码以指定所需的逻辑函数 - 不要使用其他类型的VHDL语句（如IF-ELSE或CASE语句）来完成本练习的这一部分。

2. 分别使用开关SW₇₋₄和SW₃₋₀作为输入A和B，并使用SW₈进位。将SW开关连接到相应的红灯 LEDR，并将操作A + B产生的四位总和和输出连接到绿灯LEDG。在7段显示器HEX6和HEX4上显示 A和B的BCD值，并在HEX1和HEX0上显示结果S₁ S₀
3. 由于您的电路仅处理BCD数字，因此请检查输入A或B大于9的情况。如果发生这种情况，请打开绿灯LEDG₈指示错误。
4. 包括DE2板的必要引脚分配，编译电路，并将其下载到FPGA芯片中。
5. 通过尝试数字A，B和c_{in}的不同值来测试您的电路。

Part V

设计一个可以添加两个2位BCD编号的电路，A₁ A₀和B₁ B₀，以产生三位数BCD和S₂ S₁ S₀。使用第四部分中的两个电路实例来构建这个两位数的BCD加法器。执行以下步骤：

1. 使用开关SW₁₅₋₈和SW₇₋₀分别表示2位BCD编号A₁ A₀和B₁ B₀。A₁ A₀的值应显示在7段显示器HEX7和HEX6上，而B₁ B₀应显示在HEX5和HEX4上。在7段显示器HEX2，HEX1和HEX0上显示 BCD和，S₂ S₁ S₀。
2. 进行必要的引脚分配并编译电路。
3. 将电路下载到FPGA芯片中，并测试其运行情况。

Part VI

在第五部分中，您通过使用两部分VHDL代码实例为第四部分的一位BCD加法器创建了一个两位BCD加法器的VHDL代码。在VHDL代码中描述两位数BCD加法器的另一种方法是指定一个算法，如下面的伪代码所代表的算法：

```

1   $T_0 = A_0 + B_0$ 
2  if ( $T_0 > 9$ ) then
3       $Z_0 = 10$ ;
4       $c_1 = 1$ ;
5  else
6       $Z_0 = 0$ ;
7       $c_1 = 0$ ;
8  end if
9   $S_0 = T_0 - Z_0$ 

10  $T_1 = A_1 + B_1 + c_1$ 
11 if ( $T_1 > 9$ ) then
12      $Z_1 = 10$ ;
13      $c_2 = 1$ ;
14 else
15      $Z_1 = 0$ ;
16      $c_2 = 0$ ;
17 end if
18  $S_1 = T_1 - Z_1$ 
19  $S_2 = c_2$ 

```

看看可以用什么电路来实现这个伪代码是相当简单的。第1,9,10和18行代表加法器，第2-8和11-17行对应于多路复用器，并且对条件 $T_0 > 9$ 和 $T_1 > 9$ 的测试需要比较器。您将编写与此伪代码对应的VHDL代码。请注意，您可以在VHDL代码中执行加法运算，而不是第9行和第18行中显示的减法。本部分练习的目的是检查更多依赖VHDL编译器来设计电路的效果。ELSE语句以及VHDL $>$ 和 $+$ 运算符。执行以下步骤：

1. 为您的VHDL代码创建一个新的Quartus II项目。使用与第V部分相同的开关，指示灯和显示屏。编译电路。
2. 使用Quartus II RTL Viewer工具检查通过编译VHDL代码生成的电路。将电路与您在第V部分中设计的电路进行比较。
3. 将电路下载到DE2板上，并通过尝试不同的数字 $A_1 A_0$ 和 $B_1 B_0$ 进行测试。

Part VII

设计一个组合电路，将6位二进制数转换为以BCD格式表示的2位十进制数。使用开关SW_{5:0}输入二进制数，7段显示HEX1和HEX0以显示十进制数。在DE2板上实现电路并演示其功能。

Copyright ©2006 Altera Corporation.