

Laboratório 2

Números e Displays

Este é um exercício para projetos de circuitos combinatórios que podem executar conversão de números binários em decimal e adição em números BCD (*binary-coded-decimal*).

Parte I

Desejamos mostrar no display de 7 segmentos *HEX3* até *HEX0* os valores das chaves SW_{15-0} . Mostre os valores representados pelas chaves SW_{15-12} , SW_{11-8} , SW_{7-4} e SW_{3-0} nos displays *HEX3*, *HEX2*, *HEX1* e *HEX0*, respectivamente. Seu circuito deve ser capaz de mostrar os dígitos de 0 a 9, e deve tratar as entradas de 1010 a 1111 como estados de *don't-care*.

1. Crie um novo projeto Quartus II para este circuito.
2. Escreva um arquivo VHDL que execute a funcionalidade necessária.
3. Compile e teste o circuito.

Parte II

Você deve projetar um circuito que converte um número binário de quatro bits $V = v_3v_2v_1v_0$ em seu decimal equivalente $D = d_1d_0$. A Tabela 1 mostra a saída necessária. Um projeto parcial deste circuito é mostrado na Figura 1. Ele inclui um comparador que checa quando o valor de V é maior que 9, e usa a saída deste comparador para controlar os displays de 7 segmentos. Você deve completar o projeto deste circuito criando uma entidade em VHDL que inclua o comparador, multiplexadores e o circuito *A* (não inclua o circuito *B* ou o decodificador de 7 segmentos neste ponto). Sua entidade VHDL deve ter uma entrada de 4 bits V , uma saída de 4 bits M e a saída z . O objetivo deste exercício é usar simples atribuições em VHDL para especificar a função lógica utilizando expressões Booleanas. Assim sendo, seu código VHDL não deve incluir nenhuma cláusula de IF-ELSE, CASE, ou expressões similares.

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

Tabela 1. Valores de Conversão Binária para Decimal.

Execute os seguintes passos:

1. Crie um novo projeto Quartus II para esta entidade.
2. Compile o circuito e use simulação funcional para verificar a operação correta do comparador, multiplexadores e do circuito *A*.

3. Aumente seu código VHDL para incluir o circuito *B* da Figura 1 bem como o decodificador de 7 segmentos. Use como entrada e saída do seu código as chaves SW_{3-0} para representar o número V , e os displays *HEX1* e *HEX0* para mostrar os dígitos decimais d_1 e d_0 .
4. Compile e teste o circuito.

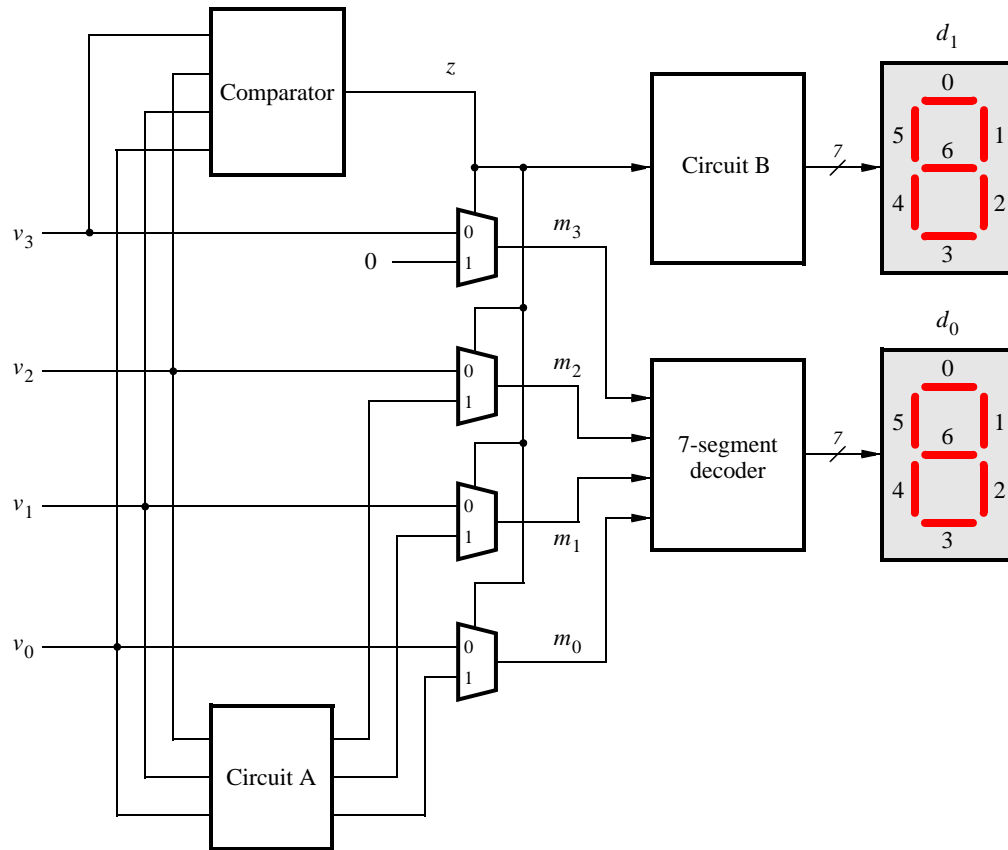


Figura 1: Projeto parcial do circuito de conversão binário para decimal.

Parte III

A Figura 2a mostra um circuito para um *somador completo*, que possui as entradas a , b , e c_i , e produz as saídas s e c_o . As partes *b* e *c* da figura mostram o símbolo do circuito e a sua tabela verdade, que produz a soma binária de 2 bits. A Figura 2d mostra como quatro instâncias deste somador completo podem ser usadas para projetar um circuito que soma dois números de 4 bits. Escreva código VHDL que implemente este circuito como descrito abaixo:

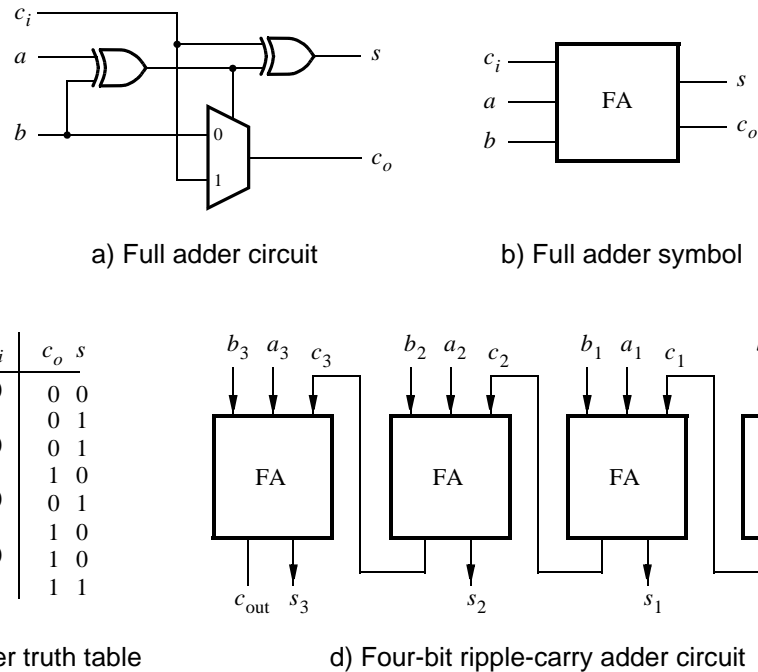


Figura 2: Um circuito somador.

1. Crie um novo projeto Quartus II para esta entidade.
2. Use as chaves SW_{7-4} e SW_{3-0} para representar as entradas A e B , respectivamente. Use SW_8 para o carry-in c_{in} do somador. Connecte as chaves SW nos seus LEDRs correspondentes, e a saída do somador c_{out} e S nos leds verdes.
3. Compile e teste o circuito.

Parte IV

Na parte II discutimos a conversão de números binários em dígitos decimais. Em alguns casos, é útil construir circuitos que usem este método de representação de números decimais, em que cada dígito decimal é representado por 4 bits. Este esquema é conhecido por representação BCD. Por exemplo, o valor decimal 59 é codificado em BCD como 0101 1001.

Você deve projetar um circuito que some dois dígitos em BCD. As entradas do circuito são números BCD A e B , além de um carry-in, c_{in} . A saída deve ser um número BCD de dois dígitos S_1S_0 . Observe que a maior soma que deve ser manipulada por este circuito é $S_1S_0 = 9 + 9 + 1 = 19$. Execute os seguintes passos:

1. Crie um novo projeto Quartus II para esta entidade. Você deve usar o circuito somador de quatro bits da Parte III para produzir a soma de quatro bits e o carry-out para a operação. Um circuito que converte este resultado de cinco bits, que possui valor máximo 19, nos dígitos BCD S_1S_0 pode ser projetado de forma similar ao conversor binário para decimal da parte II. Escreva seu código VHDL usando apenas expressões simples para especificar a função lógica – não use outros tipos de expressões como IF-ELSE ou CASE nesta parte do exercício.
2. Use as chaves SW_{7-4} e SW_{3-0} para as entradas A e B , respectivamente, e use SW_8 para o carry-in. Connecte as chaves SW nos leds vermelhos LEDR correspondentes, e conecte a soma de quatro bits e o

carry-out produzidos pela operação $A + B$ nos leds verdes $LEDG$. Mostre os valores BCD de A e B nos displays de 7 segmentos $HEX6$ e $HEX4$, e mostre o resultado S_1S_0 em $HEX1$ e $HEX0$.

3. Como seu circuito opera apenas com dígitos BCD, observe os casos quando a entrada A ou B for maior que nove. Se isto ocorrer, indique um erro ligando o led verde $LEDG_8$.
4. Compile e teste o circuito.

Parte V

Projete um circuito que efetue a soma de dois números de 2 dígitos em BCD, A_1A_0 e B_1B_0 para produzir a soma em BCD de 3 dígitos $S_2S_1S_0$. Use duas instâncias do seu circuito somador da Parte VI para construir este somador. Execute os passos a seguir:

1. Crie um novo projeto Quartus II para esta entidade.
2. Use as chaves SW_{15-8} e SW_{7-0} para representar os números BCD de 2 dígitos A_1A_0 e B_1B_0 , respectivamente. O valor de A_1A_0 deve ser mostrado no display de 7 segmentos $HEX7$ e $HEX6$, enquanto B_1B_0 deve estar em $HEX5$ e $HEX4$. Mostre a soma em BCD, $S_2S_1S_0$, nos displays de 7 segmentos $HEX2$, $HEX1$ e $HEX0$.
3. Compile e teste o circuito.