



2ª ATIVIDADE DE LABORATÓRIO

- 1) Considerando a tabela-verdade fornecida (Tabela 1), determine:
- a expressão booleana na forma de soma-de-produtos que caracteriza o circuito lógico;
 - a expressão booleana na forma de produto-de-somas que caracteriza o circuito lógico;
 - implemente os circuitos correspondentes às expressões obtidas nos Itens a) e b); e
 - verifique, usando o *software* LogiSim Evolution, a equivalência entre os circuitos.

Tabela 1.

| A | B | C | x |
|---|---|---|---|
| 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 |
| 0 | 1 | 0 | 1 |
| 0 | 1 | 1 | 0 |
| 1 | 0 | 0 | 1 |
| 1 | 0 | 1 | 1 |
| 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 0 |

- 2) Considere que um dado circuito lógico pode ser descrito por

$$x = ABCD + (A + \overline{B} + \overline{C} + \overline{D}) + A(B + \overline{C})D.$$

A partir disso,

- simplifique (se possível) a expressão usando apenas teoremas da álgebra booleana;
- implemente, usando o *software* LogiSim Evolution, o circuito correspondente; e
- demonstre, por simulação, a equivalência com o comportamento do circuito original.

- 3) Considere que se deseja projetar um multiplicador que recebe como entrada dois números binários de 2 bits cada, i.e., $x = x_1 x_0$ e $y = y_1 y_0$. Então, baseado em $x = x_1 x_0$ e $y = y_1 y_0$, o multiplicador produz como saída $z = z_3 z_2 z_1 z_0$. Nesse contexto,

- elabore a tabela-verdade que descreve o funcionamento lógico do circuito;
- obtenha expressões simplificadas, usando o Mapa de Karnaugh, para z_3 , z_2 , z_1 e z_0 ;
- implemente o circuito lógico (simplificado) usando o *software* LogiSim Evolution; e
- verifique, por simulação, se o circuito implementado tem o comportamento desejado. Vale lembrar que as condições de irrelevância (*don't care*) podem ser usadas para tratar resultados de saída não alcançáveis.

- 4) Projete um conversor binário-BCD que recebe como entrada um número binário de 4 bits, i.e., $x = x_3 x_2 x_1 x_0$, e produz como saída o equivalente número BCD de 5 bits, i.e., $y = y_4 y_3 y_2 y_1 y_0$. Nesse contexto,

- elabore a tabela-verdade que descreve o funcionamento lógico do circuito;
- obtenha expressões simplificadas, usando o método do Mapa de Karnaugh, para as saídas y_4 , y_3 , y_2 , y_1 e y_0 ;



c) implemente o circuito lógico (simplificado) usando o *software* LogiSim Evolution; e
d) verifique, por simulação, se o circuito implementado tem o comportamento desejado.
O bloco “*BFH mega funções/BCD para sete segmentos*” pode ser usado em conjunto com o elemento “*Entrada-Saída/Display de 7-segmentos*” para apresentar o resultado.

5) Circuitos geradores e verificadores de paridade são usualmente encontrados em sistemas de comunicação. Esses sistemas visam possibilitar a detecção de erros que possam ter ocorrido durante a transmissão da informação entre o transmissor e o receptor. Tais erros podem ocorrer devido à presença de ruído ou interferência em um dado barramento de comunicação. Diante disso, considere que se deseja projetar um circuito verificador de paridade “ímpar”, usando “portas” XOR e XNOR, o qual deve acusar a detecção de erro em sua saída quando cabível. Nesse projeto, assume-se que o barramento tem 8 linhas de comunicação, sendo 7 linhas (7 bits) de dados mais 1 linha (1 bit) de paridade. O bit de paridade deve ser gerado visando fazer com que o conjunto de 8 bits (dados mais paridade) exiba paridade ímpar, i.e., o número de ‘1s’ no conjunto de 8 bits deve ser ímpar. Ao concluir a implementação, verifique o comportamento do sistema para os conjuntos de bits a) 00011010; b) 11000001; c) 01111110 e d) 10000000; e, discorra sobre os resultados observados, indicando se o circuito detecta corretamente a presença ou ausência de erro. Vale comentar que um *led* pode ser inserido na saída do sistema a fim de indicar a detecção (nível alto) ou não de erro (nível baixo).