



Instituto Tecnológico de Aeronáutica - ITA

Divisão de Engenharia Eletrônica

Departamento de Eletrônica Aplicada

Laboratório de EEA-21

3ª Experiência

Projeto e montagem de Redes Iterativas

1. Objetivos

- a. Familiarização com a implementação de circuitos digitais em *protoboard*;
- b. Familiarização com as operações aritméticas básicas;
- c. Familiarização com as operações relacionais; e
- d. Familiarização com síntese de redes iterativas.

2. Instruções gerais

Para cada um dos tópicos abaixo, **observe** o que se pede em negrito. Quando a solicitação é da forma:

- “**Projete**”, espera-se que os alunos projetem os circuitos seguindo algum tipo de especificação **dada**. Assim, os cálculos combinacionais, procedimentos de minimização ou outros mecanismos utilizados para o projeto e o diagrama esquemático final do circuito devem ser mostrados (constar do relatório).
- “**Monte**”, espera-se que o circuito dado ou projetado seja montado em *protoboard* de maneira organizada, com as entradas e saídas identificadas. Os **componentes** serão fornecidos pelo almoxarifado, devendo os mesmos ser retornados após a avaliação dos circuitos pelo instrutor.
- “**Simule**”, espera-se que seja utilizado um software de captura esquemática para a obtenção dos resultados. Assim, o diagrama esquemático (no caso de captura esquemática) deve ser apresentado, bem como o diagrama de temporização contendo as entradas e as saídas.
- “**Analise**”, espera-se que sejam obtidas as expressões lógicas, tabelas verdade, a partir de um diagrama esquemático, diagrama de temporização ou outra informação sobre o circuito lógico. Dessa forma, os procedimentos de análise devem ser mostrados no relatório.

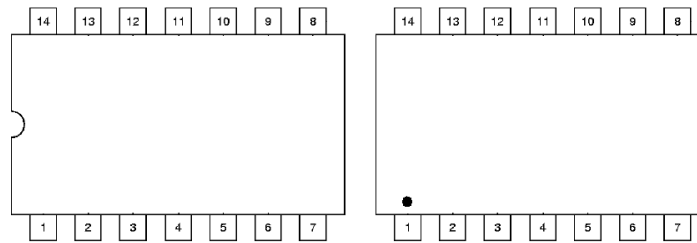
Um breve comentário sobre os resultados, observações e dificuldades é esperado para todas as tarefas.

3. Informações úteis

3.1 Circuitos integrados TTL

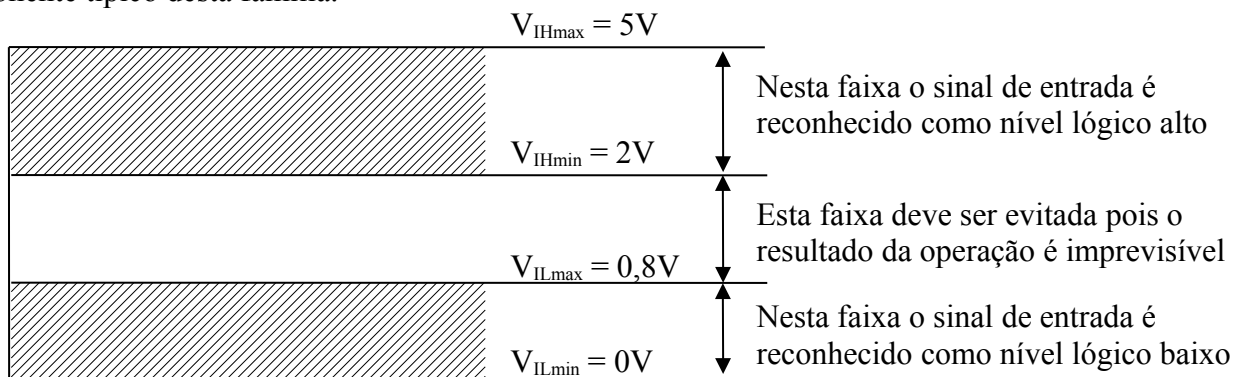
Os circuitos integrados que serão utilizados nesta prática pertencem à família TTL (*Transistor-Transistor Logic*). A **fonte de alimentação** a ser utilizada para o correto funcionamento destes dispositivos é de **5V**. É fundamental ter muito cuidado na hora de ligar os pinos de VCC (5V) e GND (terra ou 0V). **Se os pinos de alimentação forem invertidos haverá a perda do componente**. Após efetuar a montagem do circuito, verifique se as ligações estão corretas antes de alimentar o circuito. Utilize o **multímetro**, verificando a continuidade do ponto de alimentação e terra para os pinos correspondentes do CI.

Atente para a correta identificação dos pinos do componente. As formas mais comuns para enumeração dos pinos é mostrada na figura a seguir. Se houver um ponto no encapsulamento, este indica o Pino número 1. Se houver um chanfro, o pino número 1 é o primeiro abaixo do chanfro, como se vê na disposição da figura abaixo. A contagem dos pinos é feita a partir do número 1, no sentido anti-horário.

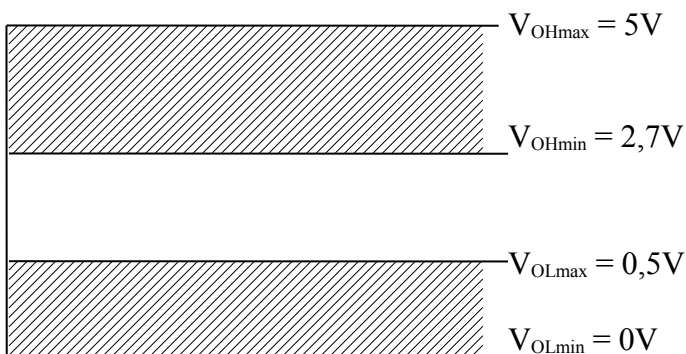


3.2 Níveis de tensão na entrada e saída da família TTL

Para a família TTL, existe uma certa faixa de **tensão na entrada** de uma porta que é tipicamente reconhecida como nível lógico baixo, e uma certa faixa de tensão na entrada da porta que é tipicamente reconhecida como nível lógico alto. A faixa aceita como nível baixo é o intervalo $[V_{ILmin}, V_{ILmax}]$, e a faixa aceita como nível alto é o intervalo $[V_{IHmin}, V_{IHmax}]$. Os valores exatos destes parâmetros devem ser consultados no *datasheet* específico de cada componente, mas abaixo é mostrado um exemplo para um componente típico desta família.



A **tensão na saída** também está sujeita a variar dentro de uma faixa. Quando o nível lógico da saída é baixo, a tensão estará entre $[V_{OLmin}, V_{OLmax}]$. Quando o nível lógico da saída for alto, a tensão estará entre $[V_{OHmin}, V_{OHmax}]$. A figura a seguir ilustra um exemplo típico para a família TTL (valores específicos devem ser consultados em *datasheet*).



Observe que é necessário que a faixa $[V_{OLmin}, V_{OLmax}]$ esteja dentro da faixa $[V_{ILmin}, V_{ILmax}]$, e que a faixa $[V_{OHmin}, V_{OHmax}]$ esteja dentro da faixa $[V_{IHmin}, V_{IHmax}]$ para o correto funcionamento da família.

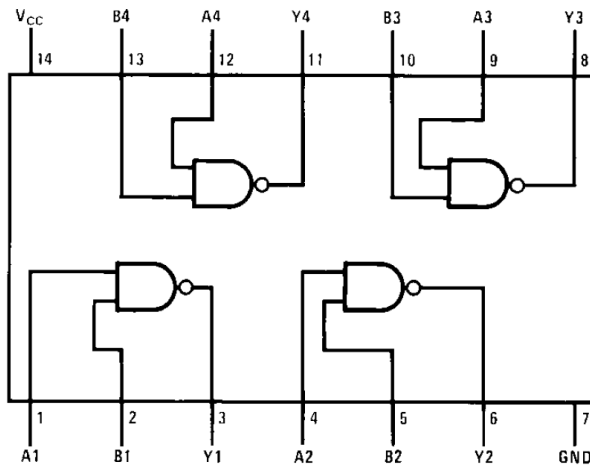
3.3 5CIs: NANDs, NOTs, ANDs, ORs e XORs

Circuitos integrados da família TTL 7400 que serão utilizados nesta prática:

NOTA1: Sobre tecnologias → 74LS00 – CI 7400, com tecnologia Low Shotcky;
74HC00 - CI 7400, com tecnologia High Speed CMOS.

- CI 7400 4 NANDs

Connection Diagram



Function Table

$$Y = \overline{AB}$$

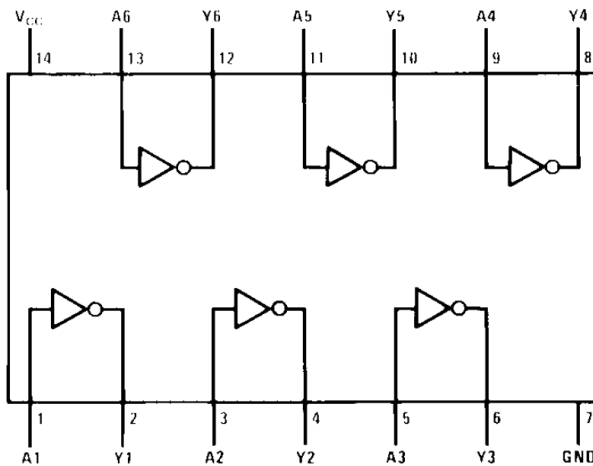
Inputs		Output
A	B	Y
L	L	H
L	H	H
H	L	H
H	H	L

H = HIGH Logic Level

L = LOW Logic Level

- CI 7404 6 NOTs

Connection Diagram



Function Table

$$Y = \overline{A}$$

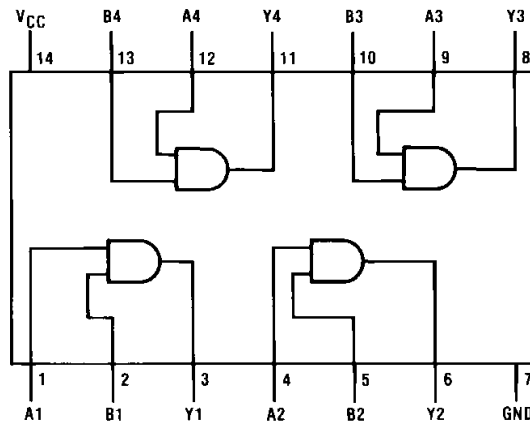
Inputs	Output
A	Y
L	H
H	L

H = HIGH Logic Level

L = LOW Logic Level

- CI 7408 4 ANDs

Connection Diagram



Function Table

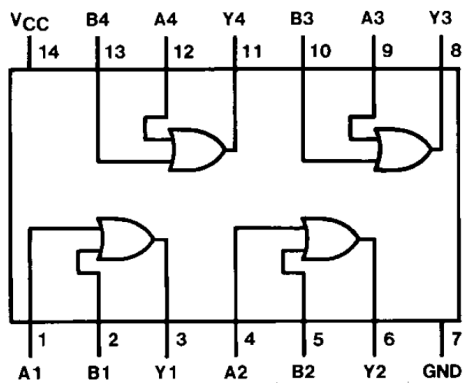
$$Y = AB$$

Inputs		Output
A	B	Y
L	L	L
L	H	L
H	L	L
H	H	H

H = High Logic Level
L = Low Logic Level

- CI 7432 4 ORs

Connection Diagram



Function Table

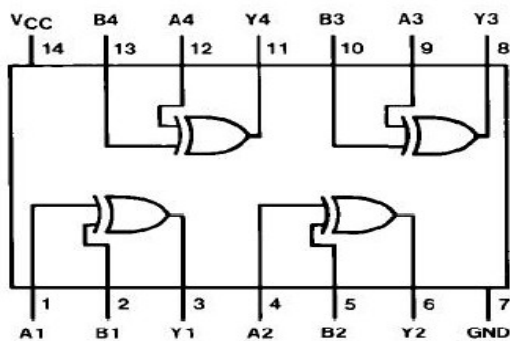
$$Y = A + B$$

Inputs		Output
A	B	Y
L	L	L
L	H	H
H	L	H
H	H	H

H = HIGH Logic Level
L = LOW Logic Level

- CI 7486 4 XORs

Connection Diagram 7486



Function Table

$$Y = A \oplus B$$

Inputs		Output
A	B	Y
L	L	L
L	H	H
H	L	H
H	H	L

H = HIGH Logic Level
L = LOW Logic Level

Não deixe de consultar os *datasheets* para mais informações.

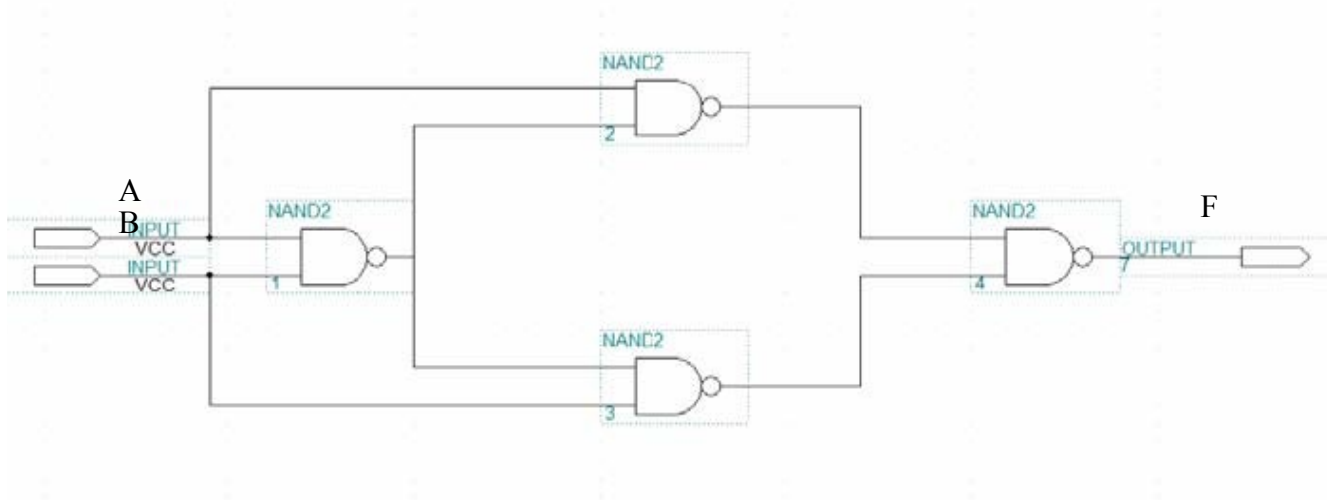
4. Material necessário para as montagens

3 CIs 7400; 1 CI 7404, 1 CI 7408, 1 CI 7432 e 1 CI 7486; fios para *proto-board*; fonte de +5V; e *proto-board*.

5. Montagens

5.1 Tarefa 5.1: Montagem de um circuito combinacional com portas NAND2

Monte a função combinacional abaixo esquematizada que contém apenas portas *NAND* de 2 entradas. Levante experimentalmente qual função básica está sendo implementada.

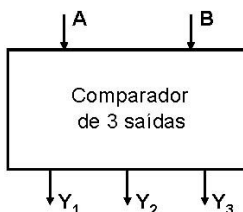


Obtenha a tabela verdade e a expressão de F em função de A e B.

Dica: não desmonte este circuito até finalizar a tarefa 5.3.

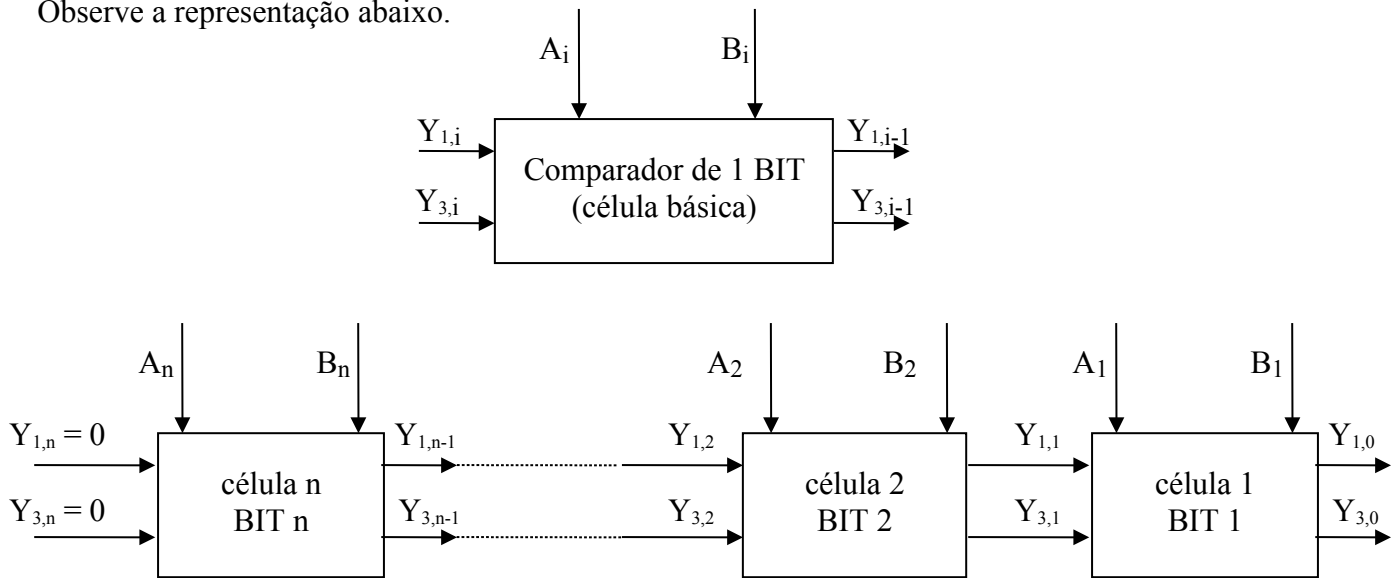
5.2 Tarefa 5.2: Projeto e montagem da célula básica de um comparador

Projete e Monte a célula básica de um comparador de duas palavras de n bits que utiliza a técnica de redes iterativas, tal que:



$Y_1 = 1$	se $A > B$,	0 caso contrário
$Y_2 = 1$	se $A = B$,	0 caso contrário
$Y_3 = 1$	se $A < B$,	0 caso contrário

Observe a representação abaixo.



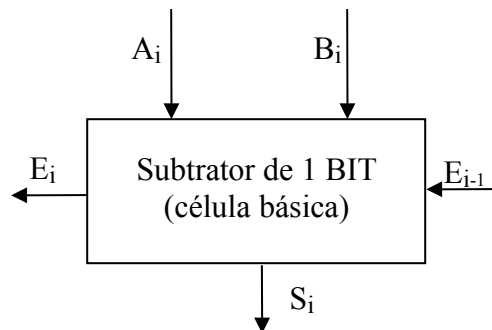
Procedimento:

- Elabore a tabela verdade da célula básica;
- Obtenha as funções lógicas da célula básica;
- Implemente em *proto-board* a célula básica;
- Demonstre o circuito funcionando e apresente os resultados em forma de tabela verdade; e
- Mostre como obter Y_1 , Y_2 e Y_3 a partir das variáveis mostradas na rede iterativa do comparador de n bits.

5.3 Tarefa 5.3: Projeto e montagem de um subtrator completo de 1 bit

Projete e Monte um subtrator completo de 1 bit do tipo propagação do empresta-um. Este subtrator pode ser usado como célula básica para projeto de subtratores de N bits.

Observe a representação abaixo.



Procedimento:

- Elabore a tabela verdade da célula básica;
- Obtenha as funções lógicas da célula básica (S_i e E_i);
- Minimize simultaneamente as expressões obtidas em (b), colocando-as num formato que utilize apenas a operação NAND com duas variáveis;
- Usando apenas portas **NAND de 2 entradas**, implemente em *proto-board* a célula básica;

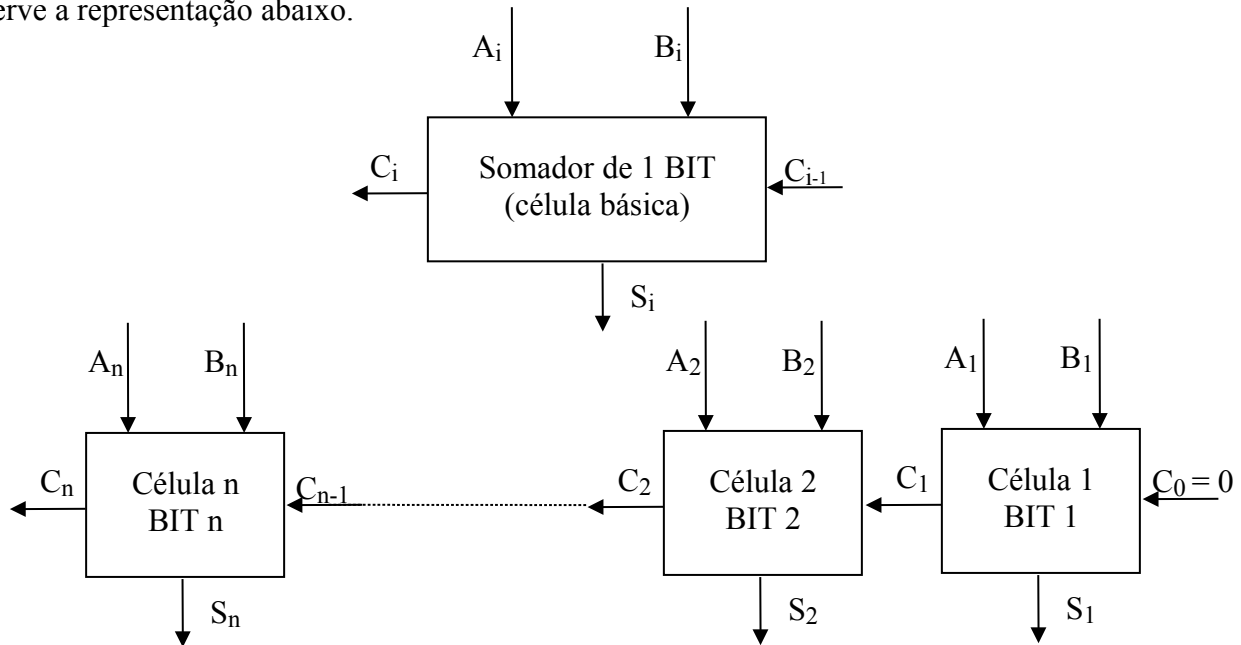
e. Demonstre o circuito funcionando e apresente os resultados em forma de tabela verdade, comparando o resultado com o esperado.

Dica: não desmonte este circuito até finalizar a tarefa 5.4.

5.4 Tarefa 5.4: Projeto e montagem de um somador completo de 1 bit

Projete e Monte um somador completo de 1 bit do tipo propagação do vai-um (carry). Este somador pode ser usado como célula básica para projeto de somadores de N bits.

Observe a representação abaixo.



Procedimento:

- Elabore a tabela verdade da célula básica;
- Obtenha as funções lógicas da célula básica (S_i e C_i);
- Minimize simultaneamente as expressões obtidas em (b), colocando-as num formato que utilize apenas a operação NAND com duas variáveis;
- Usando portas **NAND de 2 entradas** e **portas XOR**, implemente em *protoboard* a célula básica;
- Demonstre o circuito funcionando e apresente os resultados em forma de tabela verdade e comparando com o resultado esperado.

6. Simulação:

6.1 Projeto e simulação de um somador/subtrator de um dígito

Projete e simule um somador/subtrator paralelo decimal de um dígito. Os operandos e o resultante da soma (ou subtração) estão representados em BCD (4 bits). Os números negativos deverão ser representados em complemento de 2. Quando o resultado for um número de dois dígitos, apenas o dígito da unidade será apresentado, estando em complemento de 2 no caso de o resultado ser negativo.

Comentário sobre operações com números representados em BCD:

As operações aritméticas onde os operandos estão representados em BCD precisam ser corrigidas sempre que os resultados caírem fora de certo intervalo (maior que 9 para e menor que -9). Com o objetivo de visualizar este fato observe os seguintes exemplos de operações (adição/subtração) em BCD.

Exemplo de operações que não precisam correção do resultado:

$$\text{Ex1: } (2)_d + (5)_d = (7)_d$$

$$(0010)_b + (0101)_b = (0111)_b \Rightarrow (7)_d$$

$$\text{Ex2: } (9)_d - (4)_d = (5)_d$$

$$(1001)_b - (0100)_b = (0101)_b \Rightarrow (5)_d$$

Soma de dois dígitos representados em BCD: Sempre que dois números positivos representados em BCD forem somados e o resultado for maior ou igual a 10_d o resultado precisa ser corrigido. A correção é feita somando 6_d (ou 0110_b) ao resultado.

$$\text{Ex3. } (7)_d + (8)_d = (15)_d$$

$$(0111)_b + (1000)_b = \underbrace{1111}_{\text{Repre. BCD errada}} \Rightarrow \begin{array}{r} 1111 \\ +0110 \\ \hline 1 \quad 0101 \\ \text{carry} \quad 5_d \end{array}$$

Subtração de dois dígitos representados em BCD: Quando um número negativo é representado em complemento de 2 tem-se um caso crítico na subtração se o minuendo é negativo e o subtraendo positivo. Neste caso o resultado pode ser menor que -10_d e o resultado deve ser corrigido somando -6_d . Observe que -6_d equivale a 10_d em complemento de 2.

$$\text{Ex4. } (-6)_d - (7)_d = -(13)_d$$

$$\begin{array}{l} -6_d \xRightarrow{\text{Comp. 2}} (1010)_b \Rightarrow 1010 \\ -7_d \xRightarrow{\text{Comp. 2}} (1001)_b \Rightarrow +1001 \end{array} \Rightarrow \begin{array}{r} 1010 \\ +1001 \\ \hline 10011 \leftarrow \text{BCD errado} \end{array}$$

$$\begin{array}{r} 10011 \\ +1010 \leftarrow (-6 \text{ comp. 2}) \\ \hline 1101 \leftarrow (-3 \text{ comp. 2}) \end{array} \quad \text{Correção}$$

$$\text{Ex5 } (-2)_d - (3)_d = -(5)_d$$

$$\begin{array}{l} -2_d \xRightarrow{\text{Comp. 2}} (1110)_b \Rightarrow 1110 \\ -3_d \xRightarrow{\text{Comp. 2}} (1101)_b \Rightarrow +1101 \end{array} \Rightarrow \begin{array}{r} 1110 \\ +1101 \\ \hline 1011 \leftarrow (-5 \text{ comp. 2}) \end{array}$$

Restrições: com o objetivo de reduzir a complexidade do exercício considere que:

- a) o operando B é sempre positivo;
- b) o intervalo de variação do operando A é de $+7_d$ até -8_d .
- c) os resultados negativos estão representados em complemento de 2;

Observe que:

a) para o caso da soma: poderá ser necessária a correção do resultado BCD caso o operando A seja positivo. Estando o valor deste limitado a $+7_d$, é possível reconhecer esta situação pelo bit mais significativo deste operando (a_3). O circuito deverá gerar a correção se a comparação do resultado da soma com o valor 10_d exigir ou quando houver *carry* na soma.

b) para o caso da subtração: poderá ser necessária a correção do resultado BCD se o operando A for negativo. É possível reconhecer esta situação pelo bit mais significativo de A. O circuito deverá neste caso gerar a correção se a comparação do resultado da soma com o valor $(-10)_d$ exigir, ou se o *carry* for nulo.

Exemplos de resultados esperados:

A = 3	B = 6	E = 0 (soma)	Resultado: 1001 $\rightarrow (9)_d$
A = -7	B = 2	E = 0 (soma)	Resultado: 1011 $\rightarrow (-5)_d$
A = 6	B = 9	E = 0 (soma)	Resultado: 0101 $\rightarrow (5)_d$
A = 2	B = 9	E = 1 (subtração)	Resultado: 1001 $\rightarrow (-7)_d$
A = -5	B = 8	E = 1 (subtração)	Resultado: 1101 $\rightarrow (-3)_d$

Obtenha:

- a. O diagrama esquemático completo, mostrando os passos do desenvolvimento deste; no Quartus II, utilize o somador de 4 bits 74283;
- b. A simulação funcional (*Functional Simulation*) da realização das seguintes operações:

$$(4)_d + (5)_d$$

$$(7)_d + (4)_d$$

$$(7)_d + (9)_d$$

$$(-5)_d + (8)_d$$

$$(-8)_d + (4)_d$$

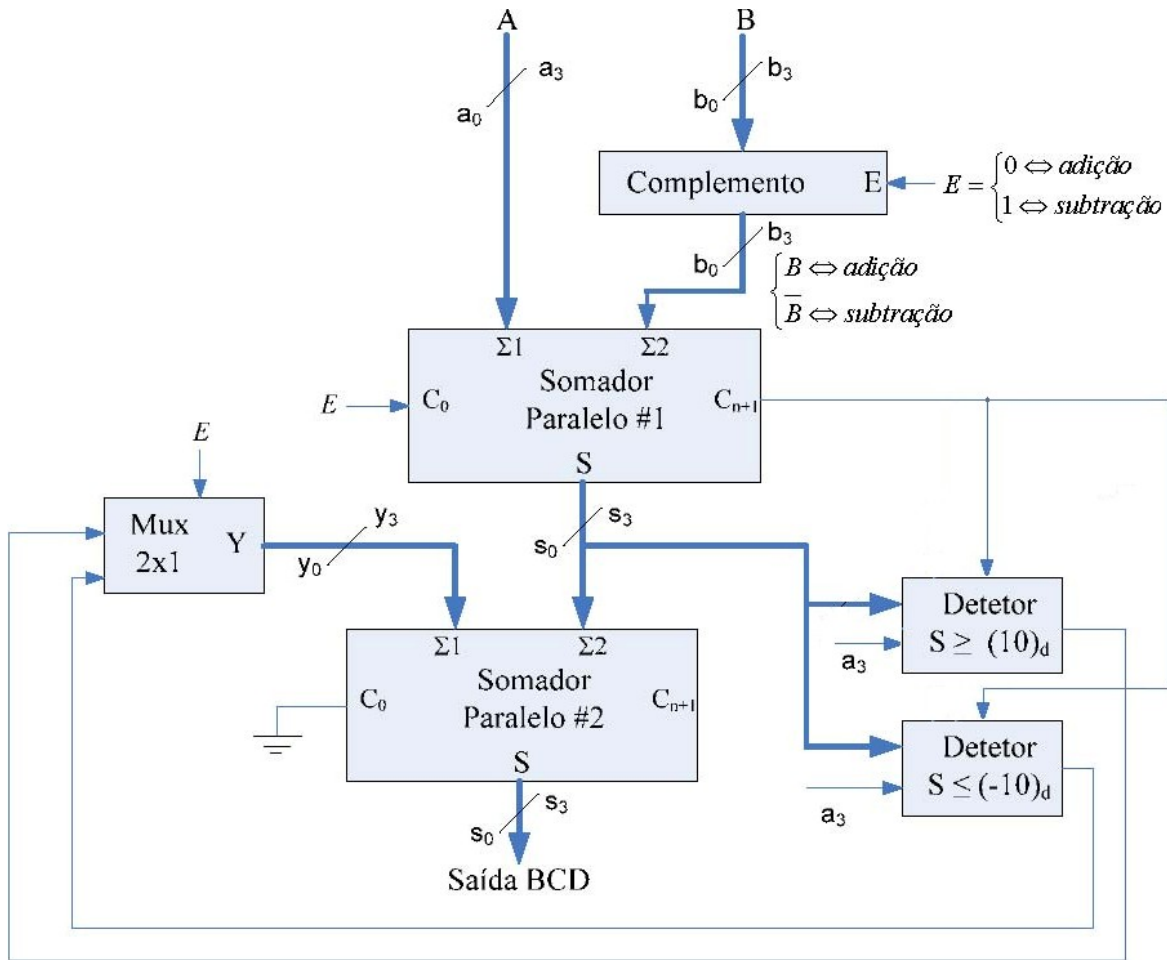
$$(1)_d - (7)_d$$

$$(-7)_d - (7)_d$$

$$(-8)_d - (9)_d$$

Mostre os resultados de forma clara para confirmar o funcionamento do circuito.

Um diagrama de blocos para o circuito é mostrado na figura a seguir, onde **E** é a entrada seletora que define se a operação é de soma ou subtração, s_3 é o bit mais significativo da saída do somador, a_3 é o bit mais significativo do operando A e C_0 é o *carry in*.



6.2 Projeto e simulação de um circuito multiplicador 2 n de 2 b

Projete e simule um circuito capaz de multiplicar 2 números de 2 bits, resultando num número de 4 bits.

Note que:

$$\begin{array}{r}
 A_1 A_0 \\
 \times B_1 B_0 \\
 \hline
 C_1 C_0 \\
 + D_1 D_0 \\
 \hline
 M_3 M_2 M_1 C_0
 \end{array}$$

Obtenha:

- O diagrama esquemático, apresentando os passos do desenvolvimento deste;
- O diagrama de temporização, resultante da simulação funcional do circuito, para as multiplicações: $(10)_b * (11)_b$, $(10)_b * (00)_b$, $(01)_b * (01)_b$ e $(11)_b * (11)_b$.