

[View on GitHub](#)

Circuitos Digitais

Repo of Digital Circuits course - CRT0384

PRÁTICA 05 - CIRCUITOS ARITMÉTICOS

[Voltar à home](#)

OBJETIVOS

- Verificar o funcionamento de um circuito somador e de um circuito comparador;
- Fixar conceitos de operações com números binários, *carry in*, *carry out* e *overflow*;

Material Necessário:

- Kit Digital;
- 01 TTL 74LS243;
- 01 TTL 74L85

Exercícios pré-laboratório:

1. Determine o significado dos conceitos de *carry in*, *carry out* e *overflow*, mostrando exemplos com números binários.

PARTE 1 - CIRCUITO SOMADOR

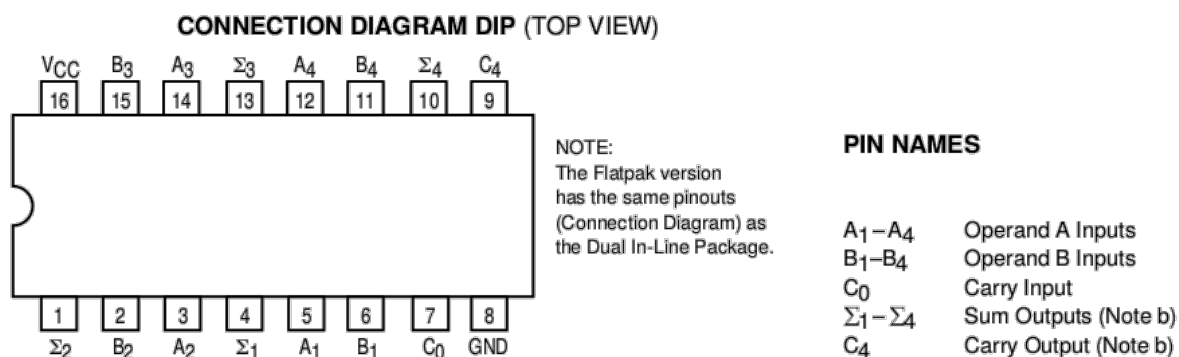
Um circuito combinacional aritmético implementa operações aritméticas como adição, subtração, multiplicação e divisão com números binários. Quando há uma soma binária em onde ambos os operandos são iguais a 1, são necessários dois dígitos para expressar seu resultado. Neste caso, o transporte (vai - um ou carry, em inglês) é somado ao próximo par mais significativo de bits. Um circuito combinacional que implementa a adição de dois bits é chamado meio-somador (half adder, em inglês). Um circuito que implementa a adição de três bits (dois bits significativos e um carry) é chamado de somador completo (full adder, em inglês). Estes nomes decorrem do fato de que com dois meio-somadores pode-se implementar um somador completo. O somador completo é um circuito aritmético básico a partir do qual todos os outros circuitos aritméticos são construídos.

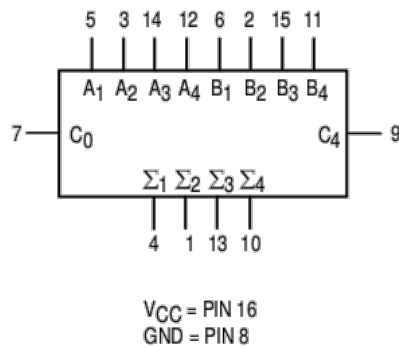
Nesta prática será utilizado o meio somador de 4 bits com *carry in* e *carry out* do circuito 74LS283.

Prática

Implemente um circuito somador de duas entradas de 4 bits $A_3 \sim A_2 \sim A_1 \sim A_0$ e $B_3 \sim B_2 \sim B_1 \sim B_0$ cuja saída seja uma palavra binária $\Sigma_3 \sim \Sigma_2 \sim \Sigma_1 \sim \Sigma_0$ e um bit de *carry out* C_1 . A entrada de carry in C_0 deve receber valor 0. Represente a saída binária nos leds do kit utilizando a sequência de exibição horizontal $C_1 \sim \Sigma_3 \sim \Sigma_2 \sim \Sigma_1 \sim \Sigma_0$. Escolha 16 pares de valores em 4 bits para as variáveis A e B e apresente uma tabela com a saída do somador.

DIAGRAMA 74LS283



LOGIC SYMBOL**Example:**

	C_0	A_1	A_2	A_3	A_4	B_1	B_2	B_3	B_4	Σ_1	Σ_2	Σ_3	Σ_4	C_4	
logic levels	L	L	H	L	H	H	L	L	H	H	H	L	L	H	
Active HIGH	0	0	1	0	1	1	0	0	1	1	1	0	0	1	(10+9=19)
Active LOW	1	1	0	1	0	0	1	1	0	0	0	1	1	0	(carry+5+6=12)

Interchanging inputs of equal weight does not affect the operation, thus C_0 , A_1 , B_1 , can be arbitrarily assigned to pins 7, 5 or 3.

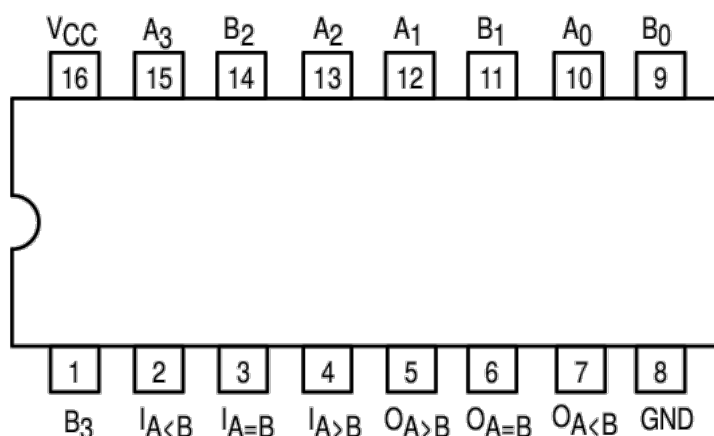
PARTE 2 - CIRCUITO COMPARADOR

Um comparador é um circuito combinatório que permite comparar o valor absoluto de dois inteiros A e B representados em binário com n-bits. Tem três saídas que indicam respectivamente se $A < B$, $A = B$ ou $A > B$.

O circuito comparador da prática é implementado pelo CI 74LS85, que dispõe de duas entradas de 4 bits, 3 pinos de entrada e 3 pinos de saída de comparação. Os pinos de entrada podem ser utilizados para utilização de múltiplos circuitos.

Implemente no Kit um circuito comparador utilizando o CI 74LS85 de duas entradas de 4 bits $A_3 \sim A_2 \sim A_1 \sim A_0$ e $B_3 \sim B_2 \sim B_1 \sim B_0$. Ligue as entradas $I \sim A = B$, $I \sim A > B$ e $I \sim A < B$ em três chaves ligadas com níveis lógicos HLL, respectivamente.

Escolha um conjunto de 16 pares de entrada e escreva a tabela de saída das portas $O \sim A = B$, $O \sim A > B$ e $O \sim A < B$.

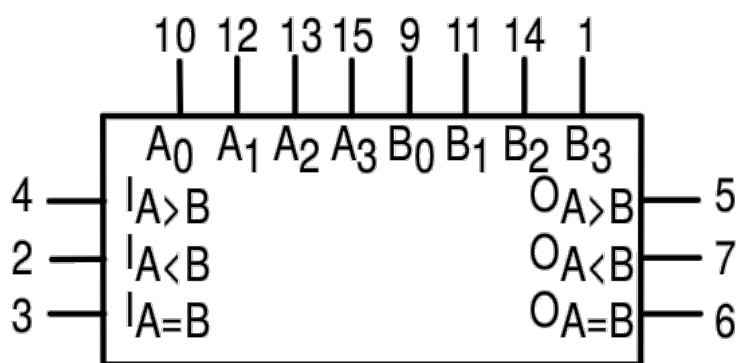
CONNECTION DIAGRAM DIP (TOP VIEW)

NOTE:
 The Flatpak version has the same pinouts (Connection Diagram) as the Dual In-Line Package.

PIN NAMES

A_0-A_3, B_0-B_3	Parallel Inputs
$I_{A=B}$	A = B Expander Inputs
$I_{A < B}, I_{A > B}$	A < B, A > B, Expander Inputs
$O_{A > B}$	A Greater Than B Output (Note b)
$O_{A < B}$	B Greater Than A Output (Note b)
$O_{A=B}$	A Equal to B Output (Note b)

LOGIC SYMBOL



$V_{CC} = \text{PIN } 16$

$GND = \text{PIN } 8$

TRUTH TABLE

COMPARING INPUTS				CASCADING INPUTS			OUTPUTS		
A_3, B_3	A_2, B_2	A_1, B_1	A_0, B_0	$I_{A>B}$	$I_{A<B}$	$I_{A=B}$	$O_{A>B}$	$O_{A<B}$	$O_{A=B}$
$A_3 > B_3$	X	X	X	X	X	X	H	L	L
$A_3 < B_3$	X	X	X	X	X	X	L	H	L
$A_3 = B_3$	$A_2 > B_2$	X	X	X	X	X	H	L	L
$A_3 = B_3$	$A_2 < B_2$	X	X	X	X	X	L	H	L
$A_3 = B_3$	$A_2 = B_2$	$A_1 > B_1$	X	X	X	X	H	L	L
$A_3 = B_3$	$A_2 = B_2$	$A_1 < B_1$	X	X	X	X	L	H	L
$A_3 = B_3$	$A_2 = B_2$	$A_1 = B_1$	$A_0 > B_0$	X	X	X	H	L	L
$A_3 = B_3$	$A_2 = B_2$	$A_1 = B_1$	$A_0 < B_0$	X	X	X	L	H	L
$A_3 = B_3$	$A_2 = B_2$	$A_1 = B_1$	$A_0 = B_0$	H	L	L	H	L	L
$A_3 = B_3$	$A_2 = B_2$	$A_1 = B_1$	$A_0 = B_0$	L	H	L	L	H	L
$A_3 = B_3$	$A_2 = B_2$	$A_1 = B_1$	$A_0 = B_0$	X	X	H	L	L	H
$A_3 = B_3$	$A_2 = B_2$	$A_1 = B_1$	$A_0 = B_0$	H	H	L	L	L	L
$A_3 = B_3$	$A_2 = B_2$	$A_1 = B_1$	$A_0 = B_0$	L	L	L	H	H	L

H = HIGH Level
L = LOW Level
X = IMMATERIAL

PÓS LABORATÓRIO - RELATÓRIO

1. Descreva os procedimentos de montagem realizados na prática. Comente

o significado dos pinos $C_{0\sim}$ e $C_{1\sim}$ do circuito somador e sobre as entradas $I_{A=B\sim}$, $I_{A>B\sim}$ e $I_{A<B\sim}$ e saídas $O_{A=B\sim}$, $O_{A>B\sim}$ e $O_{A<B\sim}$ do circuito comparador;

2. Elabore um circuito portas lógicas básicas (AND, OR e NOT)

comparador de dois bits que receba duas entradas lógicas de dois bits $A_1 \sim A_0$ e $B_1 \sim B_0$ e os três bits de entrada $I \sim A=B$, $I \sim A>B$ e $I \sim A<B$ e retorne o resultado da comparação na forma $O \sim A=B$, $O \sim A>B$ e $O \sim A<B$. Representar a tabela verdade do circuito.

3. Construa um circuito que receba uma entrada de dois bits $A_1 \sim A_0$ e

um bit de sinal S_0 e retorne uma saída de três bits $B_2 \sim B_1 \sim B_0$ que equivale à entrada em formato complemento de dois. Se a entrada for positiva, é mantida na saída da forma $B_2 \sim B_1 \sim B_0 = 0 A_1 \sim A_0$ e, caso seja negativa, $B_2 \sim B_1 \sim B_0 = \text{comp2}(A_1 \sim A_0)$

4. Elabore um circuito utilizando portas lógicas básicas (AND, OR e

NOT) **somador em complemento de dois** que receba duas entradas lógicas de três bits $A_2 \sim A_1 \sim A_0$ e $B_2 \sim B_1 \sim B_0$ retorne a soma em de A e B na forma $\Sigma_2 \sim \Sigma_1 \sim \Sigma_0$ e um bit de carry out C_1 e um bit de overflow O_1 . O circuito deve ser capaz de somar números positivos e negativos representados conforme representação a seguir:

-3 -2 -1 0 1 2 3 4

101 110 111 000 001 010 011 100

Representar a tabela verdade do circuito.

Circuitos Digitais maintained by [marcielbp](https://marcielbp.github.io)

Published with [GitHub Pages](https://github.com/marcielbp/Circuits)