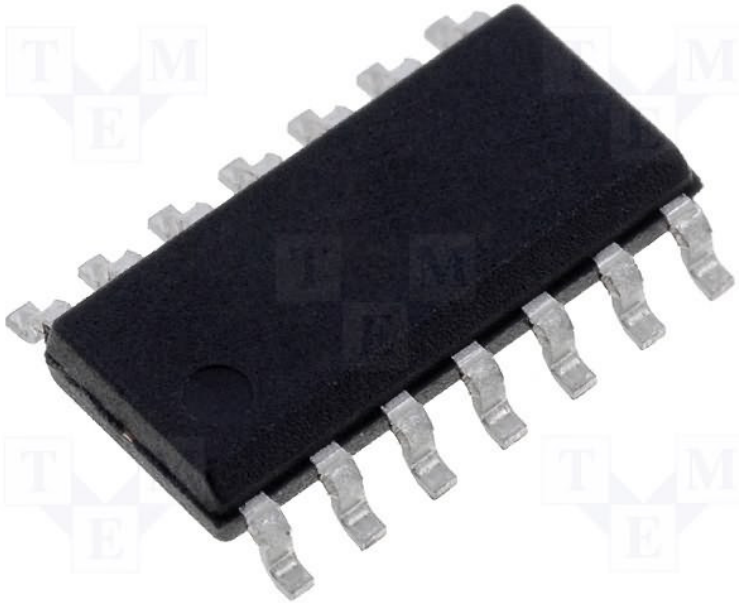
Comparadores



Índice

Descripción

- La operación lógica de comparación es esencial en todo el cálculo digital. Aparece en funciones de búsqueda, identificación selectiva de área de datos, frecuente en editores, compiladores, etc.

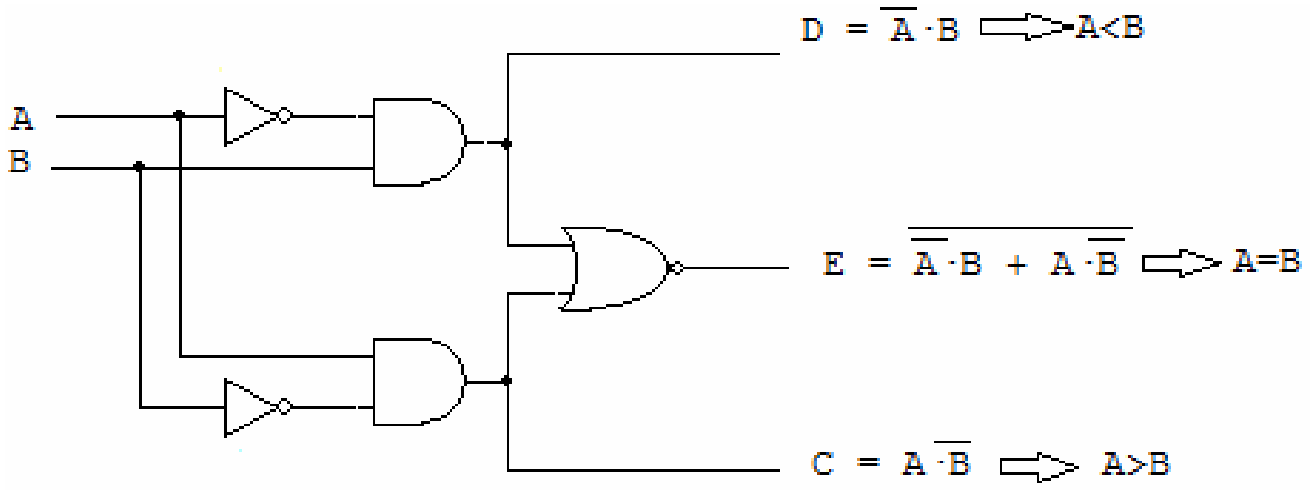
- Un comparador de dos palabras de N bits es un circuito que determina cual de estas palabras es mayor, cual es menor y cuando son iguales. Es decir, produce tres salidas:

A>B A=B A<B

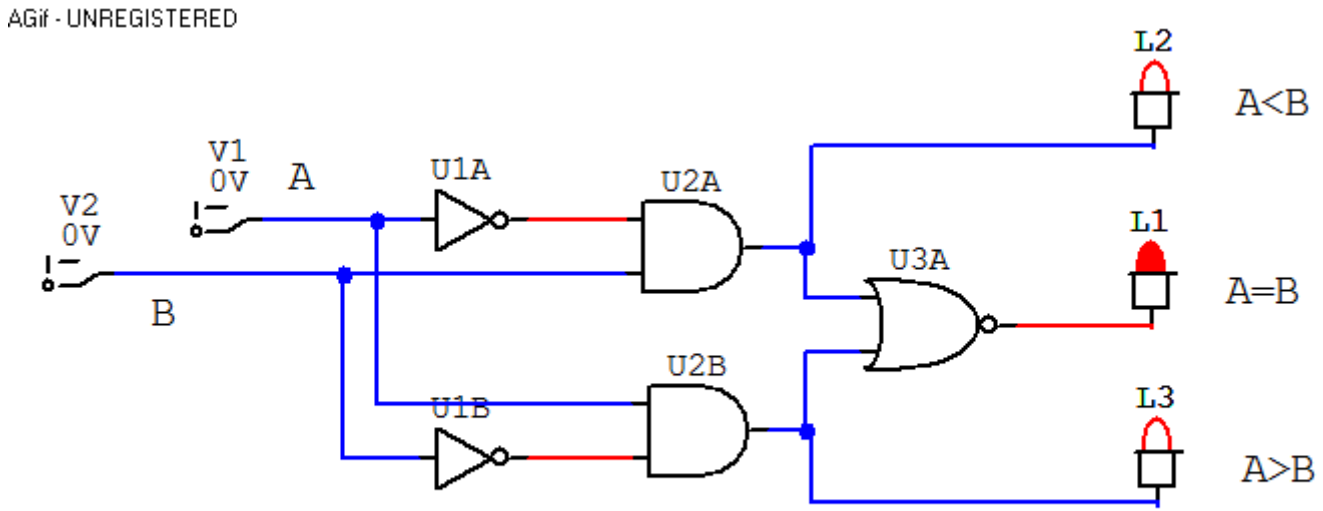
Comparador de 1 bit

Tabla de verdad y circuito lógico

A	B	A > B	A = B	A < B
0	0	0	1	0
0	1	0	0	1
1	0	1	0	0
1	1	0	1	0



Simulación comparador de 1 bit



Comparador de N bits

- Ampliamos la comparación anterior a palabras de n bits:

  Para cumplir la expresión A=B, tendremos que todos los bits de ambas palabras son iguales.

  Ejemplo: A=101111 → A=47

  B=101111 → B=47

  Para A>B, iremos recorriendo los bits de más a menos significativo hasta encontrar una diferencia que cumpla dicha expresión (bit de A mayor que bit de B).

  Ejemplo: A=110111 → A=55

  B=100111 → B=39

Comparador de N bits

Para A<B , iremos recorriendo los bits de más a menos significativo hasta encontrar una diferencia que cumpla dicha expresión (bit de A menor que bit de B).

Ejemplo: A=101111 → A=47

B=110111 → B=55

Tenemos en cuenta en todas estas expresiones que son números sin signo.

Comparadores de N bits

- A través de la lógica y el álgebra de Boole, se define el razonamiento que lleva a la formulación de un comparador de n bits.
- A = B si y solo si

  E(An-1,Bn-1)*E(An-2,Bn-2)*...*E(A1,B1)*E(A0, B0) = 1

- A > B si y solo si

  C(An-1,Bn-1) + E(An-1,Bn-1)*C(An-2,Bn-2) + E(An-1,Bn-1)*E(An-2,Bn-2)*C(An-3,Bn-3) +... + E(An-1,Bn-1)*E(An-2,Bn-2)*...*E(A1,B1)*C(A0,B0) = 1

- A < B si y solo si

  D(An-1,Bn-1) + E(An-1,Bn-1)*D(An-2,Bn-2) + E(An-1,Bn-1)*E(An-2,Bn-2)*D(An-3,Bn-3) +... + E(An-1,Bn-1)*E(An-2,Bn-2)*...*E(A1,B1)*D(A0,B0) = 1

Comparador comercial 74LS85

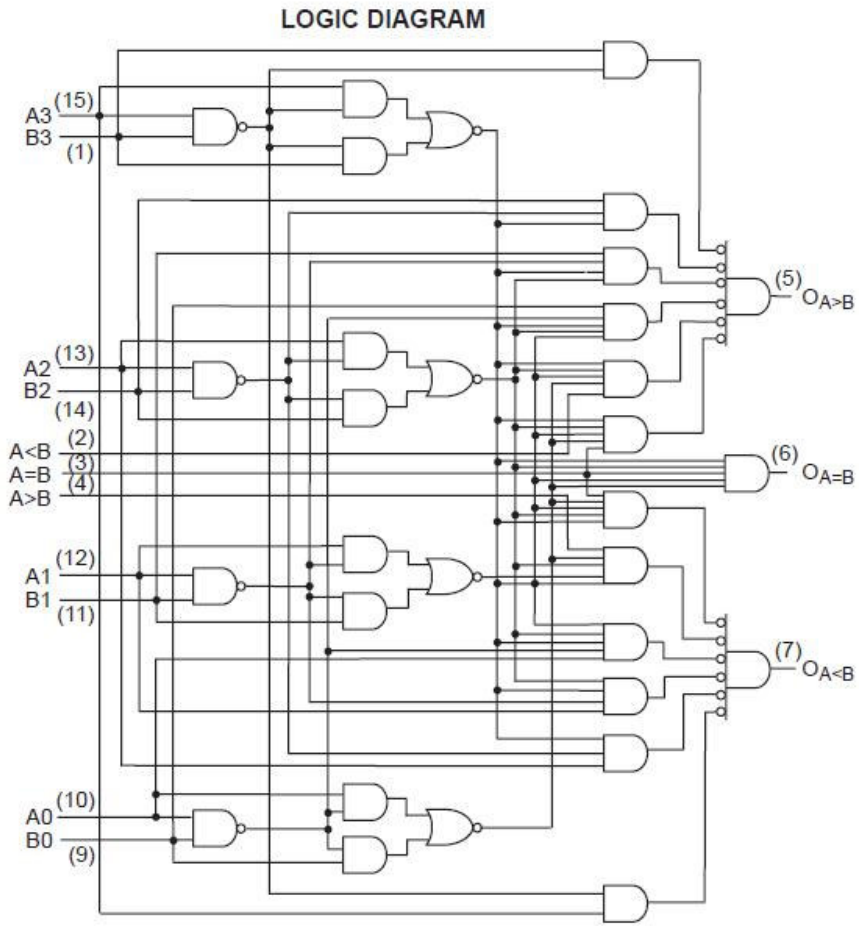
- El 74LS85 compara dos códigos binarios de 4 bits A y B aplicados en paralelo a las entradas $A_3A_2A_1A_0$ y $B_3B_2B_1B_0$ respectivamente, e indica en tres líneas de salida activas en alto sus magnitudes relativas. Es decir, cuándo A es mayor, menor o igual a B.

- Además cuenta con tres líneas de entrada adicionales que le permiten conectarse en cascada a unidades similares para comparar números de mayor longitud.

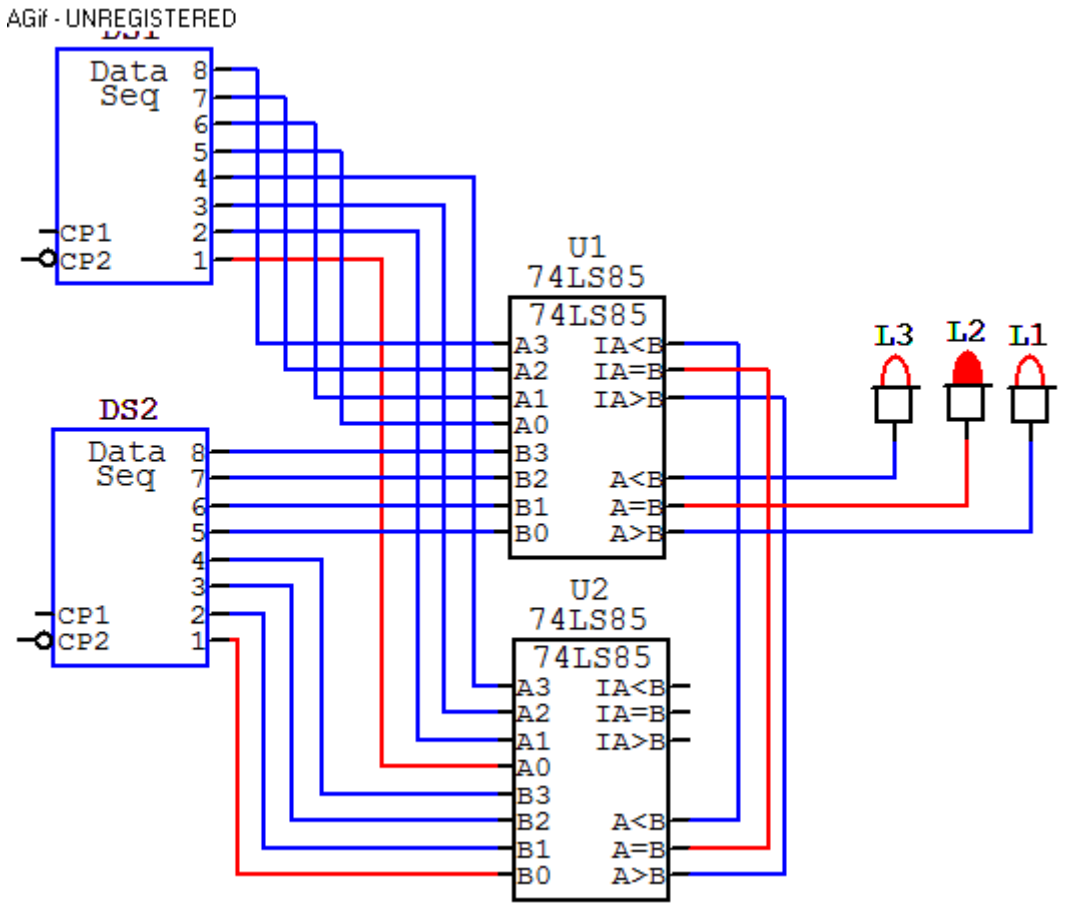
Tabla de verdad 74LS85

Comparing Inputs				Cascading Inputs			Outputs		
A3, B3	**A2, B2**	**A1, B1**	**A0, B0**	**A > B**	**A < B**	**A = B**	**A > B**	**A < B**	**A = B**
A3 > B3	X	X	X	X	X	X	H	L	L
A3 < B3	X	X	X	X	X	X	L	H	L
A3 = B3	A2 > B2	X	X	X	X	X	H	L	L
A3 = B3	A2 < B2	X	X	X	X	X	L	H	L
A3 = B3	A2 = B2	A1 > B1	X	X	X	X	H	L	L
A3 = B3	A2 = B2	A1 < B1	X	X	X	X	L	H	L
A3 = B3	A2 = B2	A1 = B1	A0 > B0	X	X	X	H	L	L
A3 = B3	A2 = B2	A1 = B1	A0 < B0	X	X	X	L	H	L
A3 = B3	A2 = B2	A1 = B1	A0 = B0	H	L	L	H	L	L
A3 = B3	A2 = B2	A1 = B1	A0 = B0	L	H	L	L	H	L
A3 = B3	A2 = B2	A1 = B1	A0 = B0	L	L	H	L	L	H
A3 = B3	A2 = B2	A1 = B1	A0 = B0	X	X	H	L	L	H
A3 = B3	A2 = B2	A1 = B1	A0 = B0	H	H	L	L	L	L
A3 = B3	A2 = B2	A1 = B1	A0 = B0	L	L	L	H	H	L

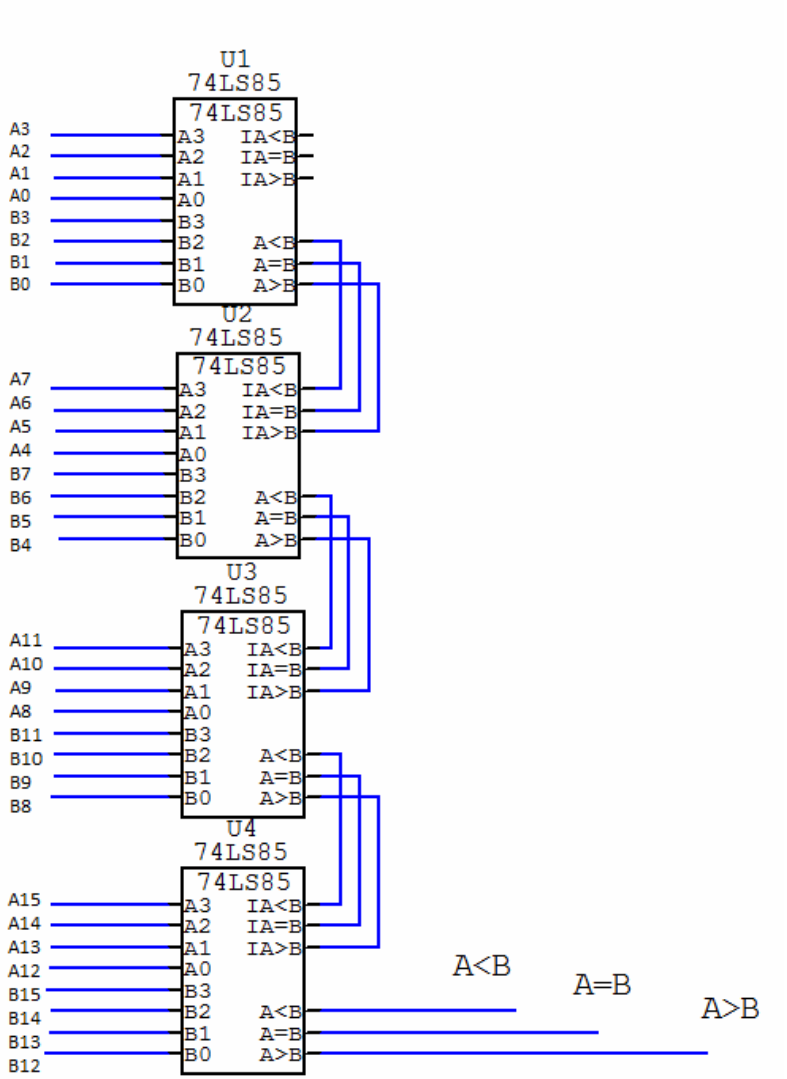
Circuito lógico 74LS85



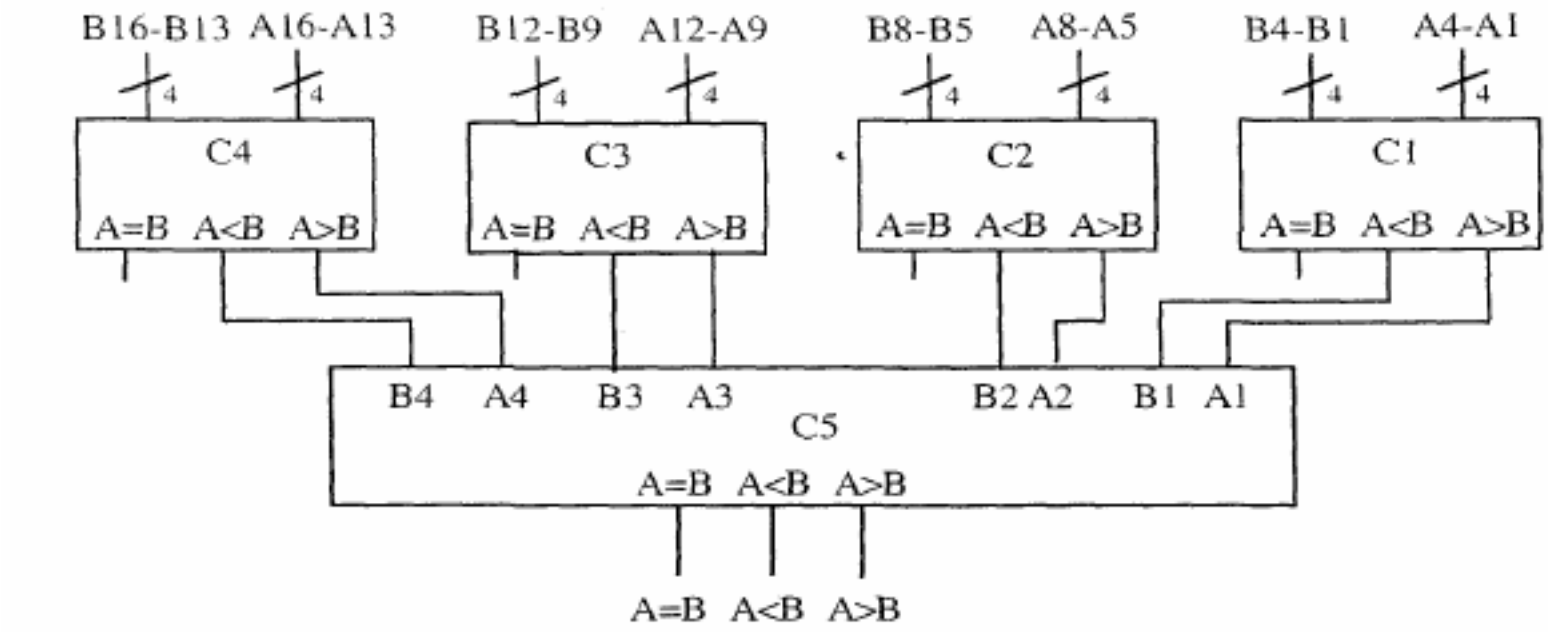
Comparador 8 bits



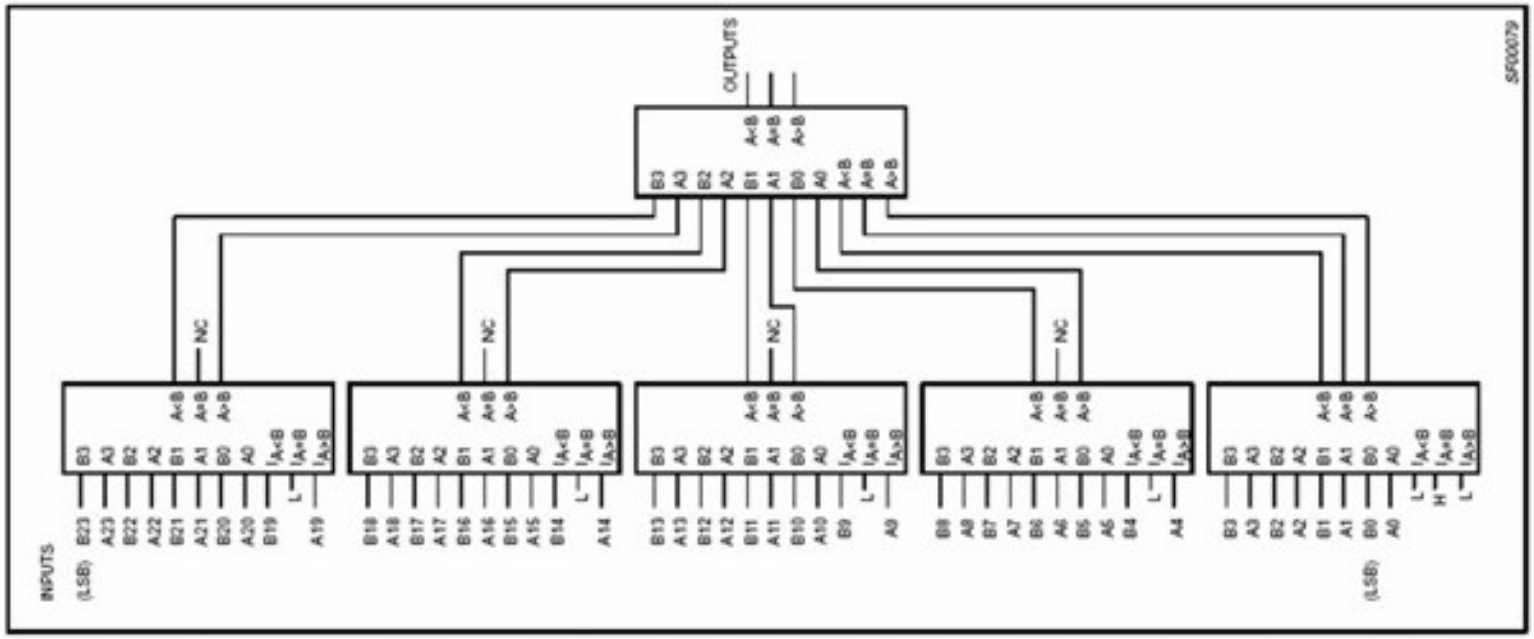
Comparador 16 bits



Comparador 16 bits



Comparador de 24 bits



Descripción VHDL

Comparador de 2 bits

```
entity comparador is
 port(a,b: in bit_vector (1 downto 0);
 mayor,menor, igual: out bit);
 end comparador;
```

```
architecture COMP of comparador is
begin
 process (a,b)
 begin
 if (a(1) = b(1)) and (a(0) = b(0)) then
      mayor <= '0';
      menor <= '0';
      igual <= '1';
elsif (a(1) = '1') and (b(1) = '0') then
      mayor <= '1';
      menor <= '0';
      igual <= '0';
elsif (a(1) = '0') and (b(1) = '1') then
      mayor <= '0';
      menor <= '1';
      igual <= '0';
elsif (a(0) = '1') and (b(0) = '0') then
     mayor <= '1';
     menor <= '0';
     igual <= '0';
elsif (a(0) = '0') and (b(0) = '1') then
     mayor <= '0';
     menor <= '1';
     igual <= '0';
end if;
end process;
end COMP;
```

Descripcion VHDL

Comparador de N bits

```
Library ieee;
use ieee.std_logic_1164.all;
use ieee.std_arith.all;

entity comparador is
port(a,b: in std_logic_vector(N downto 0);
 mayor,menor, igual: out std_logic);
 end comparador;
```

```
architecture COMP of comparador is
begin
process (a,b)
 begin
if  (a > b) then
      mayor <= '1';
      menor <= '0';
      igual <= '0';
elsif  (a < b)  then
      mayor <= '0';
      menor <= '1';
      igual <= '0';
else
      mayor <= '0';
      menor <= '0';
      igual <= '1';
end if;
end process;
end  COMP;
```

Problemas

- Diseñar un comparador de dos números A y B de 2 bits cada uno (A2A1 y B2B1). Para ello tendremos que realizar la tabla de verdad y obtener las funciones lógicas de S1,S2 y S3 a partir de los mapas de Karnaugh.

- Realizar el problema anterior a partir de comparadores de 1 bit. Una forma de hacerlo es estudiar las diferentes situaciones que se pueden dar en función de los bits de entrada (A2A1 y B2B1).

Problemas

- Se quieren diseñar circuitos digitales que realicen la comparación de dos números binarios con signo X e Y. Se deben obtener tres salidas que indiquen cuando X = Y, X > Y ó X < Y. Se debe utilizar en lo posible comparadores comerciales como el 74’85, y otros elementos lógicos cuando sea necesario. Indicar en cada caso el razonamiento o las ecuaciones lógicas que llevan al diseño final.
  Recordar: los números positivos siempre son mayores que los negativos; entre números positivos el mayor es el de mayor valor absoluto (5 > 3), entre números negativos el mayor es el de menor valor absoluto (-3 > -5).
  a) Suponer X (x3x2x1x0) e Y (y3y2y1y0) de 4 bits en complemento-2.
  b) Suponer X (Sx x3x2x1x0) e Y (Sy y3y2y1y0) de cinco bits en formato con bit de signo, donde Sx e Sy son los signos de X y de Y: 0 positivo, 1 negativo. Los otros bits contienen los módulos Mx y My de cada número en código binario. Para simplificar un poco el problema suponer que existe el +0 pero no existe el -0 (eso evita el problema de evaluar -0 = +0).

Bibliografía

- Electrónica Digital – J.Mira Mira (Ed. Sanz y Torres)
- Problemas de electrónica digital – Francisco Ojeda Cherta (Ed. Paraninfo)
- Problemas resueltos de electrónica digital – Cándido Bariáin Aisa (Universidad de Navarra)