

# COMPARADORES

- INTRODUCCION
- COMPARADORES DE 1 BIT
- COMPARADORES DE 4 BITS
  - Familia lógica
  - Descripción VHDL
- COMPARADOR SN74LS85
- COMPARADORES DE 16 BITS
- COMPARADORES DE 24 BITS

# INTRODUCCION

Un circuito digital comparador realiza la comparación de dos palabras A y B de N bits tomadas como un número entero sin signo e indica si son iguales o si una es mayor que otra en tres salidas  $A = B$ ,  $A > B$  y  $A < B$ . Solo una de estas salidas estará a 1 y las demás estarán a 0 dependiendo de los valores de las entradas.

# COMPARADORES DE 1 BIT

Para unos operandos A y B de un bit se puede desarrollar un comparador de la siguiente tabla:

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

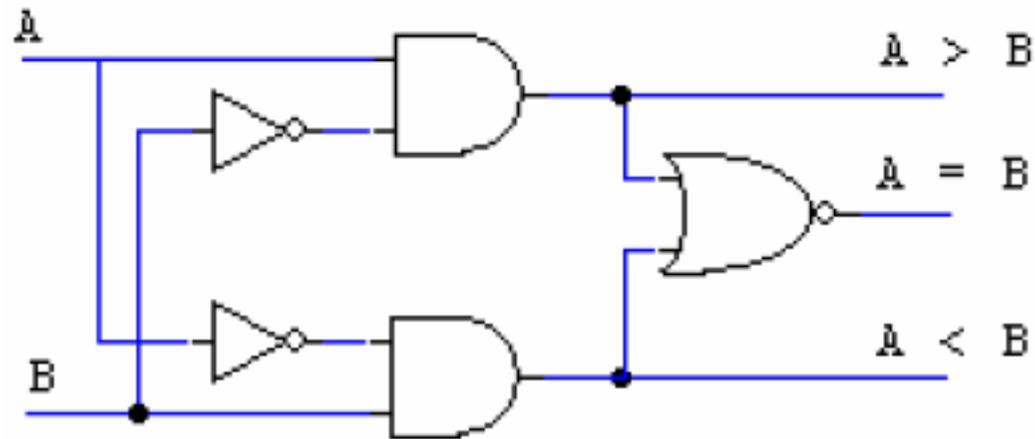
$$(A = B) = \overline{A \oplus B} = \overline{A \bar{B} + \bar{A} B}$$

$$(A > B) = A \bar{B}$$

$$(A < B) = \bar{A} B$$

# COMPARADORES DE 1 BIT

Circuito de un comparador de 1 bit implementado en puertas lógicas:



# COMPARADORES DE 4 BITS

En un comparador de 4 bits el procedimiento es el siguiente:

-  $A=B$

Deben ser iguales cada uno de sus bits.

-  $A>B$

Recorremos todos los bits de A y B desde el mas significativo hasta que A sea 1 y B sea 0.

-  $A<B$

Recorremos todos los bits de A y B desde el mas significativo hasta que A sea 0 y B sea 1.

# COMPARADORES DE 4 BITS

## Ejemplo1:

A(1010)

B(1001)

1 0 1 0  
1 0 0 1

A > B

## **NOTA:**

SE ESTUDIA BIT A BIT Y EN EL MOMENTO QUE A SEA 1 Y B O SE CUMPLE A > B Y VICEVERSA

SI NO SE DIERAN NINGUNA DE ESTAS SERIA LOS NUMEROS IGUALES

## Ejemplo2:

A(1010)

B(1010)

1 0 1 0  
1 0 1 0

A = B

## Ejemplo3:

A(1010)

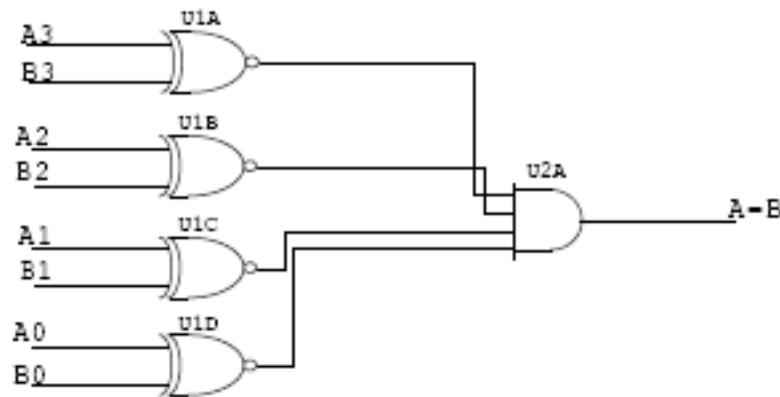
B(1110)

1 0 1 0  
1 1 1 0

A < B

# COMPARADORES DE 4 BITS

## ○ OPERACIÓN $A = B$



LA FUNCION LOGICA DE LA OPERACIÓN  $A = B$  ES :

$$F = \overline{A_3 \oplus B_3} * \overline{A_2 \oplus B_2} * \overline{A_1 \oplus B_1} * \overline{A_0 \oplus B_0}$$

# COMPARADORES DE 4 BITS

## ○ OPERACIÓN $A > B$

Partimos de la misma base, es decir, utilizar dos números de 4 bits cada uno.

Para que un número sea mayor que otro expresado en algebra Booleana sería:

$$F = A_3 * \overline{B_3} + \overline{A_3 \oplus B_3} * A_2 * \overline{B_2} + \overline{A_3 \oplus B_3} * \overline{A_2 \oplus B_2} * A_1 * \overline{B_1} + \overline{A_3 \oplus B_3} * \overline{A_2 \oplus B_2} * \overline{A_1 \oplus B_1} * A_0 * \overline{B_0}$$

Así se va comparando bit a bit teniendo en cuenta que compara de mas significativo a menos significativo

# COMPARADORES DE 4 BITS

## ○ OPERACIÓN $A < B$

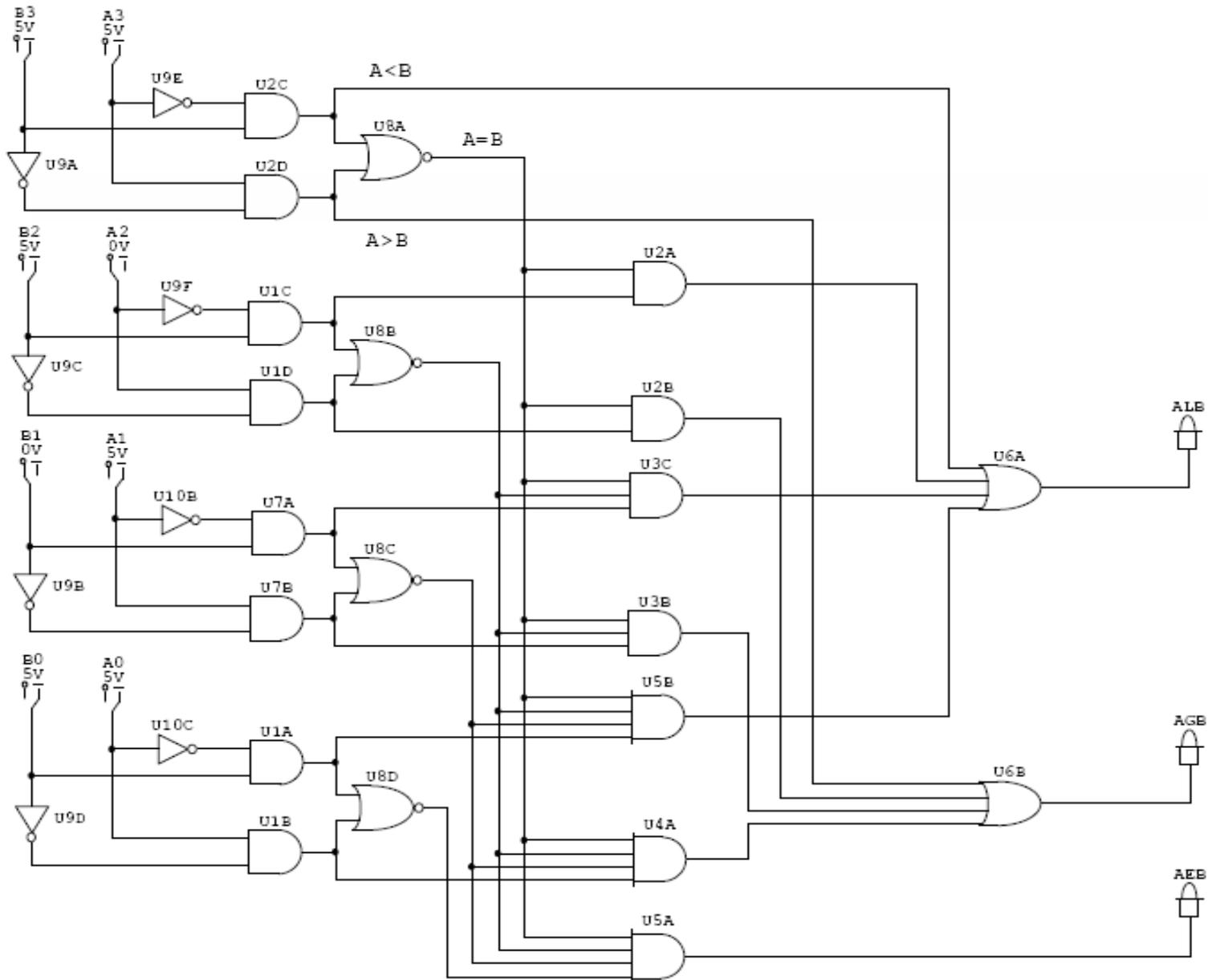
Partimos de la misma base, es decir, utilizar dos números de 4 bits cada uno.

Para que un número sea mayor que otro expresado en algebra Booleana sería:

$$F = B_3 * \overline{A_3} + \overline{B_3 \oplus A_3} * B_2 * \overline{A_2} + \overline{B_3 \oplus A_3} * \overline{B_2 \oplus A_2} * B_1 * \overline{A_1} + \overline{B_3 \oplus A_3} * \overline{B_2 \oplus A_2} * \overline{B_1 \oplus A_1} * B_0 * \overline{A_0}$$

Así se va comparando bit a bit teniendo en cuenta que compara de mas significativo a menos significativo

# COMPARADORES DE 4 BITS



# COMPARADORES DE 4 BITS

- Library ieee;
- Use ieee.std\_logic\_1164.all; --Estas sentencias reconocen el tipo std\_logic
- Entity compara is
- Port (A,B: in std\_logic\_vector(4 downto 1);
- Z: out std\_logic\_vector(3 downto 1));
- End compara;
- Architecture comportamiento of compara is
- Begin
- Process(A,B)
- Begin
- If (A>B) then
- Z<="100";
- elsif (A<B) then
- Z<="010";
- else
- Z<="001";
- End if;
- End process;
- End comportamiento;

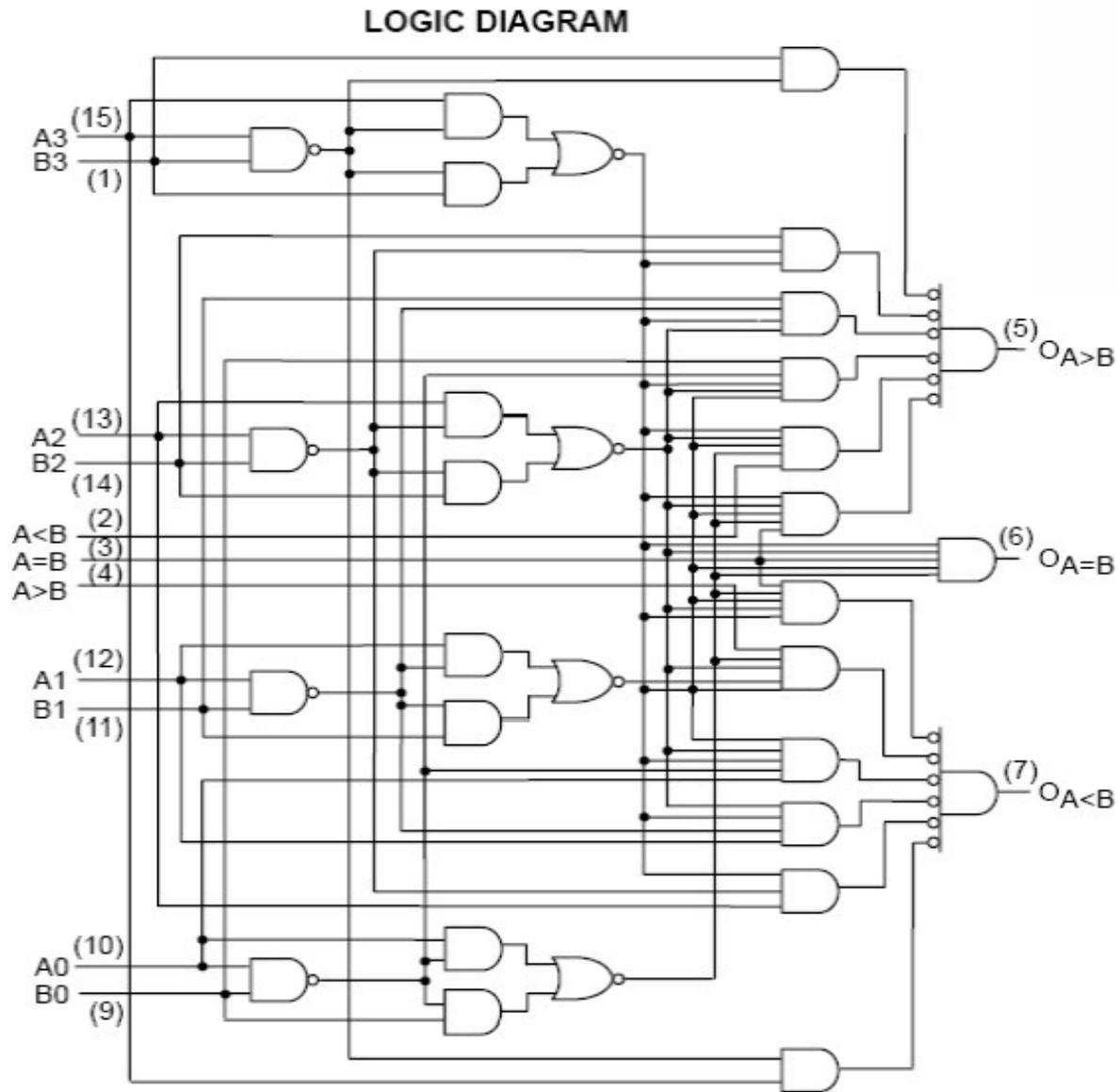
# COMPARADOR SN74LS85

TRUTH TABLE

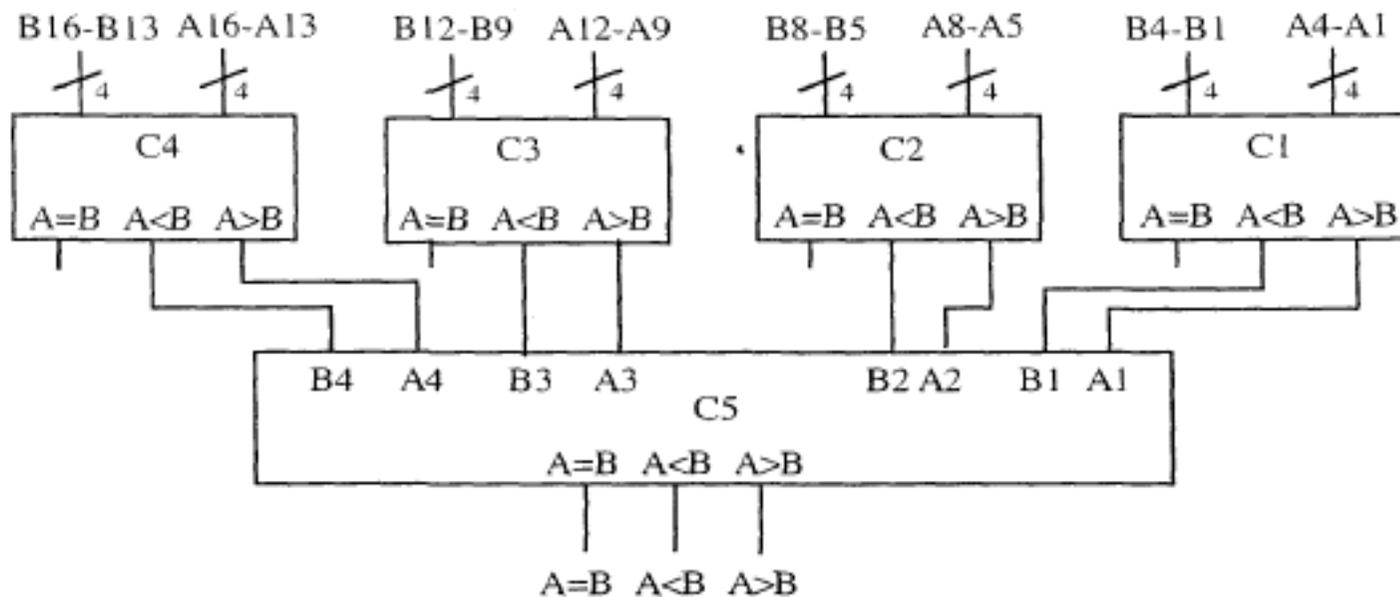
COMPARING INPUTS				CASCADING INPUTS			OUTPUTS		
A <sub>3</sub> ,B <sub>3</sub>	A <sub>2</sub> ,B <sub>2</sub>	A <sub>1</sub> ,B <sub>1</sub>	A <sub>0</sub> ,B <sub>0</sub>	I <sub>A&gt;B</sub>	I <sub>A&lt;B</sub>	I <sub>A=B</sub>	O <sub>A&gt;B</sub>	O <sub>A&lt;B</sub>	O <sub>A=B</sub>
A <sub>3</sub> >B <sub>3</sub>	X	X	X	X	X	X	H	L	L
A <sub>3</sub> <B <sub>3</sub>	X	X	X	X	X	X	L	H	L
A <sub>3</sub> =B <sub>3</sub>	A <sub>2</sub> >B <sub>2</sub>	X	X	X	X	X	H	L	L
A <sub>3</sub> =B <sub>3</sub>	A <sub>2</sub> <B <sub>2</sub>	X	X	X	X	X	L	H	L
A <sub>3</sub> =B <sub>3</sub>	A <sub>2</sub> =B <sub>2</sub>	A <sub>1</sub> >B <sub>1</sub>	X	X	X	X	H	L	L
A <sub>3</sub> =B <sub>3</sub>	A <sub>2</sub> =B <sub>2</sub>	A <sub>1</sub> <B <sub>1</sub>	X	X	X	X	L	H	L
A <sub>3</sub> =B <sub>3</sub>	A <sub>2</sub> =B <sub>2</sub>	A <sub>1</sub> =B <sub>1</sub>	A <sub>0</sub> >B <sub>0</sub>	X	X	X	H	L	L
A <sub>3</sub> =B <sub>3</sub>	A <sub>2</sub> =B <sub>2</sub>	A <sub>1</sub> =B <sub>1</sub>	A <sub>0</sub> <B <sub>0</sub>	X	X	X	L	H	L
A <sub>3</sub> =B <sub>3</sub>	A <sub>2</sub> =B <sub>2</sub>	A <sub>1</sub> =B <sub>1</sub>	A <sub>0</sub> =B <sub>0</sub>	H	L	L	H	L	L
A <sub>3</sub> =B <sub>3</sub>	A <sub>2</sub> =B <sub>2</sub>	A <sub>1</sub> =B <sub>1</sub>	A <sub>0</sub> =B <sub>0</sub>	L	H	L	L	H	L
A <sub>3</sub> =B <sub>3</sub>	A <sub>2</sub> =B <sub>2</sub>	A <sub>1</sub> =B <sub>1</sub>	A <sub>0</sub> =B <sub>0</sub>	X	X	H	L	L	H
A <sub>3</sub> =B <sub>3</sub>	A <sub>2</sub> =B <sub>2</sub>	A <sub>1</sub> =B <sub>1</sub>	A <sub>0</sub> =B <sub>0</sub>	H	H	L	L	L	L
A <sub>3</sub> =B <sub>3</sub>	A <sub>2</sub> =B <sub>2</sub>	A <sub>1</sub> =B <sub>1</sub>	A <sub>0</sub> =B <sub>0</sub>	L	L	L	H	H	L

H = HIGH Level  
 L = LOW Level  
 X = IMMATERIAL

# COMPARADOR SN74LS85



# COMPARADORES DE 16 BITS



Para que  $A = B$ , se necesita que las salidas ( $A = B$ ) de los cuatro comparadores C4-C 1 estén a 1 (todos los bits son iguales en cada grupo de 4 bits).

# COMPARADORES DE 16 BITS

Para que  $A > B$  se necesita que se cumpla una de estas condiciones:

- En C4 la salida ( $A > B$ ) sea 1 {lo que implica a su vez que  $A < B$  es 0}
- En C4 las salidas ( $A > B$ ) y ( $A < B$ ) son 0 (equivalente a que la salida ( $A = B$ ) es 1) y en C3 La salida( $A > B$ ) es 1.
- En C4 y C3 Las salidas { $A > B$ } y ( $A < B$ ) son 0 y en C2 la salida ( $A > B$ ) es 1
- En C4, C3 y C2 las salidas ( $A > B$ ) y ( $A < B$ ) son 0 y en C1 la salida ( $A > B$ ) es 1

Para que  $A < B$  se realiza el mismo razonamiento que en el punto anterior intercambiando los ( $A > B$ ) y ( $A < B$ ).

# COMPARADORES DE 24 BITS

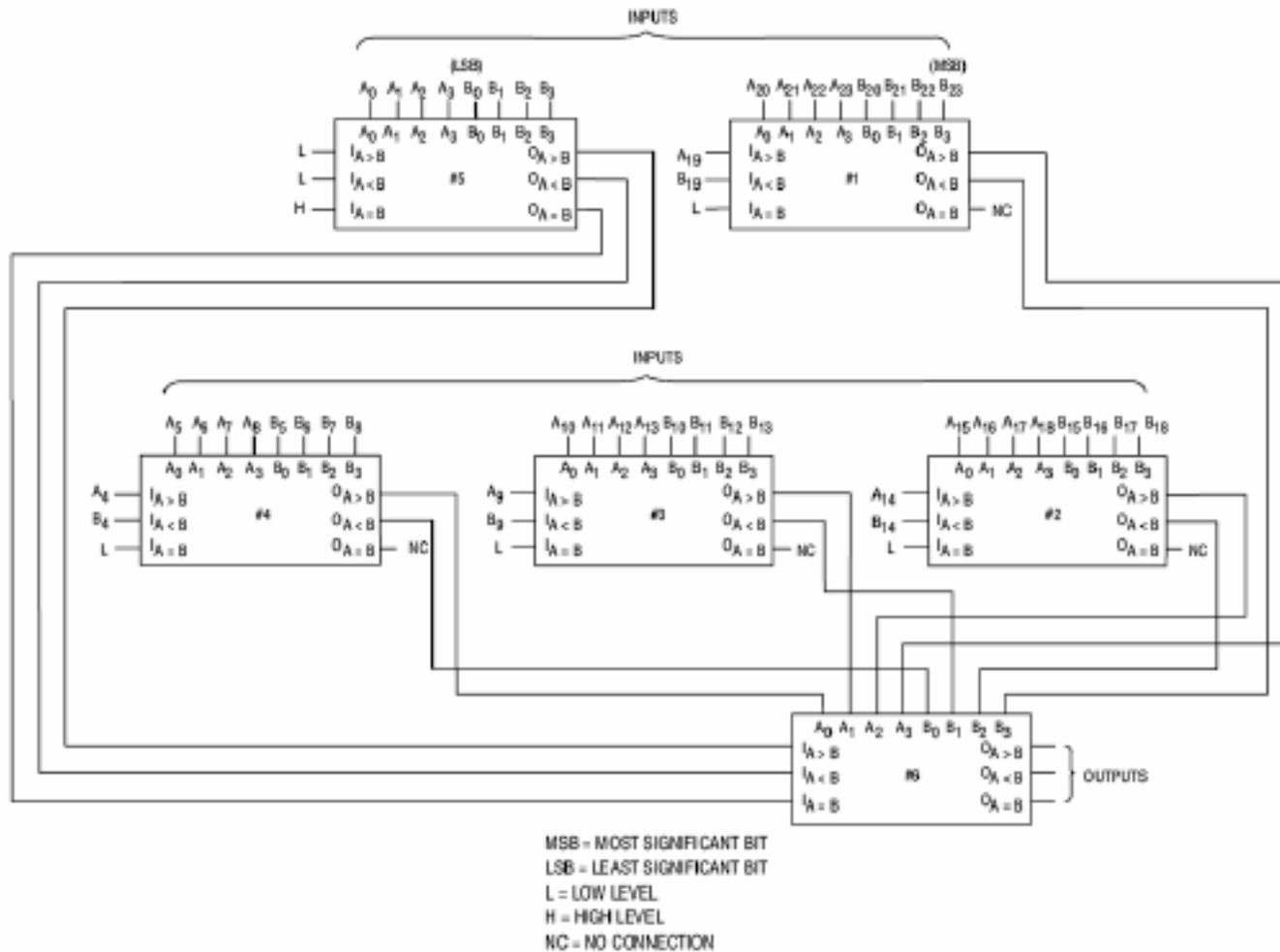


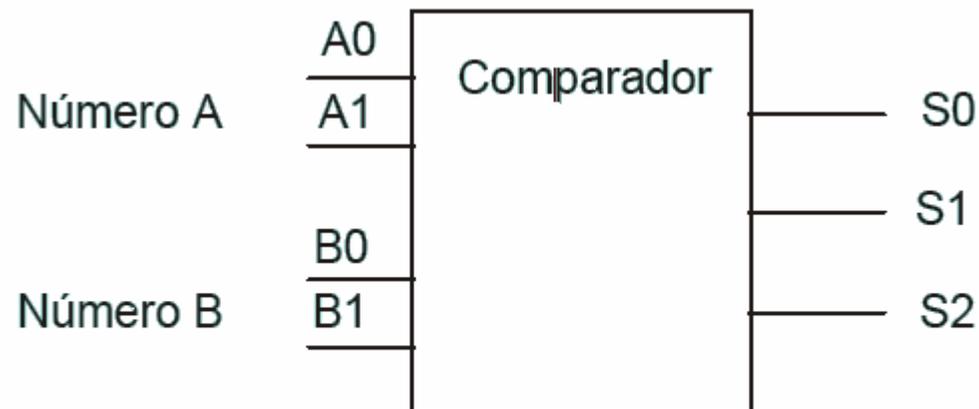
Figure 2. Comparison of Two 24-Bit Words

LAS ENTRADAS I SE UTILIZAN COMO ENTRADAS DE EXPANSION PARA ESTE TIPO DE COMPARADOR QUE NECESITA ENTRADAS EN CASCADA

# PROBLEMAS PROPUESTOS

- Problema 1:

El circuito de la figura es un comparador binario de dos números (A y B) de dos bits. Las salidas (S0, S1 y S2) toman el valor lógico "1" cuando  $A > B$ ,  $A < B$  y  $A = B$ , respectivamente. Obtenga las funciones lógicas de cada salida.



# PROBLEMAS PROPUESTOS

- Problema 2:

Realizar el diseño de un comparador de dos números A y B de cuatro bits tomando como base el sumador de números binarios de cuatro bits 74'83, utilizando puertas lógicas cuando sea necesario. El circuito debe generar tres salidas: O1 ( $A = B$ ), O2 ( $A > B$ ), O3 ( $A < B$ ).