
Don

a) Para comparar X e Y es necesario comparar cada bit de uno de ellos con el bit de igual peso del otro. Para hacer estas comparaciones utilizamos los comparadores de 1 bit propuestos. Si el resultado de la comparación de x_i e y_i es: M_i , I_i y m_i , se cumple que:

$$M_i = 1 \text{ si } x_i > y_i$$

$$I_i = 1 \text{ si } x_i = y_i$$

$$m_i = 1 \text{ si } x_i < y_i$$

A partir de M_i , I_i y m_i se generan las salidas M, I y m del comparador de 2 bits mediante las siguientes funciones booleanas:

$$M = M_1 + I_1 M_0$$

$$I = I_1 I_0$$

$$m = m_1 + I_1 m_0$$

La explicación de estas expresiones es inmediata:

$$\mathbf{X} > \mathbf{Y} \text{ si } (x_1 > y_1) \bullet ((x_1 = y_1) \mathbf{Y} (x_0 > y_0))$$

$$\mathbf{X} = \mathbf{Y} \text{ si } (x_1 = y_1) \mathbf{Y} (x_0 = y_0)$$

$$\mathbf{X} < \mathbf{Y} \text{ si } (x_1 < y_1) \bullet ((x_1 = y_1) \mathbf{Y} (x_0 < y_0))$$

La Figura 2000-2-2 muestra el circuito lógico pedido en este apartado obtenido a partir de las expresiones anteriores.

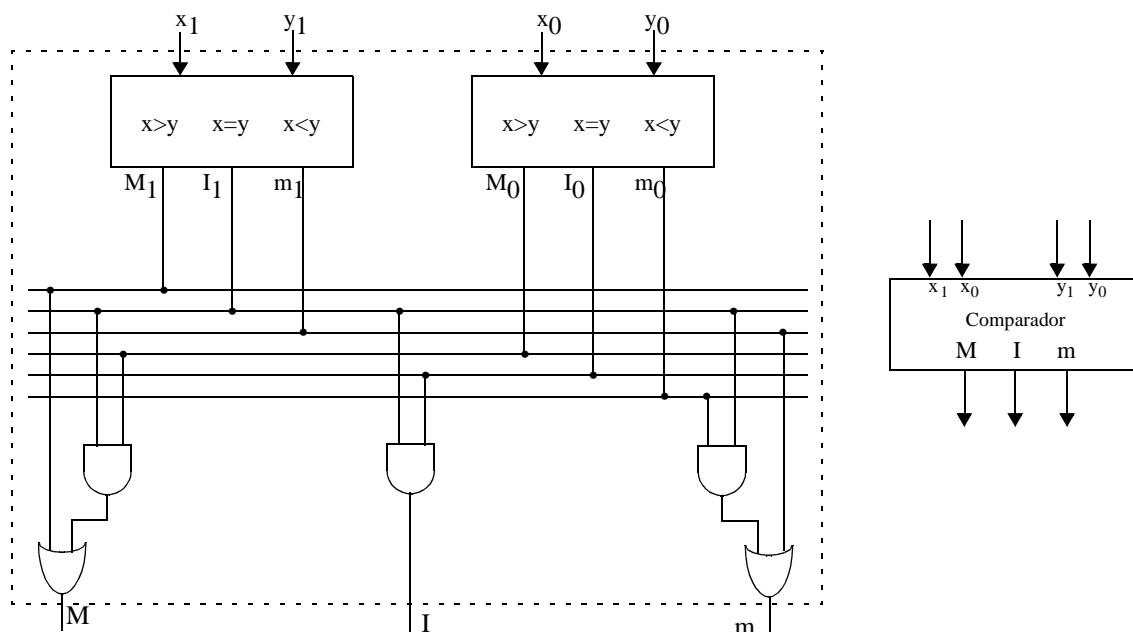


Figura 2000-2-2: Circuito lógico de un comparador de dos números de 2bits

b) Para calcular el tamaño de la memoria ROM necesaria hay que saber el número entradas y salidas del circuito.

- Entradas. Este circuito tiene **5 entradas**: 2 para la entrada de datos X (x_1x_0), 2 para la entrada de datos Y (y_1y_0) y una entrada de control (c) para la selección de la operación a realizar por el circuito. Esta última entrada de control va a tener el siguiente significado:
 - $c = 0$: Suma
 - $c = 1$: Producto
- Salidas: Este circuito tiene **4 salidas**. El resultado de la suma de dos números de dos bits tiene tan sólo 3 bits pero, para el resultado del producto de dos números de dos bits se requieren 4 bits. Como el circuito a diseñar tiene que poder hacer ambas operaciones, su salida ha de tener, necesariamente, 4 bits.

Por tanto, el tamaño de la memoria ROM necesario para implementar el sumador es de: 2^5 palabras \times 4 bits/palabra. El significado de cada una de sus entradas y salidas se muestra claramente en la Figura 2000-2-3. Finalmente, en la Tabla 2000-2-2 se muestra el contenido que debería tener la memoria ROM.

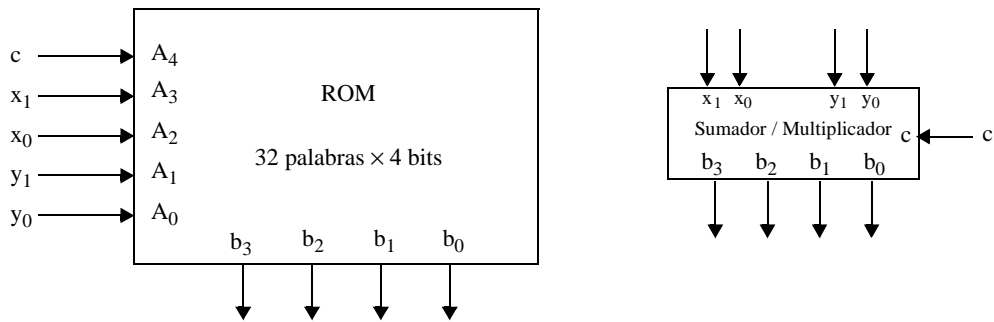


Figura 2000-2-3: Sumador/Multiplicador diseñado con memoria ROM

Dirección memoria c x ₁ x ₀ y ₁ y ₀	Contenido	Dirección memoria c x ₁ x ₀ y ₁ y ₀	Contenido
A ₄ A ₃ A ₂ A ₁ A ₀	b ₃ b ₂ b ₁ b ₀	A ₄ A ₃ A ₂ A ₁ A ₀	b ₃ b ₂ b ₁ b ₀
0 0 0 0 0	0 0 0 0	1 0 0 0 0	0 0 0 0
0 0 0 0 1	0 0 0 1	1 0 0 0 1	0 0 0 0
0 0 0 1 0	0 0 1 0	1 0 0 1 0	0 0 0 0
0 0 0 1 1	0 0 1 1	1 0 0 1 1	0 0 0 0
0 0 1 0 0	0 0 0 1	1 0 1 0 0	0 0 0 0
0 0 1 0 1	0 0 1 0	1 0 1 0 1	0 0 0 1
0 0 1 1 0	0 0 1 1	1 0 1 1 0	0 0 1 0
0 0 1 1 1	0 1 0 0	1 0 1 1 1	0 0 1 1
0 1 0 0 0	0 0 1 0	1 1 0 0 0	0 0 0 0
0 1 0 0 1	0 0 1 1	1 1 0 0 1	0 0 1 0
0 1 0 1 0	0 1 0 0	1 1 0 1 0	0 1 0 0
0 1 0 1 1	0 1 0 1	1 1 0 1 1	0 1 1 0
0 1 1 0 0	0 0 1 1	1 1 1 0 0	0 0 0 0
0 1 1 0 1	0 1 0 0	1 1 1 0 1	0 0 1 1
0 1 1 1 0	0 1 0 1	1 1 1 1 0	0 1 1 0
0 1 1 1 1	0 1 1 0	1 1 1 1 1	1 0 0 1

Tabla 2000-2-2: Contenido de la memoria ROM

c) En la Figura 2000-2-4 se muestra el diseño de la UAL pedida en el problema donde se han utilizado 4 multiplexores para seleccionar el valor adecuado a colocar en la salida, gobernados por las entradas de control c_1 y c_0 . También la entrada de control de la UAL c_0 se utiliza para seleccionar la operación a realizar por el Sumador/Multiplicador.

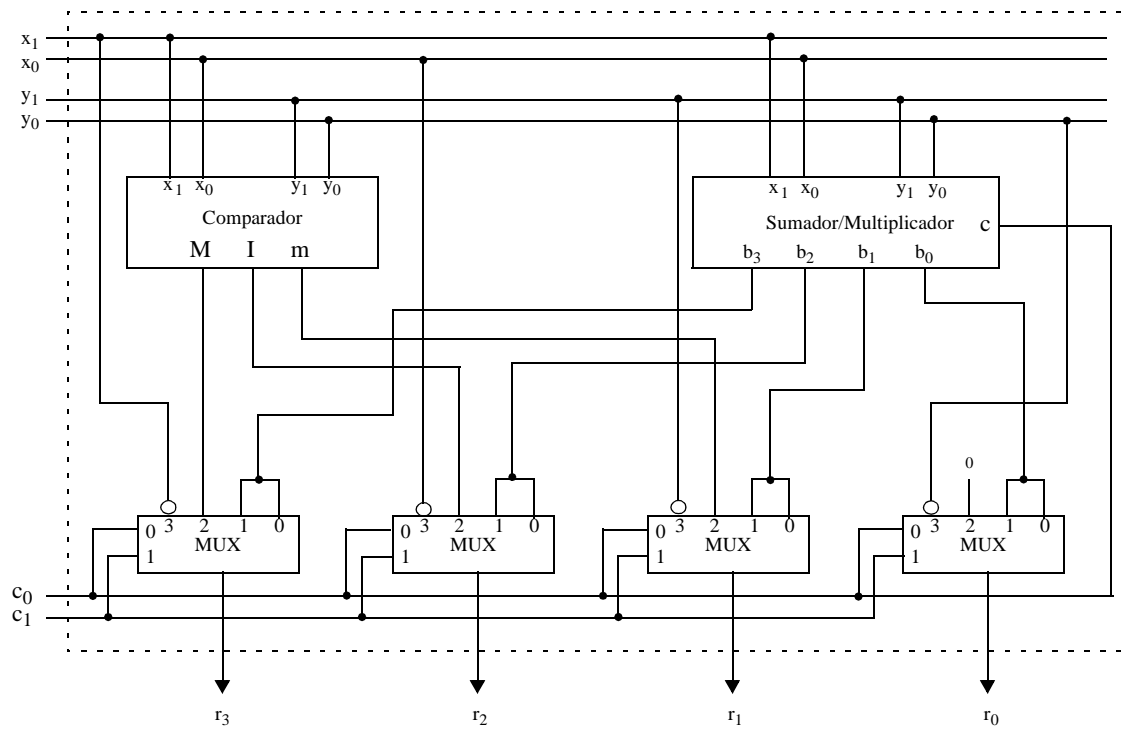


Figura 2000-2-4: UAL pedida en el problema