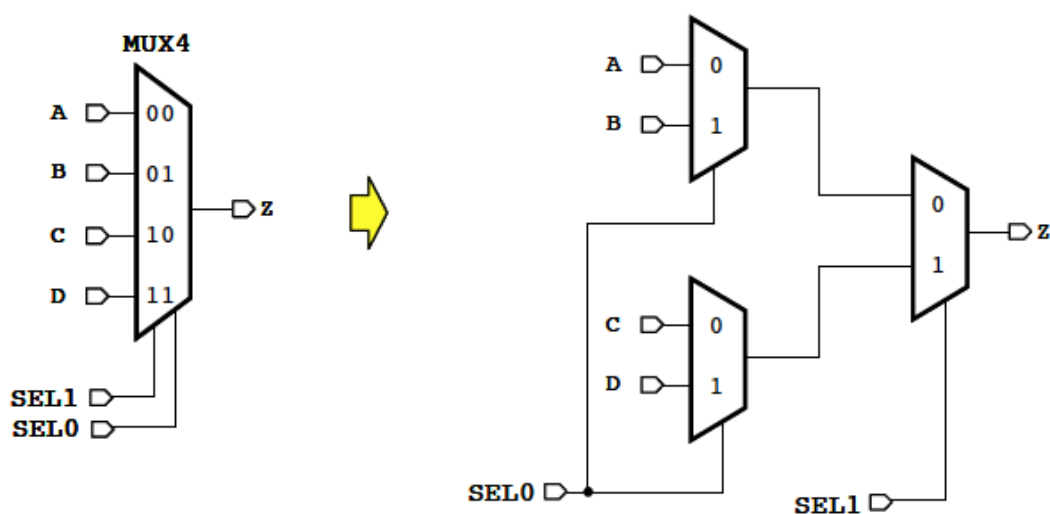


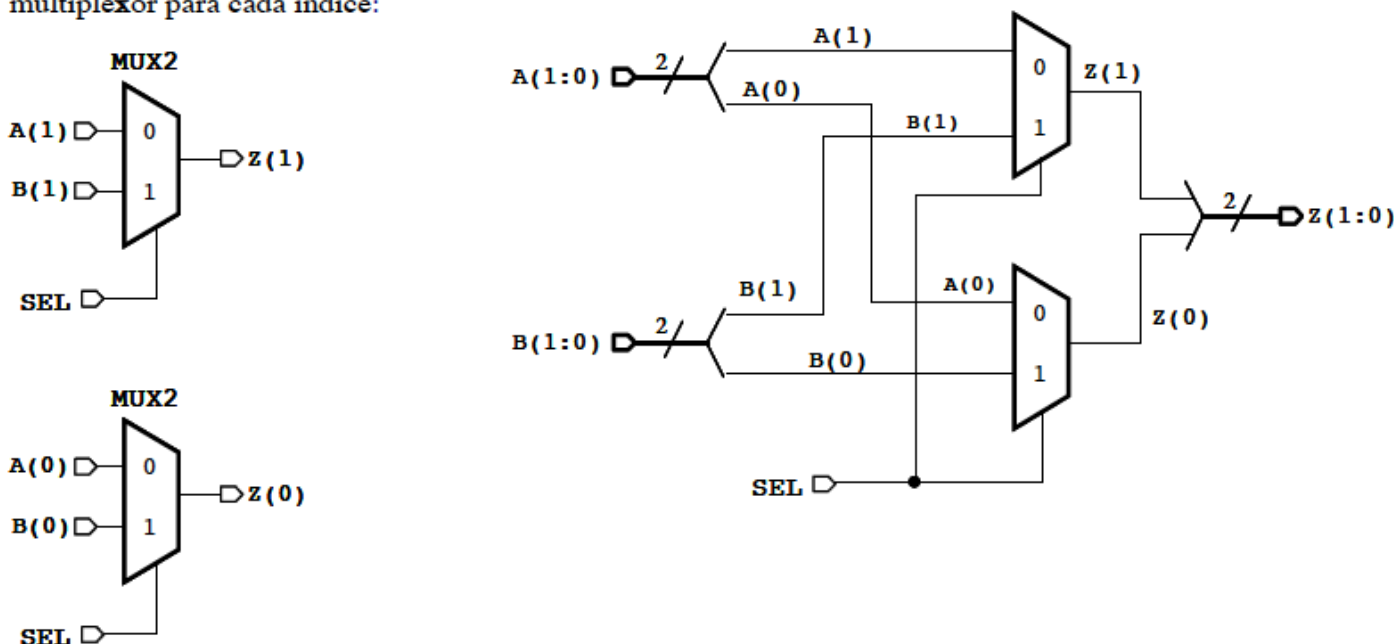
## 8.2. Soluciones

### 8.2.1. Realizar un multiplexor de 4 entradas de datos en función de multiplexores de 2 entradas de datos



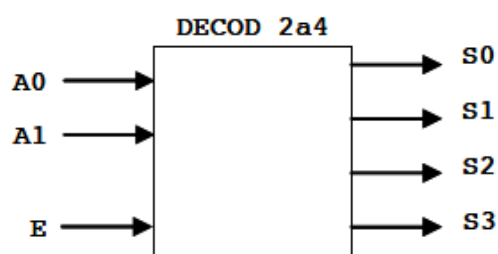
### 8.2.2. Realizar un multiplexor de 2 entradas de datos de dos bits de ancho, en función de multiplexores de 2 entradas de datos de un bit de ancho.

Simplemente hay que tomar un Dibujado todo junto queda:  
multiplexor para cada índice:



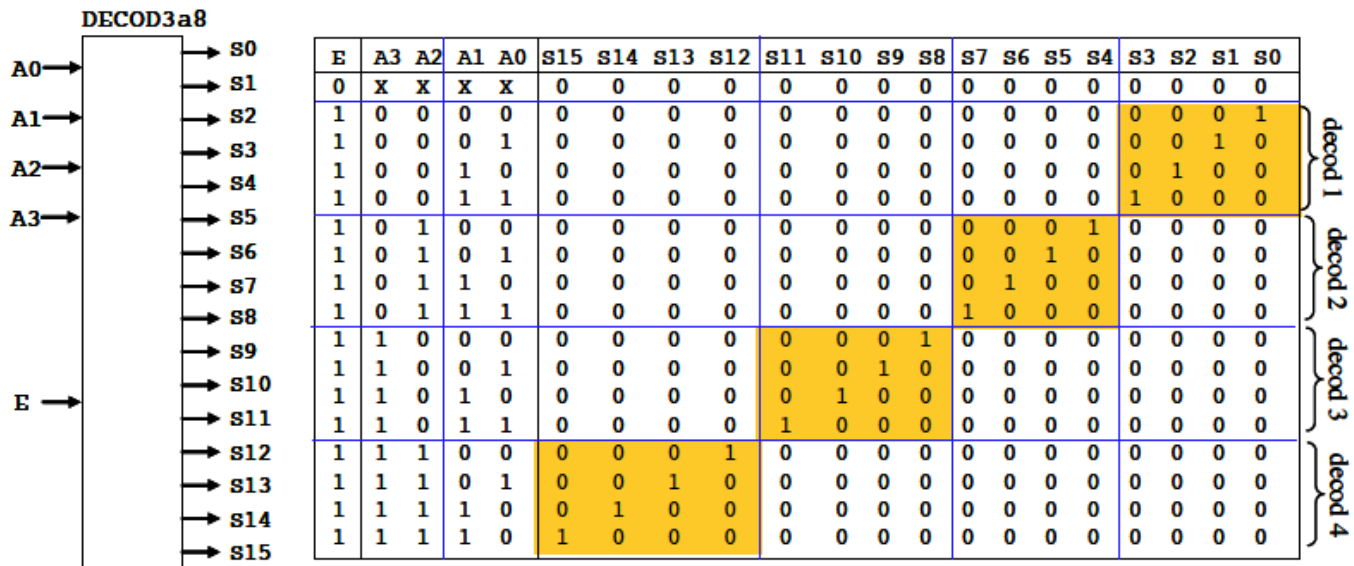
### 8.2.3. Realizar un decodificador de 4 entradas y 16 salidas a partir de decodificadores de 2 entradas y 4 salidas.

La tabla de verdad del decodificador de 2 a 4 es la siguiente:



E	A1	A0	S3	S2	S1	S0
0	X	X	0	0	0	0
1	0	0	0	0	0	1
1	0	1	0	0	1	0
1	1	0	0	1	0	0
1	1	1	1	0	0	0

Y la tabla de verdad del decodificador que queremos obteneres la siguiente:



En la tabla de verdad vemos que las señales de salida se han agrupado de cuatro en cuatro. En la siguiente figura se han separado estos grupos más el caso en que  $E = '0'$ .

[illegible]

todas '0'

E	A3	A2	A1	A0	S15	S14	S13	S12	S11	S10	S9	S8	S7	S6	S5	S4	S3	S2	S1	S0	1 <sup>er</sup> decod activo cuando  A3 = '0' A2 = '0'
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	
1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	
1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	
1	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	

$$A_2 = '0'$$

todas '0'

igual que decod2a4  
para las entradas A1 y A0

E	A3	A2	A1	A0	S15	S14	S13	S12	S11	S10	S9	S8	S7	S6	S5	S4	S3	S2	S1	S0	2° decod activo cuando  A3 = '0' A2 = '1'
1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	
1	0	1	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	
1	0	1	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	
1	0	1	1	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	

A3 = '0'

A2 = '1'

todas '0'

igual que  
decod2a4

todas '0'

[illegible]

A3 = '1'  
A2 = '0'

todas '0'

igual que  
decod2a4

todas '0'

[illegible]

A3 = '1'

A2 = '1'

igual que  
 decod2a4

todas '0'

**misma -  
secuencia**

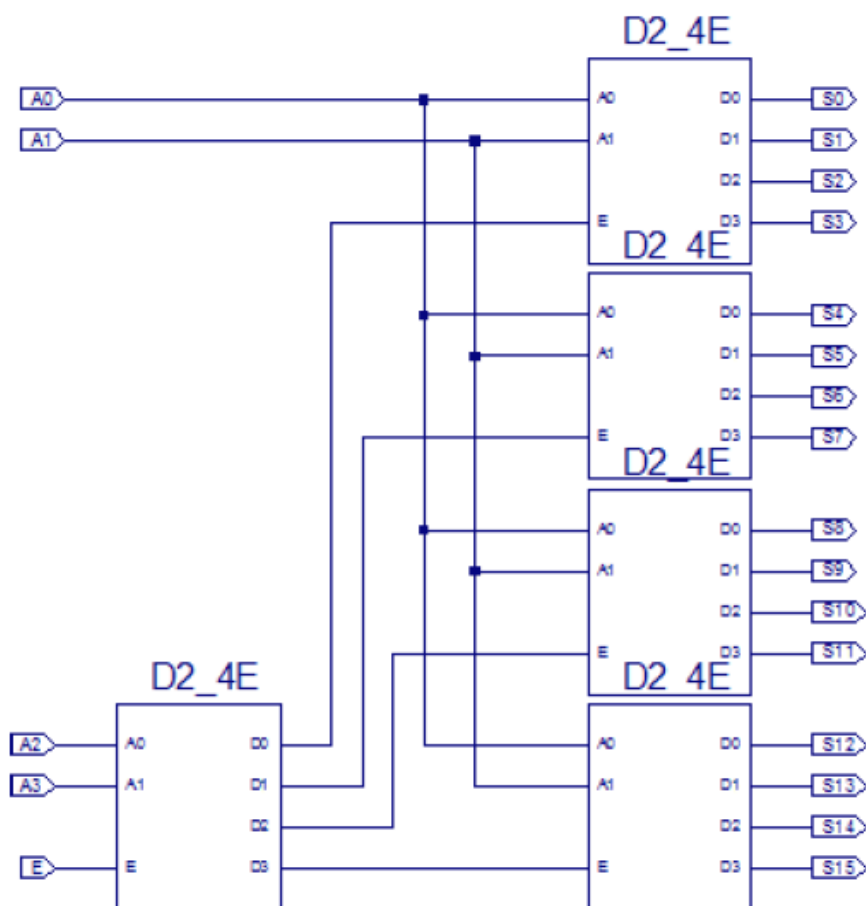
Podemos ver que para cada grupo se puede utilizar un decodificador de 2 a 4. Todos los decodificadores tendrán las mismas entradas A1 y A0. La única diferencia está en las salidas a las que se conecta cada decodificador, y la habilitación. Por ejemplo, el primer decodificador tendrá como salidas S0, S1, S2 y S3 y sólo estará habilitado cuando A3='0' y A2='0'; el segundo decodificador estará habilitado cuando A3='0' y A2='1'. Y de manera similar para el tercer y cuarto decodificador (como lo muestra la figura).

Así que usaremos cuatro decodificadores de 2 a 4 para las 16 salidas. Además necesitaremos crear la lógica que se encargue de habilitar cada decodificador. Si nos fijamos en las señales A3 y A2 de la tabla de verdad, y de cómo habilitan a cada decodificador, podemos extraer la tabla de verdad de la figura. En ella se ponen las señales de habilitación de cada decodificador (E4, E3, E2, E1) en función de la habilitación general (E) y las entradas A3 y A2. Podemos ver que la tabla de verdad se corresponde con la tabla de verdad de un decodificador de 2 a 4, así que para las señales de habilitación de los cuatro decodificadores usaremos otro decodificador de 2 a 4.

Señales de habilitación de los 4 decodificadores

E	A3	A2	E4	E3	E2	E1	
0	X	X	0	0	0	0	→ Ningún decodificador habilitado
1	0	0	0	0	0	1	→ 1 <sup>er</sup> decodificador habilitado
1	0	1	0	0	1	0	→ 2 <sup>o</sup> decodificador habilitado
1	1	0	0	1	0	0	→ 3 <sup>er</sup> decodificador habilitado
1	1	1	1	0	0	0	→ 4 <sup>o</sup> decodificador habilitado

Y conectamos los decodificadores:



#### 8.2.4. Explicar **brevemente** la implementación de una función lógica mediante un multiplexor y un decodificador. ¿Cuál de las dos opciones es preferible?

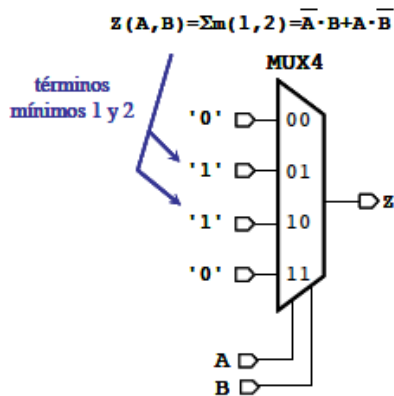
Con un multiplexor:

Si tenemos el mismo número de variables que señales de selección, simplemente hay que poner unos o ceros en las entradas del multiplexor según los términos mínimos de la función a realizar.

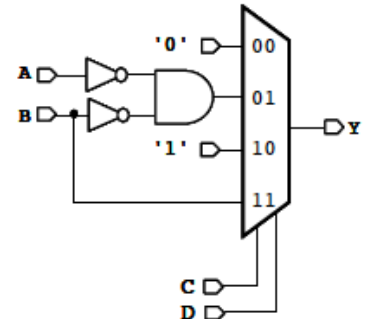
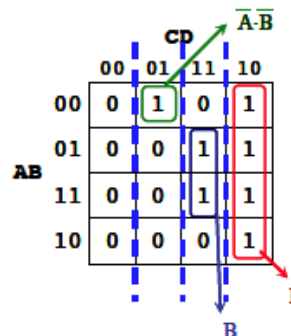
Si tenemos mayor número de variable, se puede obtener la función lógica o el mapa de Karnaugh, agrupando las variables según las variables que se hayan escogido como señales de selección.

Mismo número de variables que entradas de selección

Mayor número de variables que entradas de selección



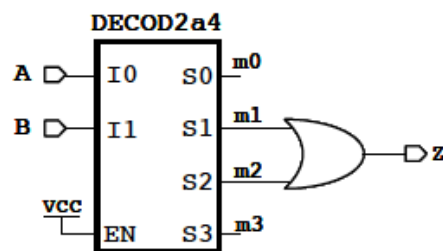
$Y(A, B, C, D) = \sum m(1, 2, 6, 7, 10, 14, 15)$



Con un decodificador:

Si tenemos el mismo número de variables que entradas del decodificador, se unen con una puerta OR todas las salidas correspondientes a los términos mínimos de la función.

$z(A, B) = \sum m(1, 2) = \bar{A} \cdot B + A \cdot \bar{B}$



Ventajas:

Si se tiene el mismo número de variables que entradas de selección, la ventaja del multiplexor es que no hace falta añadir ninguna puerta externa. Sin embargo con el decodificador necesitamos una puerta OR.

Por otro lado, con más variables que entradas de selección el multiplexor reduce la complejidad del mapa de Karnaugh.

El decodificador también podría utilizarse cuando hubiese más variables que señales de entrada, aunque el método de diseño no es tan fácil como el empleado en el multiplexor