UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA

Generació de funcions lògiques mitjançant multiplexors

Cognoms, nom	Martí Campoy, Antonio (amarti@disca.upv.es)
Departament	Informàtica de Sistemes i Computadors
Centre	Universitat Politècnica de València

1 Resum de les idees clau

En aquest article es presentarà la utilització de multiplexors per a la generació de funcions lògiques. Són moltes les formes de dissenyar una funció lògica, i una de les més senzilles és la utilització del bloc combinacional conegut com a multiplexor. Per a poder adquirir els coneixements i habilitats presentades en aquest article, has de comptar amb uns coneixements previs, llistats en la taula 1. Però tranquil, durant el text s'inclouran breus descripcions d'aquestes idees prèvies.

Taula 1 Coneixements previs

Coneixements previs
1. Què és una funció lògica i la seua aritat
2. Tipus i taules de debò de portes lògiques
3. Formes de representar una funció lògica: taula de veritat, formes canòniques i expressions algebraiques
4. Funcionament d'un multiplexor

2 Objectius

Una vegada llegit aquest article docent i reproduïts els exemples presentats, hauràs de ser capaç d'implementar una funció lògica mitjançant l'ús de multiplexors.

A més, la implementació de la funció lògica podrà prendre com a punt de partida diferents representacions de la mateixa, com la taula de veritat o una forma canònica, per la qual cosa seràs capaç de traduir des d'una representació a una altra.

Finalment, i atenent criteris de simplificació de circuits, seràs capaç de simplificar el disseny reduint la grandària del multiplexor.

3 Introducció

En primer lloc, una breu descripció dels conceptes previs més importants per a poder aconseguir els objectius proposats en aquest article. Aquestes descripcions poden ampliar-se consultant la bibliografia proposada al final del document.

- Funció lògica: expressió formal del comportament d'un circuit digital. L' aritat d'una funció lògica és el nombre de variables d'entrada.
- Taula de veritat: representació única en forma de taula d'una funció lògica.
- Formes canòniques: representació única com a suma de productes o com a producte de sumes d'una funció lògica.

- Expressió algebraica: combinació de variables i operadors lògics per a expressar una funció lògica.
- Porta lògica: circuit digital que implementa una funció lògica bàsica.
- Circuit o funció combinacional: circuit en el qual les eixides en un instant de temps depenen exclusivament de les entrades en aqueix mateix instant de temps.
- Multiplexor: circuit combinacional, amb m entrades de selecció, $n=2^m$ entrades de dades, i una única eixida.
- La funció realitzada pel multiplexor correspon amb la d'un selector o commutador, que conecta l'eixida amb una de les entrades.
- En concret, el valor de l'eixida pren el valor present en l'entrada i-esima quan en les entrades de selecció s'assigna el valor i.
- La Figura 1 mostra el símbol lògic per a un multiplexor d'm entrades de selecció i $n=2^m$ entrades de dades.

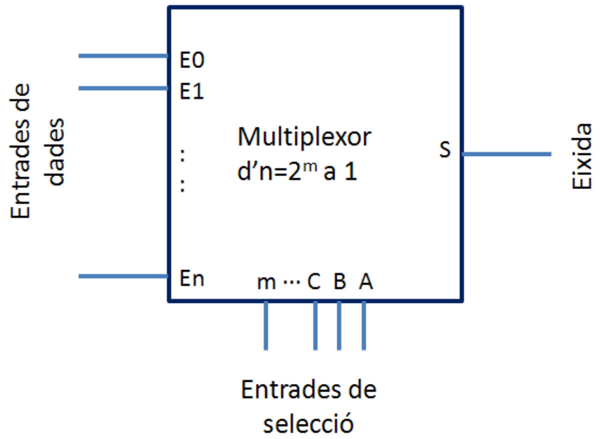

Figura 1 Símbol lògic d'un multiplexor d'n entrades de dades

Per a veure si el funcionament del multiplexor s'entén correctament, farem un parell de preguntes. Tenim un multiplexor de 2 entrades de selecció, anomenades B i A, sent A la de menor pes. Les entrades de dades, quatre, es diuen I0, I1, I2 i I3, sent I0 la de menor pes. La Taula 2 mostra en cada fila una combinació de valors per a les entrades. Quin és el valor de l'eixida per a cadascuna de les 6 combinacions? Per favor, pensa les respostes abans de veure les solucions[i]

Taula 2 Exercicis de multiplexors

	Entrades de selecció		Entrades de dades				
Combinació	B	A	E3	E2	E1	E0	Eixida
0	0	1	1	1	0	1	¿?
1	0	1	1	0	1	0	¿?
2	1	1	1	0	0	1	¿?
3	1	1	0	1	1	1	¿?
4	1	0	1	0	1	1	¿?
5	1	0	0	1	0	0	¿?

4 Generació de funcions

La generació de funcions utilitzant multiplexors és molt senzilla. En primer lloc, veurem com generar una funció utilitzant un multiplexor amb el mateix nombre d'entrades de selecció que variables d'entrada (aritat) que té la funció. En segon lloc, veurem una optimització que permet utilitzar un multiplexor amb una entrada de selecció menys que variables d'entrada té la funció.

4.1 Multiplexors amb la mateixa aritat que la funció

La forma més senzilla d'implementar una funció amb un multiplexor és utilitzar un que tinga el mateix nombre d'entrades de selecció que variables d'entrada tinga la funció. Així, per a una funció amb tres variables d'entrada (aritat 3) utilitzaríem un multiplexor amb tres entrades de selecció, és a dir, un multiplexor de 8 a 1.

D'aquesta manera, en cadascuna de les entrades de dades col·locarem el valor corresponent de l'eixida de la funció. Finalment, connectarem les variables d'entrada de la funció a les entrades de selecció del multiplexor.

Però ho veurem millor amb un exemple. La Taula 3 mostra la taula de veritat de la funció F, d'aritat 3. L'Equació 1 mostra les formes canòniques conjuntiva i disjuntiva per a la mateixa funció F.

Taula 3 Taula de veritat de la funció F

Valoració	C B A	F
0	0 0 0	0
1	0 0 1	1
2	0 1 0	0
3	0 1 1	0
4	1 0 0	1
5	1 0 1	1
6	1 1 0	1
7	1 1 1	0

Equació 1 Formes Canòniques Conjuntiva i Disjuntiva de la funció F

$$F = \prod_{C,B,A}(0,2,3,7)$$

$$F = \sum_{C,B,A}(1,4,5,6)$$

Podem veure que quan les entrades de la funció valen 1, 4, 5 o 6 l'eixida de la funció és 1. Per tant, en les entrades del multiplexor 1, 4, 5, i 6 posarem un 1, D'aquesta manera, quan seleccionem una d'aquestes entrades, tindrem en l'eixida un 1.

Igualment, podem veure que quan les entrades de la funció valen 0, 2, 3, o 7 l'eixida de la funció és 0. Per tant, en les entrades del multiplexor 0, 2, 3 i 7 posarem un zero. D'aquesta manera, quan seleccionem una d'aquestes entrades, tindrem en l'eixida un 0.

La Figura 2 mostra la implementació de la funció F emprant un multiplexor de 8 a 1. És molt important que et fixes en l'ordre en què es connecten les variables a les entrades de selecció.

Algunes funcions tenen entrades indiferents, és a dir, que l'eixida no està definida per a algunes entrades. Aquestes entrades s'identifiquen en la taula de veritat amb el símbol "X", i en les formes canòniques com un grup de minitermes o maxitermes diferenciats. La Taula 4 mostra la taula de veritat per a la funció H, amb entrades indiferents. L'Equació 2 mostra les formes canòniques conjuntiva i disjuntiva per a la mateixa funció H.

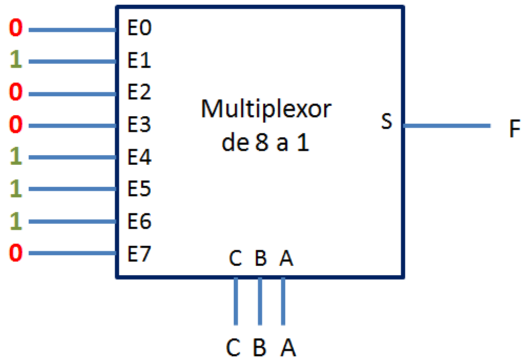

Figura 2 Implementació de la funció F emprant un multiplexor

Taula 4 Taula de veritat de la funció H

Valoració	C B A	H
0	0 0 0	1
1	0 0 1	X
2	0 1 0	1
3	0 1 1	X
4	1 0 0	X
5	1 0 1	0
6	1 1 0	1
7	1 1 1	0

Equació 2 Formes Canòniques Conjuntiva i Disjuntiva de la funció H

$$H = \prod_{C,B,A}(5,7) \cdot \prod_{\emptyset}(1,3,4)$$

$$H = \sum_{C,B,A}(0,2,6) + \sum_{\emptyset}(1,3,4)$$

Quan la funció inclou entrades indiferents, és obligatori identificar-les correctament en la taula de veritat i en les formes canòniques. Però a l'hora d'implementar la funció amb un circuit, el símbol "X" no pot utilitzar-se, per la qual cosa, cal assignar-li un valor de 0 o d'1. I precisament, com són entrades indiferents, és indiferent assignar valor 0 a totes les X, o valor 1 a totes les X, o valor 0 i 1 de forma aleatòria. El que no ha de fer-se mai és deixar una entrada a l'aire, o utilitzar valors diferents de 0 i d'1. La Figura 3 mostra una de les possibles implementacions de la funció H emprant un multiplexor de 8 a 1.

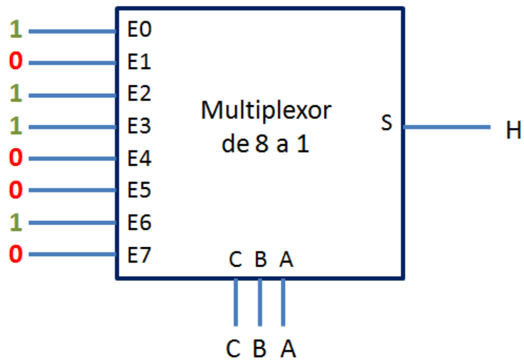

Figura 3 Implementació de la funció H amb un multiplexor de 8 a 1

La Figura 4 mostra els passos a seguir per a implementar una funció amb un multiplexor, partint de la taula de veritat o de les formes canòniques.

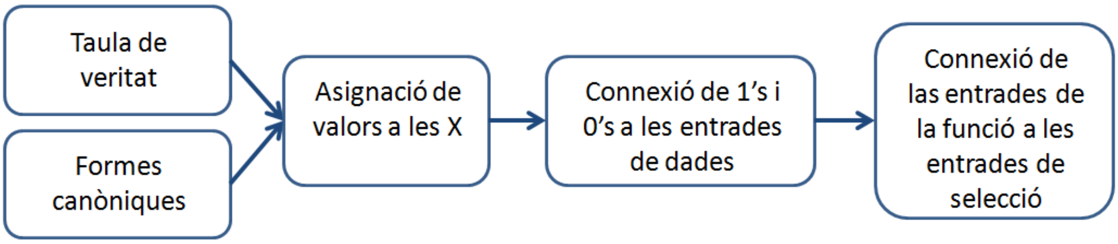

Figura 4 Passos per a la implementació d'una funció emprant un multiplexor

4.2 Multiplexors de menor aritat que la funció

En aquest apartat veurem com implementar una funció d'aritat n utilitzant un multiplexor amb n-1 entrades de selecció.

En tenir una entrada de selecció menys, el multiplexor disposa d'exactament la meitat d'entrades de dades que valoracions té la funció. En principi solament podríem implementar la meitat de la funció, la qual cosa no seria de gran utilitat. L'opció d'utilitzar dos multiplexors tampoc seria molt útil.

Com solucionar aquest problema? Fàcil. Expressar l'eixida de la funció lògica com a funció de la variable d'entrada de menor pes.

Torna a llegir el paràgraf anterior. Estic d'acord amb tu. No queda massa clar, així que ho veurem amb una funció molt senzilla, d'aritat dos. La Taula 5 mostra la taula de veritat de la funció I. Les seues variables d'entrada són B i A, sent A la de menor pes.

Taula 5 Taula de veritat de la funció E

Valoració	B A	E
0	0 0	1
1	0 1	1
2	1 0	0
3	1 1	1

Si parem atenció a la taula de veritat, veurem que quan l'entrada B val 0, l'eixida val 1 independentment del valor de l'entrada A. I quan l'entrada B val 1, l'eixida de la funció pren el valor de l'entrada A. D'aquesta manera podem canviar l'entrada A a la columna de l'eixida, i obtenir una taula reduïda, que es mostra en la Taula 6. Fixa't que ara la funció s'expressa com I(A), és a dir, s'indica que depén de l'entrada A.

Taula 6 Taula de veritat reduïda de la funció E(A)

Valoració	B	E(A)
0	0	1
1	1	A

Una vegada hem obtingut la taula reduïda, ja podem utilitzar un multiplexor amb una entrada de selecció menys. En aquest cas, l'aritat de la funció és 2, per la qual cosa utilitzarem un multiplexor d'una entrada de selecció, o el que és el mateix, un multiplexor de 2 a 1. La Figura 5 mostra la implementació de la funció I emprant un multiplexor amb una única entrada de selecció. És important que et fixes que a l'entrada de selecció, s'ha connectat l'entrada de major pes de la funció I, i no la de menor pes.

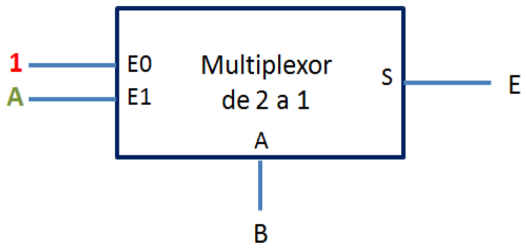

Figura 5 Implementació de la funció E amb un multiplexor d'una entrada de selecció

Si la funció compta amb entrades indiferents, aquestes poden utilitzar-se per a simplificar la implementació. La Taula 7 mostra la taula de veritat de la funció F, d'aritat 3, i la taula reduïda per a implementar-la utilitzant un multiplexor de dues entrades de selecció. Fixa't en l'agrupació de l'eixida de la funció per a les quatre combinacions de les variables C i B.

Taula 7 Taula de veritat i taula reduïda de la funció F

Valoració	C B A	F	
0	0 0 0	1	$1,\overline{A}$
1	0 0 1	X	
2	0 1 0	0	$0,A$
3	0 1 1	X	
4	1 0 0	X	$0,\overline{A}$
5	1 0 1	0	
6	1 1 0	1	$\overline{A}$
7	1 1 1	0	

C B	F(A)
0 0	$1, \overline{A}$
0 1	$0, A$
1 0	$0, \overline{A}$
1 1	$\overline{A}$

És important que pares atenció que, en ser una funció amb entrades indiferents, existeixen diverses possibilitats de reduir la funció. En la taula reduïda cal que apareguen totes les opcions, ja que en cas contrari estem ocultant informació sobre la funció lògica. Tanmateix això, a l'hora d'implementar el circuit utilitzant un multiplexor, és necessari decidir quina de les possibles entrades escollirem. Aquesta decisió es realitzarà sobre la base de criteris d'optimització i simplificació de circuits. Per exemple, si podem escollir entre posar un 1 o l'entrada negada, triarem el valor fix 1, ja que ens estalviem la porta NOT necessària per a negar l'entrada. La Figura 6 mostra una possible implementació de la funció F. És important que pares atenció a com es connecten les entrades de la funció a les entrades de selecció del multiplexor.

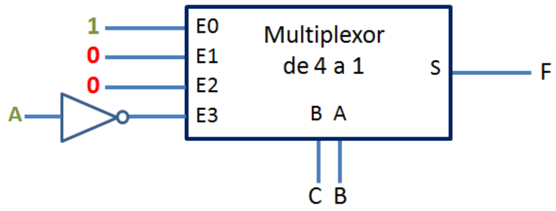

Figura 6 Implementació de la funció F emprant un multiplexor amb una entrada de selecció de menys

La Figura 7 mostra els passos per implementar una funció emprant un multiplexor amb una entrada de selecció menys que l'aritat de la funció.

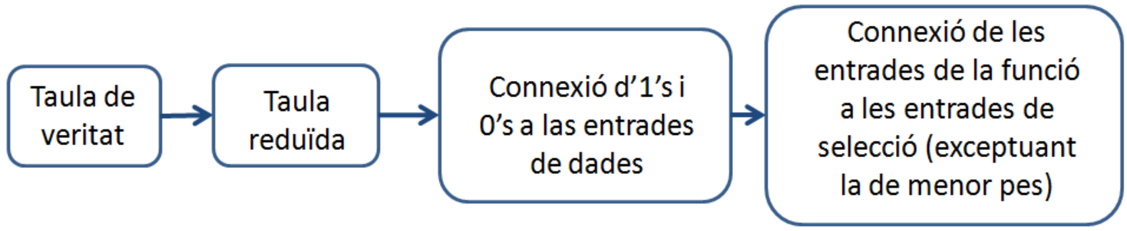

Figura 7 Passos per utilitzar un multiplexor amb una entrada de selecció de menys

4.3 Resum

La implementació d'una funció lògica per mitjà de l'ús d'un multiplexor pot fer-se de manera directa des de la taula de veritat o des de les formes canòniques, emprant un multiplexor amb el mateix nombre d'entrades de selecció que variables d'entrada tinga la funció. Es a dir, coincidint l'aritat de la funció i de la funció.

També és possible implementar una funció lògica d'n variables d'entrada utilitzant un multiplexor amb n-1 entrades de selecció. En aquest cas, és necessari partir de la taula de veritat i crear una taula reduïda, en la qual s'expressa l'eixida del circuit com a funció de l'entrada de menor pes.

En aquest últim cas, l'existència d'eixides indiferents, conegudes com Xs, permet realitzar simplificacions i optimitzacions.

4.4 Exercicis

A continuació uns exercicis senzills. Per la funció lògica J expressada per la seua forma canònica disjuntiva, respon les qüestions següents:

$$J = \sum_{Z,Y,X,W}(0,1,3,7,8,13) + \sum_{\emptyset}(2,4,5,9,12,15)$$

a. Si utilitzem un multiplexor de 4 entrades de selecció, quins seran els valors de les entrades? (Decidim donar valor 0 a les entrades indiferents)
b. Si utilitzem un multiplexor de 4 entrades de selecció, anomenades D, C, B i A, amb A la de menor pes, quina cal que siga la connexió de les entrades de la funció a les entrades de selecció del multiplexor?
c. Si utilitzem un multiplexor de 3 entrades de selecció, quins seran els valors de les entrades? Indica, per cadascuna de les entrades tots el possibles valors, tal com es mostraria en la taula reduïda.

UNIVERSITAT POLITÈCNICA DE VALÈNCIA

d. Si utilitzem un multiplexor de 3 entrades de selecció, anomenades C, B i A, amb A la de menor pes, quina cal que siga la connexió de les entrades de la funció a les entrades de selecció del multiplexor?
e. Prova de respondre les preguntes abans de veure les solucions, per favor[ii].

5 Conclusions

En aquest article has pogut conéixer una forma ràpida i senzilla d'implementar una funció lògica mitjançant l'ús de multiplexors. Per a una funció d'aritat n, és possible utilitzat un multiplexor d'n entrades de selecció, o un multiplexor d'n-1 entrades de selecció. En el primer cas, els valors que cal connectar a les entrades de dades són directament els mostrats en la taula de veritat o en les formes canòniques. En el segon cas, és necessari començar amb la taula de veritat i realitzar agrupacions, per expressar l'eixida com a funció de la variable d'entrada de menor pes.

6 Bibliografia

6.1 Llibres:

[1] John F. Wakerly *Digital design : principles and practices*, Upper Saddle River : Pearson Prentice Hall. 2006

[2] Antonio Lloris Ruiz; Alberto Prieto Espinosa; Luis Parrilla Roure *Sistemas digitales*, Aravaca, Madrid : McGraw-Hill/Interamericana de España. 2003

6.2 Referències de fonts electròniques:

[3] Martí Campoy, Antonio "Circuitos combinacionales: multiplexores" Universitat Politècnica de València. Escola Tècnica Superior d'Enginyeria Informàtica. 2011. http://politube.upv.es/play.php?vid=46987

---

[i] Les eixides, per a les combinacions 0 a 5 son: 0, 1, 1, 0, 0, 1.

[ii] Respostes als exercicis: a) Els valors de les entrades de E0 a E15 són:1,1,0,0,0,0,1,1,1,0,0,0,1,0,1; b) La connexió de les entrades de selecció és: W→A, X→B, Y→C, Z→D; c)Els possibles valors de les entrades de E0 a E7 són: E0 = 1; E1 = (1, W); E2 = (0,1,W, Not(W)); E3 = W; E4 = (1, Not(W)); E5 = 0; E6 = (1,W); E7 = (0,W); d) La connexió de les entrades de selecció és: X→A, Y→B, Z→C.