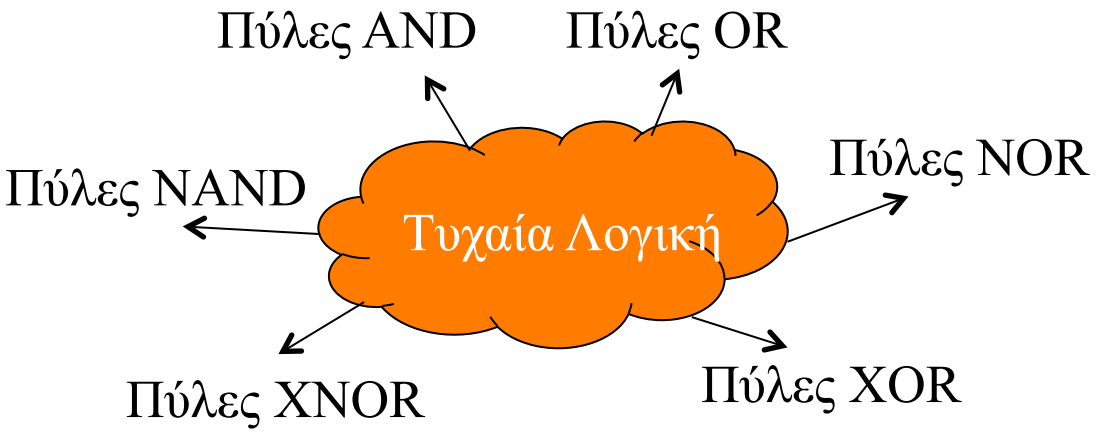
2η Θεματική Ενότητα : Σύνθετα Συνδυαστικά Κυκλώματα

Επιμέλεια διαφανειών:
Χρ. Καβουσιανός

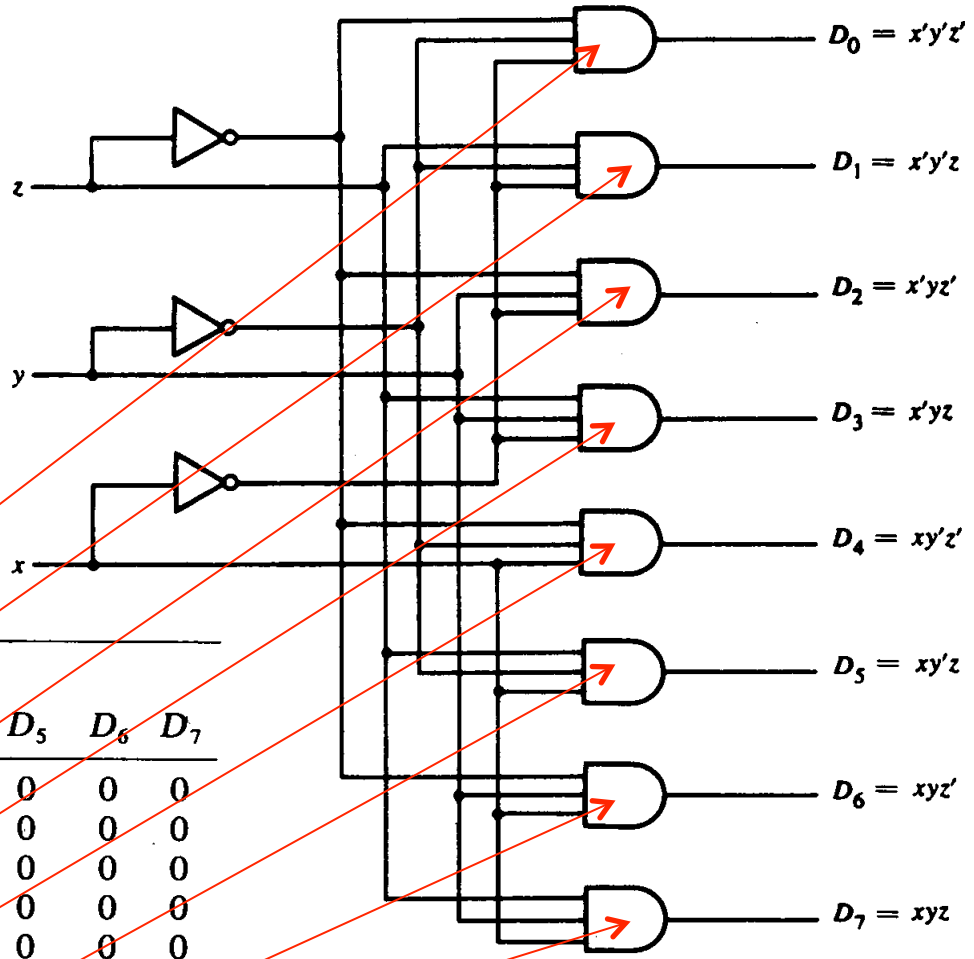
Σύνθετα Συνδυαστικά Κυκλώματα



Η ολοκληρωμένη σχεδίαση συνδυαστικών κυκλωμάτων απαιτεί τυχαία και σύνθετη λογική

Αποκωδικοποιητής (Decoder)

Αποκωδικοποιητής: κύκλωμα που μετατρέπει τη δυαδική πληροφορία των n γραμμών εισόδου σε εως 2^n μοναδικές γραμμές εξόδου (ελαχιστόροι n μεταβλητών).

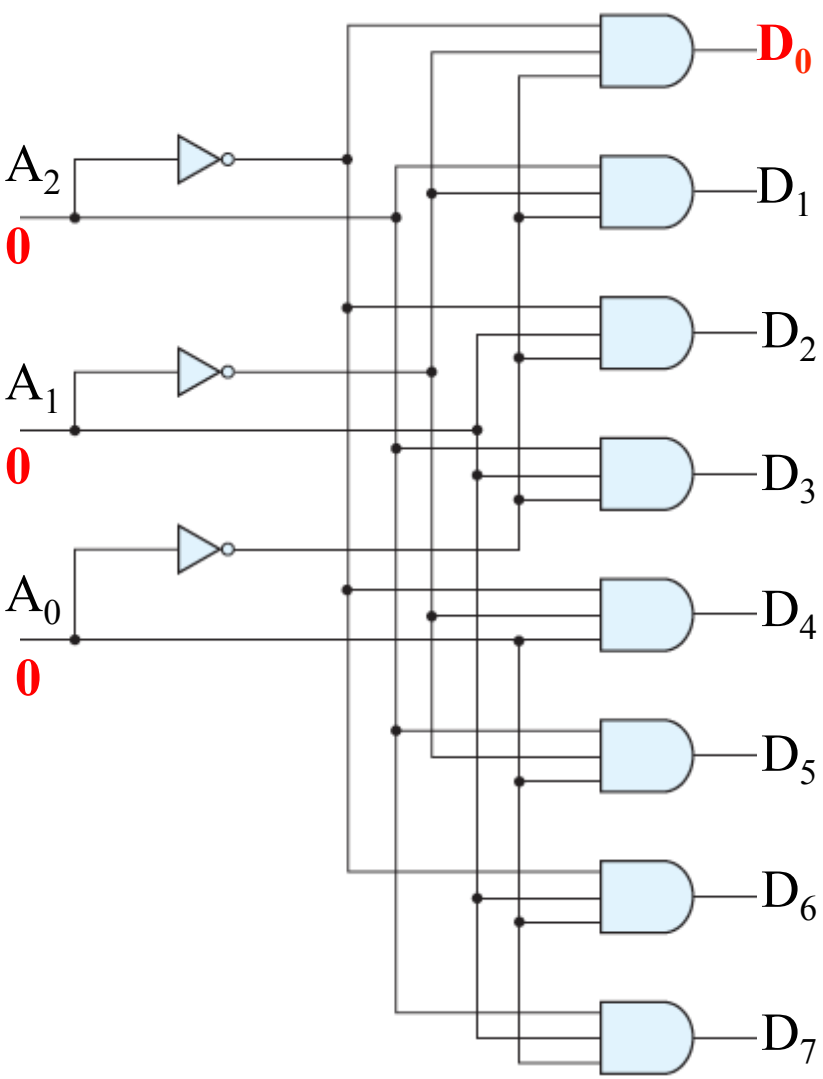


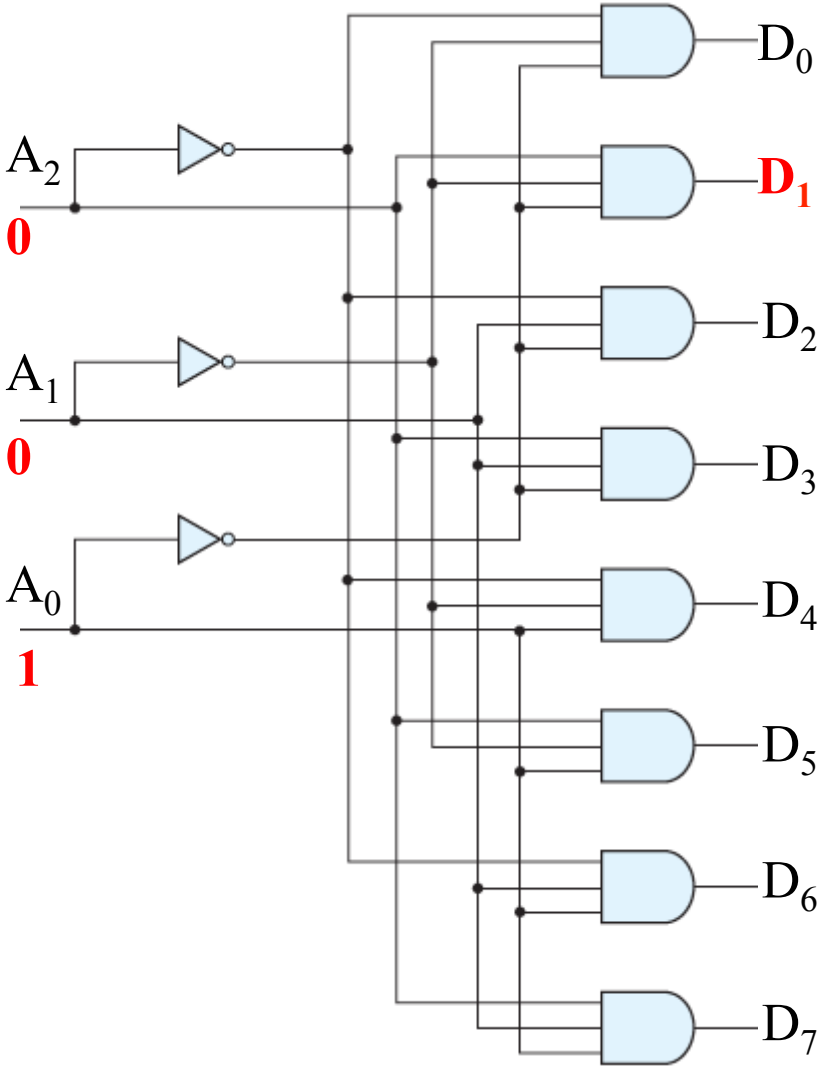
Παράδειγμα:

Αποκωδικοποιητής 3-σε-8

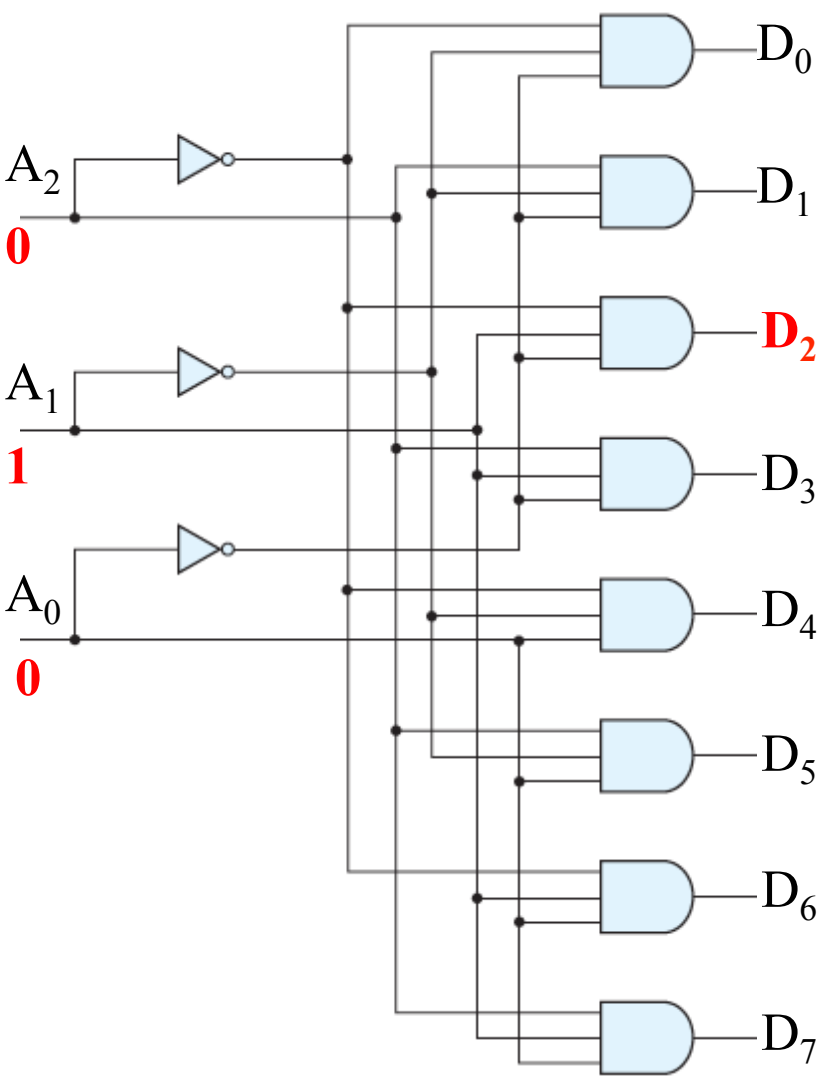
Είσοδοι			Έξοδοι							
x	y	z	D_0	D_1	D_2	D_3	D_4	D_5	D_6	D_7
0	0	0	1	0	0	0	0	0	0	0
0	0	1	0	1	0	0	0	0	0	0
0	1	0	0	0	1	0	0	0	0	0
0	1	1	0	0	0	1	0	0	0	0
1	0	0	0	0	0	0	1	0	0	0
1	0	1	0	0	0	0	0	1	0	0
1	1	0	0	0	0	0	0	0	1	0
1	1	1	0	0	0	0	0	0	0	1

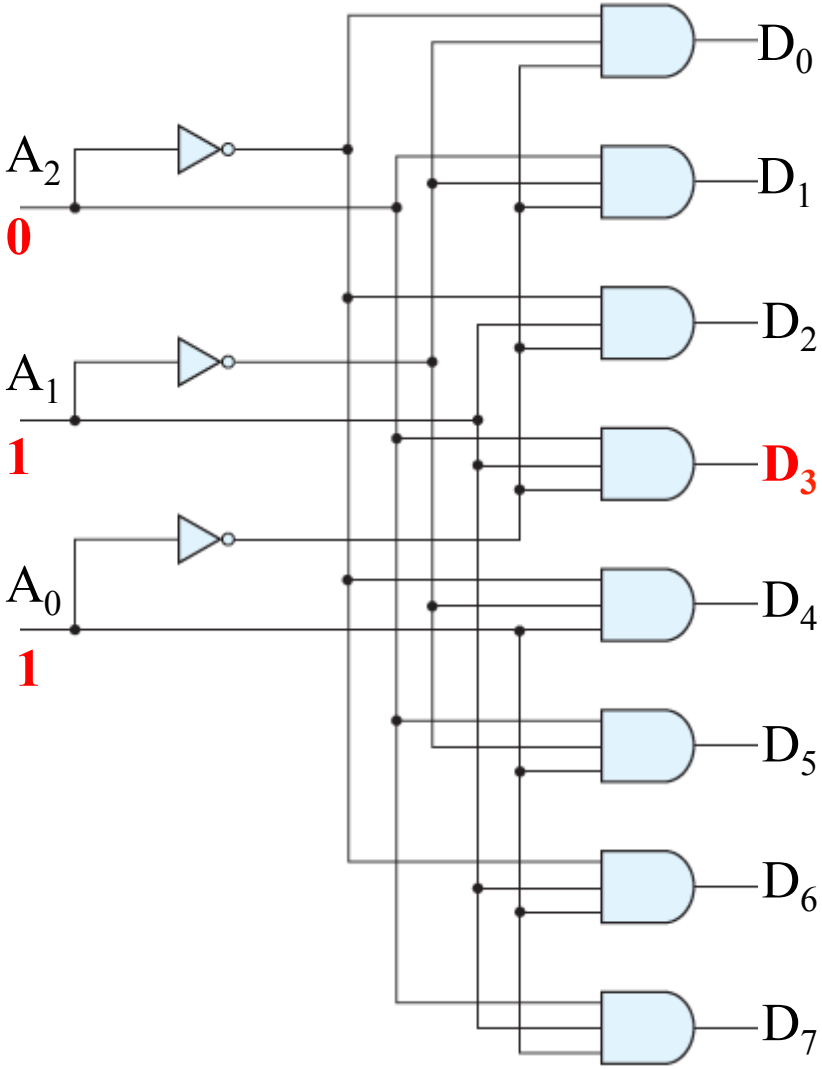
Αποκωδικοποιητές



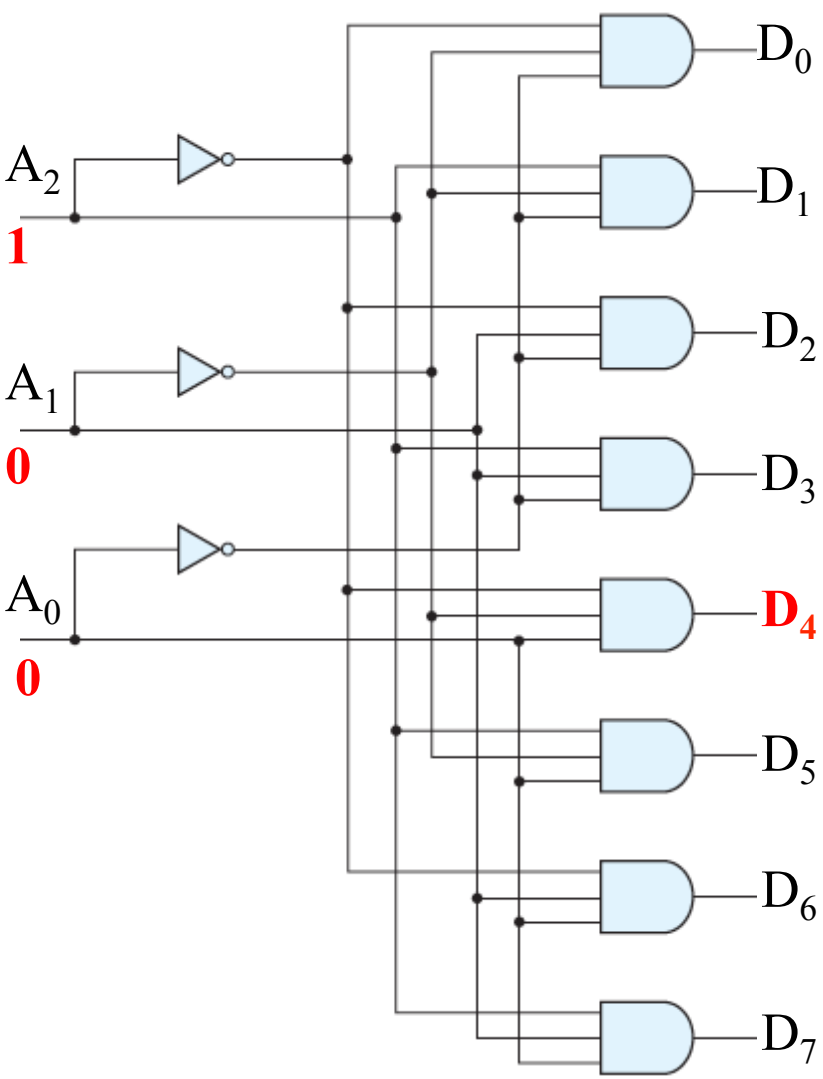


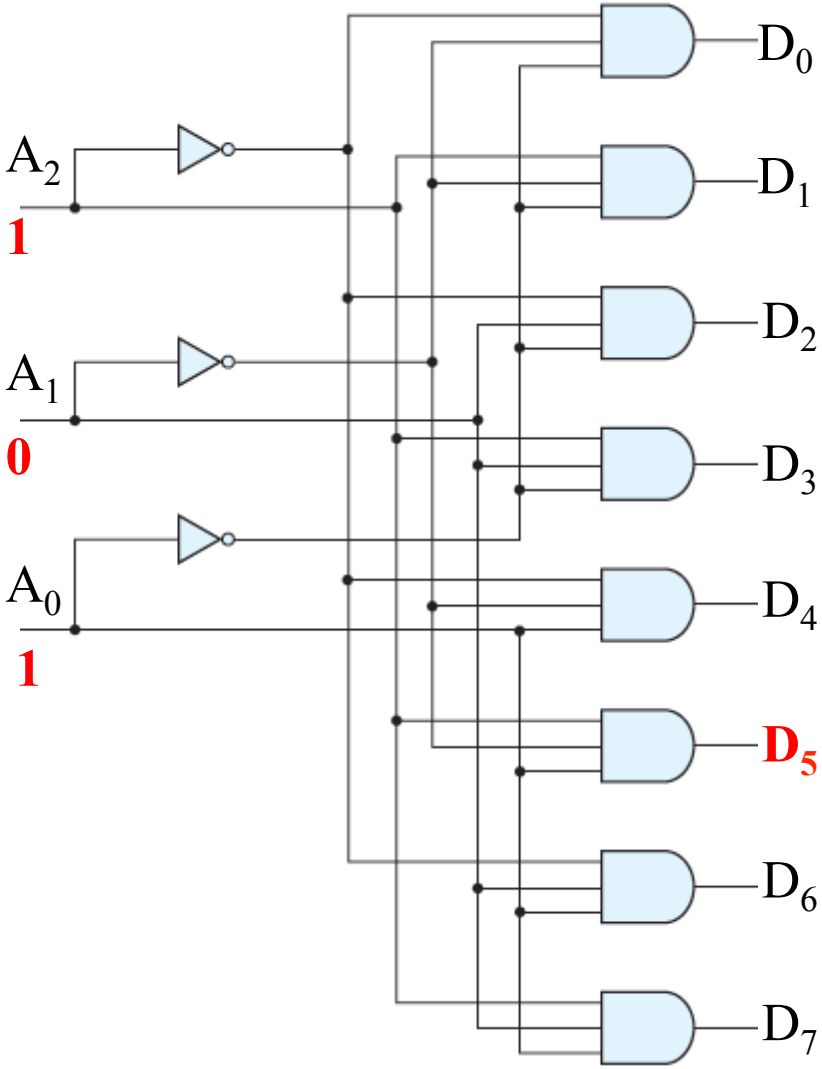
Αποκωδικοποιητές



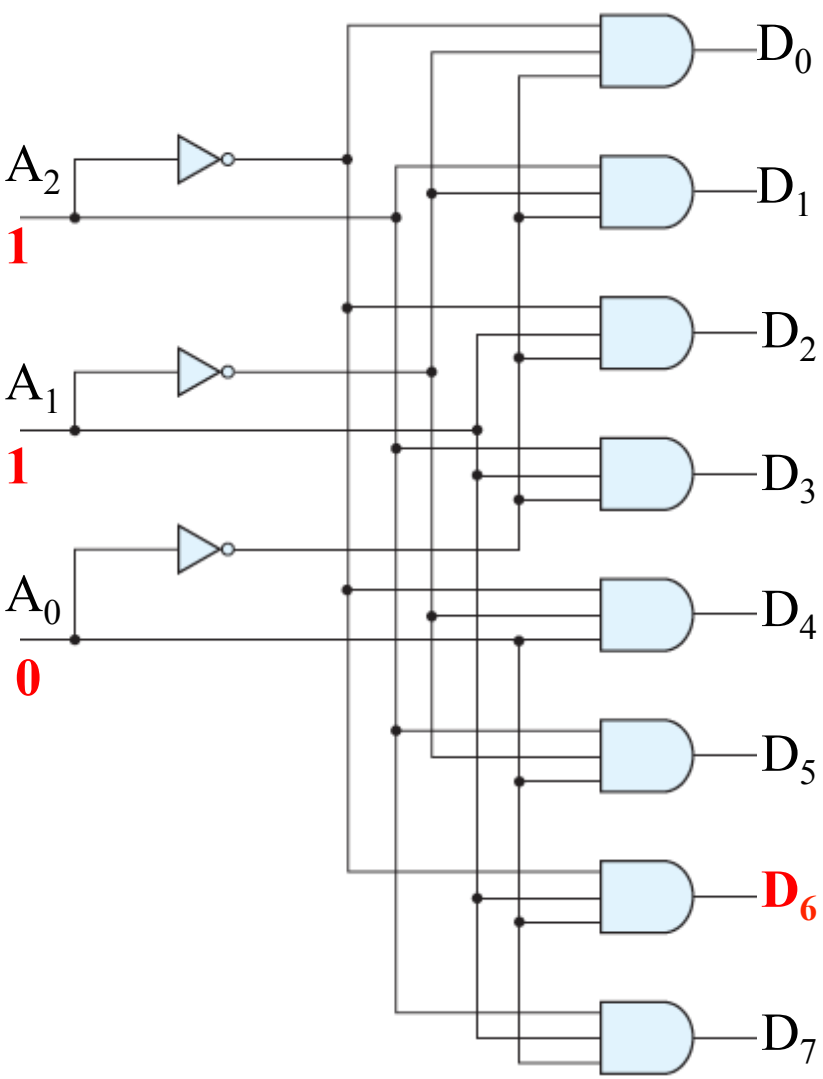


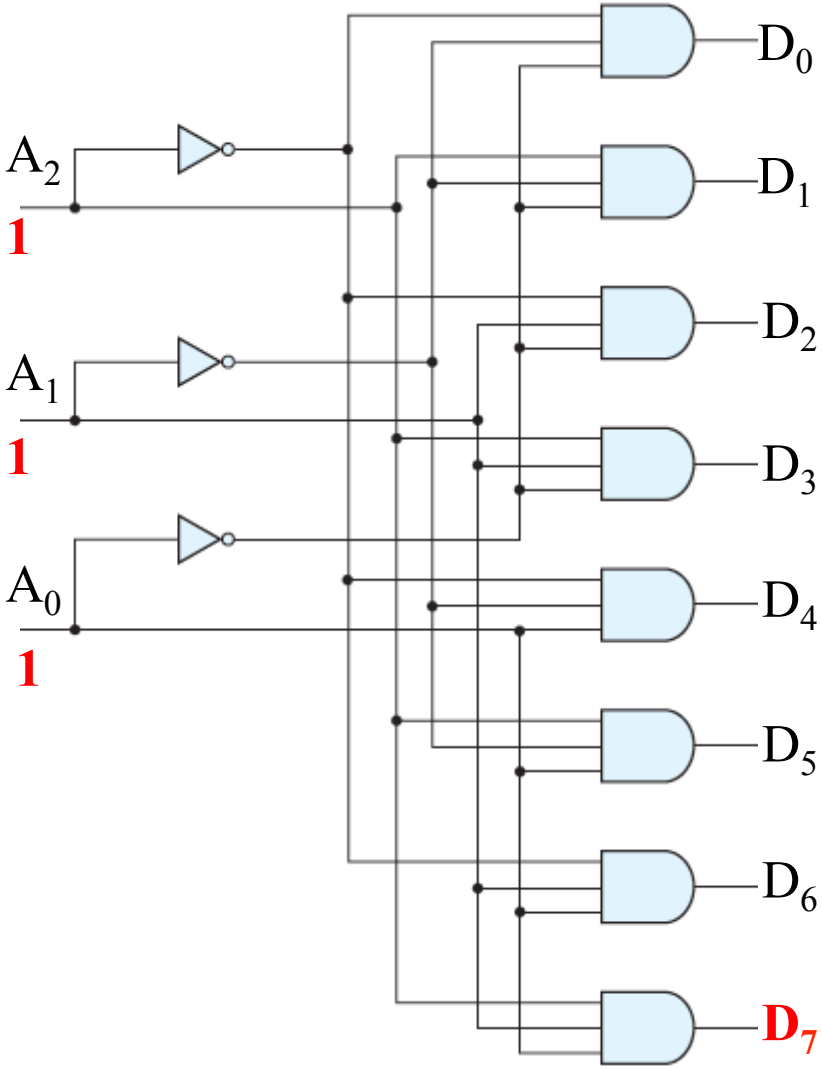
Αποκωδικοποιητές





Αποκωδικοποιητές





Υλοποίηση Συνδυαστικής Λογικής

- Αφού ο αποκωδικοποιητής παράγει τους 2^n ελαχιστόρους μπορεί να χρησιμοποιηθεί για να υλοποιήσει οποιαδήποτε συνάρτηση.

- Κάθε συνδυαστικό κύκλωμα με n εισόδους και m εξόδους μπορεί να υλοποιηθεί με έναν αποκωδικοποιητή n-σε-2^n γραμμών και m πύλες Η.

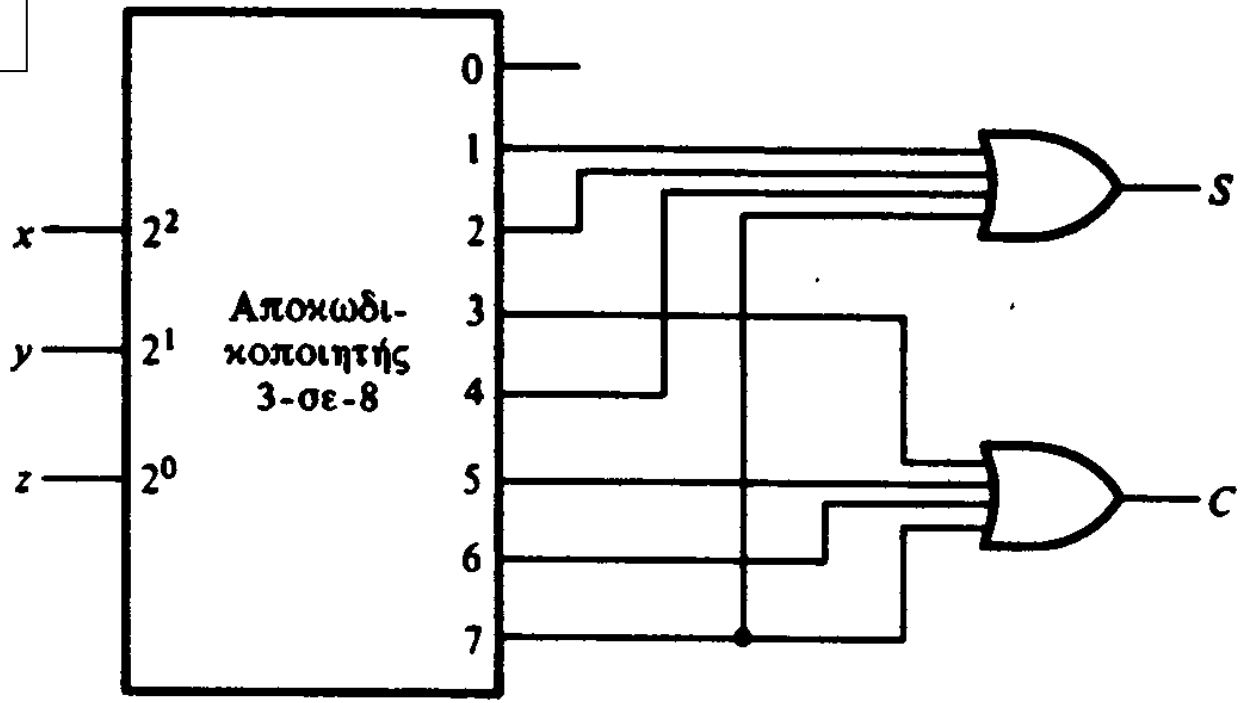
- Εάν ο αριθμός των ελαχιστόρων μιας συνάρτησης είναι μεγαλύτερος από τους μισούς ($2^n/2$), τότε μπορούμε να χρησιμοποιήσουμε μία πύλη ΟΥΤΕ για να αθροίσουμε τους ελαχιστόρους της F΄. Η έξοδος της πύλης ΟΥΤΕ δίνει τη συνάρτηση F.

Παράδειγμα

Υλοποίηση πλήρους αθροιστή με αποκωδικοποιητή.

$S(x,y,z) = \Sigma(1,2,4,7)$

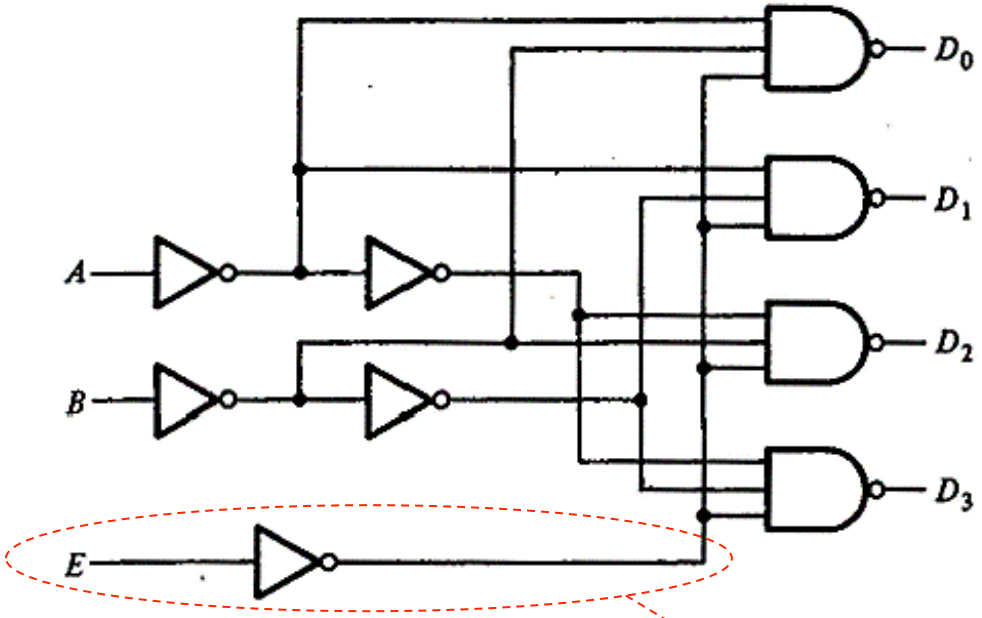
$C(x,y,z) = \Sigma(3,5,6,7)$



Αποκωδικοποιητής με Είσοδο Επίτρεψης

Ο αποκωδικοποιητής μπορεί να παράγει συμπληρωματικές εξόδους.

Ο αποκωδικοποιητής μπορεί να έχει είσοδο επίτρεψης.



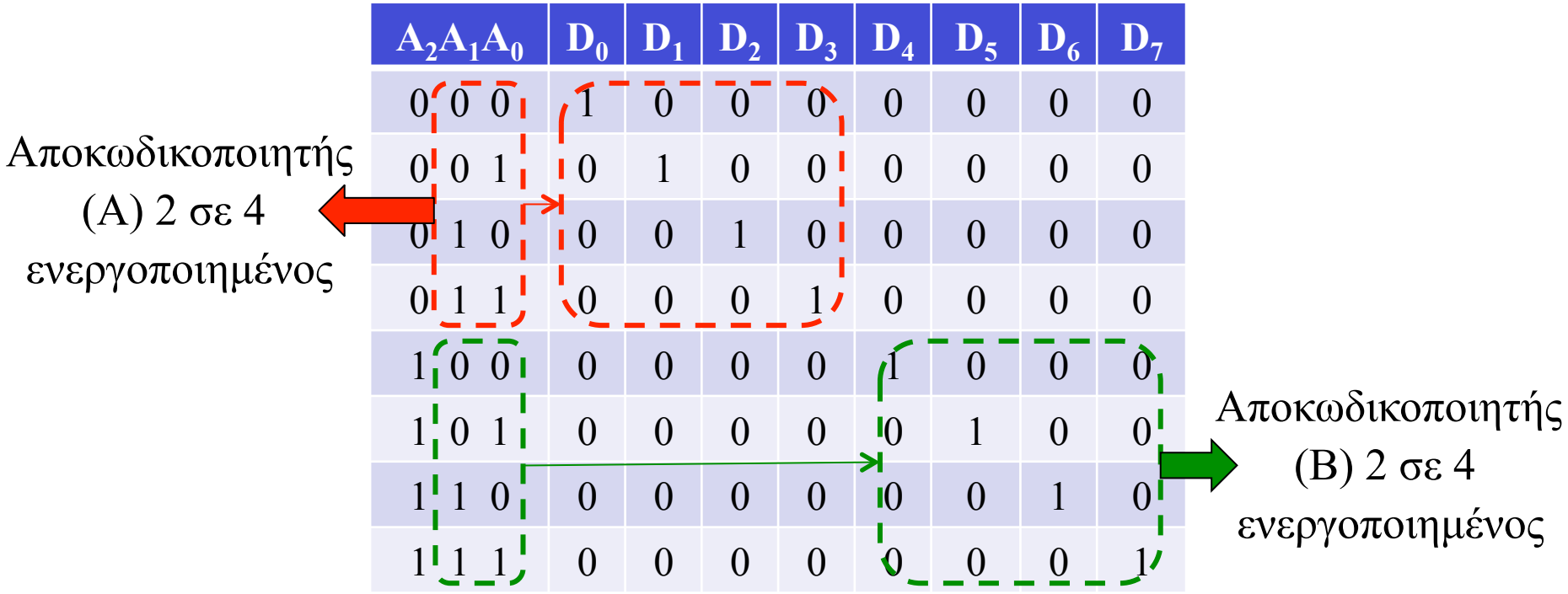
(α) Λογικό διάγραμμα

E	A	B	D_0	D_1	D_2	D_3
1	X	X	1	1	1	1
0	0	0	0	1	1	1
0	0	1	1	0	1	1
0	1	0	1	1	0	1
0	1	1	1	1	1	0

(β) Πίνακας αληθείας

Χρησιμοποιείται για λόγους επέκτασης

Αποκωδικοποιητής 3 σε 8 από δύο 2 σε 4



$A_2A_1A_0$	D_0	D_1	D_2	D_3	D_4	D_5	D_6	D_7
0 0 0	1	0	0	0	0	0	0	0
0 0 1	0	1	0	0	0	0	0	0
0 1 0	0	0	1	0	0	0	0	0
0 1 1	0	0	0	1	0	0	0	0
1 0 0	0	0	0	0	1	0	0	0
1 0 1	0	0	0	0	0	1	0	0
1 1 0	0	0	0	0	0	0	1	0
1 1 1	0	0	0	0	0	0	0	1

Η επιλογή της εξόδου γίνεται και στους δύο αποκωδικοποιητές με τις εισόδους A_1A_0.

Αποκωδικοποιητής 3 σε 8 από δύο 2 σε 4

$A_2A_1A_0$	D_0	D_1	D_2	D_3	D_4	D_5	D_6	D_7
0 0 0	1	0	0	0	0	0	0	0
0 0 1	0	1	0	0	0	0	0	0
0 1 0	0	0	1	0	0	0	0	0
0 1 1	0	0	0	1	0	0	0	0
1 0 0	0	0	0	0	1	0	0	0
1 0 1	0	0	0	0	0	1	0	0
1 1 0	0	0	0	0	0	0	1	0
1 1 1	0	0	0	0	0	0	0	1

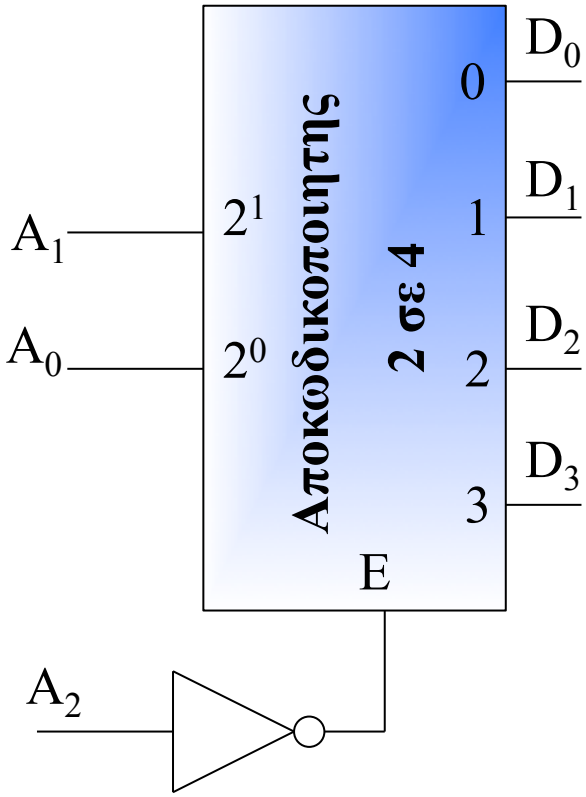
Αποκωδικοποιητής (B) 2 σε 4 απενεργοποιημένος

Αποκωδικοποιητής (A) 2 σε 4 απενεργοποιημένος

Η είσοδος A_2 ενεργοποιεί τον έναν από τους δύο αποκωδικοποιητές.

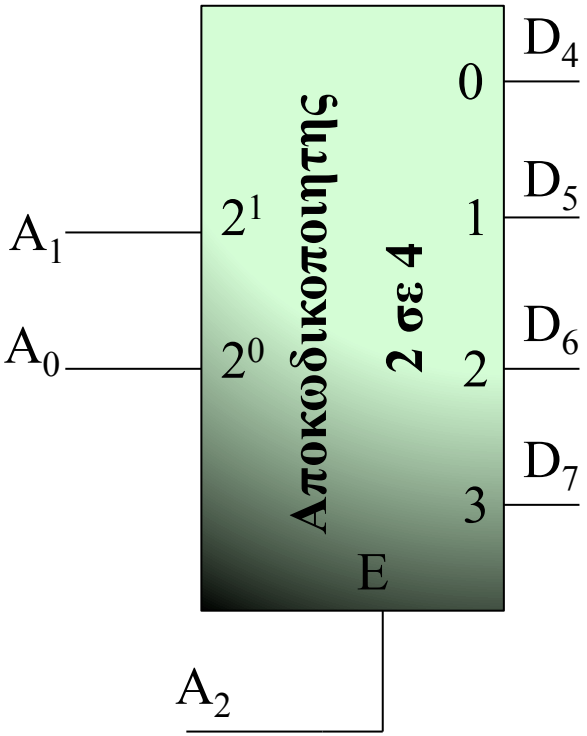
Αποκωδικοποιητής Α

$A_2A_1A_0$	D_0	D_1	D_2	D_3	D_4	D_5	D_6	D_7
0 0 0	1	0	0	0	0	0	0	0
0 0 1	0	1	0	0	0	0	0	0
0 1 0	0	0	1	0	0	0	0	0
0 1 1	0	0	0	1	0	0	0	0
1 0 0	0	0	0	0	1	0	0	0
1 0 1	0	0	0	0	0	1	0	0
1 1 0	0	0	0	0	0	0	1	0
1 1 1	0	0	0	0	0	0	0	1

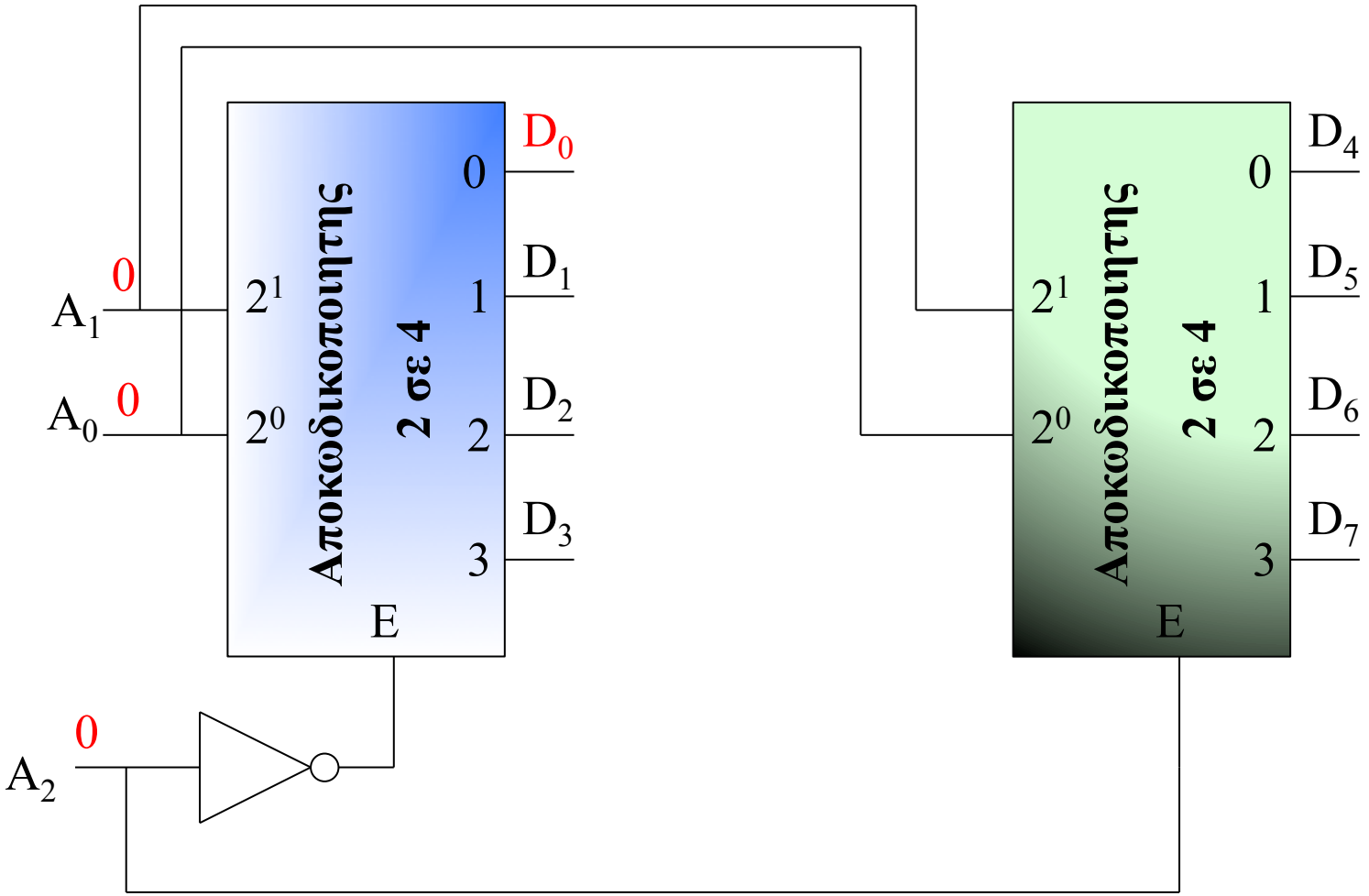


Αποκωδικοποιητής Β

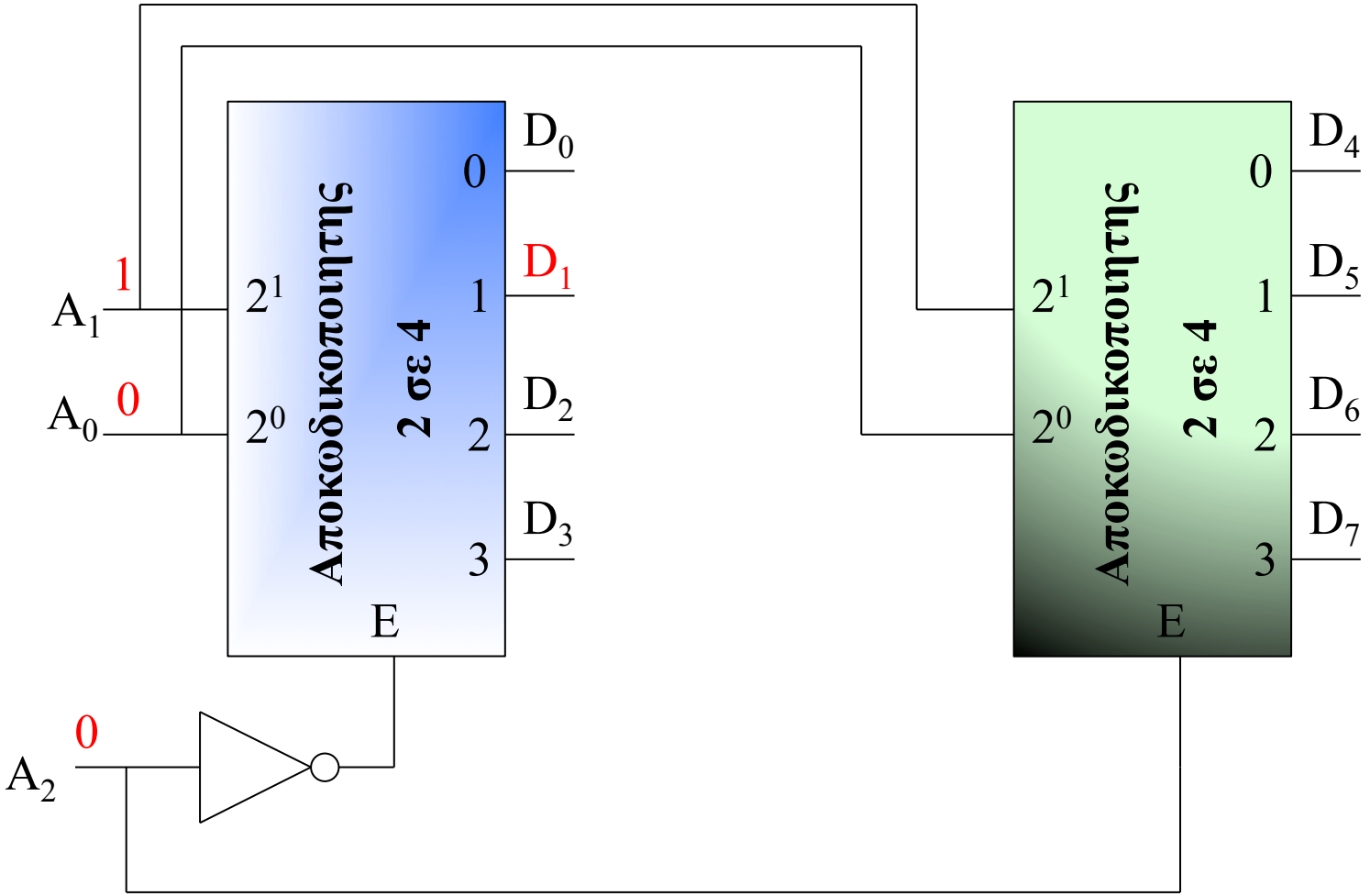
$A_2A_1A_0$	D_0	D_1	D_2	D_3	D_4	D_5	D_6	D_7
0 0 0	1	0	0	0	0	0	0	0
0 0 1	0	1	0	0	0	0	0	0
0 1 0	0	0	1	0	0	0	0	0
0 1 1	0	0	0	1	0	0	0	0
1 0 0	0	0	0	0	1	0	0	0
1 0 1	0	0	0	0	0	1	0	0
1 1 0	0	0	0	0	0	0	1	0
1 1 1	0	0	0	0	0	0	0	1



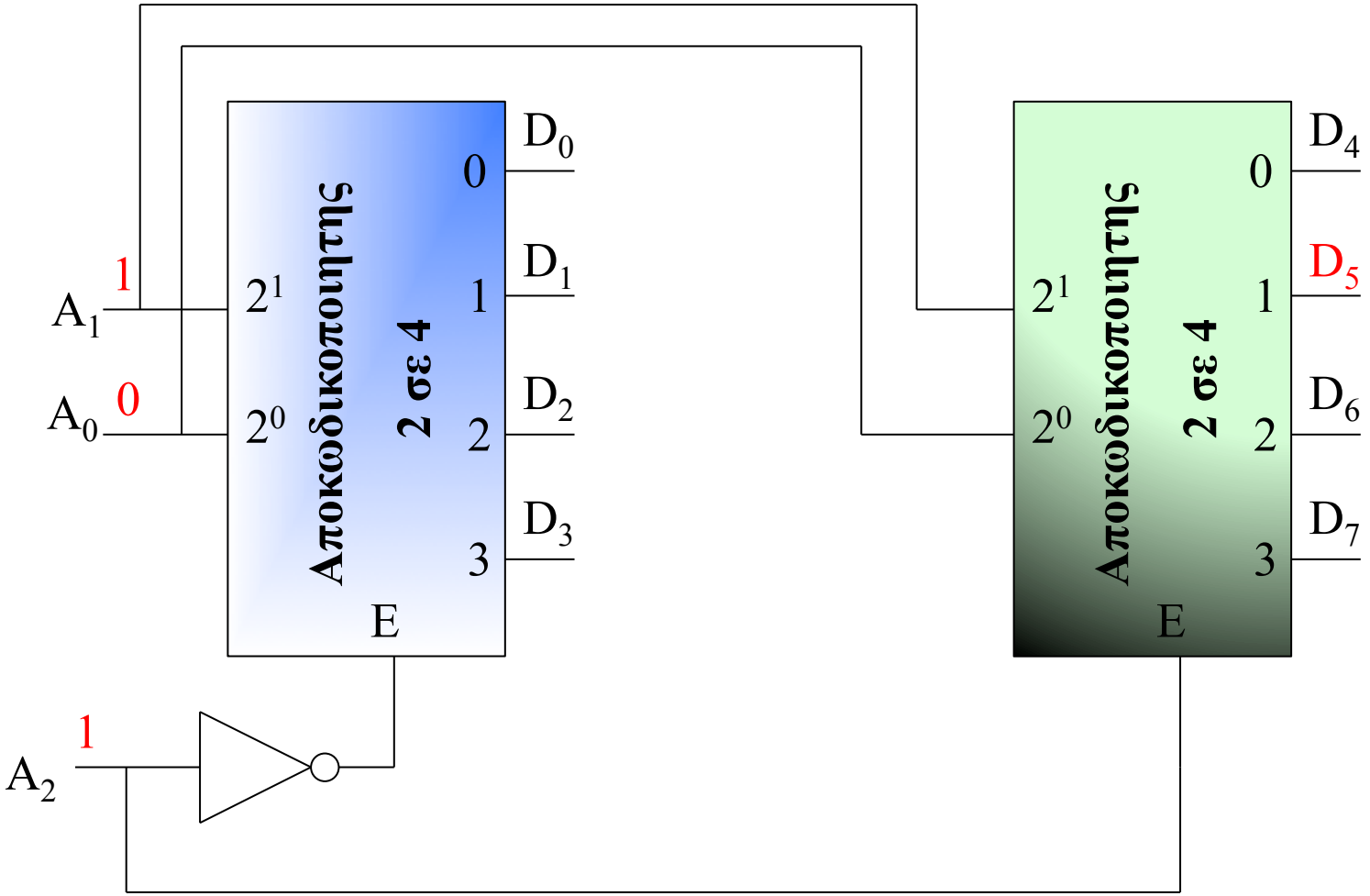
Αποκωδικοποιητής 3 σε 8



Αποκωδικοποιητής 3 σε 8

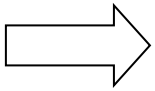


Αποκωδικοποιητής 3 σε 8

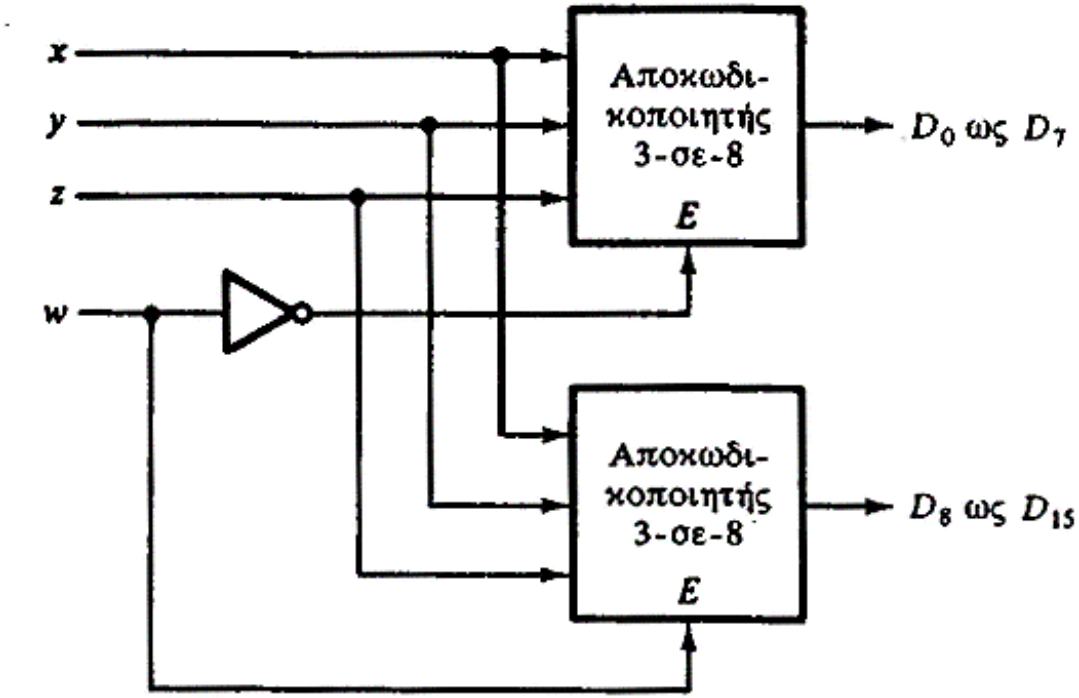


Αποκωδικοποιητής/Αποπλέκτης

Επέκταση αποκωδικοποιητή με χρήση πολλών αποκωδικοποιητών

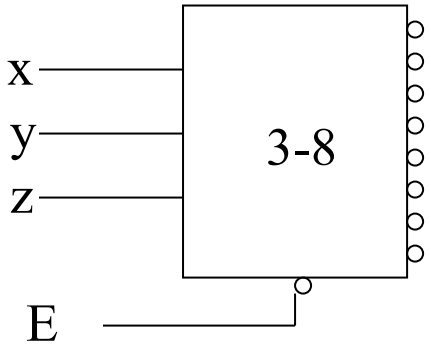


2 αποκωδικοποιητές 3 σε 8 δίνουν 1 αποκωδικοποιητή 4 σε 16



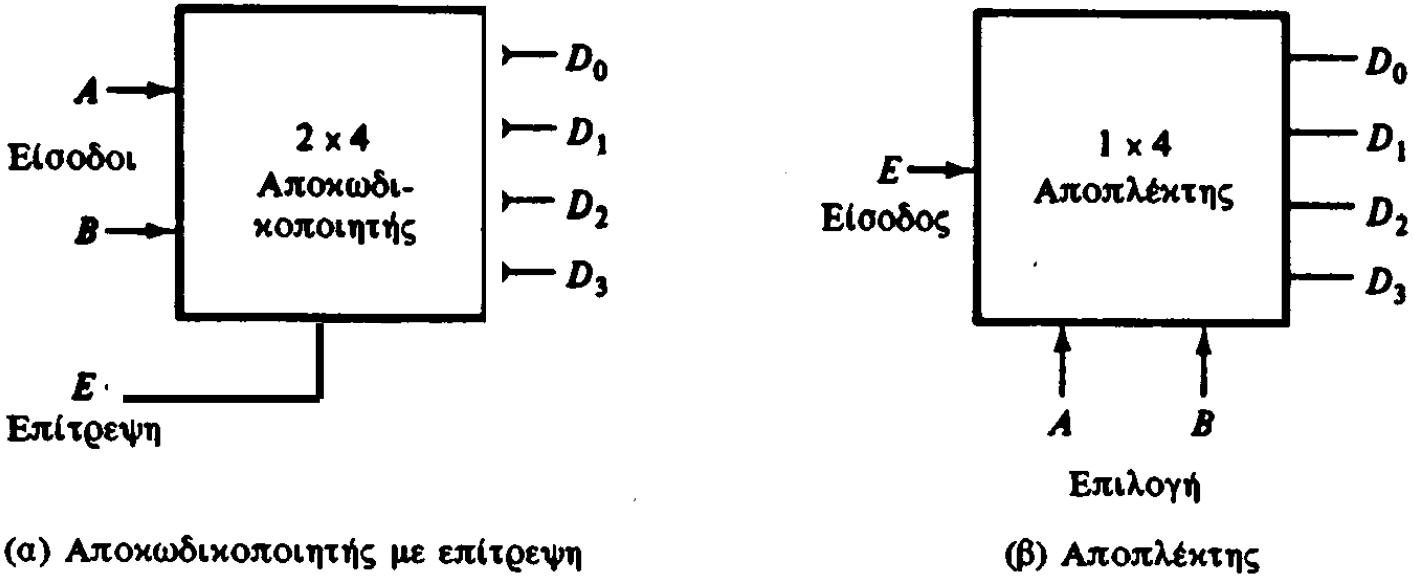
Συμβολισμοί:

Το κυκλάκι δείχνει σε ποια τιμή μία είσοδος/έξοδος είναι ενεργή.



Αποπλέκτης (Demultiplexer)

Ο αποπλέκτης δέχεται πληροφορίες από μία απλή γραμμή και τις μεταβιβάζει σε μία από τις 2^n δυνατές γραμμές εξόδου ανάλογα με τις τιμές των n γραμμών επιλογής.



(α) Αποκωδικοποιητής με επίτρεψη

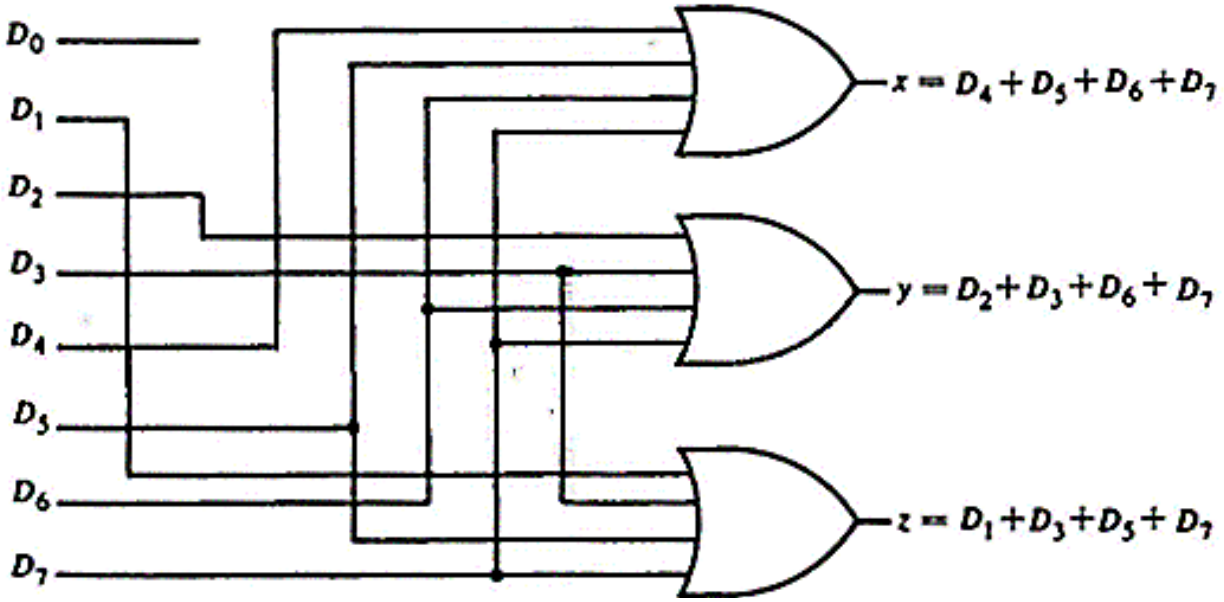
(β) Αποπλέκτης

Κωδικοποιητής από Οκταδικό σε Δυαδικό

Ο Κωδικοποιητής εκτελεί την αντίστροφη λειτουργία από τον Αποκωδικοποιητή: Έχει 2^n γραμμές εισόδου και n γραμμές εξόδου και δίνει στην έξοδο τον δυαδικό κώδικα που αντιστοιχεί στις γραμμές εισόδου.

Είσοδοι								Έξοδοι		
D_0	D_1	D_2	D_3	D_4	D_5	D_6	D_7	x	y	z
1	0	0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0	0	1
0	0	1	0	0	0	0	0	0	1	0
0	0	0	1	0	0	0	0	0	1	1
0	0	0	0	1	0	0	0	1	0	0
0	0	0	0	0	1	0	0	1	0	1
0	0	0	0	0	0	1	0	1	1	0
0	0	0	0	0	0	0	1	1	1	1

Κωδικοποιητής (Encoder)



Προβλήματα:

- Όταν περισσότερες της μίας είσοδοι είναι 1 τότε η έξοδος είναι απροσδιόριστη. (Λύση: προτεραιότητα)
- Όταν όλες οι είσοδοι είναι 0 τότε η έξοδος είναι 0 που δεν είναι σωστό αφού η $D_0 \neq 1$. (Λύση: διάκριση της κατάστασης)

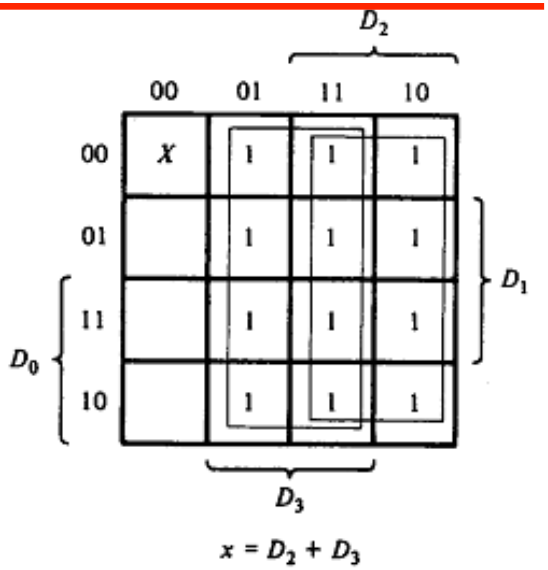
Κωδικοποιητής Προτεραιότητας

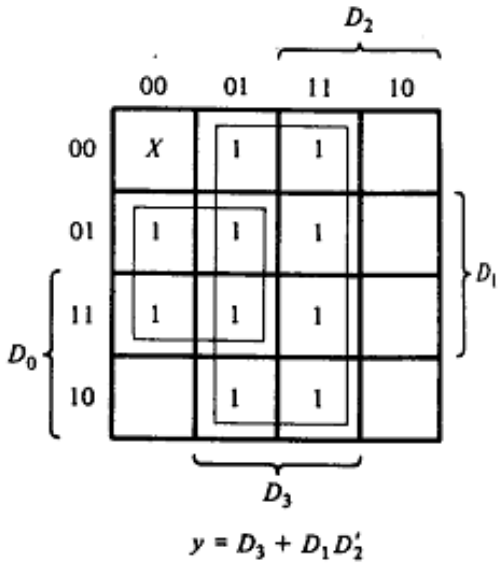
Το πρόβλημα προέρχεται από τις αδιάφορες καταστάσεις.

Ο κωδικοποιητής προτεραιότητας είναι ένα κύκλωμα κωδικοποιητή που περιλαμβάνει συνάρτηση προτεραιότητας και καθορίζει όλες τις αδιάφορες καταστάσεις.

Παράδειγμα: Κωδικοποιητής προτεραιότητας 4 εισόδων

Είσοδοι				Έξοδοι		
D_0	D_1	D_2	D_3	X	Y	V
0	0	0	0	X	X	0
1	0	0	0	0	0	1
X	1	0	0	0	1	1
X	X	1	0	1	0	1
X	X	X	1	1	1	1



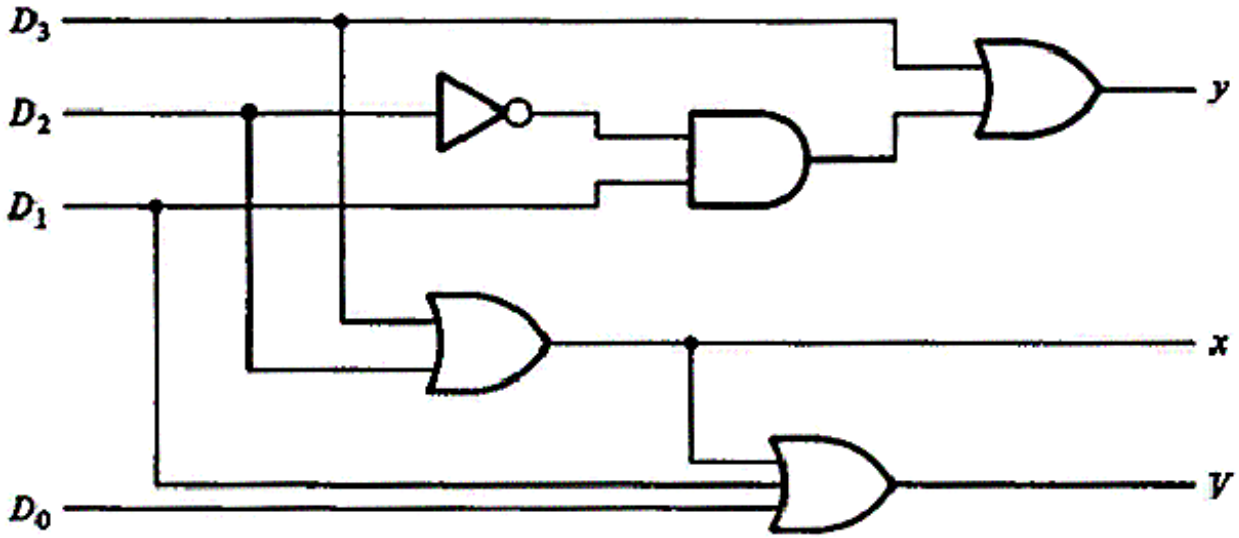


Κωδικοποιητής Προτεραιότητας 4 Εισόδων

$x = D_2 + D_3$

$y = D_3 + D_1 D_2'$

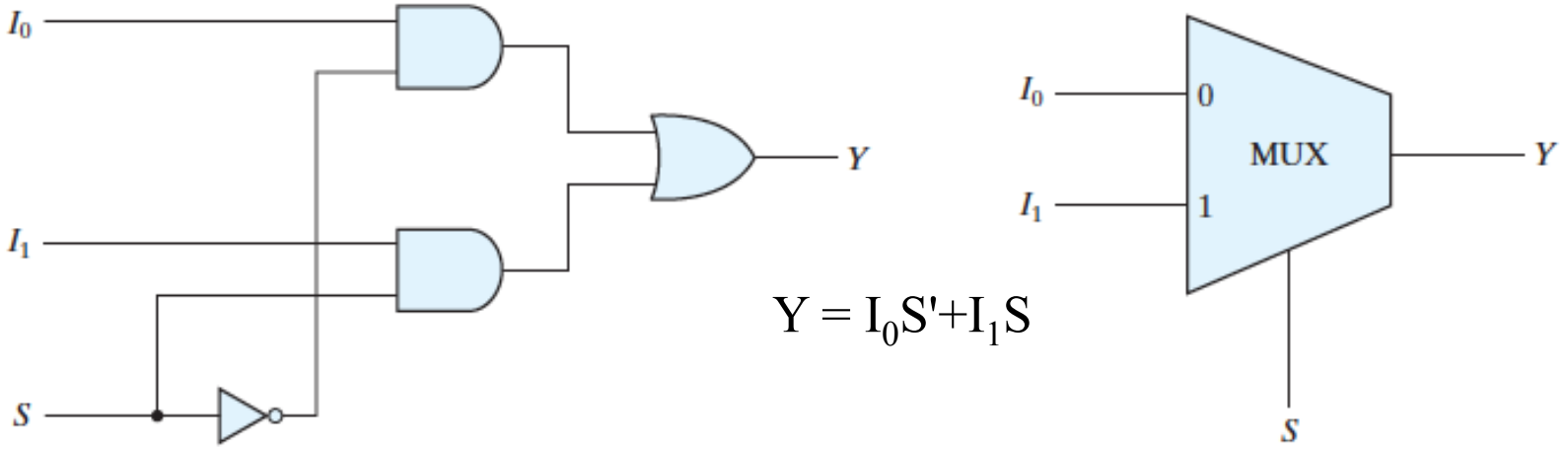
$V = D_0 + D_1 + D_2 + D_3$



Λύνει το πρόβλημα της επιλογής όταν περισσότερες της μίας εισόδων είναι 1 επιλέγοντας αυτή με τη μεγαλύτερη προτεραιότητα.

Πολυπλέκτης (Multiplexer)

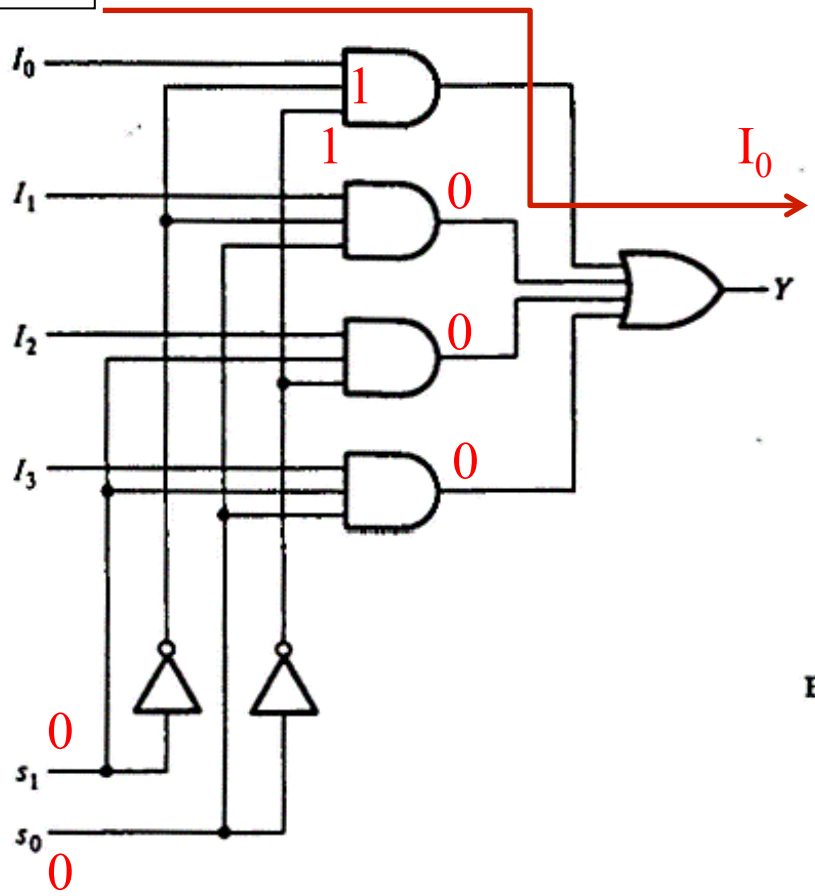
- Ο πολυπλέκτης **επιλέγει** δυαδικές πληροφορίες από πολλές γραμμές εισόδου και τις κατευθύνει σε **μία γραμμή εξόδου**.
- Η επιλογή της μιας συγκεκριμένης γραμμής εισόδου γίνεται μέσω μερικών **γραμμών επιλογής**.
- Ένας πολυπλέκτης 2^n-σε-1 γραμμή κατασκευάζεται από έναν αποκωδικοποιητή n-σε-2^n προσθέτοντας σε αυτόν 2^n εισόδους μια για κάθε πύλη ΚΑΙ.



$$Y = I_0S' + I_1S$$

Παράδειγμα

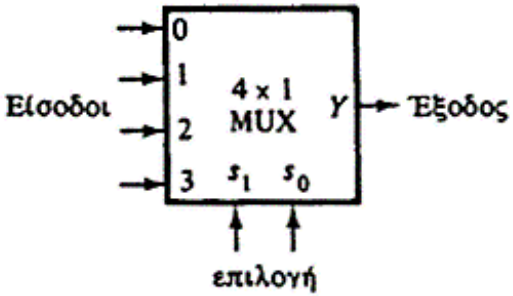
Πολυπλέκτης 4-σε-1



(α) Λογικό διάγραμμα

s_1	s_0	Y
0	0	I_0
0	1	I_1
1	0	I_2
1	1	I_3

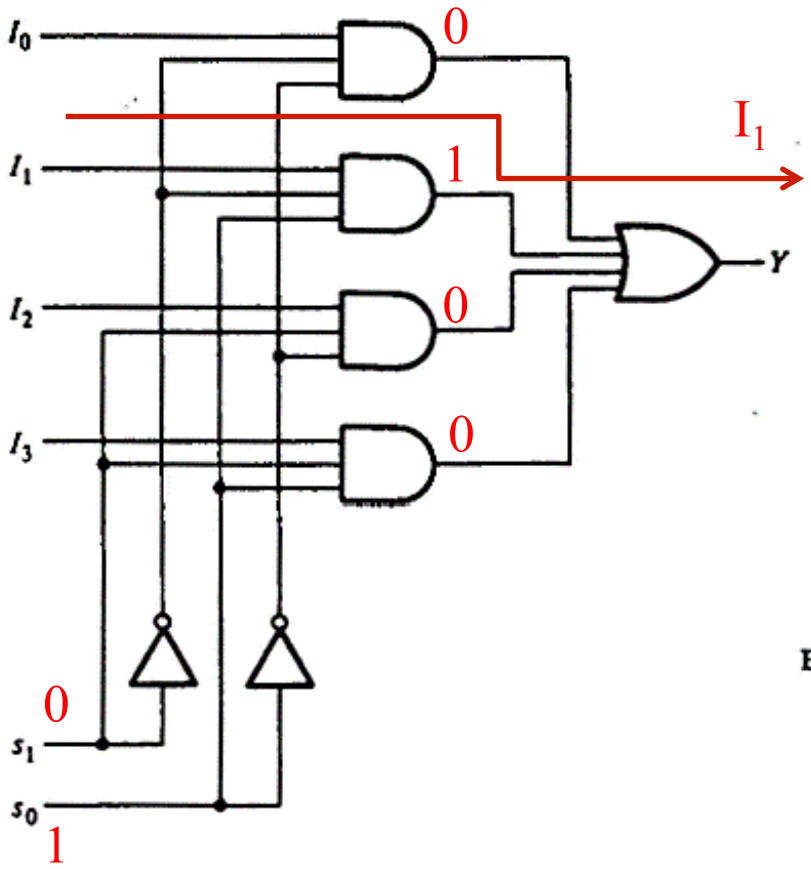
(β) Πίνακας της συνάρτησης



(γ) Σχηματικό διάγραμμα

Παράδειγμα

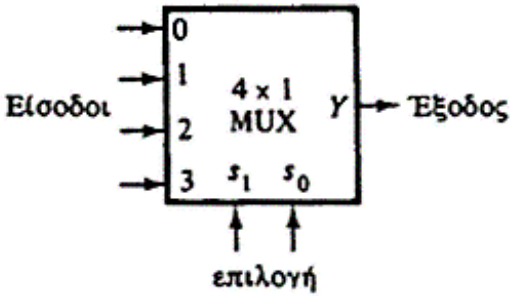
Πολυπλέκτης 4-σε-1



(α) Λογικό διάγραμμα

s_1	s_0	Y
0	0	I_0
0	1	I_1
1	0	I_2
1	1	I_3

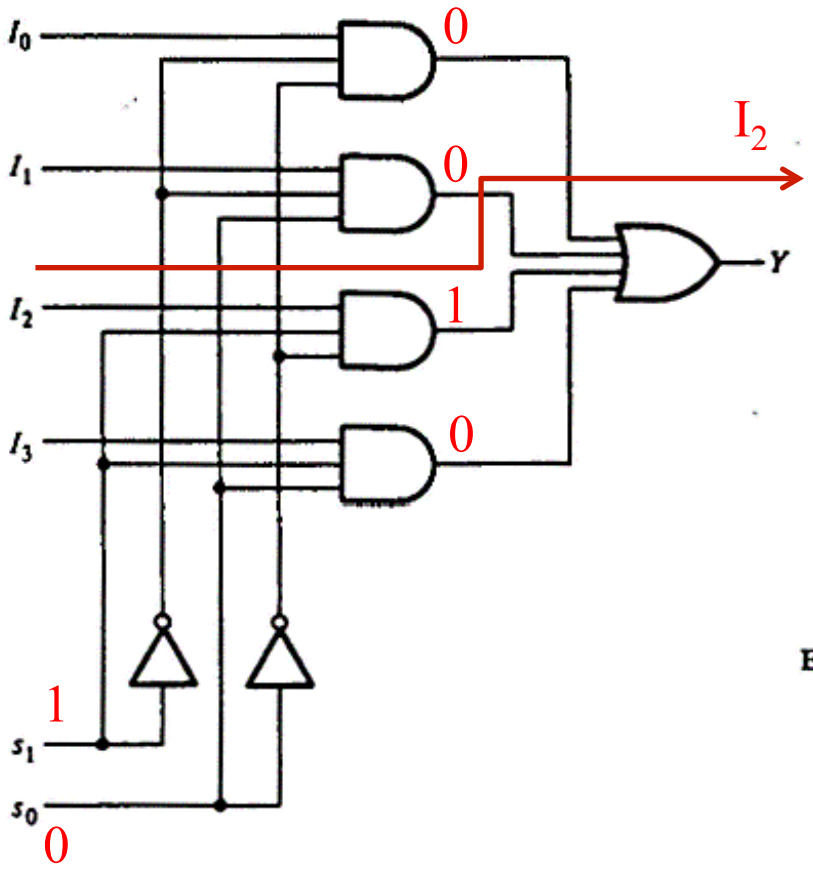
(β) Πίνακας της συνάρτησης



(γ) Σχηματικό διάγραμμα

Παράδειγμα

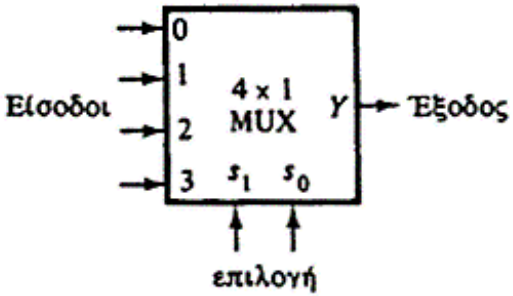
Πολυπλέκτης 4-σε-1



(α) Λογικό διάγραμμα

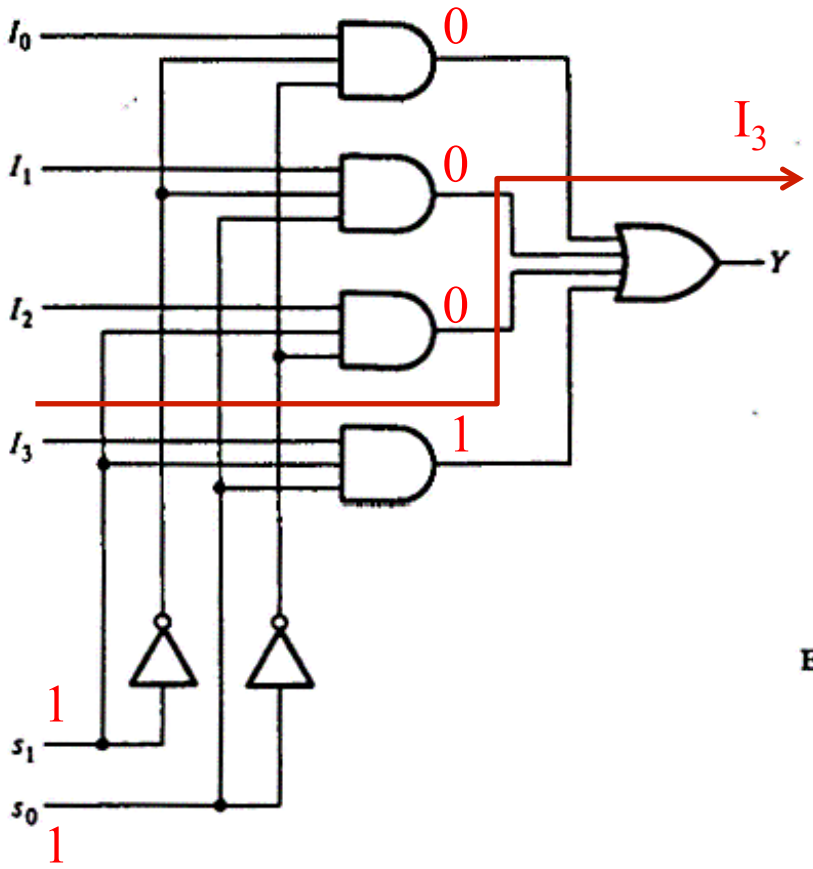
s_1	s_0	Y
0	0	I_0
0	1	I_1
1	0	I_2
1	1	I_3

(β) Πίνακας της συνάρτησης



(γ) Σχηματικό διάγραμμα

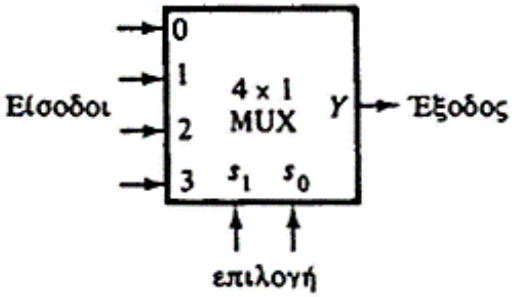
Πολυπλέκτης 4 σε 1



(α) Λογικό διάγραμμα

s_1	s_0	Y
0	0	I_0
0	1	I_1
1	0	I_2
1	1	I_3

(β) Πίνακας της συνάρτησης



(γ) Σχηματικό διάγραμμα

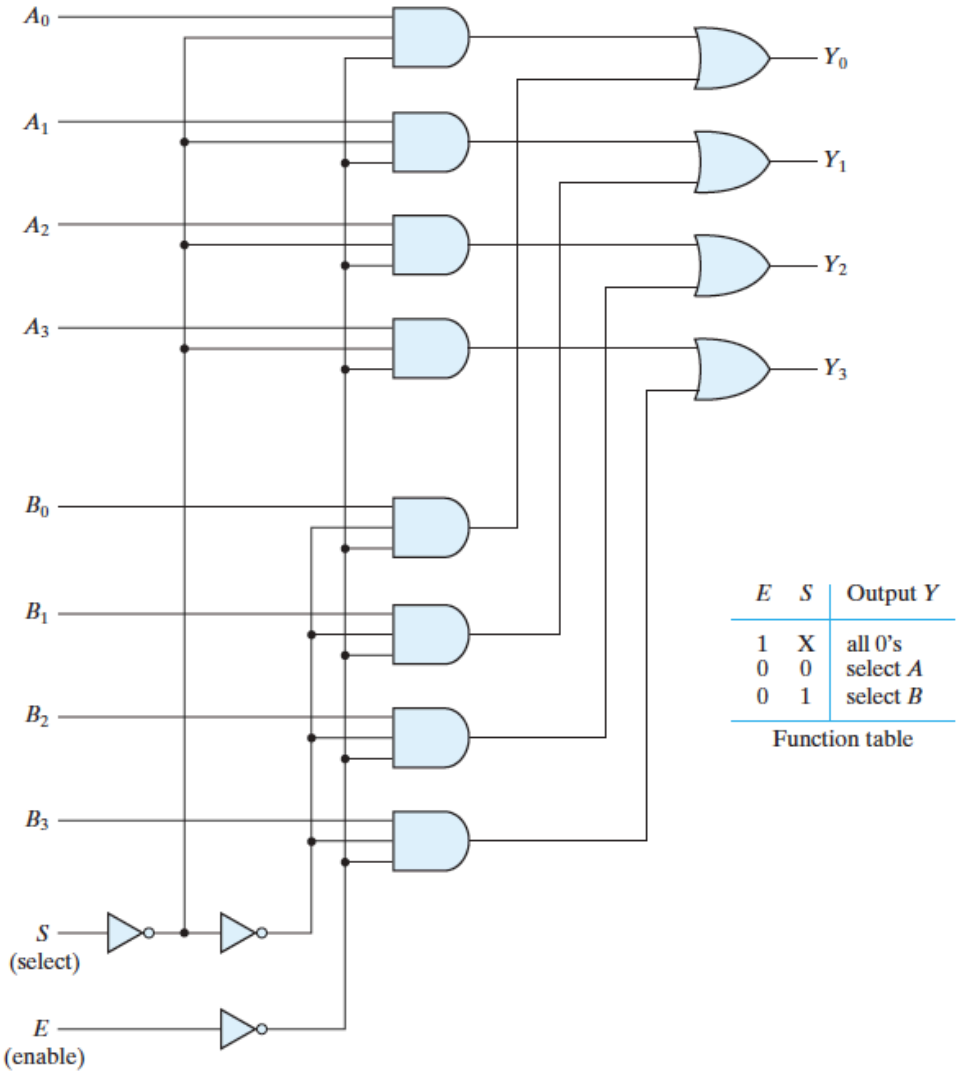
Τετραπλός Πολυπλέκτης 2 σε 1

Η είσοδος Επίτρεψης (E) τοποθετείται για λόγους επέκτασης.

Πολυπλέκτης 8-σε-4

ή

4 Πολυπλέκτες 2-σε-1



E	S	Output Y
1	X	all 0's
0	0	select A
0	1	select B

Function table

Υλοποίηση Συνάρτησης Boole με Πολυπλέκτη

Κάθε πολυπλέκτης 2^n σε 1 μπορεί να υλοποιήσει οποιαδήποτε συνάρτηση n+1 μεταβλητών ως εξής:

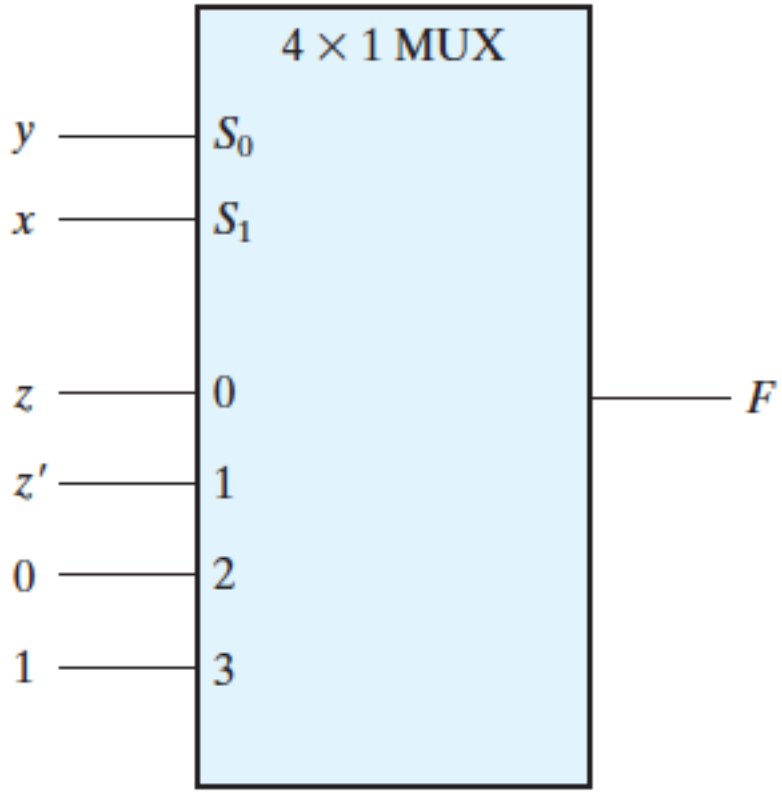
1. Βάζουμε τις n μεταβλητές στις εισόδους επιλογής.

2. Χρησιμοποιούμε την τελευταία μεταβλητή για τις εισόδους.

Παράδειγμα 1

$F(x, y, z) = \Sigma(1, 2, 6, 7)$

x	y	z	F	
0	0	0	0	$F = z$
0	0	1	1	
0	1	0	1	$F = z'$
0	1	1	0	
1	0	0	0	$F = 0$
1	0	1	0	
1	1	0	1	$F = 1$
1	1	1	1	



Υλοποίηση Συναρτήσεων Boole

Αλγόριθμος υλοποίησης συνάρτησης με χρήση πολυπλέκτη:

1. Εκφράζουμε τη συνάρτηση σε άθροισμα ελαχιστόρων.

2. Συνδέουμε τις *n* - 1 μεταβλητές στις γραμμές επιλογής και κρατάμε την αριστερότερη (πιο σημαντική) έστω Α.

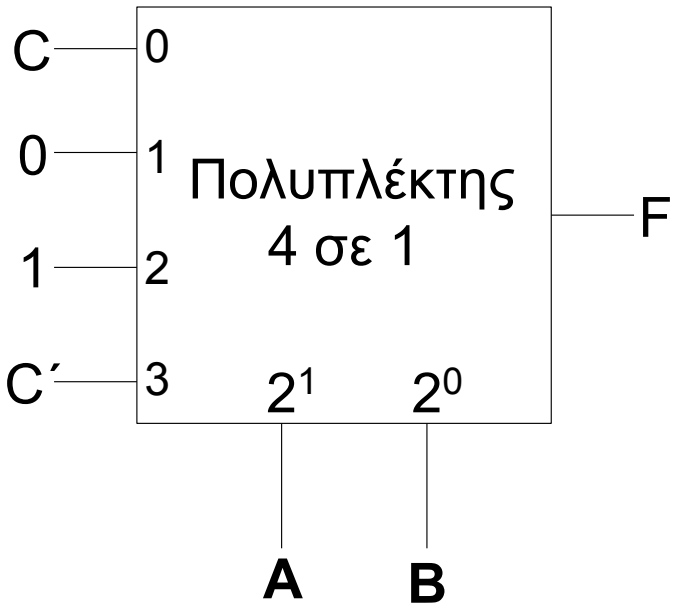
3. Καταγράφουμε τις εισόδους του πολυπλέκτη και κάτω από αυτές όλους τους ελαχιστόρους σε δύο σειρές (αντίστοιχα για Α=0 και Α=1).

4. Σημειώνουμε τους ελαχιστόρους που έχει η συνάρτηση.

5. Σε κάθε στήλη βάζουμε 0 αν δεν έχει σημειωθεί ελαχιστόρος, 1 αν έχουν σημειωθεί και οι δύο, Α΄ αν έχει σημειωθεί ο πάνω και Α αν έχει σημειωθεί ο κάτω ελαχιστόρος.

Παράδειγμα

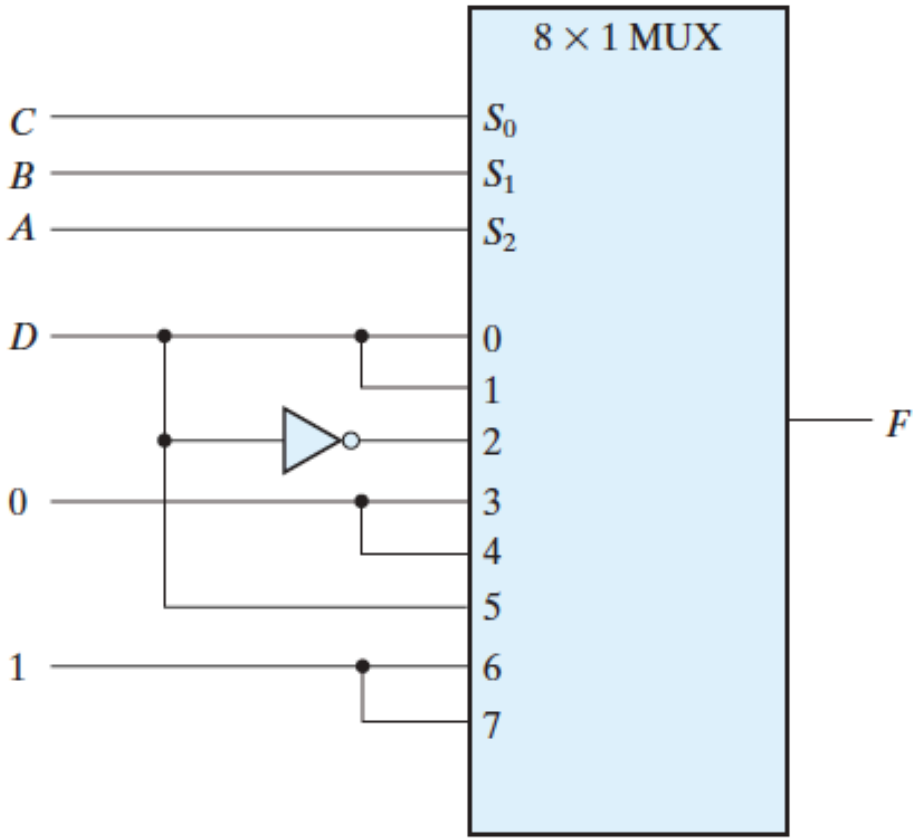
F(A, B, C)=Σ(1,4,5,6)



A	B	C	F	
0	0	0	0	**F = C**
0	0	1	1	
0	1	0	0	**F = 0**
0	1	1	0	
1	0	0	1	**F = 1**
1	0	1	1	
1	1	0	1	**F = C´**
1	1	1	0	

Παράδειγμα

A	B	C	D	F	
0	0	0	0	0	$F = D$
0	0	0	1	1	
0	0	1	0	0	$F = D$
0	0	1	1	1	
0	1	0	0	1	$F = D'$
0	1	0	1	0	
0	1	1	0	0	$F = 0$
0	1	1	1	0	
1	0	0	0	0	$F = 0$
1	0	0	1	0	
1	0	1	0	0	$F = D$
1	0	1	1	1	
1	1	0	0	1	$F = 1$
1	1	0	1	1	
1	1	1	0	1	$F = 1$
1	1	1	1	1	



Σύγκριση

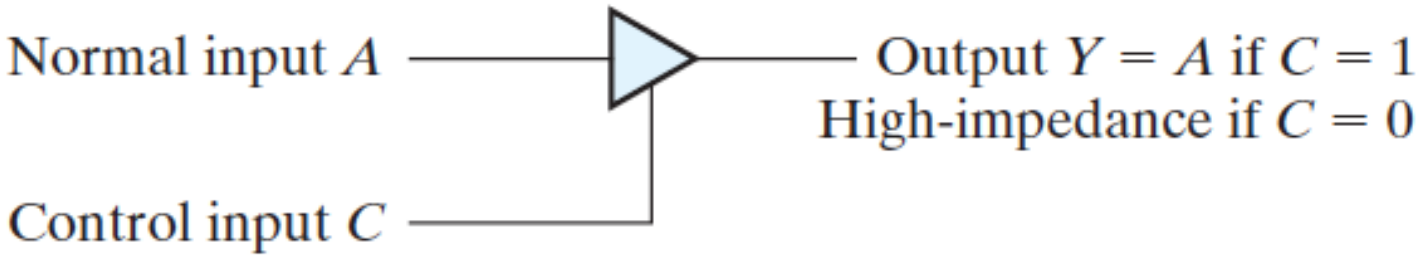
- Η μέθοδος του αποκωδικοποιητή απαιτεί μια πύλη Η για κάθε συνάρτηση εξόδου αλλά μόνο ένας αποκωδικοποιητής απαιτείται για όλες τις συναρτήσεις.

- Η μέθοδος του πολυπλέκτη χρησιμοποιεί μονάδες μικρότερου μεγέθους αλλά χρειάζεται έναν πολυπλέκτη για κάθε συνάρτηση εξόδου.

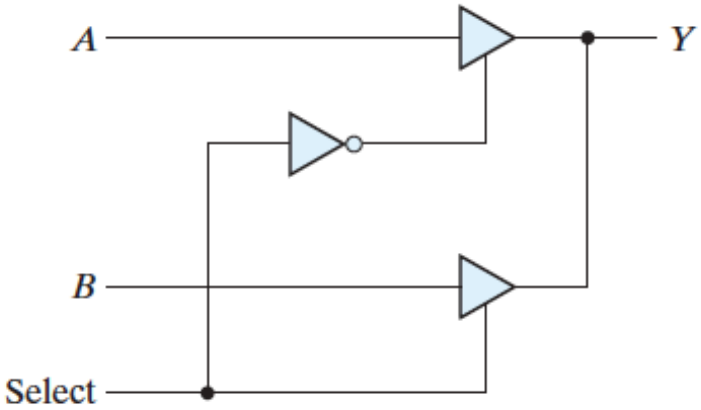
- Τα κυκλώματα με λίγες εξόδους υλοποιούνται καλύτερα με πολυπλέκτες, ενώ αυτά με πολλές εξόδους υλοποιούνται καλύτερα με αποκωδικοποιητές.

Απομονωτές τριών καταστάσεων

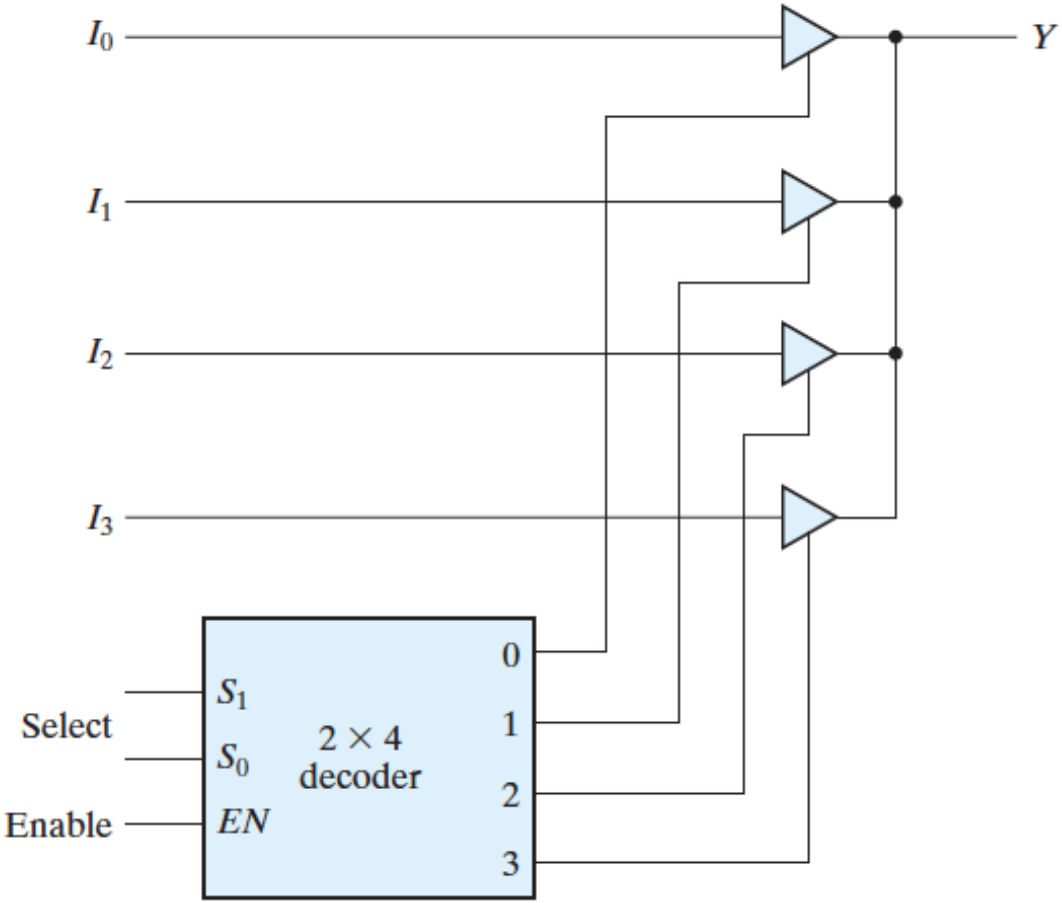
Οι απομονωτές χρησιμοποιούνται για να συνδέονται διαφορετικές μονάδες σε κοινά μέσα μετάδοσης (πχ σε διαύλους)



Παράδειγμα: υλοποίηση πολυπλέκτη



(a) 2-to-1-line mux



(b) 4-to-1-line mux