Κ03 - Λογική Σχεδίαση
Δεύτερο Εξάμηνο Φοίτησης

Συνδυαστική λογική και βασικά λογικά κυκλώματα

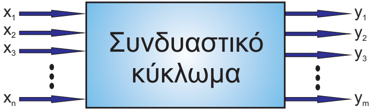
Γεώργιος Χ. Αλεξανδρόπουλος
Λέκτορας Π.Δ. 407/80
e-mail: alexandg@uop.gr
URL: http://users.iit.demokritos.gr/~alexandg

Πανεπιστήμιο Πελοποννήσου
Τμήμα Επιστήμης και Τεχνολογίας Τηλεπικοινωνιών

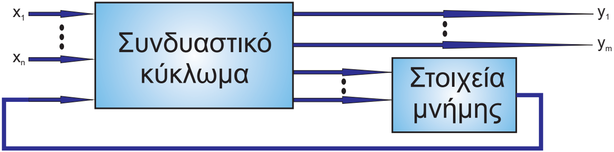
Περιεχόμενα

Εισαγωγή

- Συνδυαστικά λογικά κυκλώματα



- Ακολουθιακά λογικά κυκλώματα



Ανάλυση και Σχεδίαση Ψηφιακών Κυκλωμάτων

- 2-δικοί αριθμοί και 2-δικοί κώδικες → αναπαράσταση διακριτής πληροφορίας
- Άλγεβρα Boole → αλγεβρικός τρόπος έκφρασης λογικών συναρτήσεων
- Χάρτης Karnaugh → απλοποίηση λογικών/ψηφιακών κυκλωμάτων

Ανάλυση ψηφιακού κυκλώματος

Προσδιορισμός των συναρτήσεων Boole των εξόδων του

Σχεδίαση ψηφιακού κυκλώματος

Σύνθεση κυκλώματος με προδιαγεγραμμένη συμπεριφορά

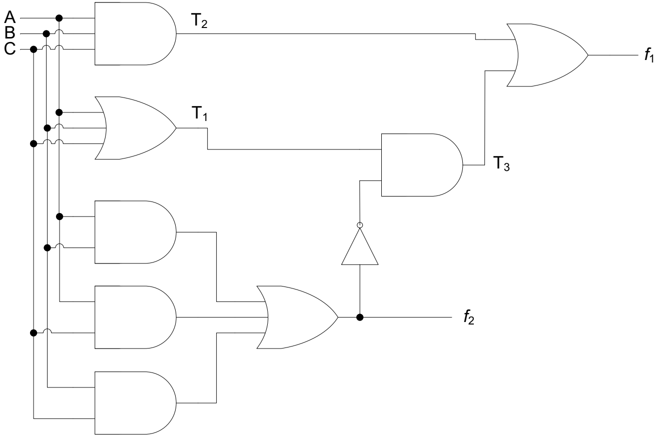
Ανάλυση Συνδυαστικών Κυκλωμάτων

Βήμα 1: Βεβαιώνω ότι το κύκλωμα είναι συνδυαστικό (απουσία βρόγχων ανάδρασης ή στοιχείων μνήμης) κι όχι ακολουθιακό

Βήμα 2: Ονομάζω τις εξόδους των πυλών με εισόδους τις μεταβλητές εισόδου κι υπολογίζω τις συναρτήσεις Boole που τις περιγράφουν. Επαναλαμβάνω τη διαδικασία μέχρι να προσδιορίσω τις συναρτήσεις Boole όλων των εξόδων του κυκλώματος

Βήμα 3: Με επανειλημμένες αντικαταστάσεις, από το τέλος προς την αρχή, των αλγεβρικών μορφών των συναρτήσεων του προηγούμενου βήματος προκύπτουν οι συναρτήσεις Boole των εξόδων του κυκλώματος ως προς τις μεταβλητές εισόδου του

Παράδειγμα Ανάλυσης Συνδυαστικού Κυκλώματος



- $f_1 = \mathrm{A'BC' + A'B'C + AB'C' + ABC}$
- $f_2 = \mathrm{AB + AC + BC}$

Πίνακας Αληθείας Συνδυαστικών Κυκλωμάτων

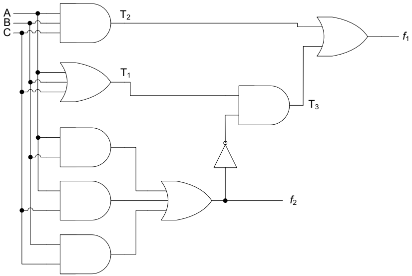
Βήμα 1: Για n μεταβλητές εισόδου σχηματίζω τους 2^n πιθανούς συνδυασμούς τιμών τους

Βήμα 2: Ονομάζω τις εξόδους των ενδιάμεσων πυλών

Βήμα 3: Προσδιορίζω, αρχικά, τον πίνακα αληθείας των εξόδων των πυλών εκείνων που εξαρτώνται από τις μεταβλητές εισόδου

Βήμα 4: Προσδιορίζω τον πίνακα αληθείας για τις εξόδους των πυλών των υπολοίπων επιπέδων μέχρι να προσδιορίσω τους πίνακες αληθείας όλων των εξόδων

Πίνακας Αληθείας Συνδυαστικών Κυκλωμάτων



A	B	C	f_2	f_2'	T_1	T_2	T_3	f_1
0	0	0	0	1	0	0	0	0
0	0	1	0	1	1	0	1	1
0	1	0	0	1	1	0	1	1
0	1	1	1	0	1	0	0	0
1	0	0	0	1	1	0	1	1
1	0	1	1	0	1	0	0	0
1	1	0	1	0	1	0	0	0
1	1	1	1	0	1	1	0	1

Σχεδίαση Συνδυαστικών Κυκλωμάτων

Βήμα 1: Με βάση τις προδιαγραφές του προς σχεδίαση κυκλώματος, καθορίζω τον απαιτούμενο αριθμό εισόδων κι εξόδων δίνοντάς τους κατάλληλα ονόματα

Βήμα 2: Κατασκευάζω τον πίνακα αληθείας που περιγράφει τη σχέση μεταβλητών εισόδου κι εξόδου

Βήμα 3: Υπολογίζω τις απλοποιημένες συναρτήσεις Boole συναρτήσει των μεταβλητών εισόδου

Βήμα 4: Σχεδιάζω το λογικό διάγραμμα των συναρτήσεων κι επαληθεύω την ορθότητα της σχεδίασης

Σχεδίαση Συνδυαστικών Κυκλωμάτων

Πιθανοί περιορισμοί σχεδίασης πρακτικού κυκλώματος:

- Πλήθος πυλών και το πλήθος είσοδων τους
- Καθυστέρηση διάδοσης σήματος
- Πλήθος διασυνδέσεων ανάμεσα στις πύλες
- Ικανότητα οδήγησης των πυλών κ.ά.

Παραδείγματα Σχεδίασης Συνδυαστικών Κυκλωμάτων

- Σχεδιάστε ένα συνδυαστικό κύκλωμα τριών εισόδων με μια έξοδο, η οποία θα είναι 1 όταν στις μεταβλητές εισόδου υπάρχουν περισσότερα 1 από 0, αλλιώς θα είναι 0 (κύκλωμα πλειοψηφίας)
- Σχεδιάστε ένα συνδυαστικό κύκλωμα τριών εισόδων με μια έξοδο, η οποία θα είναι 1 όταν ο συνδυασμός των 2-δικών τιμών των εισόδων (ως μια δυαδική λέξη) είναι μικρότερος του 3, αλλιώς θα είναι 0

Κυκλώματα Μετατροπής Κωδικών

Μετατροπέας Κώδικα

Το κύκλωμα εκείνο που αποκαθιστά τη συμβατότητα ανάμεσα σε δύο συστήματα που χρησιμοποιούν διαφορετικούς κώδικες για την αναπαράσταση της 2-δικής πληροφορίας

Παραδείγματα:

- Κώδικας BCD σε Excess-3
- Κώδικας BCD σε 8 4 -2 -1

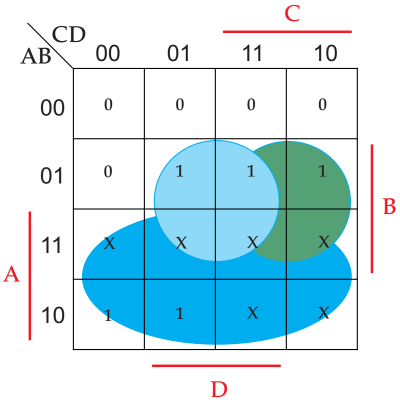
Μετατροπέας BCD σε Excess-3

BCD				Excess-3			
A	B	C	D	w	x	y	z
0	0	0	0	0	0	1	1
0	0	0	1	0	1	0	0
0	0	1	0	0	1	0	1
0	0	1	1	0	1	1	0
0	1	0	0	0	1	1	1
0	1	0	1	1	0	0	0
0	1	1	0	1	0	0	1
0	1	1	1	1	0	1	0
1	0	0	0	1	0	1	1
1	0	0	1	1	1	0	0

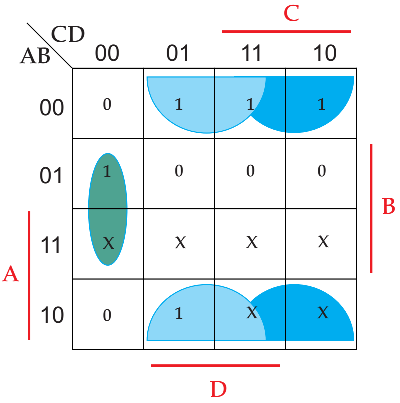
$$d(w,x,y,z) = \sum(10,11,12,13,14,15)$$

- $w = \sum(5,6,7,8,9)$
- $x = \sum(1,2,3,4,9)$
- $y = \sum(0,3,4,7,8)$
- $z = \sum(0,2,4,6,8)$

Μετατροπέας BCD σε Excess-3



$w = A + BC + BD$



$x = B'C + B'D + BC'D'$

Μετατροπέας BCD σε Excess-3

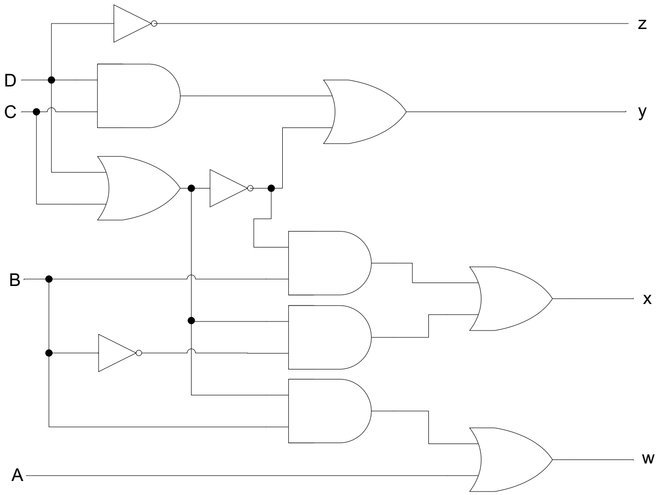
AB \ CD	00	01	11	10
00	1	0	1	0
01	1	0	1	0
11	X	X	X	X
10	1	0	X	X

$y = \mathrm{CD} + \mathrm{C'D'}$

AB \ CD	00	01	11	10
00	1	0	0	1
01	1	0	0	1
11	X	X	X	X
10	1	0	X	X

$z = \mathrm{D'}$

Μετατροπέας BCD σε Excess-3



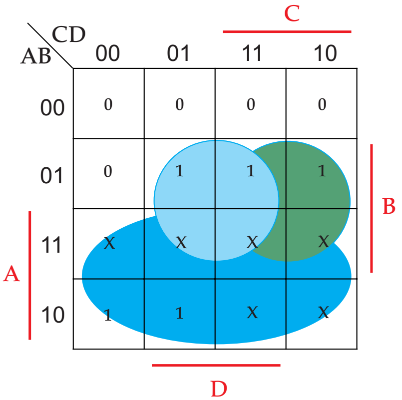
Μετατροπέας BCD σε $8\ 4\ -2\ -1$

BCD				8 4 -2 -1			
A	B	C	D	w	x	y	z
0	0	0	0	0	0	0	0
0	0	0	1	0	1	1	1
0	0	1	0	0	1	1	0
0	0	1	1	0	1	0	1
0	1	0	0	0	1	0	0
0	1	0	1	1	0	1	1
0	1	1	0	1	0	1	0
0	1	1	1	1	0	0	1
1	0	0	0	1	0	0	0
1	0	0	1	1	1	1	1

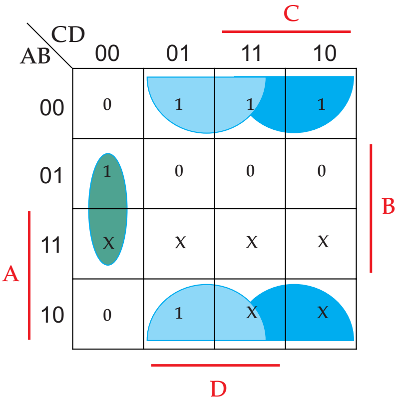
$$d(w,x,y,z) = \sum(10,11,12,13,14,15)$$

- $w = \sum(5,6,7,8,9)$
- $x = \sum(1,2,3,4,9)$
- $y = \sum(1,2,5,6,9)$
- $z = \sum(1,3,5,7,9)$

Μετατροπέας BCD σε 8 4 − 2 − 1



$w = A + BC + BD$



$x = B'C + B'D + BC'D'$

Μετατροπέας BCD σε 8 4 −2 −1

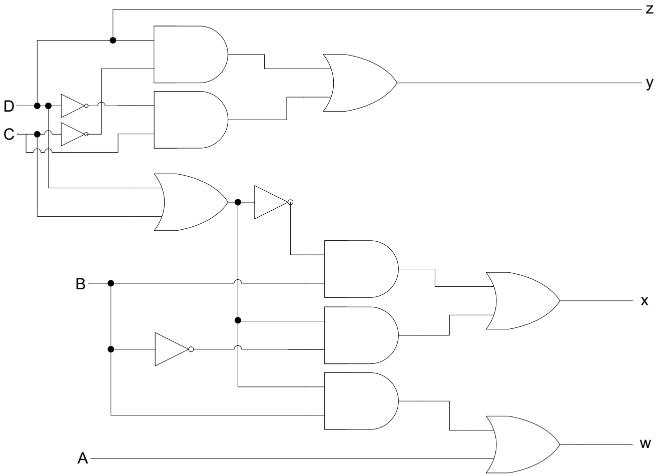
AB \ CD	00	01	11	10
00	0	1	0	1
01	0	1	0	1
11	X	X	X	X
10	0	1	X	X

$$y = CD' + C'D$$

AB \ CD	00	01	11	10
00	0	1	1	0
01	0	1	1	0
11	X	X	X	X
10	0	1	X	X

$$z = D$$

Μετατροπέας BCD σε 8 4 − 2 − 1



Εισαγωγή

Αρκετά συνδυαστικά κυκλώματα είναι διαθέσιμα ως MSI ICs, αποτελούν δηλαδή τυπικά υλικά σύνθεσης ψηφιακών κυκλωμάτων και χρησιμοποιούνται αρκετά συχνά στη σχεδίαση ψηφιακών συστημάτων:

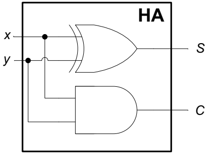
- Αθροιστές/αφαιρέτες
- Πολλαπλασιαστές
- Συγκριτές
- Αποκωδικοποιητές/κωδικοποιητές
- Πολυπλέκτες

Βασικά ψηφιακά κυκλώματα σε τυποποιημένα κύτταρα

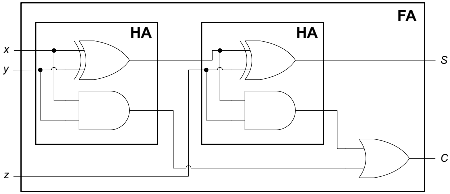
Τα παραπάνω κυκλώματα χρησιμοποιούνται σε δομή πολυσύνθετων VLSI κυκλωμάτων

Δομικά Στοιχεία Αθροιστών

Ημιαθροιστής: το συνδυαστικό κύκλωμα που υλοποιεί την πρόσθεση δύο bits

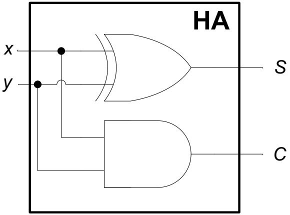


Πλήρης αθροιστής: το συνδυαστικό κύκλωμα που υλοποιεί την πρόσθεση τριών bits



Ημιαθροιστής

x	y	C	S
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0



$$S = x'y + xy'$$

$$C = xy$$

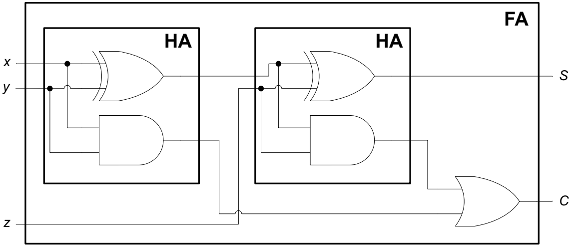
Πλήρης Αθροιστής

x	y	z	C	S
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

$$S = x'y'z + x'yz' + xy'z' + xyz$$

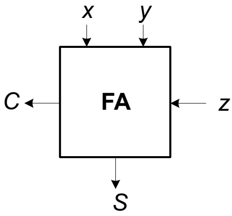
$$C = xy + xz + yz$$

Πλήρης Αθροιστής με Δύο Ημιαθροιστές



$$S = (x \oplus y) \oplus z$$

$$C = (x \oplus y)z + xy$$



2-δικός Αθροιστής

Ο 2-δικός αθροιστής είναι το ψηφιακό κύκλωμα που υλοποιεί το άθροισμα δύο 2-δικών αριθμών

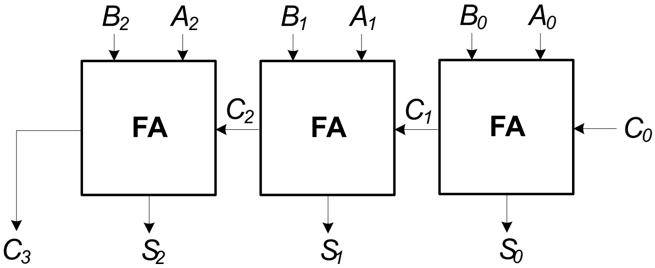
2-δικός Αθροιστής

Υλοποιείται εύκολα τοποθετώντας σε σειρά FAs: το κρατούμενο εξόδου κάθε FA συνδέεται στο κρατούμενο εισόδου του επόμενου FA

Για την κατασκευή ενός n-bit 2-δικού αθροιστή χρειάζονται n FAs σε μια διάταξη κατά την οποία κάθε κρατούμενου εξόδου συνδέεται με το κρατούμενο εισόδου του FA της αμέσως μεγαλύτερης τάξης

3-bit 2-δικός Αθροιστής Ριπής Κρατουμένου

3-bit 2-δικός αθροιστής ριπής κρατουμένου των $A_2A_1A_0$ και $B_2B_1B_0$:



Η πρόσθεση των $A_2A_1A_0$ και $B_2B_1B_0$ γίνεται σειρακά $\longrightarrow$ Όταν παραχθούν όλα τα κρατούμενα, οι έξοδοι του κυκλώματος περιέχουν τα σωστά ψηφία του αθροίσματος $C_3S_2S_1S_0$

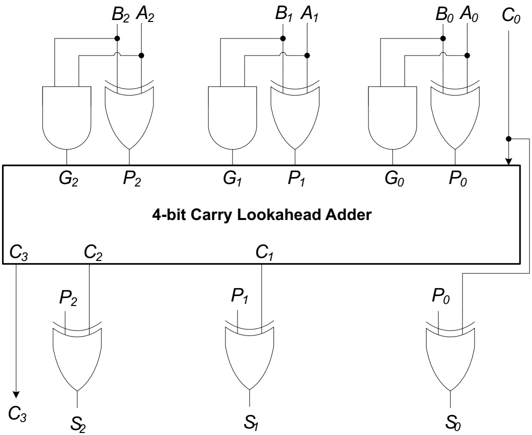
Αθροιστής Πρόβλεψης Κρατουμένου

- Σε έναν n-bit παράλληλο αθροιστή υπάρχουν $2n$ επίπεδα πυλών ανάμεσα στην είσοδο και στην έξοδο, μέσα από τα οποία διαδίδονται τα κρατούμενα
- Ο χρόνος διάδοσης των κρατουμένων αποτελεί σημαντικό χαρακτηριστικό ενός αθροιστή ριπής κρατουμένου

Αθροιστής Πρόβλεψης Κρατουμένου

Με τον αθροιστή πρόβλεψης κρατουμένου αυξάνεται μεν η πολυπλοκότητα του κυκλώματος της άθροισης αλλά μειώνεται ο συνολικός χρόνος καθυστέρησης διάδοσης των κρατουμένων, τα οποία πλέον υπολογίζονται ταυτόχρονα

3-bit Αθροιστής Πρόβλεψης Κρατουμένου



$$S_i = P_i \oplus C_i \qquad C_{i+1} = G_i + P_i C_i$$

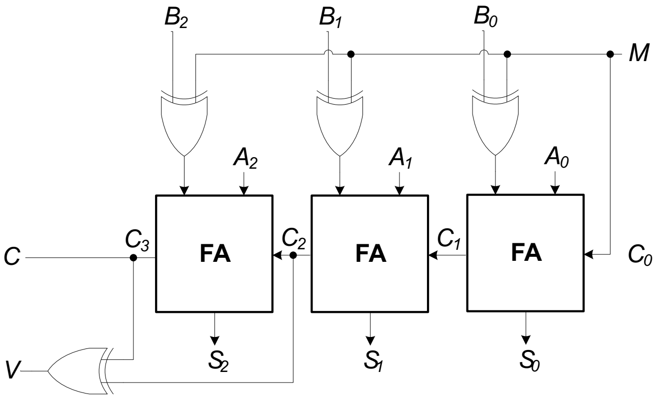
2-δικός Αφαιρέτης

Η 2-δική αφαίρεση επιτυγχάνεται εύκολα με πρόσθεση στο μειωτέο του συμπληρώματος ως προς 2 του αφαιρετέου

Για την υλοποίηση της 2-δικής αφαίρεσης χρειάζονται:

- Αντιστροφείς για την υλοποίηση του συμπληρώματος ως προς 1
- Η πρόσθεση ενός ακόμα 1 (ώστε να προκύψει το συμπλήρωμα ως προς 2), η οποία μπορεί να λάβει χώρα στο τελικό άθροισμα μέσω του κρατουμένου εισόδου

3-bit 2-δικός Αθροιστής/Αφαιρέτης

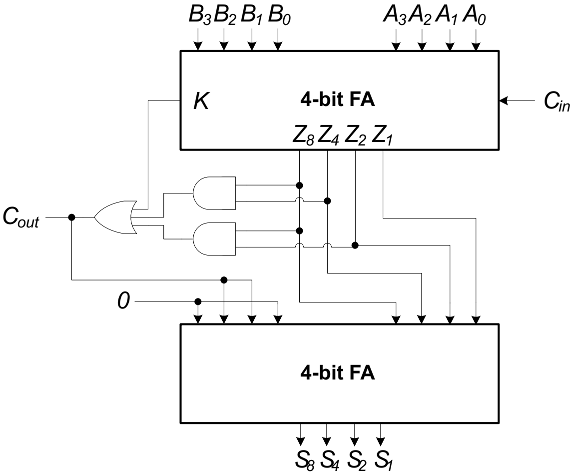


- $M = 0 \longrightarrow$ Πρόσθεση
- $M = 1$ $(C_0 = 1) \longrightarrow$ Αφαίρεση
- Όταν $V = 1$ για προσημασμένους $\longrightarrow$ Υπερχείλιση

10-δικός Αθροιστής/Αφαιρέτης

- Για την εκτέλεση αριθμητικών πράξεων στο 10-δικό σύστημα σε ένα ψηφιακό σύστημα χρησιμοποιούνται 2-δικοί κώδικες για την αναπαράσταση των 10-δικών αριθμών
- Ένα στοιχειώδες κύκλωμα πρόσθεσης δύο 10-δικών αριθμών θα έχει εννέα εισόδους και πέντε εξόδους
- Υπάρχει μεγάλη ποικιλία 10-δικών αθροιστών ανάλογα με το χρησιμοποιούμενο κώδικα αναπαράστασης των 10-δικών αριθμών προς πρόσθεση

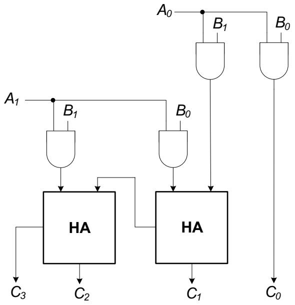
Αθροιστής BCD



2-δικός Πολλαπλασιαστής

$$B_1B_0 \text{x} A_1A_0 = C_3C_2C_1C_0$$

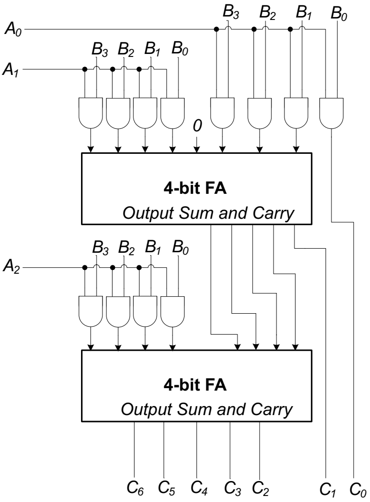
- 4 πύλες AND
- 2 HAs



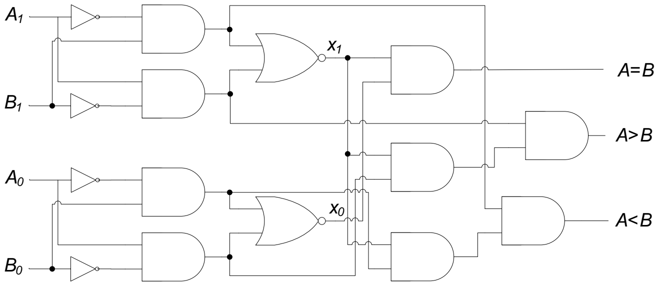
2-δικός Πολλαπλασιαστής

Για m-bit πολλαπλασιαστή και n-bit πολλαπλασιαστέο χρειάζονται mn πύλες AND και $m-1$ n-bit αθροιστές ώστε να παραχθεί το $(m+n)$-bit γινόμενο

4-bit επί 3-bit 2-δικός Πολλαπλασιαστής



2-bit 2-δικός Συγκριτής Μεγέθους



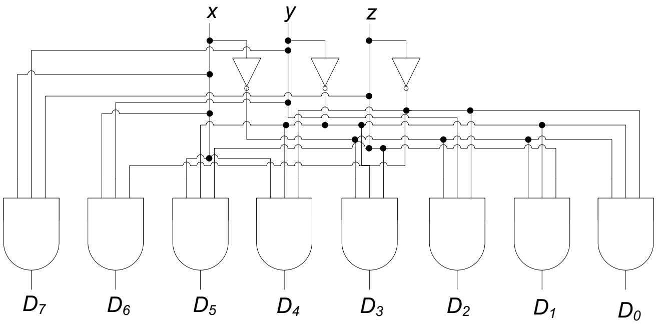
- $(A = B) \longrightarrow (A_0B_0 + A_0'B_0')(A_1B_1 + A_1'B_1')$
- $(A > B) \longrightarrow A_1B_1' + (A_1B_1 + A_1'B_1')A_0B_0'$
- $(A < B) \longrightarrow A_1'B_1 + (A_1B_1 + A_1'B_1')A_0'B_0$

Αποκωδικοποιητής/Κωδικοποιητής

- Ο αποκωδικοποιητής είναι το συνδυαστικό κύκλωμα μετατροπής της n-bit κωδικοποιημένης πληροφορίας εισόδου σε ισοδύναμη έως και 2^n bits πληροφορία στην έξοδό του
- Για n-bit είσοδο και m-bit $(m \leqslant 2^n)$ έξοδο το κύκλωμα ονομάζεται αποκωδικοποιητής n-σε-m
- Ο κωδικοποιητής είναι το συνδυαστικό κύκλωμα που εκτελεί την ανάστροφη λειτουργία από αυτή του αποκωδικοποιητή
- Ο κωδικοποιητής μετατρέπει τη m-bit $(m \leqslant 2^n)$ πληροφορία εισόδου σε κωδικοποιημένη n-bit πληροφορία στην έξοδό του

Αποκωδικοποιητής 3-σε-8

Αποκωδικοποίηση 3 εισόδων σε 8
(πχ αποκωδικοποίηση 2-δικού αριθμού σε 8-δικό)



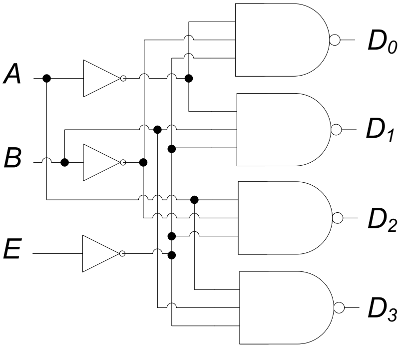
Αποκωδικοποιητής 3-σε-8

Ο αποκωδικοποιητής 3-σε-8 αντιστοιχίζει τις εισόδους του στους ελαχιστόρους τους

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

Αποκωδικοποιητής 2-σε-4 με Είσοδο Επίτρεψης

Οι αποκωδικοποιητές κατασκευάζονται μόνο με πύλες NAND και συνήθως περιλαμβάνουν τουλάχιστον μια είσοδο επίτρεψης

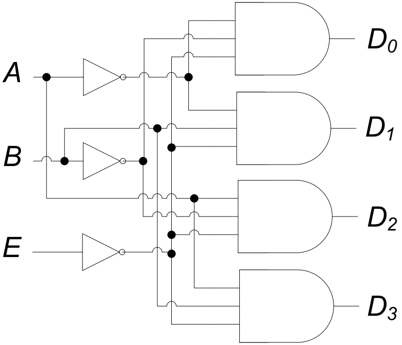


E	A	B	D_0	D_1	D_2	D_3
1	X	X	1	1	1	1
0	0	0	0	1	1	1
0	0	1	1	0	1	1
0	1	0	1	1	0	1
0	1	1	1	1	1	0

Αποκωδικοποιητής/Αποπλέκτης

Ως αποπλέκτης είναι γνωστό το κύκλωμα εκείνο που λαμβάνει πληροφορία από μια γραμμή εισόδου και την προωθεί σε μια από τις έως και 2^n γραμμές εξόδου

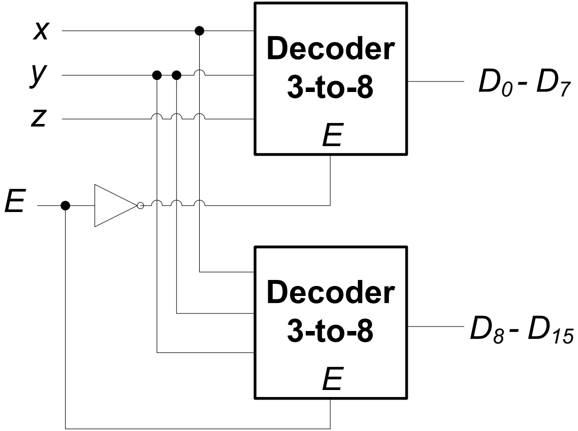
Ποια γραμμή εξόδου θα επιλεγεί εξαρτάται από συνδυασμό τιμών των εισόδων (*γραμμές ή είσοδοι επιλογής*)



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

Διασύνδεση Αποκωδικοποιητών

Αποκωδικοποιητής 4-σε-16 από δύο αποκωδικοποιητές 3-σε-8:



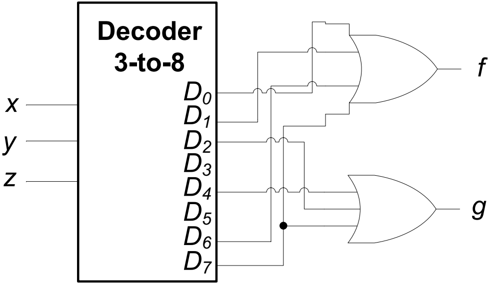
Συνδυαστικά Κυκλώματα με Αποκωδικοποιητές

- Η βασική μορφή του αποκωδικοποιητή παράγει στην έξοδό του τους 2^n ελαχιστόρους των n μεταβλητών εισόδου
- Κάθε συνάρτηση Boole εκφράζεται σε μορφή αθροίσματος ελαχιστόρων, συνεπώς, μπορώ να την υλοποιήσω χρησιμοποιώντας έναν αποκωδικοποιητή και πύλες OR
- Κάθε συνδυαστικό κύκλωμα n εισόδων και m εξόδων δύναται να υλοποιηθεί με έναν αποκωδικοποητή n-σε-2^n και m πύλες OR

Συνδυαστικά Κυκλώματα με Αποκωδικοποιητές

Υλοποιείστε με αποκωδικοποιητή τις συναρτήσεις:

- $f(x, y, z) = \sum(0, 1, 6, 7)$
- $g(x, y, z) = \sum(2, 4, 7)$



Κωδικοποιητής 8-σε-3

Ο κωδικοποιητής μετατρέπει τη m-bit ($m \leqslant 2^n$) πληροφορία εισόδου σε κωδικοποιημένη n-bit πληροφορία στην έξοδό του

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 8-σε-3

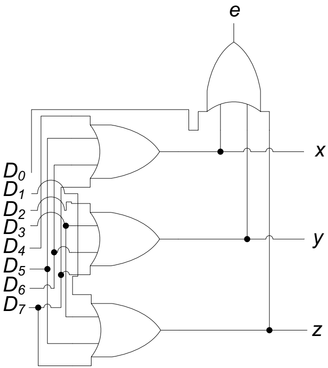
Ο κωδικοποιητής αυτός υλοποιείται εύκολα από τον πίνακα αληθείας του, χρησιμοποιώντας πύλες OR:

$$x = D_4 + D_5 + D_6 + D_7$$
$$y = D_2 + D_3 + D_6 + D_7$$
$$z = D_1 + D_3 + D_5 + D_7$$

- Η έξοδος αυτού του κωδικοποιητή έχει νόημα μόνο όταν μία εκ των εισόδων είναι 1
- Συνήθως, χρησιμοποιείται σειρά προτεραιότητας των εισόδων για να αποφευχθεί αυτή η ασάφεια

Κωδικοποιητής 8-σε-3

Ο κωδικοποιητής 8-σε-3 υλοποιείται κυκλωματικά ως ακολούθως:



Όταν η έξοδος e είναι 1, ο κωδικοποιητής παράγει ένα 2-δικό αριθμό στην έξοδό του με τουλάχιστον ένα 1 στην είσοδό του

Κωδικοποιητής Προτεραιότητας

- Πρόκειται για τον κωδικοποιητή εκείνο που χρησιμοποιεί την έννοια της προτεραιότητας στις εισόδους του
- Εάν μία ή περισσότερες είσοδοι είναι ταυτόχρονα 1, κωδικοποιείται η είσοδος με τη μεγαλύτερη προκαθορισμένη προτεραιότητα
- Εκτός των ζητουμένων εισόδων, ο κωδικοποιητής προτεραιότητας περιέχει και μια επιπλέον έξοδο ενδεικτική της εγκυρότητας των εισόδων

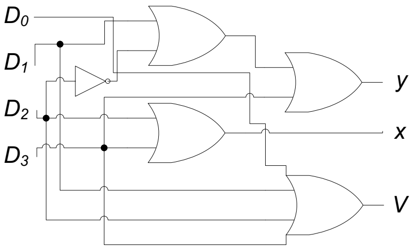
Κωδικοποιητής Προτεραιότητας Τεσσάρων Εισόδων

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

D_3	D_2	D_1	D_0	x	y	V
0	0	0	0	X	X	0
0	0	0	1	0	0	1
0	0	1	0	0	1	1
0	0	1	1	0	1	1
0	1	0	0	1	0	1
0	1	0	1	1	0	1
0	1	1	0	1	0	1
0	1	1	1	1	0	1
1	0	0	0	1	1	1
1	0	0	1	1	1	1
1	0	1	0	1	1	1
1	0	1	1	1	1	1
1	1	0	0	1	1	1
1	1	0	1	1	1	1
1	1	1	0	1	1	1
1	1	1	1	1	1	1

Κωδικοποιητής Προτεραιότητας Τεσσάρων Εισόδων

Από τον πίνακα αληθείας και χρησιμοποιώντας τη μέθοδο του χάρτη Karnaugh προκύπτει η ακόλουθη κυκλωματική υλοποίηση:



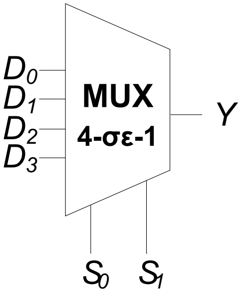
- $x = D_2 + D_3$
- $y = D_3 + D_1 D_2'$
- $V = D_0 + D_1 + D_2 + D_3$

Πολυπλέκτης

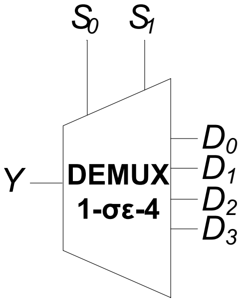
- Πρόκειται για το συνδυαστικό κύκλωμα που επιλέγει δυαδική πληροφορία από μία από τις πολλές γραμμές εισόδου του και την κατευθύνει σε μια και μόνο γραμμή εξόδου του
- Ένα σύνολο γραμμών επιλογής είναι υπεύθυνο για τον έλεγχο της επιλογής μιας συγκεκριμένης γραμμής εισόδου
- Οι συνδυασμοί τιμών των 2^n γραμμών εισόδου και των n γραμμών επιλογής καθορίζουν την είσοδο του πολυπλέκτη που επιλέγεται στην έξοδό του

Πολυπλέκτης/Αποπλέκτης

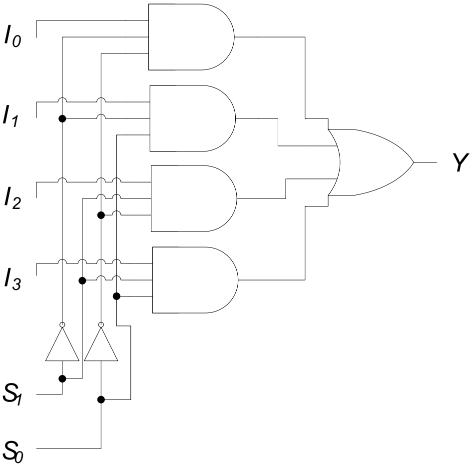
Σχηματικό διάγραμμα πολυπλέκτη 4-σε-1:



Σχηματικό διάγραμμα αποπλέκτη 1-σε-4:

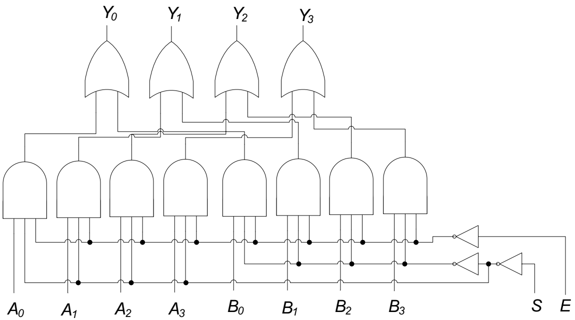


Πολυπλέκτης 4-σε-1 (*Επιλογέας Δεδομένων*)



S_0	S_1	Y
0	0	I_0
0	1	I_1
1	0	I_2
1	1	I_3

Τετραπλός Πολυπλέκτης 2-σε-1



E	S	Y_0	Y_1	Y_2	Y_3
1	X	0	0	0	0
0	0	A_0	A_1	A_2	A_3
0	1	B_0	B_1	B_2	B_3

Συνδυαστικά Κυκλώματα με Πολυπλέκτες

Υλοποίηση συνάρτησης Boole με πολυπλέκτη

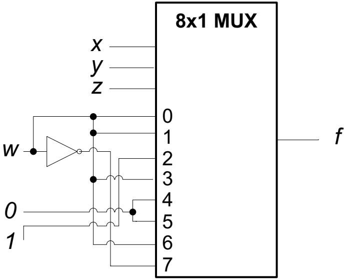
Οποιαδήποτε συνάρτηση Boole n μεταβλητών δύναται να υλοποιηθεί με έναν πολυπλέκτη $n-1$ εισόδων επιλογής και 2^{n-1} εισόδων δεδομένων

- Βήμα 1: Υπολογίζω τον πίνακα αληθείας της συνάρτησης Boole
- Βήμα 2: Συνδέω τις $n-1$ MSB εισόδους της συνάρτησης με τις εισόδους επιλογής του πολυπλέκτη
- Βήμα 3: Για κάθε συνδυασμό τιμών των μεταβλητών επιλογής υπολογίζω την επιθυμητή έξοδο της συνάρτησης ως αλγεβρική έκφραση είτε της εναπομείνασας μεταβλητής είτε του 1 είτε του 0
- Βήμα 4: Τις τιμές του βήματος 3 τις τοποθετώ στις αντίστοιχες εισόδους δεδομένων του πολυπλέκτη

Υλοποίηση συνάρτησης Boole με πολυπλέκτη

Παράδειγμα: Υλοποίηση της $f(x,y,z,w) = \sum(1,3,4,5,7,13,14)$ με πολυπλέκτη

x	y	z	w	f	
0	0	0	0	0	f=w
0	0	0	1	1	
0	0	1	0	0	f=w
0	0	1	1	1	
0	1	0	0	1	f=1
0	1	0	1	1	
0	1	1	0	0	f=w
0	1	1	1	1	
1	0	0	0	0	f=0
1	0	0	1	0	
1	0	1	0	0	f=0
1	0	1	1	0	
1	1	0	0	0	f=w
1	1	0	1	1	
1	1	1	0	1	f=w'
1	1	1	1	0	



Τέλος Θεματικής Ενότητας

Ευχαριστώ για την προσοχή σας

Γεώργιος Χ. Αλεξανδρόπουλος

e-mail: alexandg@uop.gr
URL: http://users.iit.demokritos.gr/~alexandg

eclass: http://eclass.uop.gr/courses/TST289/