

5.4.2 ΑΠΟΚΩΔΙΚΟΠΟΙΗΤΕΣ

Ο αποκωδικοποιητής είναι το κύκλωμα που μετατρέπει την κωδικοποιημένη πληροφορία στην αρχική της μορφή. Γενικά, πρόκειται για ένα συνδυαστικό κύκλωμα που λαμβάνει στην είσοδο κωδικοποιημένη πληροφορία μεγέθους n bits, και την μετατρέπει στην αρχική της μορφή μεγέθους m bits, $n < m$. Συμβολικά, ο κωδικοποιητής αναφέρεται και ως n -σε- m .

Ο δυαδικός αποκωδικοποιητής είναι ένα κύκλωμα, το οποίο λαμβάνει n bits πληροφορίας στις γραμμές εισόδου του, τα οποία μετατρέπει σε αποκωδικοποιημένη πληροφορία μεγέθους 2^n bits, όπου όμως ένα μόνον bit είναι ίσο με τη μονάδα. Η αποκωδικοποιημένη πληροφορία εξάγεται ως αποτέλεσμα στις γραμμές εξόδου του κυκλώματος. Δηλαδή, ο δυαδικός αποκωδικοποιητής διαθέτει n γραμμές εισόδου τις οποίες κατευθύνει σε 2^n γραμμές εξόδου. Ο δυαδικός κωδικοποιητής αναφέρεται και ως αποκωδικοποιητής n -σε- 2^n , δηλώνοντας με τον τρόπο αυτό το πλήθος εισόδων και εξόδων του. Οι αποκωδικοποιητές έχουν αρκετές εφαρμογές, οι σημαντικότερες εκ των οποίων είναι ως επιλογείς μονάδων (chip select) και λέξεων (word select) κατά τη διαδικασία ανάγνωσης και εγγραφής από και προς τη μνήμη, ως γεννήτριες ελαχιστόρων για την υλοποίηση άλλων ψηφιακών συστημάτων, ή ως μετατροπείς από μία μορφή κωδικοποίησης σε μία άλλη. Η χρήση των αποκωδικοποιητών στις λειτουργίες της μνήμης παρουσιάζεται στο Κεφάλαιο 14. Οι άλλες εφαρμογές παρουσιάζονται στη συνέχεια της ενότητας.

Έστω ένας αποκωδικοποιητής με $n = 3$, δηλαδή 3-σε-8. Οι τρεις εισοδοί x , y , z αντιπροσωπεύουν έναν αριθμό μήκους τριών bits που οδηγεί τον αποκωδικοποιητή να ενεργοποιήσει μία από τις 8 εξόδους του $d_7 - d_0$. Ο πίνακας αληθείας του κυκλώματος δίνεται από τον Πίνακα 5.4. Για κάθε συνδυασμό εισόδων, υπάρχει μια μόνο έξοδος, η οποία είναι ίση με 1.

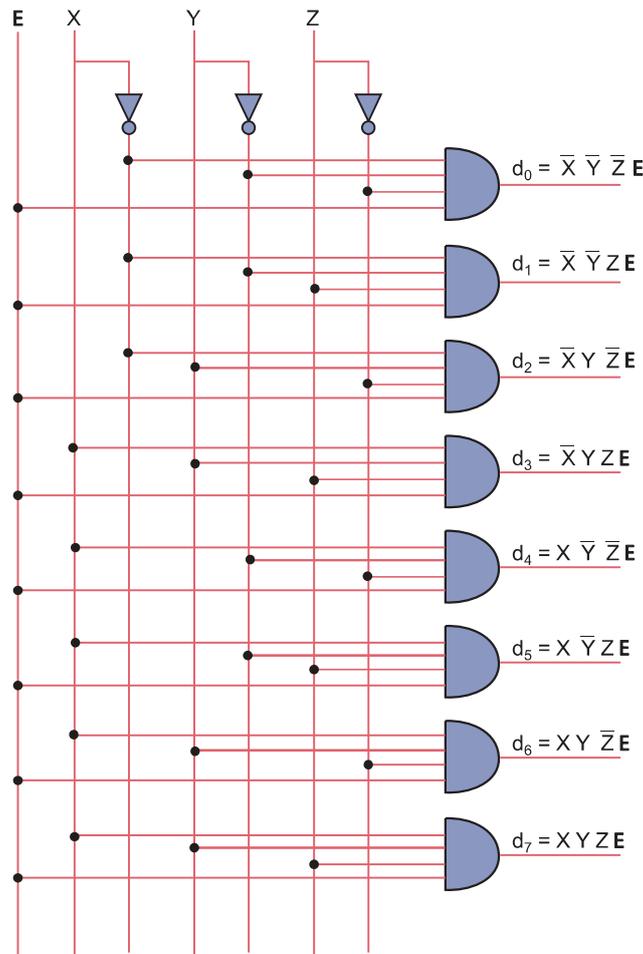
Πίνακας 5.4: Πίνακας αληθείας του αποκωδικοποιητή 3-σε-8

x	y	z	d_7	d_6	d_5	d_4	d_3	d_2	d_1	d_0
0	0	0	0	0	0	0	0	0	0	1
0	0	1	0	0	0	0	0	0	1	0
0	1	0	0	0	0	0	0	1	0	0
0	1	1	0	0	0	0	1	0	0	0
1	0	0	0	0	0	1	0	0	0	0
1	0	1	0	0	1	0	0	0	0	0
1	1	0	0	1	0	0	0	0	0	0
1	1	1	1	0	0	0	0	0	0	0

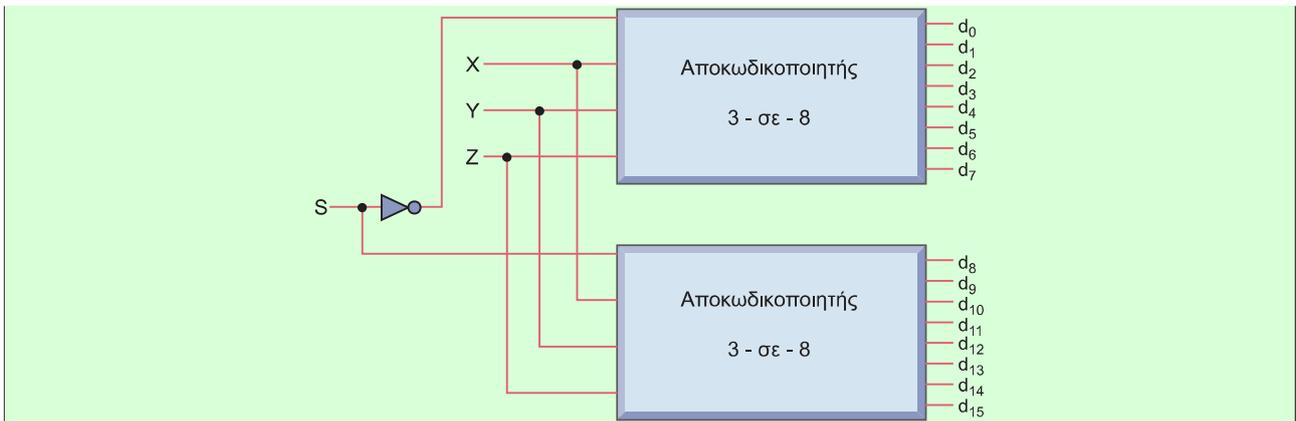
Η υλοποίηση του αποκωδικοποιητή 3-σε-8 δίνεται στο Σχήμα 5.27. Όταν η είσοδος E είναι ίση με 0, τότε αποτρέπεται οποιαδήποτε διάδοση σήματος από τις πύλες ΚΑΙ, άρα όλες οι έξοδοι είναι ίσες με 0. Όταν η είσοδος E είναι ίση με 1, τότε επιτρέπεται η διάδοση σήματος από κάποια εκ των πυλών ΚΑΙ. Ειδικότερα, όταν οι εισοδοί x , y , και z είναι ίσες με 0, από τους τρεις αντιστροφείς περνάει μονάδα, με αποτέλεσμα η έξοδος d_0 να είναι ίση με 1. Οι υπόλοιπες έξοδοι παραμένουν στο λογικό 0. Ομοίως, για κάθε συνδυασμό εισόδων, υπάρχει μία μόνον έξοδος ίση με μονάδα. Στο Παράδειγμα 5.8, χρησιμοποιούνται δύο αποκωδικοποιητές 3-σε-8 για να υλοποιηθεί ένας αποκωδικοποιητής 4-σε-16.

ΠΑΡΑΔΕΙΓΜΑ 5.8 Να υλοποιηθεί ένας αποκωδικοποιητής 4-σε-16 χρησιμοποιώντας 2 αποκωδικοποιητές 3-σε-8 και μια είσοδο-επιλογή.

Η υλοποίηση δίνεται στο Σχήμα 5.28. Ο επιλογέας συμβολίζεται με s και η τιμή του καθορίζει ποιος από τους αποκωδικοποιητές επιλέγεται. Ειδικότερα, όταν $s = 0$ επιλέγεται ο πάνω αποκωδικοποιητής και οι έξοδοι του δίνουν όλες τις τιμές των s, x, y , και z από 0000-0111. Όταν $s = 1$, επιλέγεται ο κάτω αποκωδικοποιητής και οι έξοδοι του δίνουν τις τιμές των s, x, y , και z από 1000-1111.



Σχήμα 5.27: Λογικό διάγραμμα του αποκωδικοποιητή 3-σε-8.

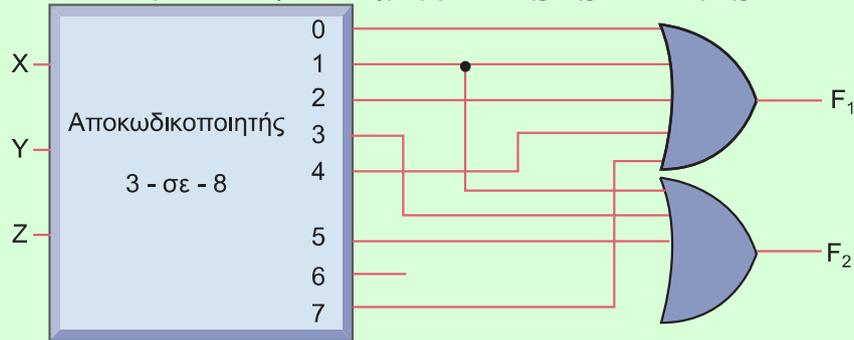


Σχήμα 5.28: Σχηματικό διάγραμμα ενός αποκωδικοποιητή 4-σε-16

Οι Αποκωδικοποιητές Ως Γεννήτριες Ελαχιστόρων Επειδή οι έξοδοι ενός αποκωδικοποιητή παράγουν όλους τις δυνατούς συνδυασμούς των εισόδων, μπορούν να χρησιμοποιηθούν για να παράγουν τους ελαχιστόρους μίας λογικής συνάρτησης.

ΠΑΡΑΔΕΙΓΜΑ 5.9 Να υλοποιηθούν με ένα κύκλωμα αποκωδικοποιητή 3-σε-8 οι λογικές συναρτήσεις F_1 και F_2 τριών εισόδων, $F_1(x, y, z) = \Sigma(0, 1, 2, 4, 7)$ και $F_2(x, y, z) = S(1, 3, 5)$.
 Η υλοποίηση βασίζεται στην παραγωγή των ελαχιστόρων που θέτουν κάθε μια από τις F_1 και F_2 στο λογικό 1. Οι ελαχιστόροι κάθε συνάρτησης τοποθετούνται ως είσοδοι σε μια πύλη Ή, η οποία

παράγει το τελικό αποτέλεσμα. Το λογικό διάγραμμα αυτής της υλοποίησης δίνεται στο Σχήμα 5.29.



Σχήμα 5.29: Υλοποίηση των F_1, F_2 με έναν αποκωδικοποιητή 3-σε-8.

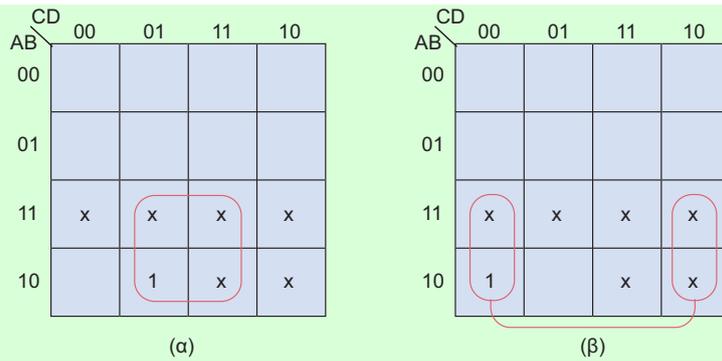
Οι Αποκωδικοποιητές Ως Μετατροπείς Κωδικών Γενικά, τα κυκλώματα αποκωδικοποιητή και κωδικοποιητή, με εξαίρεση τους δυαδικούς κωδικοποιητές και αποκωδικοποιητές, έχουν ως σκοπό να μετατρέπουν μία μορφή κωδικοποίησης που υπάρχει στην είσοδο σε μία άλλη μορφή κωδικοποίησης, η οποία εξάγεται στην έξοδο. Στην παράγραφο αυτή θα δώσουμε δύο παραδείγματα: (α) μετατροπή από κώδικα BCD σε δεκαδικό αριθμό, και (β) μετατροπή από κώδικα BCD σε κώδικα 7 τμημάτων.

ΠΑΡΑΔΕΙΓΜΑ 5.10 Να υλοποιηθούν κυκλώματα για τις εξής κωδικοποιήσεις: (α) μετατροπή από BCD σε δεκαδικό, (β) μετατροπή από κώδικα BCD σε κώδικα 7 τμημάτων.

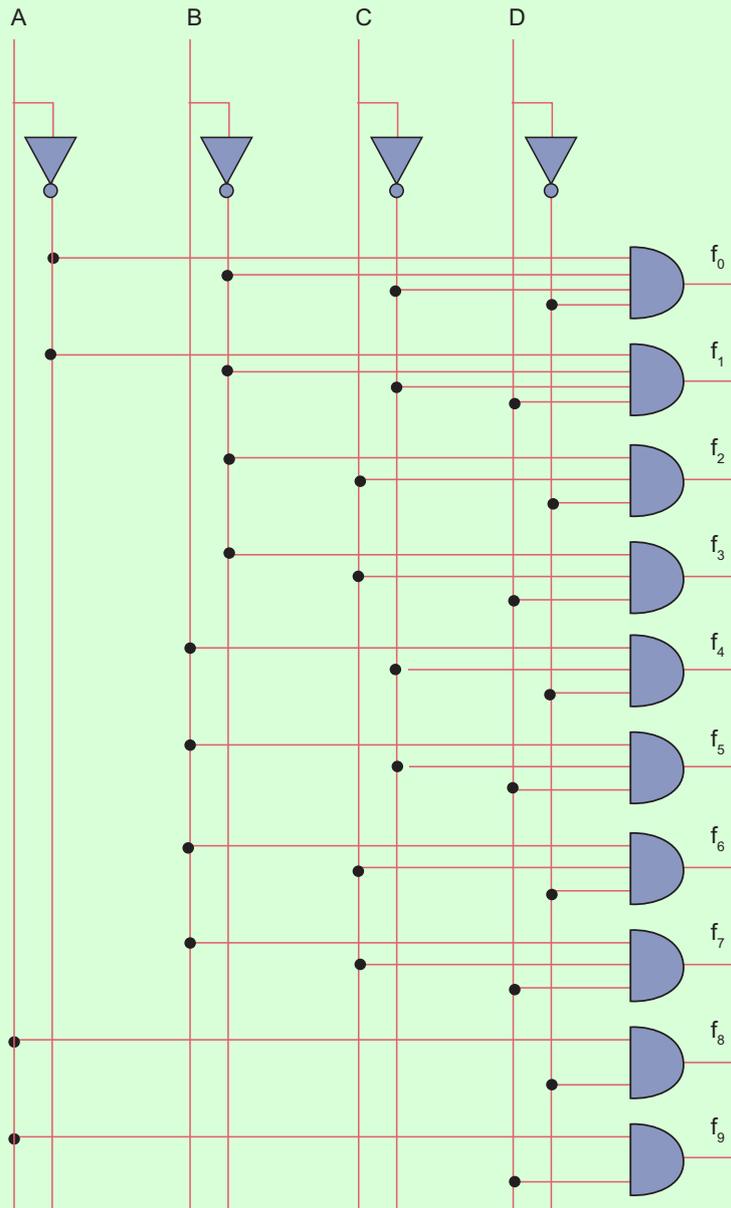
(α) Ο αποκωδικοποιητής αυτός θα είναι ίδιος με τον αποκωδικοποιητή 4-σε-16, με τη διαφορά ότι οι γραμμές είναι 10, δηλαδή μία για κάθε ψηφίο του δεκαδικού συστήματος. Οι ελαχιστόροι m_{10} ως m_{15} είναι αδιάφοροι. Επομένως, αν οι εισοδοί είναι A, B, C, D και οι έξοδοι είναι οι $f_0 - f_9$, τότε ο πίνακας αληθείας του κυκλώματος είναι ο ακόλουθος:

A	B	C	D	$f_{15} - f_{10}$	f_9	f_8	f_7	f_6	f_5	f_4	f_3	f_2	f_1	f_0
0	0	0	0	X	0	0	0	0	0	0	0	0	0	1
0	0	0	1	X	0	0	0	0	0	0	0	0	1	0
0	0	1	0	X	0	0	0	0	0	0	0	1	0	0
0	0	1	1	X	0	0	0	0	0	0	1	0	0	0
0	1	0	0	X	0	0	0	0	0	1	0	0	0	0
0	1	0	1	X	0	0	0	0	1	0	0	0	0	0
0	1	1	0	X	0	0	0	1	0	0	0	0	0	0
0	1	1	1	X	0	0	1	0	0	0	0	0	0	0
1	0	0	0	X	0	1	0	0	0	0	0	0	0	0
1	0	0	1	X	1	0	0	0	0	0	0	0	0	0
1	0	1	0	X	0	0	0	0	0	0	0	0	0	0
1	0	1	1	X	0	0	0	0	0	0	0	0	0	0
1	1	0	0	X	0	0	0	0	0	0	0	0	0	0
1	1	0	1	X	0	0	0	0	0	0	0	0	0	0
1	1	1	0	X	0	0	0	0	0	0	0	0	0	0
1	1	1	1	X	0	0	0	0	0	0	0	0	0	0

Για κάθε έξοδο μπορεί να κατασκευαστεί ένας χάρτη Karnaugh, ώστε να απλοποιηθεί η λογική της έκφραση. Οι αδιάφοροι όροι μπορούν να χρησιμοποιηθούν στο σχηματισμό ομάδων πάνω στο χάρτη, ώστε να καταλήξουμε σε όσο το δυνατόν πιο απλές εκφράσεις.



Σχήμα 5.30: Χάρτες Karnaugh για τις εξόδους f_9 (α) και f_8 (β).



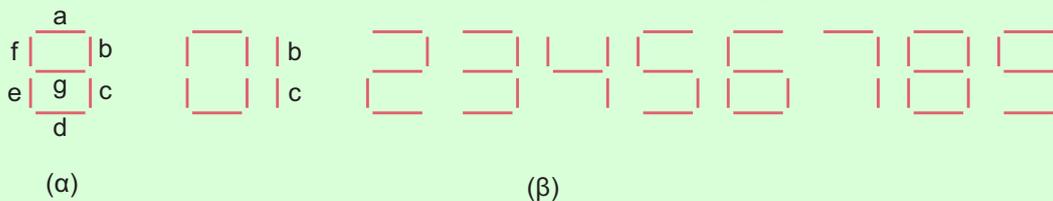
Σχήμα 5.31: Λογικό διάγραμμα αποκωδικοποιητή από BCD σε δεκαδικό.

Στο Σχήμα 5.30 δίνονται οι χάρτες Karnaugh για τις εξόδους f_9 και f_8 . Παρατηρείστε ότι οι μονάδες που υπάρχουν στα τετράγωνα των δύο ελαχιστόρων ομαδοποιούνται με τρία τετράγωνα των αδιάφορων όρων σχηματίζοντας τετράδες ελαχιστόρων. Επομένως, εξαλείφονται δύο όροι γινομένου. Το ίδιο συμβαίνει στους χάρτες Karnaugh των εξόδων $f_2, f_3, f_4, f_5, f_6,$ και f_7 , όπου δημιουργούνται δυάδες ελαχιστόρων, εξαλείφοντας έναν όρο γινομένου. Τέλος, στους χάρτες Karnaugh των εξόδων f_0 και f_1 δεν είναι δυνατή η δημιουργία ομάδων. Συνοπτικά, οι εξισώσεις που περιγράφουν τις 10

εξόδους δίνονται παρακάτω, ενώ το λογικό διάγραμμα του αποκωδικοποιητή από BCD σε δεκαδικό δίνεται στο Σχήμα 5.31.

Έξοδος	Εξίσωση	Έξοδος	Εξίσωση
f_0	$\overline{A} \overline{B} \overline{C} \overline{D}$	f_5	$B \overline{C} D$
f_1	$\overline{A} \overline{B} \overline{C} D$	f_6	$B C \overline{D}$
f_2	$\overline{B} C \overline{D}$	f_7	$B C D$
f_3	$\overline{B} C D$	f_8	$A \overline{D}$
f_4	$B \overline{C} \overline{D}$	f_9	$A D$

(β) Ο αποκωδικοποιητής από BCD σε κώδικα 7 τομέων μετατρέπει ένα δεκαδικό ψηφίο το οποίο είναι γραμμένο σε κώδικα BCD σε μία μορφή κατάλληλη για προβολή σε αριθμομηχανές ή ψηφιακά ρολόγια. Ειδικότερα, κάθε ψηφίο BCD μετατρέπεται σε επτά σήματα, τα οποία οδηγούν τους 7 τομείς. Οι τομείς είναι δύοδοι φωτοεκπομπής (LED), οι οποίες φωτοβολούν όταν δέχονται ένα σήμα. Για κάθε συνδυασμό των εισόδων $x_3..x_0$ οι έξοδοι του αποκωδικοποιητή $a, b, c, d, e, f,$ και g ενεργοποιούν τις κατάλληλες διόδους, ώστε να σχηματίζεται το αντίστοιχο ψηφίο. Το Σχήμα 5.32(α) δείχνει την ονομασία των 7 τομέων και το Σχήμα 5.32(β) το σχηματισμό των δεκαδικών ψηφίων. Για κάθε δεκαδικό αριθμό που σχηματίζουν τα 4 bits της εισόδου, ενεργοποιούνται (γίνονται 1) οι έξοδοι που αντιστοιχούν στα τμήματα που σχηματίζουν τον δεκαδικό αριθμό. Έτσι, αν οι εισοδοί $x_3, x_2, x_1, x_0 = 0, 0, 0, 1$, δηλαδή σχηματίζουν τον αριθμό 1, θα πρέπει να ενεργοποιηθούν τα τμήματα b και c . Ομοίως, αν οι εισοδοί $x_3, x_2, x_1, x_0 = 1, 0, 0, 1$, δηλαδή σχηματίζουν τον αριθμό 9, τότε θα πρέπει να ενεργοποιηθούν τα τμήματα $a, b, c, d, f,$ και g .

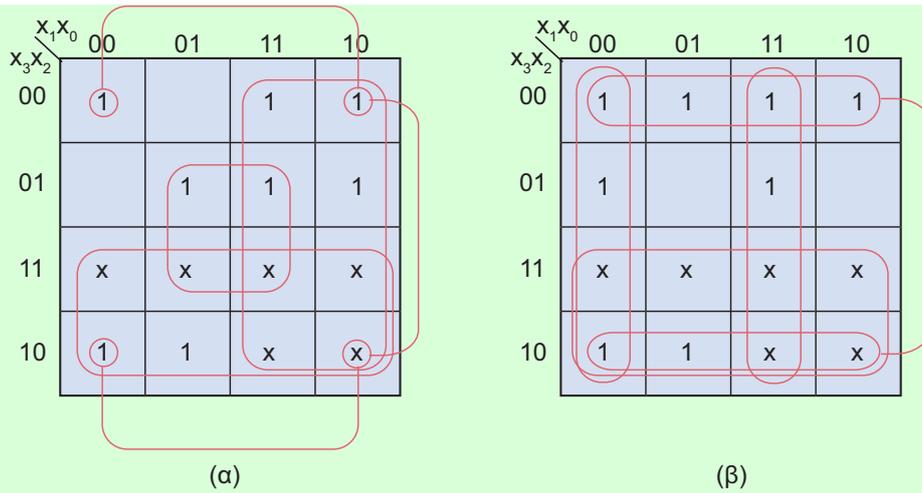


Σχήμα 5.32: Τομείς και σχηματισμός των δεκαδικών ψηφίων

Βάσει των παραπάνω, ο πίνακας αληθείας του αποκωδικοποιητή από BCD σε κώδικα 7 τομέων είναι ο ακόλουθος:

x_3	x_2	x_1	x_0	a	b	c	d	e	f	g
0	0	0	0	1	1	1	1	1	1	0
0	0	0	1	0	1	1	0	0	0	0
0	0	1	0	1	1	0	1	1	0	1
0	0	1	1	1	1	1	1	0	0	1
0	1	0	0	0	1	1	0	0	1	1
0	1	0	1	1	0	1	1	0	1	1
0	1	1	0	1	0	1	1	1	1	1
0	1	1	1	1	1	1	0	0	0	0
1	0	0	0	1	1	1	1	1	1	1
1	0	0	1	1	1	1	1	0	1	1

Οι ελαχιστόροι $m_{10} - m_{15}$ που δεν χρησιμοποιούνται, μπορούν να χρησιμοποιηθούν στους χάρτες Karnaugh για να επιτύχουμε μεγαλύτερη ελαχιστοποίηση για τις λογικές εκφράσεις των εξόδων. Στο Σχήμα 5.33(α) παρατίθεται ο χάρτης Karnaugh για την έξοδο a και στο Σχήμα 5.33(β) ο χάρτης Karnaugh για την έξοδο b . Ο αναγνώστης μπορεί εύκολα να δημιουργήσει τους χάρτες και για τις υπόλοιπες εξόδους, να εξάγει τις υπόλοιπες ελαχιστοποιημένες εκφράσεις και να σχεδιάσει το αντίστοιχο λογικό διάγραμμα που προκύπτει.



Σχήμα 5.33: Χάρτες Karnaugh για τις εξόδους a και b .

Οι ελαχιστοποιημένες εκφράσεις που προκύπτουν για τις εξόδους a και b του αποκωδικοποιητή από BCD σε κώδικα 7 τομέων είναι οι ακόλουθες:

$$a = x_3 + x_1 + x_2 x_0 + \overline{x_2} \overline{x_0}$$

$$b = x_3 + \overline{x_2} + \overline{x_1} \overline{x_0} + x_1 x_0$$

5.5 ΠΟΛΥΠΛΕΚΤΕΣ-ΑΠΟΠΟΛΥΠΛΕΚΤΕΣ

5.5.1 ΠΟΛΥΠΛΕΚΤΕΣ

Με τον όρο **πολυπλεξία** εννοούμε την επιλογή μίας εκ των πολλών εισόδων ενός ψηφιακού συστήματος και τη μεταφορά της στην έξοδο. Το κύκλωμα που υλοποιεί αυτή τη λειτουργία ονομάζεται πολυπλέκτης. Οι πολυπλέκτες αποτελούν ένα πολύ ισχυρό εργαλείο για έναν σχεδιαστή, επειδή με πολυπλέκτες μπορούμε να υλοποιήσουμε κάθε άλλο συνδυαστικό κύκλωμα. Επίσης, οι πολυπλέκτες αποτελούν σημαντικό δομικό στοιχείο των επί τόπου προγραμματιζόμενων πινάκων πυλών (FPGA), όπως θα παρουσιαστεί στο Κεφάλαιο 15. Στην ενότητα αυτή, αρχικά περιγράφεται το βασικό κύκλωμα του πολυπλέκτη και στη συνέχεια παρουσιάζονται παραδείγματα εφαρμογών του.

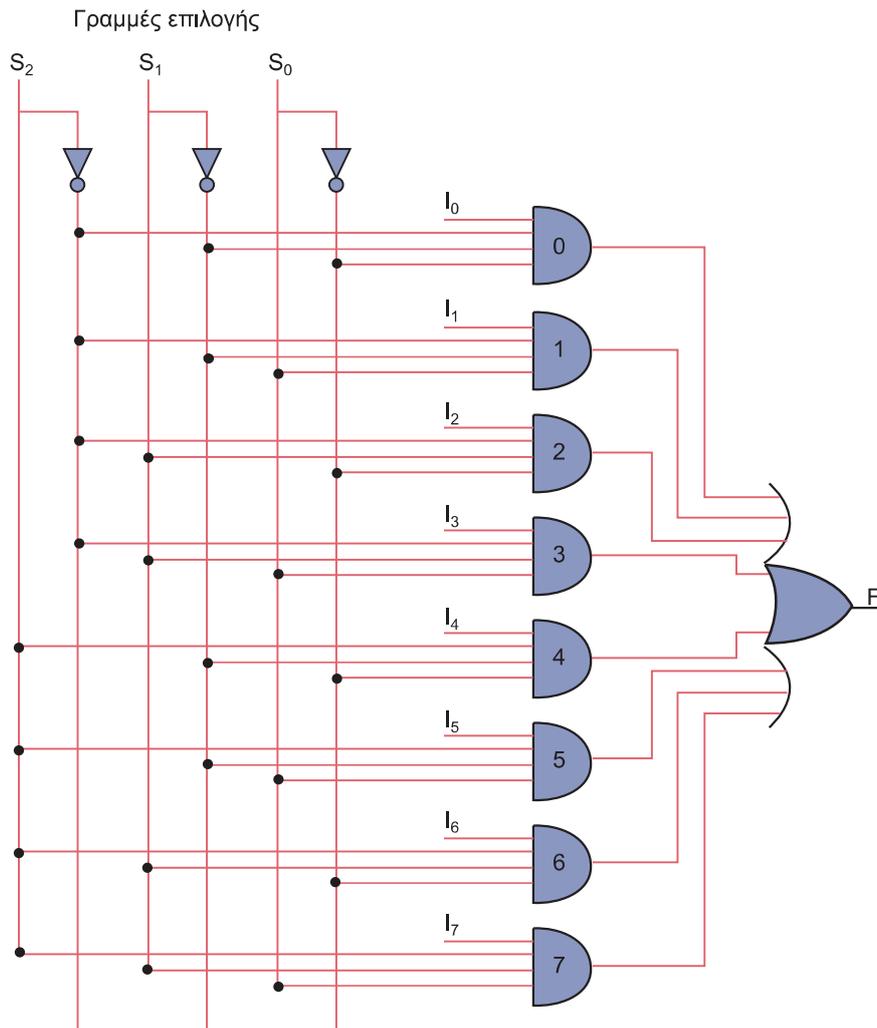
Όπως αναφέρθηκε ήδη, ο πολυπλέκτης επιλέγει τις πληροφορίες μίας από τις πολλές γραμμές εισόδου και τις μεταβιβάζει σε μια και μοναδική γραμμή εξόδου. Η επιλογή της γραμμής εισόδου γίνεται με τη βοήθεια γραμμών επιλογής. Τυπικά, το πλήθος των γραμμών εισόδου είναι 2^n . Επομένως, απαιτούνται n γραμμές επιλογής, ώστε να επιλεγεί μία από τις 2^n γραμμές εισόδου και να οδηγηθεί στην έξοδο. Ένας πολυπλέκτης με 2^n γραμμές εισόδου, n γραμμές επιλογής και μία έξοδο, αναφέρεται και ως πολυπλέκτης 2^n -σε-1.

Έστω ότι θέλουμε να υλοποιήσουμε έναν πολυπλέκτη 8-σε-1. Αυτό σημαίνει ότι απαιτούνται $n = 3$ γραμμές επιλογής, οι οποίες συμβολίζονται με S_0, S_1 , και S_2 . Οι 8 εισοδοί συμβολίζονται με $I_0 - I_7$. Η λεκτική περιγραφή της λειτουργίας αυτού του πολυπλέκτη είναι η εξής: “από το σύνολο των εισόδων, επιλέγεται για να βγει στην έξοδο F εκείνη η γραμμή, της οποίας η τιμή σχηματίζεται από τις γραμμές επιλογής”. Με βάση την παραπάνω περιγραφή, ο πίνακας αληθείας του κυκλώματος ενός πολυπλέκτη 8-σε-1 δίνεται στον Πίνακα 5.5. Η υλοποίηση του πολυπλέκτη 8-σε-1 δίνεται στο Σχήμα 5.34.

Για κάθε είσοδο χρησιμοποιείται έλεγχος διάδοσης όπως περιγράφηκε στην Ενότητα 5.2. Πιο συγκεκριμένα, σε κάθε πύλη ΚΑΙ υπάρχουν τρεις είσοδοι ελέγχου διάδοσης. Όταν οι γραμμές επιλογής S_2, S_1 , και S_0 είναι ίσες με 0, τότε οι αντιστροφείς που συνδέονται με τις γραμμές επιλογής μεταφέρουν τρεις μονάδες στην είσοδο της πύλης ΚΑΙ με αριθμό 0. Ταυτόχρονα, σε όλες τις υπόλοιπες πύλες ΚΑΙ υπάρχει τουλάχιστο ένα μηδενικό στην είσοδο τους, το οποίο αναγκάζει την έξοδο τους να πάει στο 0. Επομένως, η έξοδος F του πολυπλέκτη εξαρτάται αποκλειστικά από την τιμή του σήματος I_0 . Παρομοίως, όταν οι γραμμές S_2, S_1 , και S_0 είναι ίσες με 1, τότε οι αντιστροφείς που συνδέονται με τις γραμμές επιλογής μεταφέρουν τρεις μονάδες στην είσοδο της πύλης ΚΑΙ με αριθμό 7. Ταυτόχρονα, σε όλες τις υπόλοιπες πύλες ΚΑΙ υπάρχει τουλάχιστο ένα μηδενικό στην είσοδο τους, το οποίο αναγκάζει την έξοδο τους να

Πίνακας 5.5: Πίνακας αληθείας του πολυπλέκτη 8-σε-1

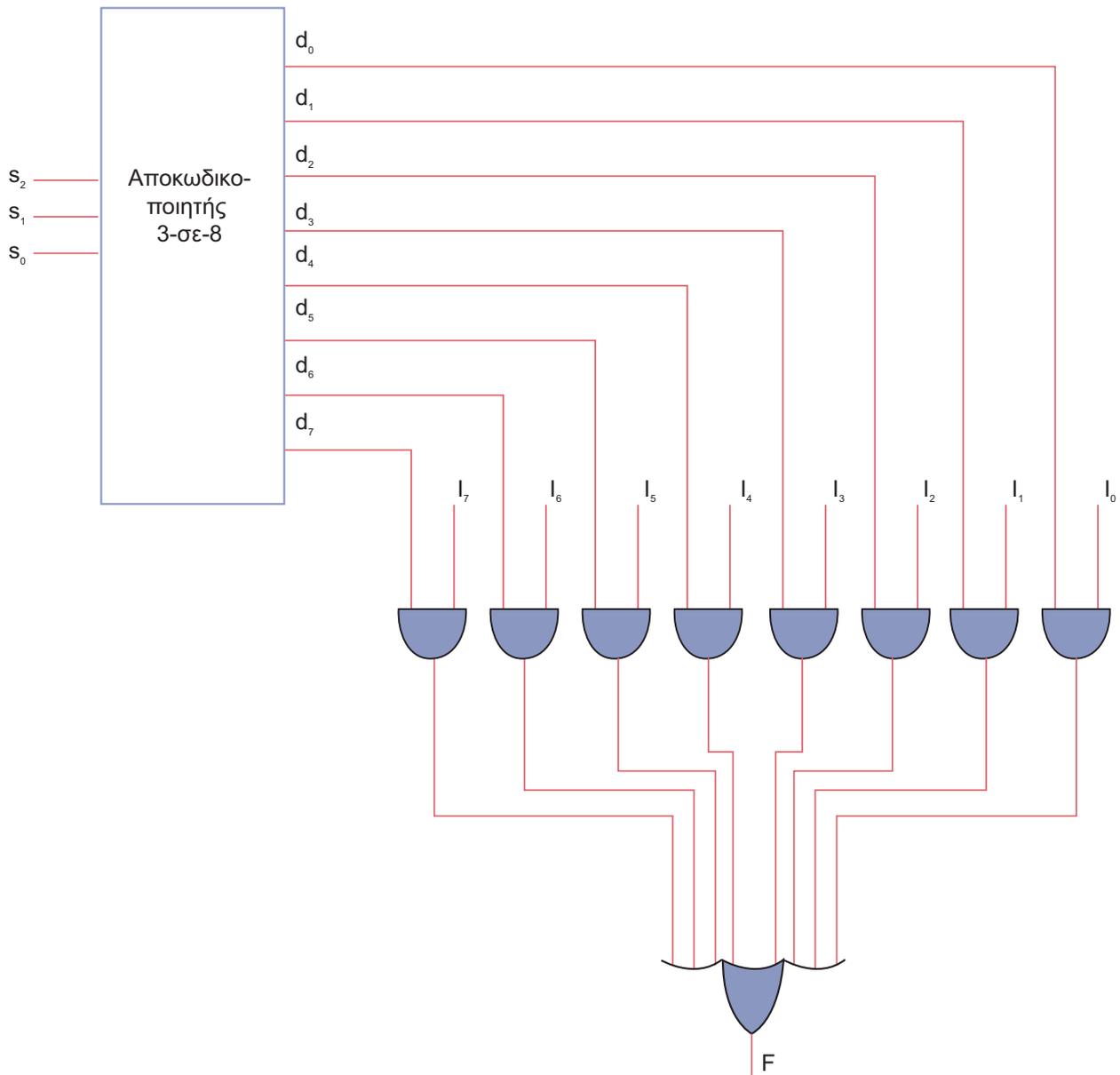
S_2	S_1	S_0	F
0	0	0	I_0
0	0	1	I_1
0	1	0	I_2
0	1	1	I_3
1	0	0	I_4
1	0	1	I_5
1	1	0	I_6
1	1	1	I_7



Σχήμα 5.34: Λογικό διάγραμμα του πολυπλέκτη 8-σε-1.

πάει στο 0. Με παρόμοιο σκεπτικό, επαληθεύεται ο πίνακας αληθείας του πολυπλέκτη και για τους υπόλοιπους συνδυασμούς των γραμμών επιλογής.

Ο τρόπος λειτουργίας του πολυπλέκτη μοιάζει πολύ με τον αντίστοιχο του αποκωδικοποιητή. Ένας πολυπλέκτης 2^n -σε-1 είναι στην πραγματικότητα ένας αποκωδικοποιητής n -σε- 2^n , του οποίου οι έξοδοι επιτρέπουν μια από τις 2^n γραμμές εισόδου, $I_0 - I_{2^n-1}$. Αυτό απεικονίζεται στο λογικό διάγραμμα του Σχήματος 5.35. Οι γραμμές επιλογής S_2, S_1, S_0 τοποθετούνται ως εισοδοί ενός αποκωδικοποιητή 3-σε-8. Κάθε συνδυασμός των γραμμών επιλογής ενεργοποιεί μία από τις εξόδους του αποκωδικοποιητή $d_0 - d_7$. Κάθε μία από τις εξόδους d_j τοποθετείται ως είσοδος σε μία πύλη ΚΑΙ δύο



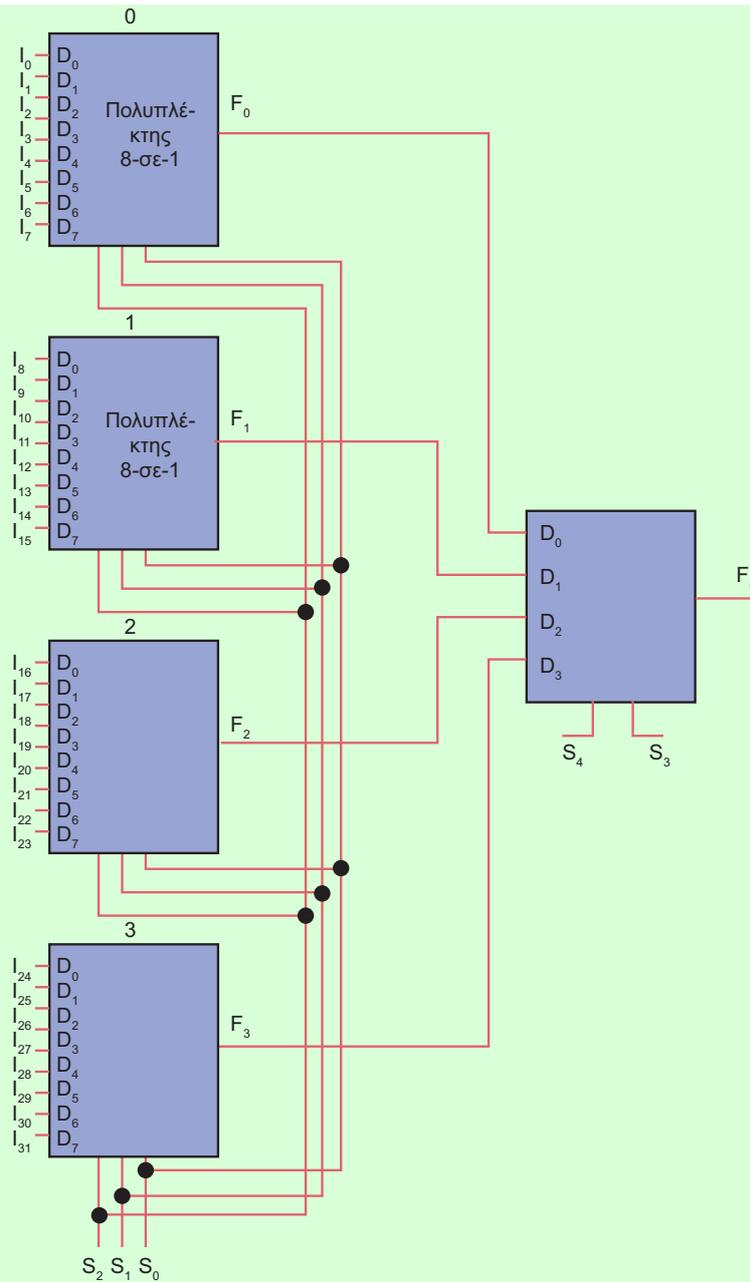
Σχήμα 5.35: Ισοδύναμη υλοποίηση πολυπλέκτη 8-σε-1

εισόδων μαζί με την είσοδο I_j . Κάθε φορά, μόνον μία εκ των d_j είναι ίση με 1, οπότε η έξοδος F εξαρτάται από την τιμή της εισόδου I_j . Οι υλοποιήσεις των Σχημάτων 5.34 και 5.35 είναι ισοδύναμες.

Οι πολυπλέκτες μπορούν να συνδυαστούν μεταξύ τους, σχηματίζοντας μεγαλύτερα κυκλώματα πολυπλεκτών. Στο Παράδειγμα 5.11, τέσσερις πολυπλέκτες 8-σε-1 συνδέονται στις εισόδους ενός πέμπτου πολυπλέκτη 4-σε-1, υλοποιώντας έναν πολυπλέκτη 32-σε-1.

ΠΑΡΑΔΕΙΓΜΑ 5.11 Να υλοποιηθεί ένας πολυπλέκτης 32-σε-1 με τέσσερις πολυπλέκτες 8-σε-1 και έναν πολυπλέκτη 4-σε-1.

Η υλοποίηση του κυκλώματος δίνεται στο Σχήμα 5.36.



Σχήμα 5.36: Πολυπλέκτης 32-σε-1

Οι εισοδοι επιλογής S_2 , S_1 και S_0 είναι συνδεδεμένες με τους τέσσερις πολυπλέκτες 8-σε-1, οι οποίοι αριθμούνται από 0-3 (ο αριθμός κάθε πολυπλέκτη βρίσκεται στο πάνω μέρος του). Οι εισοδοι S_2 , S_1 και S_0 χρησιμοποιούνται για να καθορίσουν ποιά από τις 8 εισόδους κάθε πολυπλέκτη θα μεταφερθεί στις εξόδους F_0 ως F_3 . Οι εξοδοι F_0 ως F_3 τοποθετούνται ως εισοδοι στον πολυπλέκτη 4-σε-1. Οι γραμμές επιλογής S_3 και S_2 αυτού του πολυπλέκτη επιλέγουν ποιά από τις εισόδους F_0 ως F_3 θα μεταφερθεί στην έξοδο F_5 . Με άλλα λόγια, οι τιμές των S_3 , S_2 , S_1 , S_0 σχηματίζουν την τιμή του δείκτη της εισόδου η οποία θα μεταφερθεί στην έξοδο F_5 . Για να φανεί αυτό καλύτερα, ας εξηγήσουμε τον τρόπο επιλογής της εισόδου I_{14} . Η γραμμή εισόδου I_{14} συνδέεται με την είσοδο D_6 του πολυπλέκτη 1. Παρατηρείστε ότι οι εισοδοι D_i των τεσσάρων πολυπλεκτών 8-σε-1 έχουν τους ίδιους δείκτες. Για να επιλεγεί η είσοδος D_6 , θα πρέπει οι εισοδοι S_2 , S_1 , S_0 να σχηματίζουν τον αριθμό 6 ή $(110)_2$, δηλαδή $S_2 = 1$, $S_1 = 1$, και $S_0 = 0$. Τότε, η έξοδος F_0 θα ισούται με I_6 , η έξοδος F_1 θα ισούται με I_{14} , η έξοδος F_2 θα ισούται με I_{22} , και η έξοδος F_3 θα ισούται με I_{30} . Κατά συνέπεια, οι εισοδοι D_0 , D_1 , D_2 , D_3 του πολυπλέκτη 4-σε-1 θα είναι:

$$D_0 = I_6, D_1 = I_{14}, D_2 = I_{22}, D_3 = I_{30}.$$

Για να επιλεγεί η είσοδος I_{14} , θα πρέπει να επιλεγεί η είσοδος D_1 του πολυπλέκτη. Επομένως, θα πρέπει οι γραμμές επιλογής S_4, S_3 να σχηματίζουν τον αριθμό 1 ή $(01)_2$, δηλαδή $S_4 = 0, S_3 = 1$. Αν λάβουμε συνολικά τις τιμές των S_4, S_3, S_2, S_1, S_0 , δηλαδή 0, 1, 1, 1, 0, σχηματίζεται ο αριθμός $(01110)_2 = (14)_{10}$.

Υλοποίηση Συνδυαστικών Κυκλωμάτων με Πολυπλέκτες Οι πολυπλέκτες μπορούν να χρησιμοποιηθούν σε μια πληθώρα εφαρμογών των ψηφιακών συστημάτων. Στην παράγραφο αυτή εξετάζεται ο τρόπος που μπορεί να υλοποιηθεί ένα συνδυαστικό κύκλωμα εκφρασμένο σε μορφή αθροίσματος γινομένων. Η κεντρική ιδέα βασίζεται στο γεγονός ότι το τμήμα επιλογής της γραμμής εισόδου ενός πολυπλέκτη 2^n -σε-1 είναι ένας αποκωδικοποιητής n -σε- 2^n . Επομένως, οι γραμμές επιλογής μπορούν να χρησιμοποιηθούν ως γεννήτριες ελαχιστόρων, ενώ οι γραμμές εισόδου μπορούν να χρησιμοποιηθούν ώστε να επιλεγούν οι κατάλληλοι ελαχιστοί της συνάρτησης. Έστω μία λογική συνάρτηση f , με $k + 1$ μεταβλητές εισόδου, άρα 2^{k+1} ελαχιστόρους. Για να υλοποιήσουμε τη συνάρτηση αυτή, ακολουθούμε την εξής διαδικασία :

1. Χρησιμοποιούμε έναν πολυπλέκτη n -σε-1, όπου $n = 2^k$. Ο πολυπλέκτης διαθέτει $\log_2 2^k = k$ γραμμές επιλογής.
2. Συνδέουμε τις k γραμμές επιλογής του πολυπλέκτη με τις k μεταβλητές εισόδου της συνάρτησης.
3. Η μεταβλητή της συνάρτησης που περισσεύει χρησιμοποιείται μαζί με τις σταθερές 0 και 1 για τις τιμές των γραμμών εισόδου.

Η παραπάνω διαδικασία γίνεται πιο σαφής με τα παραδείγματα που ακολουθούν :

ΠΑΡΑΔΕΙΓΜΑ 5.12 Να υλοποιηθούν με χρήση κατάλληλων πολυπλεκτών οι παρακάτω λογικές συναρτήσεις

- (α) $F_1(A, B, C) = \sum(0, 1, 3, 6)$
 (β) $F_2(A, B, C, D) = \sum(0, 1, 3, 5, 6, 9, 10, 12)$

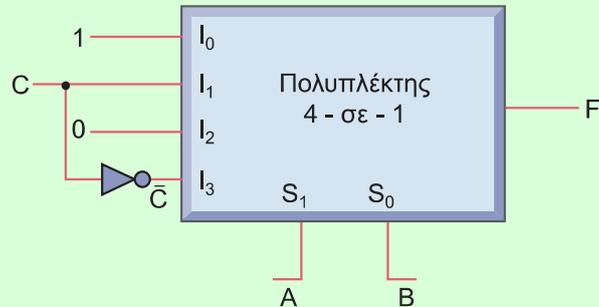
(α) Είναι $k = 2$. Βάσει του πρώτου βήματος, θα χρησιμοποιήσουμε έναν πολυπλέκτη 4-σε-1 ($n = 2^2$). Ο πολυπλέκτης περιλαμβάνει 4 εισόδους $I_0 - I_3$ και 2 γραμμές επιλογής, S_1 και S_0 . Από το βήμα 2, η σχεδίαση βασίζεται στην αντιστοίχιση των 2 γραμμών επιλογής με τις δύο μεταβλητές A και B . Από το βήμα 3, η τρίτη μεταβλητή C χρησιμοποιείται για τη σύνδεση με τις 4 εισόδους του πολυπλέκτη. Για να επεξηγηθεί ο τρόπος με τον οποίο η παραπάνω διασύνδεση υλοποιεί την F_1 , θα πρέπει αρχικά να παρατεθεί ο πίνακας αληθείας της συνάρτησης, ο οποίος προκύπτει άμεσα από τη λογική έκφρασή της.

Από τον πίνακα αληθείας, είναι φανερό ότι όταν $A = B = 0$, η συνάρτηση F_1 δίνει έξοδο 1, ανεξάρτητα από την τιμή που θα έχει η μεταβλητή C . Αντιστοιχίζοντας με την λειτουργία του πολυπλέκτη, όταν οι γραμμές S_1 και S_0 είναι ίσες με 0, επιλέγεται η γραμμή I_0 . Αυτό σημαίνει ότι αν ορίσουμε ότι το σήμα I_0 θα είναι σταθερά ίσο με 1, υλοποιούνται οι 2 πρώτες γραμμές του πίνακα αληθείας.

A	B	C	F_1
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	0

Όταν $A = 0$ και $B = 1$, η έξοδος F_1 καθορίζεται από την τιμή του C . Επομένως, η λειτουργία του πολυπλέκτη θα επαληθεύει τις γραμμές 3 και 4 του πίνακα αληθείας αν συνδέσουμε το σήμα

εισόδου I_1 με τη μεταβλητή C . Όταν $A = 1$ και $B = 0$, η συνάρτηση F_1 δίνει έξοδο 0, ανεξάρτητα με την τιμή της εισόδου C . Επομένως, ο πολυπλέκτης υλοποιεί τις γραμμές 5-6 του πίνακα αληθείας της συνάρτησης, αν θέσουμε το σήμα εισόδου του I_2 ίσο σταθερά με 0. Τέλος, όταν $A = B = 1$, η τιμή της F_1 είναι ίση με \bar{C} . Άρα οι δύο τελευταίες γραμμές του πίνακα αληθείας επαληθεύονται από τη λειτουργία του πολυπλέκτη, αν συνδέσουμε το σήμα εισόδου I_3 με την είσοδο \bar{C} . Το λογικό διάγραμμα της υλοποίησης της συνάρτησης $F_1(A, B, C) = \sum(0, 1, 3, 6)$, δίνεται στο Σχήμα 5.37.

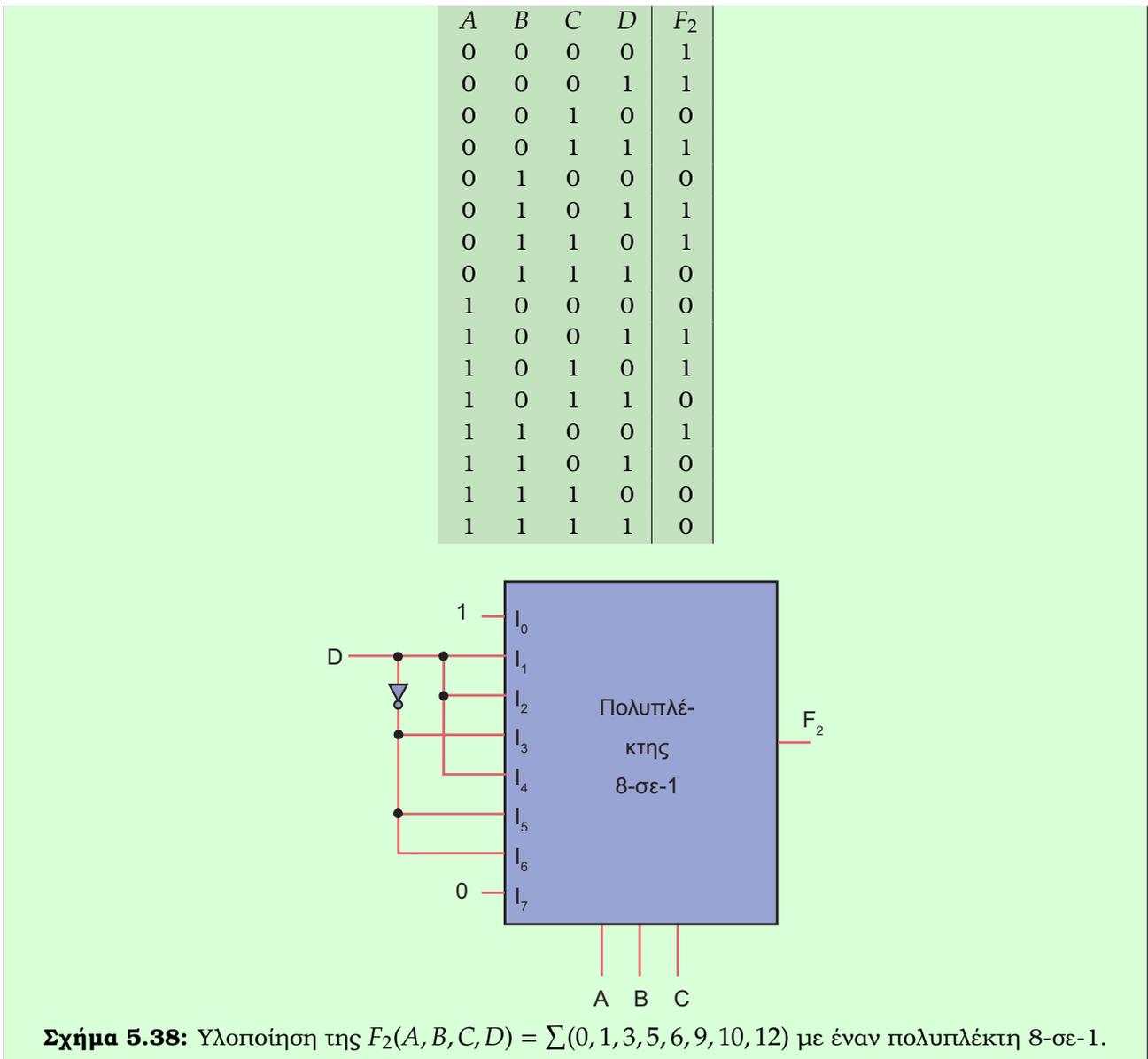


Σχήμα 5.37: Υλοποίηση της $F_1(A, B, C) = \sum(0, 1, 3, 6)$ με έναν πολυπλέκτη 4-σε-1.

(β) Είναι $k = 3$. Βάσει του πρώτου βήματος, θα χρησιμοποιήσουμε έναν πολυπλέκτη 8-σε-1 ($n = 2^3 = 8$). Ο πολυπλέκτης περιλαμβάνει 8 εισόδους $I_0 - I_7$ και 3 γραμμές επιλογής, S_2, S_1 και S_0 . Από το βήμα 2, η σχεδίαση βασίζεται στην αντιστοίχιση των τριών γραμμών επιλογής με τις μεταβλητές A, B και C . Βάσει του βήματος 3, η τέταρτη μεταβλητή D χρησιμοποιείται για τη σύνδεση με τις 8 εισόδους του πολυπλέκτη. Για να επεξηγηθεί ο τρόπος με τον οποίο η παραπάνω διασύνδεση υλοποιεί την F_2 , θα πρέπει αρχικά να παρατεθεί ο πίνακας αληθείας της συνάρτησης, ο οποίος προκύπτει άμεσα από τη λογική έκφρασή της.

Από τον πίνακα αληθείας, είναι φανερό ότι όταν $A = B = C = 0$, η συνάρτηση F_2 δίνει έξοδο 1, ανεξάρτητα με την τιμή που θα έχει η μεταβλητή D . Αντιστοιχίζοντας με την λειτουργία του πολυπλέκτη, όταν οι γραμμές S_2, S_1 και S_0 είναι ίσες με 0, επιλέγεται η γραμμή I_0 . Αυτό σημαίνει ότι αν ορίσουμε ότι το σήμα I_0 είναι σταθερά ίσο με 1, υλοποιούνται οι 2 πρώτες γραμμές του πίνακα αληθείας. Όταν $(A, B, C) = (0, 0, 1)$ ή όταν $(A, B, C) = (0, 1, 0)$ ή όταν $(A, B, C) = (1, 0, 0)$, η έξοδος F_2 καθορίζεται από την τιμή του C . Επομένως, η λειτουργία του πολυπλέκτη θα επαληθεύει τις γραμμές 3, 4, 5, 6, 9, και 10 του πίνακα αληθείας αν συνδέσουμε τα σήματα εισόδου I_1, I_2, I_4 με τη μεταβλητή C . Όταν $(A, B, C) = (0, 1, 1)$ ή όταν $(A, B, C) = (1, 0, 0)$ ή όταν $(A, B, C) = (1, 1, 0)$ η τιμή της F_2 είναι ίση με \bar{C} . Επομένως, η λειτουργία του πολυπλέκτη θα επαληθεύει τις γραμμές 7, 8, 11, 12, 13, και 14 του πίνακα αληθείας αν συνδέσουμε τα σήματα εισόδου I_3, I_5, I_6 με τη μεταβλητή \bar{C} . Τέλος, όταν $A = B = C = 0$, η συνάρτηση F_2 δίνει έξοδο 1, ανεξάρτητα με την τιμή που θα έχει η μεταβλητή D . Επομένως, η λειτουργία του πολυπλέκτη θα επαληθεύει τις γραμμές 15 και 16 του πίνακα αληθείας αν ορίσουμε ότι το σήμα I_7 είναι σταθερά ίσο με 0. Το λογικό διάγραμμα της υλοποίησης της $F_2(A, B, C, D) = \sum(0, 1, 3, 5, 6, 9, 10, 12)$, δίνεται στο Σχήμα 5.38.

Πίνακας αληθείας



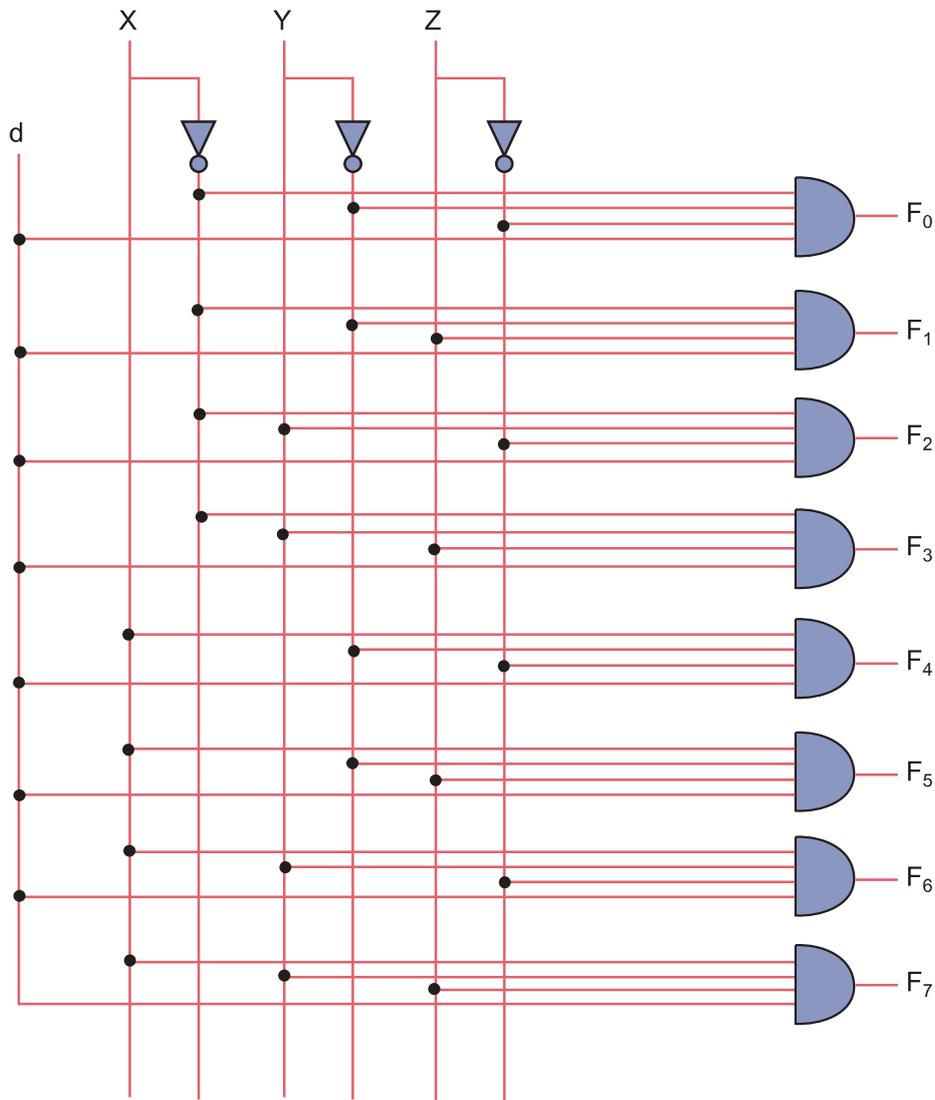
Σχήμα 5.38: Υλοποίηση της $F_2(A, B, C, D) = \sum(0, 1, 3, 5, 6, 9, 10, 12)$ με έναν πολυπλέκτη 8-σε-1.

5.5.2 ΑΠΟΠΟΛΥΠΛΕΚΤΕΣ

Με τον όρο **αποπολυπλεξία** εννοούμε την μεταφορά μίας και μοναδικής εισόδου σε μία από τις πολλές διαθέσιμες εξόδους ενός ψηφιακού συστήματος. Η αποπολυπλεξία είναι η αντίστροφη λειτουργία από την πολυπλεξία και υλοποιείται από ένα κύκλωμα, το οποίο ονομάζεται αποπολυπλέκτης. Τυπικά, ένας αποπολυπλέκτης δέχεται πληροφορίες από μία γραμμή εισόδου δεδομένων, η οποία συμβολίζεται ως d (από τη λέξη “data”), και τη μεταβιβάζει σε μια από τις 2^n διαθέσιμες γραμμές εξόδου. Το κύκλωμα αυτό αναφέρεται και ως αποπολυπλέκτης 1-σε- 2^n . Για να επιλεγεί η γραμμή εξόδου που θα λάβει την τιμή της εισόδου, υπάρχουν n γραμμές επιλογής. Εκτός της επιλεγμένης εξόδου η οποία λαμβάνει την τιμή της εισόδου, όλες οι άλλες εξοδοί μηδενίζονται.

Για παράδειγμα, ας θεωρήσουμε ότι $n = 3$. Ο πίνακας αληθείας του κυκλώματος αποπολυπλέκτη 1-σε-8, με 3 γραμμές επιλογής X, Y, Z και 8 εξόδους $F_0 - F_7$, δίνεται στον Πίνακα 5.6. Η είσοδος δεδομένων d θεωρείται ότι έχει μία τιμή, έστω c , όπου $c = 0$ ή 1 . Η τιμή αυτή μεταφέρεται σε μία από τις εξόδους $F_0 - F_7$ με βάση την τιμή των εισόδων X, Y , και Z . Το λογικό διάγραμμα ενός αποπολυπλέκτη 3-σε-8 δίνεται στο Σχήμα 5.39.

Το κύκλωμα του αποπολυπλέκτη λειτουργεί πανομοιότυπα με ένα κύκλωμα αποκωδικοποιητή ο οποίος περιέχει μια είσοδο επίτρεψης. Πράγματι, έστω ότι ένας αποκωδικοποιητής διαθέτει ένα σήμα επίτρεψης E (Enable), κάτι το οποίο συμβαίνει αρκετά συχνά. Η λειτουργία του σήματος επίτρεψης είναι



Σχήμα 5.39: Λογικό διάγραμμα του αποπολυπλέκτη 1-σε-8.

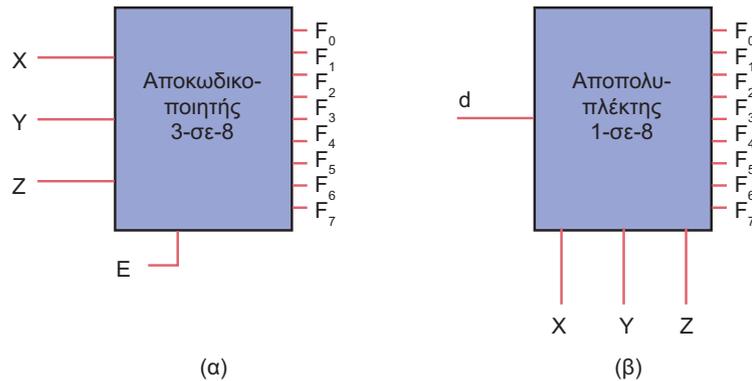
Πίνακας 5.6: Πίνακας αληθείας του αποπολυπλέκτη 1-σε-8

X	Y	Z	d	F ₀	F ₁	F ₂	F ₃	F ₄	F ₅	F ₆	F ₇
0	0	0	c	c	0	0	0	0	0	0	0
0	0	1	c	0	c	0	0	0	0	0	0
0	1	0	c	0	0	c	0	0	0	0	0
0	1	1	c	0	0	0	c	0	0	0	0
1	0	0	c	0	0	0	0	c	0	0	0
1	0	1	c	0	0	0	0	0	c	0	0
1	1	0	c	0	0	0	0	0	0	c	0
1	1	1	c	0	0	0	0	0	0	0	c

να επιτρέπει ή να αποτρέπει τη λειτουργία του αποκωδικοποιητή. Ειδικότερα, ας θεωρήσουμε ότι :

- α. Όταν $E = 0$ απενεργοποιείται η λειτουργία του αποκωδικοποιητή και όλες οι εξοδοι λαμβάνουν τιμή 0.
- β. Όταν $E = 1$ ενεργοποιείται η λειτουργία του αποκωδικοποιητή και οι τιμές των εξόδων καθορίζονται από τον τρόπο που γίνεται η αποκωδικοποίηση.

Αν θεωρήσουμε έναν αποκωδικοποιητή 3-σε-8 με είσοδο επίτρεψης, όπου X , Y , Z είναι οι γραμμές εισόδου και d η είσοδος επίτρεψης, το κύκλωμα αυτού του αποκωδικοποιητή είναι ισοδύναμο με αυτό του αποπολυπλέκτη του Σχήματος 5.39. Αν $d = 0$, το κύκλωμα απενεργοποιείται και οι εξοδοί του είναι ίσες με 0. Αν $d = 1$, τότε μία από τις εξόδους λαμβάνει τιμή ίση με 1, ανάλογα με τις τιμές των εισόδων X , Y , Z . Τα παραπάνω μπορούν να γίνουν κατανοητά και με τη βοήθεια ενός σχηματικού διαγράμματος του αποκωδικοποιητή 3-σε-8 με σήμα επίτρεψης και του αποπολυπλέκτη 1-σε-8. Το σχηματικό διάγραμμα δίνεται στο Σχήμα 5.40. Παρατηρείστε ότι με μία απλή αντιμετάθεση της θέσης των X , Y , Z με το σήμα d του Σχήματος 5.40(β), λαμβάνουμε το Σχήμα 5.40(α). Ομοίως, με μία απλή αντιμετάθεση της θέσης των X , Y , Z με το σήμα E του Σχήματος 5.40(α), λαμβάνουμε το Σχήμα 5.40(β).



Σχήμα 5.40: Σχηματικά διαγράμματα αποκωδικοποιητή 3-σε-8 με σήμα επίτρεψης και αποπολυπλέκτη 1-σε-8.

5.6 ΑΘΡΟΙΣΤΕΣ

Στο πρώτο κεφάλαιο εξετάστηκε η θεωρία πάνω στην οποία βασίζονται οι αριθμητικές πράξεις στο δυαδικό σύστημα. Σε αυτήν την ενότητα θα παρουσιάσουμε τη σχεδίαση των κυκλωμάτων που υλοποιούν τις βασικές αριθμητικές πράξεις της πρόσθεσης και της αφαίρεσης.

5.6.1 ΗΜΙΑΘΡΟΙΣΤΕΣ

Το κύκλωμα, το οποίο υλοποιεί την πρόσθεση 2 bits χωρίς να λαμβάνει υπόψη την ύπαρξη πιθανού κρατούμενου, ονομάζεται *ημιαθροιστής (half-adder)*. Η λεκτική περιγραφή του κυκλώματος του ημιαθροιστή είναι η εξής: "ο ημιαθροιστής είναι ένα κύκλωμα, το οποίο προσθέτει 2 bits εισόδου και δίνει ως αποτέλεσμα το άθροισμα τους και ένα κρατούμενο". Με βάση αυτή την περιγραφή, ο ημιαθροιστής περιλαμβάνει 2 εισόδους, έστω x και y , οι οποίες υποδέχονται τα 2 bits που προστίθενται και 2 εξόδους, μία για το άθροισμα που συμβολίζεται με S (sum) και μία για το κρατούμενο που συμβολίζεται με C (carry), όπως φαίνεται στον Πίνακα 5.7.

Ο πίνακας προκύπτει με βάση τις 4 δυνατές περιπτώσεις ως προς την απλή πρόσθεση δύο δυαδικών ψηφίων, δηλαδή $0+0=0$, $0+1=1$, $1+0=1$, $1+1=0$ και 1 κρατούμενο. Από τον πίνακα αληθείας είναι φανερό ότι το κρατούμενο είναι 1 μόνον όταν τα 2 ψηφία που προστίθενται είναι ίσα με τη μονάδα, διαφορετικά είναι 0. Επίσης, το άθροισμα είναι 1 όταν μεταξύ των 2 προστιθέμενων bits υπάρχει μια μόνο μονάδα, σε διαφορετική περίπτωση, το άθροισμα είναι 0. Από την παραπάνω διαπίστωση μπορούν να προκύψουν απλοποιημένες μορφές των λογικών εκφράσεων των 2 εξόδων:

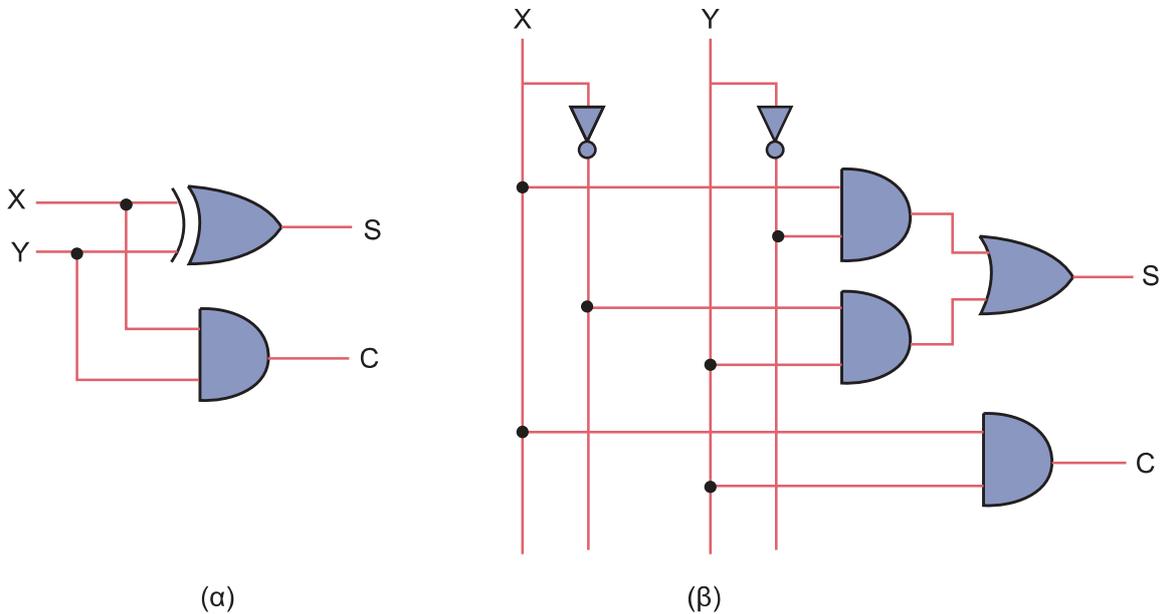
$$S = \bar{x}y + x\bar{y} = x \oplus y$$

$$C = xy$$

Πίνακας 5.7: Πίνακας αληθείας του ημιαθροιστή

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

Το λογικό διάγραμμα του κυκλώματος του ημιαθροιστή με χρήση μιας πύλης αποκλειστικό Ή και μιας ΚΑΙ δίνεται στο Σχήμα 5.41(α). Στο Σχήμα 5.41(β) δίνεται η υλοποίηση με πύλες ΚΑΙ και Ή. Δύο κυκλώματα ημιαθροιστή μπορούν να συνδυαστούν υλοποιώντας έναν πλήρη αθροιστή, όπως θα δούμε στην επόμενη παράγραφο.



Σχήμα 5.41: Δύο υλοποιήσεις του κυκλώματος του ημιαθροιστή

5.6.2 ΠΛΗΡΕΙΣ ΑΘΡΟΙΣΤΕΣ

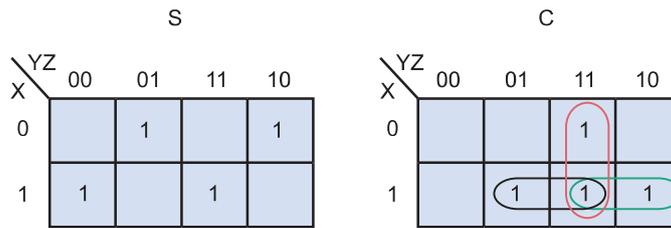
Ο πλήρης αθροιστής είναι ένα κύκλωμα, το οποίο προσθέτει 3 bits, συμπεριλαμβανομένου και του bit κρατούμενου. Αυτό σημαίνει ότι ο πλήρης αθροιστής έχει τρεις εισόδους. Οι δύο από αυτές, x και y υποδέχονται τα προστιθέμενα bits ενώ τρίτη είσοδος z υποδέχεται την τιμή που έχει το κρατούμενο που προέκυψε από τυχόν πρόσθεση στην αμέσως λιγότερο σημαντική θέση. Οι δύο έξοδοι C και S χρησιμοποιούνται για το άθροισμα και τυχόν κρατούμενο που θα προκύψει από την εκτέλεση της πράξης, αντίστοιχα. Ο πίνακας αληθείας του πλήρους αθροιστή δίνεται στον Πίνακα 5.8. Είναι φανερό ότι η έξοδος αθροίσματος S είναι ίσο με 1, όταν το πλήθος των μονάδων που υπάρχει στις 3 εισόδους είναι περιττό, διαφορετικά το άθροισμα είναι 0. Επίσης, η έξοδος κρατούμενου C είναι 1 αν το πλήθος των μονάδων στις εισόδους είναι 2 ή 3.

Για να απλοποιήσουμε τις λογικές εκφράσεις για τις εξόδους S και C, χρησιμοποιούμε 2 χάρτες Karnaugh, όπως φαίνεται στο Σχήμα 5.42 Η υλοποίηση του πλήρους αθροιστή με πύλες ΚΑΙ και Ή, δίνεται στο Σχήμα 5.43(α).

Από το χάρτη Karnaugh είναι φανερό ότι η λογική έκφραση για την έξοδο S δεν επιδέχεται καμία απλοποίηση, δεδομένου ότι δεν μπορεί να δημιουργηθεί καμία ομάδα από μονάδες. Είναι: $S = \bar{x}\bar{y}z +$

Πίνακας 5.8: Πίνακας αληθείας του πλήρους αθροιστή

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



Σχήμα 5.42: Χάρτες Karnaugh για τις εξόδους S και C του κυκλώματος του πλήρους αθροιστή

$\bar{x} y \bar{z} + x \bar{y} \bar{z} + x y z = \bar{z} (\bar{x} y + x \bar{y}) + z (\bar{x} \bar{y} + x y) = \bar{z} (x \oplus y) + z \overline{(x \oplus y)} = z \oplus (x \oplus y) = z \oplus x \oplus y$. Αντίθετα, η έξοδος C μπορεί να απλοποιηθεί δημιουργώντας 3 ομάδες των 2 μονάδων. Οι τελικές μορφές των λογικών εκφράσεων των S και C είναι οι ακόλουθες:

$$S = z \oplus x \oplus y. \tag{5.1}$$

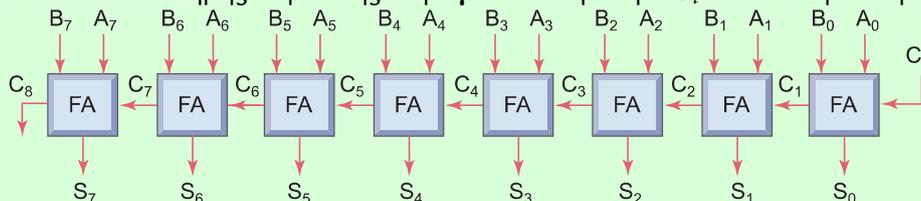
$$C = xz + xy + yz. \tag{5.2}$$

Όπως αναφέρθηκε στην προηγούμενη παράγραφο, ένας πλήρης αθροιστής μπορεί να υλοποιηθεί χρησιμοποιώντας 2 ημιαθροιστές. Για να υλοποιηθεί αυτή η σχεδίαση, θεωρήστε τον πίνακα αληθείας του πλήρους αθροιστή, από τον οποίο προκύπτει ότι το κρατούμενο εξόδου είναι το άθροισμα των ελαχιστόρων m_3, m_5, m_6 , και m_7 . Από τους ελαχιστόρους m_3 και m_5 προκύπτει το άθροισμα $x \bar{y} z + \bar{x} y z$ ή $(x \oplus y) z$. Από τους ελαχιστόρους m_3 και m_5 προκύπτει το γινόμενο $x y$. Άρα, $C = x y + (x \oplus y) z$. Επίσης, από την (5.3) έχουμε ότι $S = z \oplus x \oplus y$. Η υλοποίηση των δύο αυτών συναρτήσεων δίνεται στο Σχήμα 5.43(β), το οποίο αποτελείται από 2 ημιαθροιστές και μία πύλη Η.

Αν συνδεθούν σε μορφή αλυσίδας πολλοί πλήρεις αθροιστές, μπορεί να υλοποιηθεί ένα κύκλωμα πλήρους αθροιστή των n bits. Το Παράδειγμα 5.13 παρουσιάζει αυτήν την υλοποίηση.

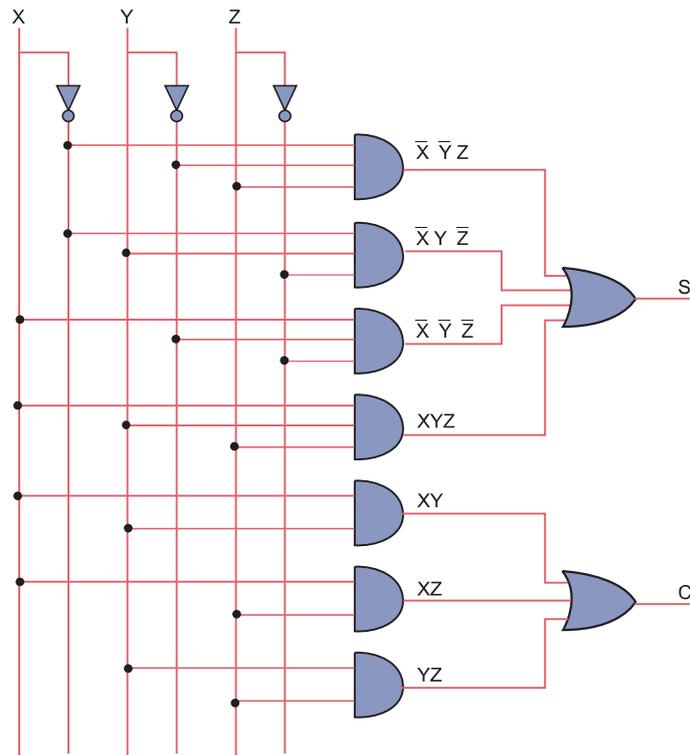
ΠΑΡΑΔΕΙΓΜΑ 5.13 Υλοποίηση αθροιστή 8 bit αποτελούμενου από 8 απλούς αθροιστές.

Το λογικό διάγραμμα του αθροιστή 8 bit δίνεται στο Σχήμα 5.44. Το κρατούμενο εισόδου του δεξιότερου αθροιστή είναι $C_0 = 0$. Τα κρατούμενα διαδίδονται από τα δεξιά προς τα αριστερά μεταξύ των αθροιστών και κάθε πλήρης αθροιστής παράγει ένα άθροισμα S_i και ένα κρατούμενο C_{i+1} .

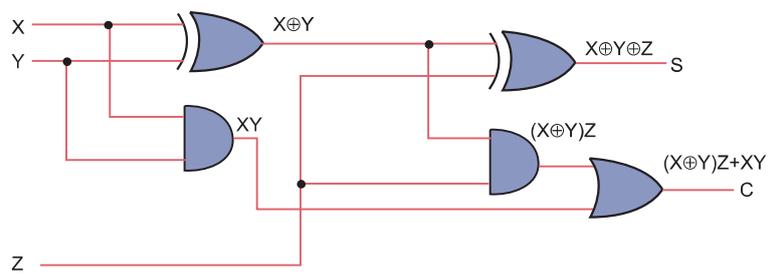


Σχήμα 5.44: Υλοποίηση αθροιστή 8 bit με 8 πλήρεις αθροιστές.

Για να επεξηγηθεί καλύτερα η λειτουργία του αθροιστή 8 bit, έστω 2 προστιθέμενοι αριθμοί $A=11001101$ και $B=10011101$. Η παραγωγή του αθροίσματος γίνεται βηματικά, ξεκινώντας από



(α)



(β)

Σχήμα 5.43: Δύο υλοποιήσεις πλήρους αθροιστή: (α) ως άθροισμα γινομένων, (β) με 2 ημιαθροιστές

τον δεξιότερο πλήρη αθροιστή, όπως φαίνεται στον παρακάτω πίνακα:

Δείκτης i	8	7	6	5	4	3	2	1	0
A_i		1	1	0	0	1	1	0	1
B_i		1	0	0	1	1	1	0	1
C_i	1	0	0	1	1	1	0	1	0
S_i		0	1	1	0	1	0	1	0

Το κρατούμενο εισόδου C_0 ισούται με 0. Στη θέση $i = 0$ προστίθενται τα bits $A_0 = 1$ και $B_0 = 1$. Το αποτέλεσμα της πράξης είναι 0, ενώ μια μονάδα θα μεταφερθεί ως κρατούμενο στη θέση $i = 1$. Ομοίως, στη θέση $i = 1$ προστίθενται τα bits $A_1 = 0$, $B_1 = 0$, και $C_1 = 1$. Το αποτέλεσμα της πρόσθεσης είναι 1 και ένα μηδενικό μεταφέρεται ως κρατούμενο στην επόμενη θέση $i = 2$. Προχωρώντας με παρόμοιο τρόπο, λαμβάνουμε, για κάθε θέση i , τις τιμές του παραπάνω πίνακα.

5.6.3 ΑΘΡΟΙΣΤΕΣ-ΑΦΑΙΡΕΤΕΣ

Μια ενδιαφέρουσα υλοποίηση, είναι αυτή ενός κυκλώματος, το οποίο λειτουργεί ως αθροιστής-αφαιρέτης 8 bit. Το κύκλωμα αποτελείται από 8 πλήρεις αθροιστές σε διάταξη παρόμοια με αυτή που περιγράφηκε στο κύκλωμα του αθροιστή 8 bit του Σχήματος 5.44. Η αφαίρεση μεταξύ των δυαδικών αριθμών γίνεται