

Κεφάλαιο 4

Σύνοψη

Αρχικά, στο κεφάλαιο αυτό, θα περιγραφούν οι βασικές λογικές πύλες AND, OR, NOT και θα δοθούν οι πίνακες αληθείας τους. Στη συνέχεια, ακολουθούν οι πύλες NAND και NOR καθώς και οι πύλες αποκλειστικής διάζευξης XOR και XNOR. Ο μετασχηματισμός των βασικών πυλών AND, OR, NOT με πύλες NAND, NOR είναι βασικός στο σχεδιασμό. Τέλος γίνεται μια περιγραφή των πυλών δύο επιπέδων τύπου AOI και OAI. Ακολουθεί η βασική ανάλυση και σχεδίαση κυκλωμάτων. Τα κυκλώματα των βασικών κυκλωμάτων αριθμητικών πράξεων της άλγεβρας (κύκλωμα ημιαθροιστή, κύκλωμα αθροιστή σειριακού και παράλληλου, κύκλωμα αφαιρέτη, κύκλωμα δυαδικού πολλαπλασιασμού). Το κεφάλαιο ολοκληρώνεται με το κύκλωμα του δυαδικού συγκριτή και το κύκλωμα μετατροπής κωδίκων με τη χρήση πυλών.

Προαπαιτούμενη γνώση

Βασικές γνώσεις της Άλγεβρας Boole και των θεωρημάτων της.

4. Λογικές Πύλες- Κυκλώματα βασικών πράξεων

4.1. Λογικές Πύλες

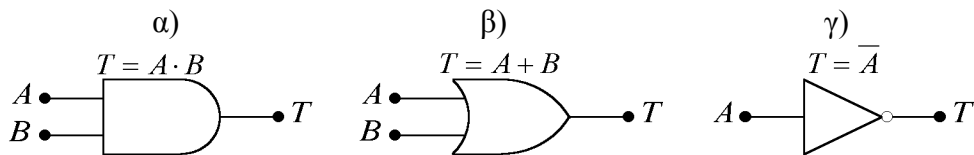
Τα θεμελιώδη λογικά κυκλώματα (λογικές πύλες), με τη βοήθεια των οποίων είναι δυνατόν να κατασκευαστούν άλλα συνθετότερα, είναι τρία:

- α. Η πύλη λογικού γινομένου **AND** (ΚΑΙ). Σύμβολο πράξης « \cdot » (γινόμενο) ή τομή δύο συνόλων.
- β. Η πύλη λογικής πρόσθεσης **OR** (Η). Σύμβολο πράξης « $+$ » (πρόσθεση) ή ένωση δύο συνόλων.
- γ. Η πύλη λογικής άρνησης **NOT** (ΟΧΙ). Σύμβολο πράξης « \neg ».

Στον πίνακα 4.1 που ακολουθεί παρατίθενται όλοι οι γνωστοί τρόποι γραφικής απεικόνισης των λογικών πυλών. Η συνήθης πρακτική είναι η χρήση τους κατά IEC 60617-12. (Nelson et al., 1995)

	IEC 60617-12	ANSI/IEEE 91-1984 ANSI/IEEE Std 91a-1991	DIN 40700
BUFFER			
NOT			
OR			
NOR			
AND			
NAND			
XOR			
XNOR			

Πίνακας 4.1. Γραφικά σύμβολα πυλών λογικών συναρτήσεων



Σχήμα 4.1. Συμβολισμός και συναρτήσεις των βασικών πυλών

Στο σχήμα 4.1 σημειώνονται τα σύμβολα των τριών παραπάνω πυλών, μαζί με τα ονόματα των λογικών μεταβλητών εισόδου και το αντίστοιχο λογικό αποτέλεσμα στην έξοδο T. (σχήματα.4.1 α, β, γ). (Nashelsky, 1994; Κοσσίδης, 1996; Holdsworth, & Woods, 2002; Givone, 2002; Balch, 2003; Katz, 2005; Κοσσίδης, & Γιαννακόπουλος, 2006; Wakerly, 2006; Balabanian, & Carlson, 2007; Maxfield, 2009; Floyd, 2013; Mano, & Ciletti, 2014; Roth, & Kinney, 2014; Pritchard, 2015)

Για το συμβολισμό των βασικών πυλών του σχήματος 4.1 χρησιμοποιήθηκαν πύλες δύο εισόδων, αριθμός που μπορεί να αυξηθεί, ώστε να ανταποκρίνεται στις ανάγκες των εκάστοτε εφαρμογών.

Στον πίνακα 4.2 δίνονται οι πίνακες αλήθειας των τριών βασικών λογικών πυλών για όλους τους δυνατούς συνδυασμούς των μεταβλητών εισόδου.

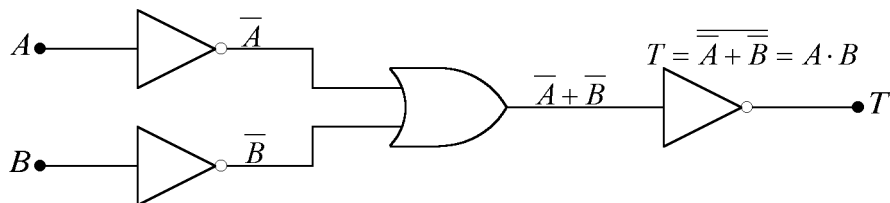
A	B	A+B
0	0	0
0	1	1
1	0	1
1	1	1

A	B	A · B
0	0	0
0	1	0
1	0	0
1	1	1

A	A-bar
0	1
1	0

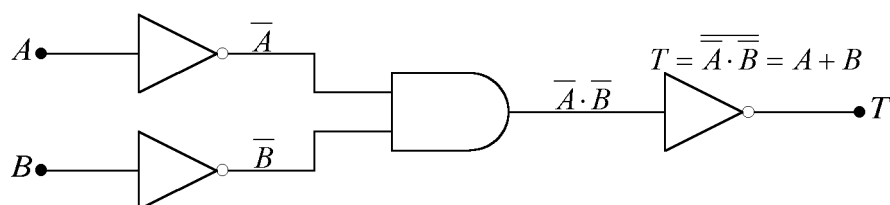
Πίνακας 4.2. Πίνακες αλήθειας των πυλών OR, AND και NOT

Στη πράξη όλα τα λογικά κυκλώματα μπορούν να κατασκευαστούν με τη βοήθεια δύο ζευγών πυλών: NOT - OR ή NOT - AND. Το παραπάνω εύκολα αποδεικνύεται, χρησιμοποιώντας το θεώρημα του De Morgan. Έτσι το κύκλωμα λογικού γινομένου $A \cdot B$ μπορεί να αντικατασταθεί από το κύκλωμα του σχήματος 4.2, που περιλαμβάνει μόνο πύλες OR και NOT.



Σχήμα 4.2. Πύλη AND με τη χρήση των πυλών NOT και OR

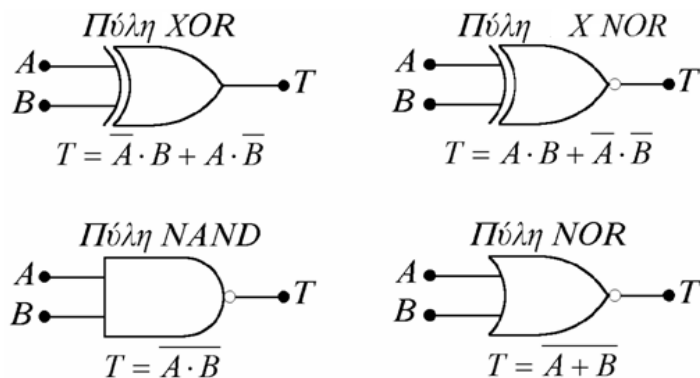
Αντίστοιχα το κύκλωμα λογικού αθροίσματος OR μπορεί να αντικατασταθεί από πύλες AND και NOT, όπως φαίνεται στο σχήμα 4.3.



Σχήμα 4.3. Πύλη OR με τη χρήση των πυλών NOT και AND

Εκτός από τις παραπάνω λογικές πύλες, υπάρχουν και άλλες εξίσου σημαντικές, με τις οποίες επιτυγχάνουμε απλούστευση των υπό μελέτη και κατασκευή λογικών κυκλωμάτων. Οι πύλες αυτές είναι οι:

- α. Η πύλη **NAND** (OXI-KAI). Σύμβολο πράξης « $\bar{\cdot}$ ».
- β. Η πύλη **NOR** (OXI-H). Σύμβολο πράξης « $\bar{\vee}$ ».
- γ. Η πύλη **XOR** ή Αποκλειστικού Ή (Exclusive-OR ή modulo-2). Σύμβολο πράξης « \oplus ».
- δ. Η πύλη **XNOR** ή Σύμπτωσης ή Αποκλειστικού NOR (Coincidence). Σύμβολο πράξης « \odot ».



Σχήμα 4.4. Συμβολισμός και συναρτήσεις των πυλών NAND, NOR, XOR, XNOR

Στο σχήμα 4.4 δίνονται τα αντίστοιχα σύμβολα των πυλών. Οι αντίστοιχοι πίνακες αλήθειας των λογικών αυτών πυλών δίνονται στον πίνακα 4.3.

A	B	T
0	0	1
0	1	0
1	0	0
1	1	1

A	B	T
0	0	0
0	1	1
1	0	1
1	1	0

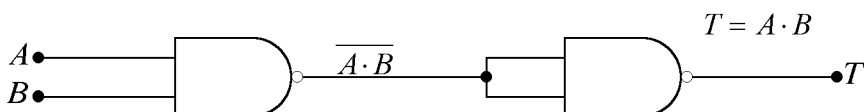
A	B	T
0	0	1
0	1	0
1	0	0
1	1	0

A	B	T
0	0	1
0	1	1
1	0	1
1	1	0

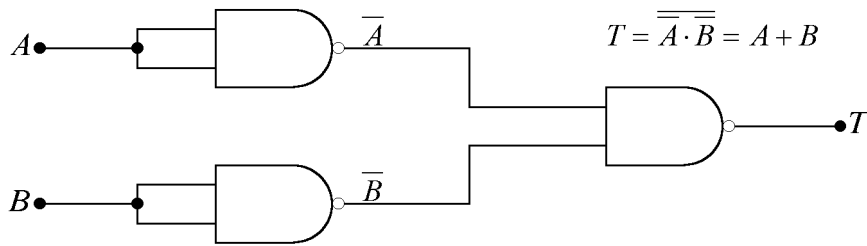
Πίνακας 4.3. Πίνακες αλήθειας των πυλών XOR, XNOR, NOR, NAND

4.2 Μετασχηματισμός των πυλών AND, OR, NOT με πύλες NAND, NOR

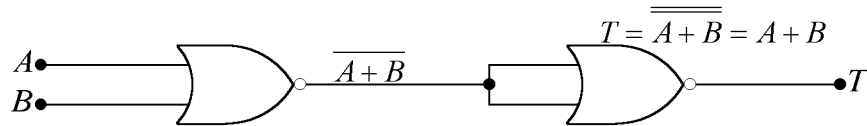
Ο μετασχηματισμός ενός κυκλώματος που έχει μελετηθεί με πύλες AND, OR, NOT σε ισοδύναμο κύκλωμα με πύλες NAND, NOR μπορεί να γίνει, χρησιμοποιώντας τις ισοδύναμες πύλες του σχήματος 4.5. Το κύκλωμα που θα προκύψει, θα πρέπει να απλοποιηθεί αμέσως μετά.



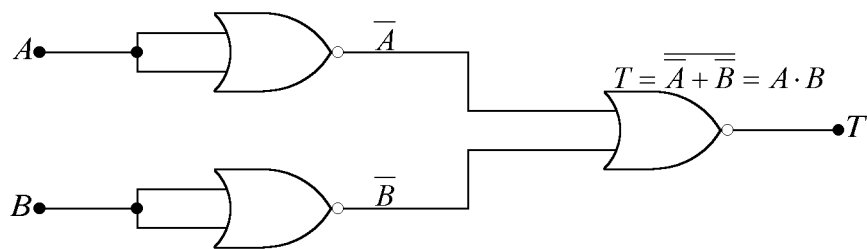
Σχήμα 4.5α. Υλοποίηση της πύλης AND με πύλες NAND.



Σχήμα 4.5β. Υλοποίηση της πύλης OR με πύλες NAND



Σχήμα 4.5γ. Υλοποίηση της πύλης OR με πύλες NOR



Σχήμα 4.5δ. Υλοποίηση της πύλης AND με πύλες NOR

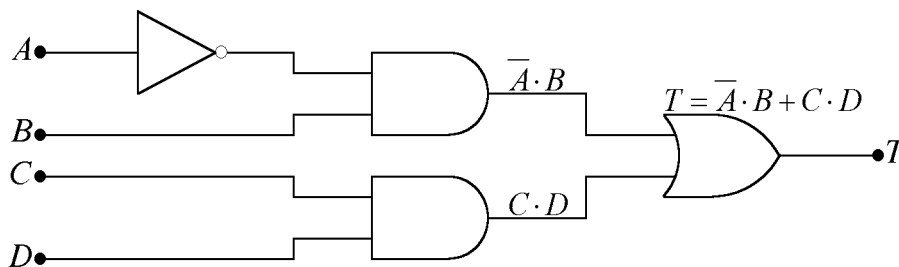
Στα σχήματα 4.5 α,β,γ,δ δίνουμε μερικά απλά κυκλώματα υλοποίησης των βασικών πυλών AND, OR, NOT με πύλες NAND ή πύλες NOR. (Στο σχήμα 4.5α το δεύτερο τμήμα του κυκλώματος είναι μια πύλη NOT). Ένας άλλος τρόπος απεικόνισης του τελικού σχήματος είναι να απλοποιήσουμε πρώτα τη συνάρτηση χρησιμοποιώντας θεωρήματα και τέλος να σχεδιάσουμε το κύκλωμα με τις πύλες NAND ή NOR:

Παράδειγμα 1ο:

Έστω ότι πρέπει να κατασκευάσουμε το λογικό κύκλωμα της συνάρτησης που ακολουθεί με πύλες NAND ή πύλες NOR.

$$T = \bar{A} \cdot B + C \cdot D \tag{4-1}$$

Το κύκλωμα με πύλες τις βασικές πύλες NOT, AND, OR έχει την παρακάτω μορφή:

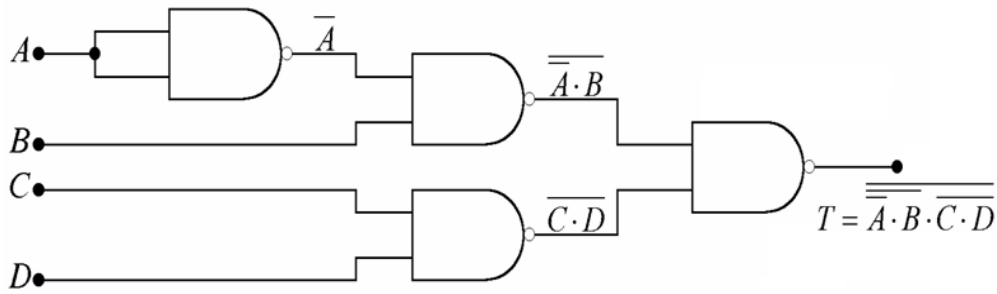


Σχήμα 4.6α. Υλοποίηση της συνάρτησης με κλασσικές πύλες

Η συνάρτηση T, εφαρμόζοντας το θεώρημα του De Morgan, γράφεται:

$$T = \overline{\overline{\bar{A} \cdot B + C \cdot D}} = \overline{\overline{\bar{A} \cdot B} \cdot \overline{C \cdot D}} = \overline{A \cdot B \cdot \bar{C} \cdot \bar{D}} \tag{4-2}$$

από όπου προκύπτει το παρακάτω κύκλωμα το οποίο είναι κατασκευασμένο εξ' ολοκλήρου με πύλες NAND.

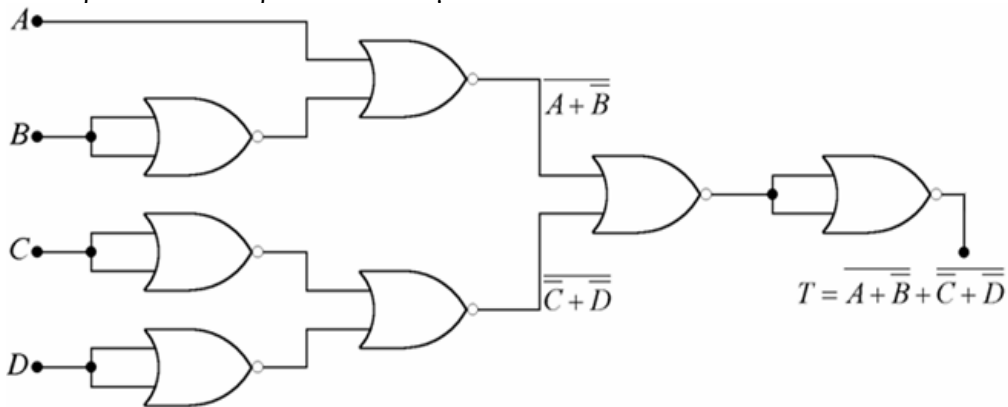


Σχήμα 4.6β. Υλοποίηση της συνάρτησης με πύλες NAND

Αντίστοιχα για το παραπάνω κύκλωμα με πύλες NOR έχουμε:

$$T = \overline{\overline{A} \cdot \overline{B} \cdot \overline{C} \cdot \overline{D}} = \overline{\overline{A} \cdot \overline{B}} + \overline{\overline{C} \cdot \overline{D}} = \overline{\overline{A+B}} + \overline{\overline{C+D}} \quad (4-3)$$

απ' όπου προκύπτει το παρακάτω κύκλωμα:



Σχήμα 4.6γ. Υλοποίηση της συνάρτησης με πύλες NOR

4.3. Πύλη αποκλειστικής διάζευξης (XOR)

Μια πύλη αποκλειστικής διάζευξης (Exclusive OR-XOR) δύο εισόδων έχει έξοδο αληθή (1), όταν μόνο μια από τις δύο εισόδους είναι αληθής. Η XOR είναι ιδιαίτερα χρήσιμη στο σχεδιασμό αθροιστών και στα κυκλώματα ανίχνευσης / διόρθωσης σφαλμάτων.

Μερικές ενδιαφέρουσες ιδιότητες της XOR είναι οι:

$$\begin{aligned} x \oplus 0 &= x & x \oplus 1 &= x' \\ x \oplus x &= 0 & x \oplus x' &= 1 \\ x \oplus (y \oplus z) &= (x \oplus y) \oplus z & & \text{[Προσεταιριστική]} \\ x \oplus y &= y \oplus x & & \text{[Επιμεριστική]} \end{aligned}$$

Η γενική συνάρτηση της XOR είναι αληθής, όταν έχουμε περιττό αριθμό ορισμάτων της αληθής (1) (δηλαδή περιττό αριθμό άσων).

Για παράδειγμα, μπορούμε να χρησιμοποιήσουμε την άλγεβρα Boole, για να απλοποιήσουμε μια πύλη XOR τριών εισόδων στην παρακάτω έκφραση, με δεδομένο τον πίνακα αληθείας της XOR. (Πίνακας 4.4)

$$\begin{aligned} x \oplus (y \oplus z) &= x \oplus (y'z + yz') & & \text{[Ορισμός της XOR]} \\ &= x'(y'z + yz') + x(y'z + yz')' & & \text{[Ορισμός της XOR]} \\ &= x'y'z + x'yz' + x(y'z + yz')' & & \text{[Επιμεριστική]} \\ &= x'y'z + x'yz' + x((y'z)' (yz')') & & \text{[Θεώρημα De Morgan]} \\ &= x'y'z + x'yz' + x((y + z')(y' + z)) & & \text{[Θεώρημα De Morgan]} \end{aligned}$$

$$\begin{aligned}
 &= x'y'z + x'yz' + x(yz + y'z') && \text{[Επιμεριστική ιδιότητα]} \\
 &= x'y'z + x'yz' + xyz + xy'z' && \text{[Επιμεριστική ιδιότητα]}
 \end{aligned}$$

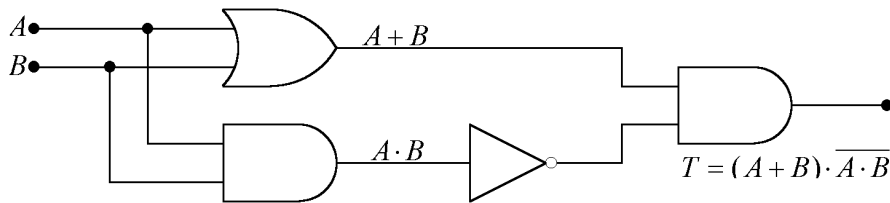
x	y	z	$x \oplus y \oplus z$
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	1

Πίνακας 4.4. Γενική συνάρτηση της πύλης XOR

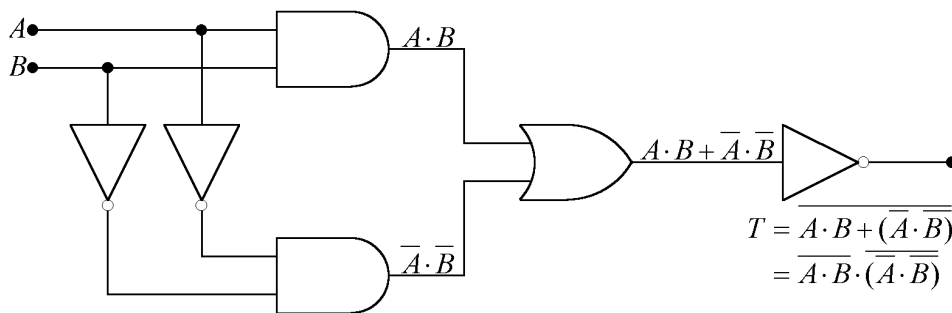
(Υπενθυμίζουμε ότι η αναπαράσταση της άρνησης γίνεται στο παρόν κεφάλαιο είτε με τονούμενη είτε με υπεργραμμισμένη μεταβλητή).

Με βάση τον πίνακα 4.4 της πύλης αποκλειστικής διάζευξης, XOR, η πύλη θα μπορούσε να κατασκευασθεί με διάφορους τρόπους με τη βοήθεια των γνωστών βασικών πυλών, που έχουμε ήδη γνωρίσει. Στο σχήμα 4.6α βλέπουμε ένα τέτοιο κύκλωμα όπου:

$$T = \bar{A} \cdot B + A \cdot \bar{B} = (A + B) \cdot (\bar{A} + \bar{B}) = (A + B) \cdot \overline{A \cdot B}$$



Σχήμα 4.7α. Πύλη XOR



Σχήμα 4.7β. Ισοδύναμη υλοποίηση πύλης XOR

Στο κύκλωμα του σχήματος 4.6β αποδίδεται το κύκλωμα της πύλης XOR, κατασκευασμένο, όμως, από διαφορετικές πύλες, όπως προκύπτει από μια ακόμα επεξεργασία της λογικής συνάρτησης που περιγράφει τις πύλες αυτού του είδους. Η σχεδιάσή του είναι απλή και εύκολο να κατανοηθεί, αν παρατηρήσουμε τις λογικές ποσότητες που εμφανίζονται στις εξόδους των πυλών και λάβουμε υπόψη μας τα θεωρήματα της άλγεβρας Boole.

Στη συνέχεια αποδεικνύουμε, για εκπαιδευτικούς λόγους, ότι η συνάρτηση $T = \overline{\bar{A} \cdot B \cdot \bar{A} \cdot B}$ περιγράφει πράξη XOR.

$$T = \overline{\bar{A} \cdot B \cdot \bar{A} \cdot B} \Rightarrow T = \overline{\bar{A} \cdot B} \cdot \overline{\bar{A} \cdot B} = (\bar{A} + B) \cdot (A + B)$$

$$T = \bar{A} \cdot A + \bar{A} \cdot B + A \cdot \bar{B} + B \cdot B \text{ και επειδή } \bar{A} \cdot A = B \cdot B = 0 \text{ έχουμε τελικά } T = A \cdot \bar{B} + \bar{A} \cdot B$$

Αποδείξαμε δηλαδή ότι: $T = A \oplus B = \overline{A \cdot B} + \overline{\overline{A} \cdot \overline{B}} = \overline{A \cdot B} + \overline{\overline{A} \cdot \overline{B}}$

Με ανάλογο τρόπο μπορεί να αποδειχθεί ότι: $\overline{A \cdot B} + \overline{\overline{A} \cdot \overline{B}} = A \oplus B$

4.3.1. Πύλη XNOR

Η πύλη XNOR ή Σύμπτωσης ή Αποκλειστικού NOR (Coincidence), είναι η δυική πράξη της XOR. Η πύλη αναφέρεται και ως συγκριτής, διότι η έξοδός της είναι «1» όταν και δύο είσοδοι έχουν την ίδια τιμή. Ανάλογα, όταν οι είσοδοι έχουν διαφορετικές τιμές η έξοδος της πύλης είναι «0». Η πράξη απεικονίζεται με το σύμβολο «⊕».

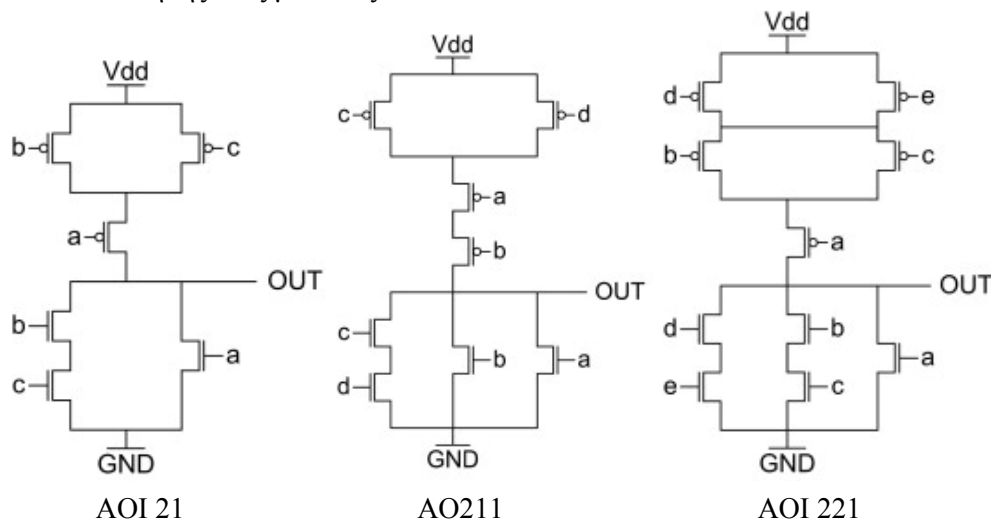
4.4. Σύνθετες πύλες δύο επιπέδων (AOI και OAI)

Με δεδομένη η ιεραρχία των λογικών πράξεων που είναι: NOT, AND, OR κρίνεται σκόπιμο να αναφέρουμε και την ύπαρξη σύνθετων πυλών δύο επιπέδων όπως, οι πύλες AOI(AND-OR-Invert) ή οι πύλες OAI(OR-AND-Invert) οι οποίες υλοποιούν όλες τις εκφράσεις SOP ή POS, που έχουν ήδη αναφερθεί. (Katz, 2005; Wakerly, 2006)

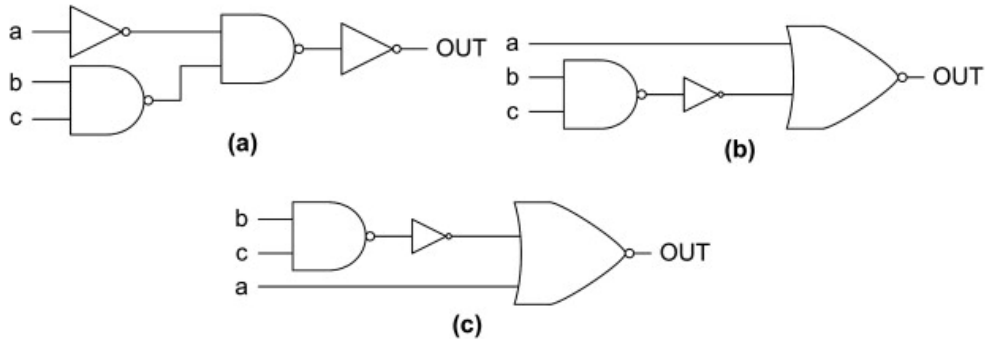
Οι δύο αυτοί τύποι πυλών είναι συμπληρωματικοί μεταξύ τους και κατασκευάζονται οι πύλες AOI από πύλες AND, που ακολουθούνται από πύλες NOR, και οι πύλες OAI από πύλες OR, που ακολουθούνται από πύλες NAND. Τα νούμερα που ακολουθούν τη σύνθετη πύλη αντιστοιχούν στον αριθμό των πυλών του πρώτου επιπέδου και στον αριθμό των εισόδων, που η καθεμία έχει. Συνήθως απεικονίζονται με τη μορφή π.χ. AND-OR-Invert-2-1 ή AOI-21 που αποδίδεται από την έκφραση: $X = \overline{(A \cdot B)} + C$. Αντίστοιχα η πύλη AOI-33 (έχει στο πρώτο επίπεδο δύο πύλες AND που η καθεμία έχει τρεις εισόδους) αντιστοιχεί στην έκφραση $X = \overline{(A \cdot B \cdot C)} + \overline{(D \cdot E \cdot F)}$. Για τις πύλες OAI αντίστοιχα παραδείγματα είναι τα π.χ. OAI και μια πύλη OAI222 έχει τρεις πύλες OR με την κάθε μία να έχει δύο εισόδους. (Η αρίθμηση γίνεται πάντα με φθίνουσα τάξη, καθώς είναι ισοδύναμη η απεικόνιση OAI222 ή OAI221).

Οι σύνθετες πύλες συχνά χρησιμοποιούνται κατά το σχεδιασμό των VLSI chip με CMOS, επειδή μπορεί να υλοποιηθούν αποτελεσματικά, βασισμένες σε έξυπνους συνδυασμούς σε σειρά και παράλληλα συνδεδεμένων τρανζίστορ. Για παράδειγμα, η πύλη AOI33 μπορεί να κατασκευαστεί, χρησιμοποιώντας μόνο 12 τρανζίστορ, ενώ η αντίστοιχη διακριτή υλοποίηση θα απαιτήσει 4 τρανζίστορ για την πύλη NOR, και 8 τρανζίστορ για κάθε πύλη AND τριών εισόδων, απαιτώντας έτσι ένα σύνολο από 20 τρανζίστορ – υλοποίηση σχεδόν δύο φορές πιο ακριβή και δύο φορές πιο αργή. (Hendrich 2006)

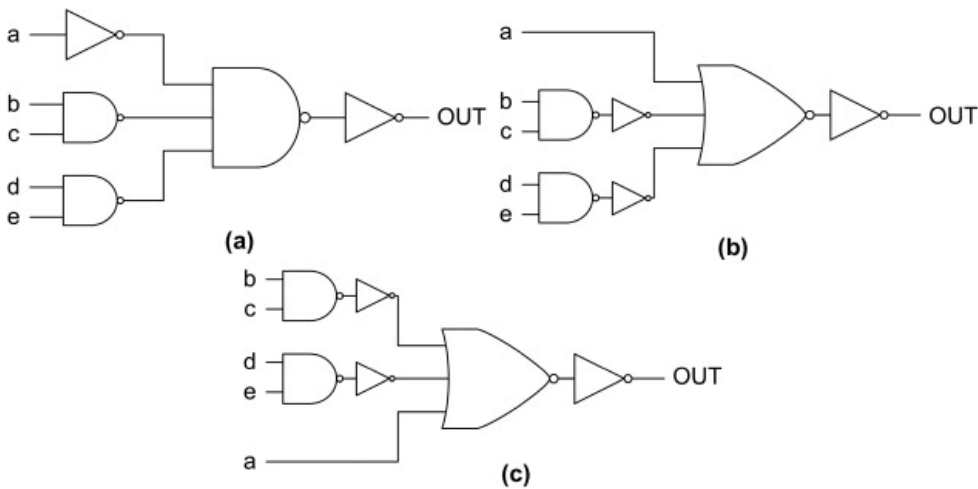
Στο διάγραμμα που ακολουθεί αποδίδονται ενδεικτικά τα δομικά διαγράμματα των AOI 21, AOI 211, AOI 221 με CMOS. (Butzen, Dal Bem, Reis & Ribas 2010). Ακολουθούν οι γραφικές απεικονίσεις των κυρίων διαγραμμάτων υλοποίησής τους με πύλες.



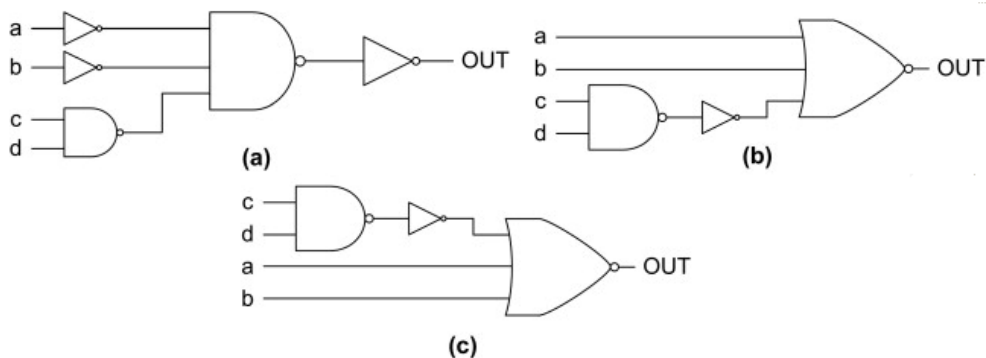
Σχήμα 4.8α. Απεικονίσεις με CMOS των πυλών AOI21, AOI211, AOI221



Σχήμα 4.8β. Διαγράμματα πυλών της AOI-21



Σχήμα 4.8γ. Διαγράμματα πυλών της AOI-221



Σχήμα 4.8δ. Διαγράμματα πυλών της AOI-211

4.5. Βασική Ανάλυση και Σχεδίαση Κυκλωμάτων

Μάθαμε ήδη ότι είναι απαραίτητο για να προχωρήσουμε στη σχεδίαση κυκλώματος:

- Οι πίνακες αληθείας και οι εκφράσεις της άλγεβρας Boole περιγράφουν συναρτήσεις.
- Οι εκφράσεις μπορούν να μετατραπούν σε κυκλώματα.
- Η άλγεβρα Boole και οι χάρτες Karnaugh μας βοηθούν να απλοποιήσουμε τις εκφράσεις και τα κυκλώματα.

Τώρα μπορούμε να χρησιμοποιήσουμε αυτές τις γνώσεις, για να αναλύσουμε και να σχεδιάσουμε μεγαλύτερα κυκλώματα.

Σχεδίαση κυκλώματος

Ο στόχος της σχεδίασης κυκλωμάτων είναι η κατασκευή υλικού το οποίο να υλοποιεί κάποια συγκεκριμένη συνάρτηση. Η βασική ιδέα είναι να γράψουμε τη συνάρτηση σαν μία Boolean έκφραση και κατόπιν να τη μετατρέψουμε σε ένα κύκλωμα. Τα βήματα, που ακολουθούμε κατά την υλοποίηση, είναι:

- Βήμα 1:** Προσδιορισμός του αριθμού των εισόδων και των εξόδων.
- Βήμα 2:** Πρέπει να είμαστε βέβαιοι ότι έχουμε τη συνάρτηση περιγραφής, είτε με τη μορφή του πίνακα αλήθειας είτε σαν μία Boolean έκφραση.
- Βήμα 3:** Μετατροπή αυτής σε μία απλοποιημένη Boolean έκφραση. (Βασικός στόχος είναι η εύρεση του ΕΑΓ, εκτός αν ζητείται κάτι διαφορετικό).
- Βήμα 4:** Κατασκευή του κυκλώματος, βασισμένη στην απλοποιημένη έκφραση.

Ανάλυση Κυκλωμάτων

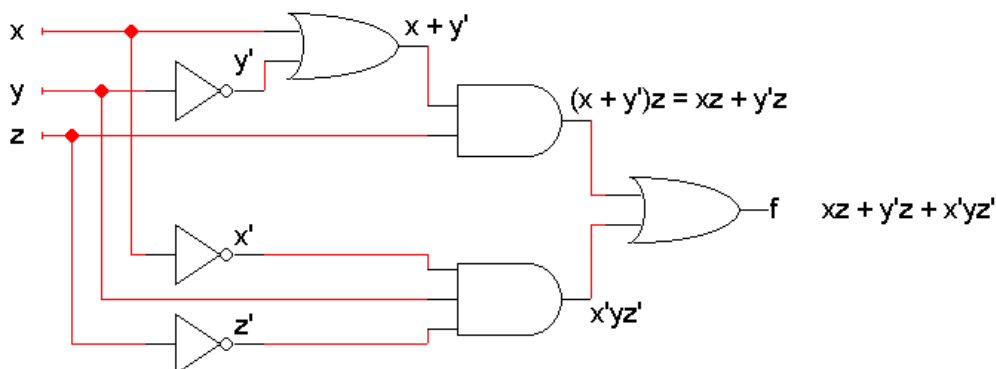
Η ανάλυση κυκλωμάτων περιλαμβάνει την κατανόηση τις λειτουργίας κάποιων κυκλωμάτων. Κάθε κύκλωμα υπολογίζει μία συνάρτηση, η οποία μπορεί να περιγραφεί με «Boolean» εκφράσεις ή με πίνακες αληθείας.

Έτσι ο στόχος μας είναι να βρούμε μια έκφραση ή ένα πίνακα αληθείας για το κύκλωμα.

Αρχικά πρέπει να προσδιορίσουμε όλες τις εισόδους και τις εξόδους του κυκλώματος. Το κύκλωμα της συνάρτησης $f(x, y, z) = xz + \bar{y}z + \bar{x}y\bar{z}$ που ακολουθεί (σχήμα 4.9α) έχει, όπως εύκολα διαπιστώνουμε, τρεις εισόδους τις x, y, z και μία έξοδο την f .

Στη συνέχεια, γράφουμε εκφράσεις για τις εξόδους κάθε πύλης, οι οποίες βασίζονται στις εισόδους των πυλών. Ξεκινάμε από τις εισόδους και εργαζόμαστε κατευθυνόμενοι προς τις εξόδους. Αν είναι δυνατόν κάνουμε και τυχόν απλοποιήσεις που θα μας βοηθήσουν στην περαιτέρω πορεία μας.

Στο διάγραμμα του σχήματος 4.9α παραθέτουμε, επίσης, τις συναρτήσεις (αλγεβρικές εκφράσεις) της καθεμίας πύλης ξεχωριστά καθώς και της τελικής εξόδου.



Σχήμα 4.9α. Υλοποίηση της συνάρτησης $f(x, y, z) = xz + \bar{y}z + \bar{x}y\bar{z}$

Πίνακας αλήθειας

x	y	z	xz	$\bar{y}z$	$\bar{x}y\bar{z}$	f
0	0	0	0	0	0	0
0	0	1	0	1	0	1
0	1	0	0	0	1	1
0	1	1	0	0	0	0
1	0	0	0	0	0	0
1	0	1	1	1	0	1
1	1	0	0	0	0	0
1	1	1	1	0	0	1

Πίνακας 4.5. Πίνακας αλήθειας της συνάρτησης $f(x, y, z) = xz + \bar{y}z + \bar{x}y\bar{z}$

Τον πίνακα αληθείας τον φτιάχνουμε απευθείας από το κύκλωμα. Όταν γνωρίζουμε τον αριθμό των εισόδων και το εξόδων, καταγράφουμε όλους τους πιθανούς συνδυασμούς εισόδων στον πίνακα αληθείας.

Ένα κύκλωμα με n εισόδους πρέπει να έχει ένα πίνακα αληθείας με 2^n σειρές. Το παράδειγμά μας έχει τρεις εισόδους. Άρα ο πίνακας αληθείας θα έχει $2^3 = 8$ σειρές.

Δίνοντας τις τιμές στις μεταβλητές x, y, z και δημιουργώντας τους όρους της συνάρτησης, παίρνουμε τον πίνακα 4.5.

Εκφράσεις και πίνακες αλήθειας

Είδαμε ήδη ότι το κύκλωμα του σχήματος 4.9α υπολογίζει τη συνάρτηση $f(x, y, z) = xz + \bar{y}z + \bar{x}y\bar{z}$. Μπορούμε να χρησιμοποιήσουμε τη συνάρτηση αυτή, για να συμπληρώσουμε τον πίνακα αλήθειας.

Δημιουργούμε ξεχωριστές στήλες για καθένα από τους όρους xz , $\bar{y}z$ και $\bar{x}y\bar{z}$. Τελικά παίρνουμε τη συνάρτηση f , κάνοντας την πράξη OR μεταξύ των ενδιαμέσων στηλών. (Πίνακας 4.5).

Μπορεί να γίνει και η αντίθετη μετατροπή. Είναι εύκολο να φτιάξουμε μια αλγεβρική έκφραση, αν ήδη έχουμε τον πίνακα αληθείας.

Είδαμε πόσο γρήγορα μετατρέψαμε έναν πίνακα αληθείας σε μία έκφραση αθροίσματος ΕΛΒΟ. Οι ΕΛΒΟ αντιστοιχούν στις γραμμές του πίνακα που έχουν την τιμή 1.

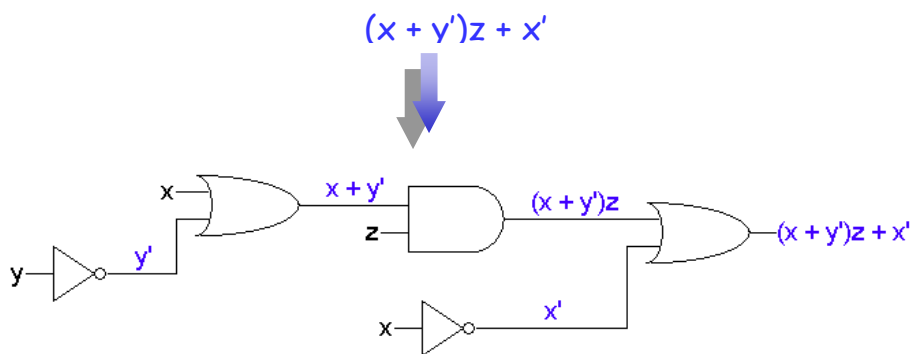
x	y	z	f
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	1

Η συνάρτηση είναι η: $f(x,y,z) = x'y'z + x'yz' + xy'z + xyz = m_1 + m_2 + m_5 + m_7$

Αν θέλουμε, μπορούμε να απλοποιήσουμε αυτό το άθροισμα των ΕΛΒΟ, χρησιμοποιώντας π.χ. ένα χάρτη Karnaugh. (Ποιο θα είναι το αποτέλεσμα;)

Παράδειγμα υλοποίησης απλών κυκλωμάτων με τη χρήση πυλών

Οποιαδήποτε Boolean έκφραση μπορεί να μετατραπεί σε κύκλωμα, συνδυάζοντας τις βασικές πύλες με έναν σχετικά εύκολο τρόπο.



Σχήμα 4.9β. Υλοποίηση της συνάρτησης $f = (x + \bar{y})z + \bar{x}$

Έστω η συνάρτηση: $f = (x + \bar{y})z + \bar{x}$. Η προτεραιότητα είναι ρητά καθορισμένη σε ένα κύκλωμα. Πρέπει να είμαστε σίγουροι ότι το κύκλωμα εκτελεί τις λειτουργίες με τον σωστό τρόπο, δηλαδή με τη σωστή σειρά.

Στο σχήμα 4.9 δίνεται η υλοποίησή της με λογικές πύλες και απεικονίζονται οι εισοδοί και οι έξοδοι κάθε πύλης.

4.6. Δυαδικός ημιαθροιστής

Πρόσθεση δύο δυαδικών ψηφίων

Ο δυαδικός ημιαθροιστής (Half adder) είναι ένα συνδυαστικό λογικό κύκλωμα ικανό να εκτελεί την πράξη της πρόσθεσης δυαδικών πληροφοριών μήκους ενός δυαδικού ψηφίου (bit) η κάθε μία. Το αποτέλεσμα λαμβάνεται από δύο εξόδους, εκ των οποίων η πρώτη παρέχει το άθροισμα και η δεύτερη το τυχόν κρατούμενο. *Nashelsky, 1994; Κοσσίδης, 1996; Holdsworth, & Woods, 2002; Givone, 2002; Predko, 2005; Κοσσίδης, & Γιαννακόπουλος, 2006; Wakerly, 2006; Godse, & Godse, 2011; Roth, & Kinney, 2014)*

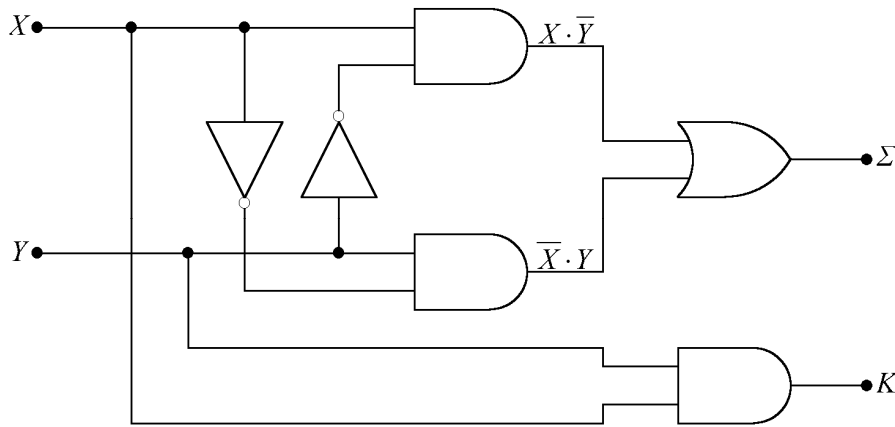
Από τον πίνακα πρόσθεσης δυαδικών ψηφίων υπολογίζουμε τις λογικές εξισώσεις που προσδιορίζουν τη λειτουργία του ημιαθροιστή:

$$\Sigma = \overline{X}Y + X\overline{Y} = X \oplus Y \quad (4-4) \quad \text{και} \quad K = X \cdot Y \quad (4-5)$$

Από αυτές προκύπτει το κύκλωμα του σχήματος 4.10α.

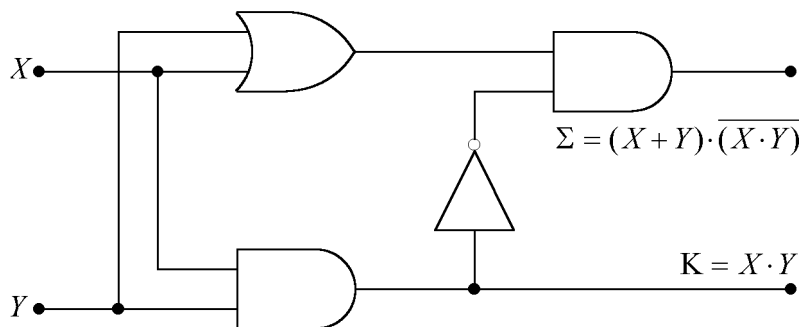
Το κύκλωμα του σχήματος 4.10α μπορεί να απλοποιηθεί, λαμβάνοντας υπόψη ότι το Σ έχει τιμή «1» μόνο όταν η μεταβλητή X ή η μεταβλητή Y είναι «1», και όχι και οι δύο ταυτόχρονα. Κατά συνέπεια, η παραπάνω σχέση που προσδιορίζει το άθροισμα Σ μπορεί να εκφραστεί και ως εξής:

$$\Sigma = (X + Y) \cdot \overline{X \cdot Y} \quad (4-6) \quad \text{και} \quad K = X \cdot Y \quad (4-7)$$

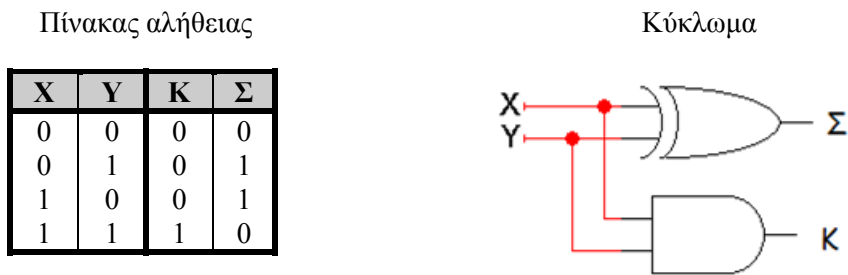


Σχήμα 4.10α. Αναλυτικό διάγραμμα του δυαδικού ημιαθροιστή

Κάνοντας πράξεις, προκύπτει ότι οι δύο σχέσεις 4-4 και 4-6 που προσδιορίζουν το Σ είναι ισοδύναμες. Το κύκλωμα που αντιστοιχεί σε αυτή την περίπτωση είναι το Σχήμα 4.11α, το οποίο με τη χρήση της πύλης της αποκλειστικής διάζευξης αντιστοιχεί στο κύκλωμα του σχήματος 4.11β που ακολουθεί.



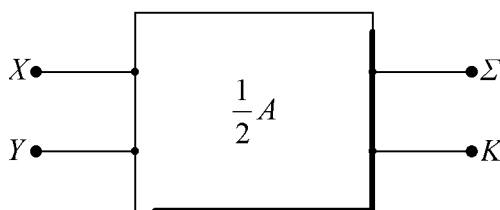
Σχήμα 4.11α. Δυαδικός ημιαθροιστής



Σχήμα 4.11β. Πίνακας αλήθειας και διάγραμμα του ημιαθροιστή

Στο ανωτέρω, Σχήμα 4.11β, δίνεται ο πίνακας αληθείας και το κύκλωμα του ημιαθροιστή, με τη χρήση της πύλης της αποκλειστικής διάζευξης (XOR).

Σχηματικά το δομικό διάγραμμα του ημιαθροιστή δίνεται στο σχήμα 4.12.



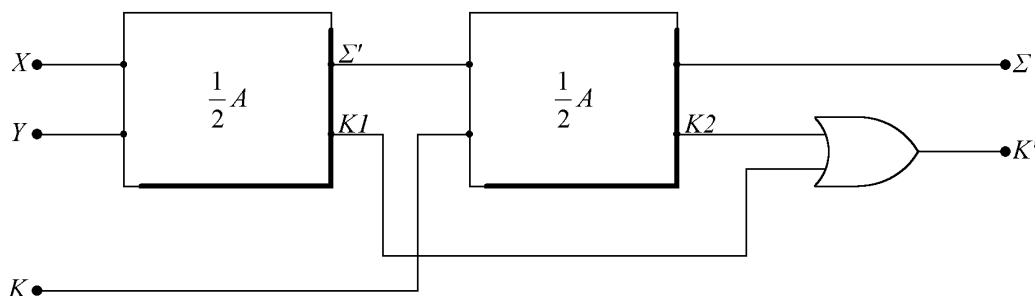
Σχήμα 4.12. Δομικό διάγραμμα ημιαθροιστή

4.7. Δυαδικός αθροιστής τριών εισόδων ή πλήρης αθροιστής (Full adder)

Ο ημιαθροιστής είναι σε θέση να προσθέσει δύο πληροφορίες του ενός bit η καθεμία, χωρίς να λαμβάνει υπόψη τυχόν κρατούμενο που μπορεί να έχει παραχθεί από ένα άλλο κύκλωμα πρόσθεσης. Με το δυαδικό αθροιστή τριών εισόδων, ή πλήρη αθροιστή, λαμβάνεται υπόψη και το πιθανό κρατούμενο. (Nashelsky, 1994; Holdsworth, & Woods, 2002; Givone, 2002; Predko, 2005; Κοσσίδης, & Γιαννακόπουλος, 2006; Godse, & Godse, 2011)

Η πρόσθεση των X, Y και K δίνει σαν αποτέλεσμα το άθροισμα Σ και το κρατούμενο Κ'. Η πρόσθεση αυτή εύκολα μπορεί να χωριστεί σε δύο επιμέρους προσθέσεις: στην πρώτη X+Y που δίνει άθροισμα Σ' και κρατούμενο Κ₁ και στη δεύτερη Σ'+K που δίνει άθροισμα Σ και κρατούμενο Κ₂. Πρέπει να ληφθεί υπόψη ότι, όταν το άθροισμα X+Y δίνει κρατούμενο Κ₁=1, το Σ' θα ισούται με 0, και, επομένως, το άθροισμα Σ'+K δεν μπορεί να δώσει κρατούμενο. Κατά συνέπεια, εάν τοποθετήσουμε δύο ημιαθροιστές (Σχήμα 4.12) σε σειρά, επιτυγχάνουμε το επιθυμητό κύκλωμα που φαίνεται στα Σχήματα 4.13α και 4.13β. Οι εξισώσεις που προκύπτουν είναι οι:

$$\begin{aligned} \Sigma' &= (X + Y) \cdot (\overline{X \cdot Y}) & K_1 &= X \cdot Y \\ \Sigma &= (\Sigma' + K) \cdot (\overline{\Sigma' \cdot K}) & K_2 &= \Sigma' \cdot K \end{aligned}$$

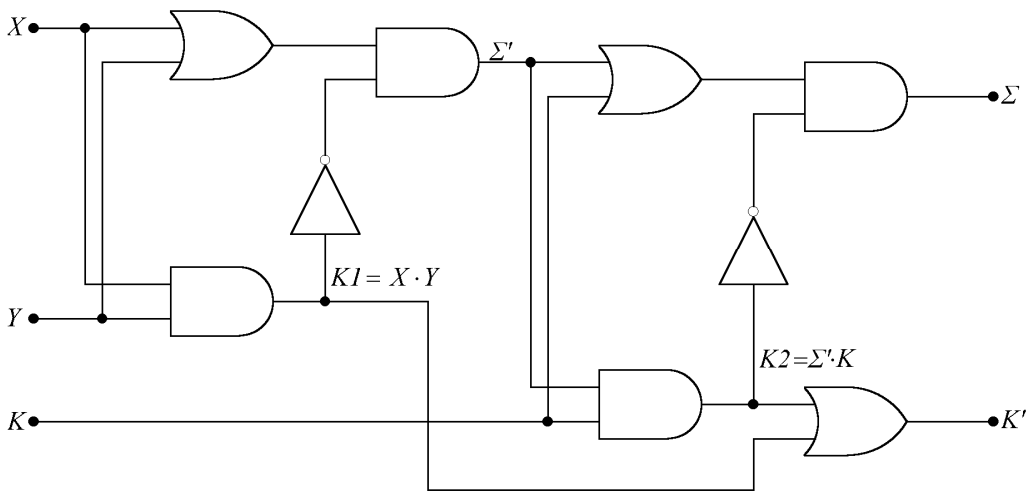


Σχήμα 4.13α. Πλήρης αθροιστής με χρήση του δομικού διαγράμματος του ημιαθροιστή

Αντικαθιστώντας, θα έχουμε:

$$\Sigma = [(X + Y) \cdot (\overline{X \cdot Y}) + K] \cdot \overline{[(X + Y) \cdot (\overline{X \cdot Y}) \cdot K]} \quad (4-8)$$

$$K' = K_1 + K_2 = X \cdot Y + (X + Y) \cdot (\overline{X \cdot Y}) \cdot K \quad (4-9)$$



Σχήμα 4.13β. Αναλυτικό διάγραμμα πλήρους αθροιστή

Πρόσθεση τριών δυαδικών ψηφίων

Με ανάλογο τρόπο μπορεί να μελετηθεί και ο αθροιστής τριών εισόδων. Τι πρέπει να κάνουμε, για να προσθέσουμε 3 bits; Απλά να αθροίσουμε τους δύο προσθετέους και το τυχόν κρατούμενο, που μεταφέρεται από τα δεξιά.

Ο πίνακας 4.6 συσχετίζει τις τρεις μεταβλητές εισόδου (**X**, **Y**, **K**) και δίνει σαν αποτέλεσμα δύο στήλες: μια για το άθροισμα (**Σ**) και μια για το κρατούμενο (**K'**).

Χρησιμοποιώντας την άλγεβρα Boole, παίρνουμε τις εξισώσεις που ακολουθούν. Η χρήση της πύλης XOR απλοποιεί περαιτέρω τη μορφή των εξισώσεων. (Χρησιμοποιούμε την άλγεβρα διότι δε μπορούμε εύκολα, να εξάγουμε τις εξισώσεις της XOR από τους χάρτες Karnaugh).

X	Y	K	Σ	K'
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

Πίνακας 4.6: Πίνακας αλήθειας πλήρους αθροιστή

$$\begin{aligned} \Sigma = \Sigma m(1,2,4,7) &= \overline{X} \overline{Y} K + \overline{X} Y \overline{K} + X \overline{Y} \overline{K} + XYK = \overline{X} (\overline{Y} K + Y \overline{K}) + X (\overline{Y} \overline{K} + YK) \\ &= \overline{X} (K \oplus Y) + X (\overline{Y \oplus K}) = X \oplus Y \oplus K \end{aligned} \quad (4-10)$$

και

$$\begin{aligned} K' = \Sigma m(3,5,6,7) &= \overline{X} Y K + X \overline{Y} K + X Y \overline{K} + XYK = (\overline{X} Y + X \overline{Y}) K + XY(\overline{K} + K) \\ &= (Y \oplus X) K + XY \end{aligned} \quad (4-11)$$

Συνεπώς οι εξισώσεις του πλήρους αθροιστή δίνονται από τις σχέσεις:

$$\Sigma = X \oplus Y \oplus K \text{ και } K' = (X \oplus Y) K + XY$$

Εναλλακτικά από τον πίνακα αλήθειας του αθροιστή προκύπτουν οι παρακάτω εξισώσεις:

$$\Sigma = \bar{X} \cdot \bar{Y} \cdot K + \bar{X} \cdot Y \cdot \bar{K} + X \cdot \bar{Y} \cdot \bar{K} + X \cdot Y \cdot K$$

$$K' = \bar{X} \cdot Y \cdot K + X \cdot \bar{Y} \cdot K + X \cdot Y \cdot \bar{K} + X \cdot Y \cdot K$$

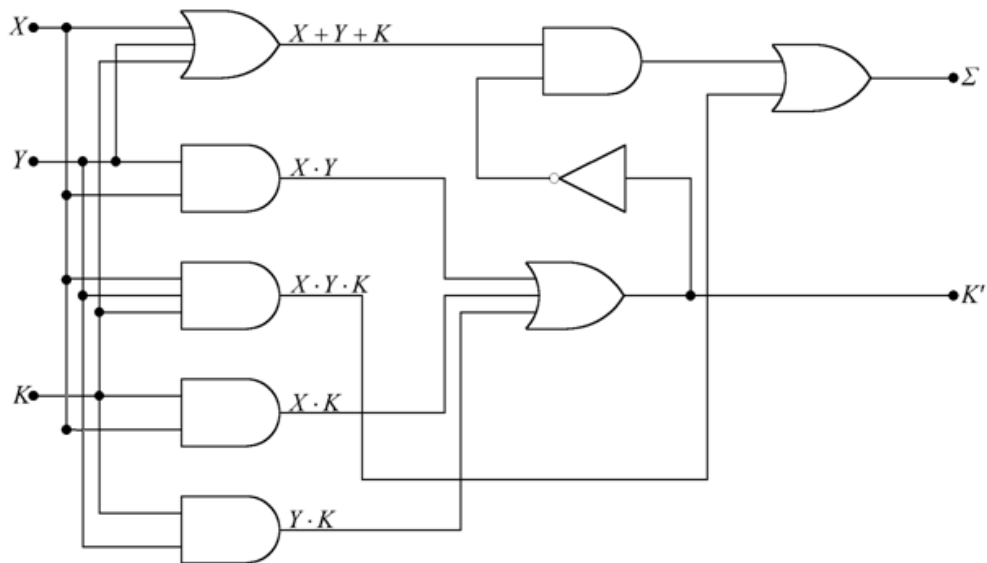
Η απλοποίηση των παραπάνω σχέσεων μπορεί να γίνει με τους χάρτες Karnaugh. Εύκολα επιτυγχάνουμε μια άλλη έκφραση του Σ και K' , αν ληφθεί υπόψη ότι το Σ έχει τιμή «1», όταν, όπως ήδη αναφέραμε περιττός αριθμός εισόδων έχει τιμή «1». Δηλαδή έχουμε:

$$\Sigma = (X + Y + K) \cdot (X \cdot Y + X \cdot K + Y \cdot K) + X \cdot Y \cdot K \quad (4-12)$$

Αντίστοιχα το κρατούμενο έχει τιμή «1», όταν δύο από τις τρεις εισόδους είναι «1» ή και οι τρεις έχουν τιμή «1».

$$K' = X \cdot Y + X \cdot K + Y \cdot K \quad (4-13)$$

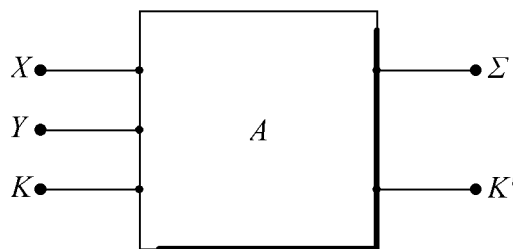
Σύμφωνα με τις παραπάνω σχέσεις προκύπτει το κύκλωμα του Σχήματος 4.14.



Σχήμα 4.14. Αναλυτικό διάγραμμα πλήρους αθροιστή με βασικές πύλες

(Να συγκρίνετε τις σχέσεις 4-10 και 4-12 καθώς και τις 4-11 και 4-13 και να σχεδιάσετε το διάγραμμα του πλήρους αθροιστή χρησιμοποιώντας την πύλη της αποκλειστικής διάζευξης).

Το δομικό διάγραμμα του πλήρους αθροιστή δίνεται στο Σχήμα 4.15.

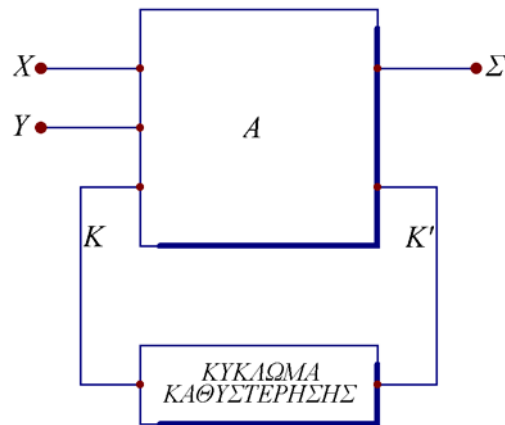


Σχήμα 4.15. Δομικό διάγραμμα πλήρους αθροιστή

4.8. Σειριακός πλήρης αθροιστής (Serial full adder)

Εάν έχουμε να αθροίσουμε δύο αριθμούς, που αποτελούνται από περισσότερα του ενός bit ο κάθε ένας, ο αθροιστής που αναφέραμε προηγουμένως (Σχήμα 4.15) είναι ικανός να εκτελέσει την πρόσθεση υπό την προϋπόθεση ότι τα δυαδικά ψηφία ακολουθούν το ένα το άλλο, ξεκινώντας από το ψηφίο μικρότερης τάξης και καταλήγοντας στο ψηφίο της μεγαλύτερης τάξης. Επίσης το κρατούμενο K_i που προκύπτει από την πρόσθεση των X_i και Y_i δυαδικών ψηφίων, επανεμφανίζεται στην είσοδο σαν K_{i+1} και συνυπολογίζεται κατά την πρόσθεση των ψηφίων X_{i+1} και Y_{i+1} των δυαδικών αριθμών.

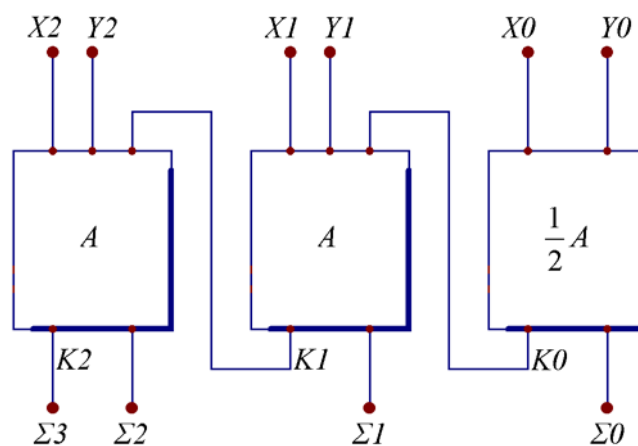
Αυτό επιτυγχάνεται συνδέοντας την έξοδο K' του αθροιστή στην είσοδο K του ίδιου αθροιστή δια μέσου ενός κυκλώματος καθυστέρησης (Delay f-f, $(D f-f)$), το οποίο συγκρατεί την πληροφορία για το απαιτούμενο χρονικό διάστημα που μεσολαβεί κατά την εφαρμογή δύο ψηφίων στην είσοδο του αθροιστή. Στο Σχήμα 4.16 δίνεται σχηματικά ο σειριακός πλήρης αθροιστής. Αποτελείται από δύο μόνον εισόδους X , Y , στις οποίες εισάγονται με σταθερή συχνότητα εξόδου τα δυαδικά ψηφία (bit) που προορίζονται να προστεθούν και από μία έξοδο Σ από την οποία εξέρχονται ένα - ένα τα ψηφία του αποτελέσματος. Τα ψηφία αυτά μπορούν στη συνέχεια να αποθηκευτούν σε έναν καταχωρητή / ολισθητή.



Σχήμα 4.16. Σειριακός δυαδικός αθροιστής

4.9. Παράλληλος πλήρης αθροιστής (Parallel full adder)

Ένας παράλληλος πλήρης αθροιστής επιτυγχάνεται, συνδέοντας κατάλληλα δυαδικούς πλήρεις αθροιστές. Κατά τη σύνδεση, πρέπει να ληφθεί υπόψη ότι το κρατούμενο της προηγούμενης βαθμίδας προστίθεται στην επόμενη βαθμίδα κ.ο.κ. Επειδή το άθροισμα δύο αριθμών n bit είναι δυνατόν να δώσει αποτέλεσμα με $n+1$ bit, θα πρέπει το κρατούμενο της νιοστής βαθμίδας, αντίθετα από τα υπόλοιπα κρατούμενα, να ληφθεί υπόψη κατά την παρουσίαση του αποτελέσματος. Στο σχήμα 4.17 δίνεται ένας αθροιστής δύο αριθμών που ο καθένας αποτελείται από τρία δυαδικά ψηφία. (Κοσσιδάς, 1996; Holdsworth, & Woods, 2002; Givone, 2002; Predko, 2005; Κοσσιδάς, & Γιαννακόπουλος, 2006; Godse, & Godse, 2011)



Σχήμα 4.17. Παράλληλος πλήρης αθροιστής

Η συνδεσμολογία αυτή παρουσιάζει ορισμένα προβλήματα. Ο αθροιστής, που προσθέτει τα δύο bit υψηλότερης τάξης, προκειμένου να εκτελέσει την πράξη της πρόσθεσης χρειάζεται να έχει και το κρατούμενο που προέκυψε από την προηγούμενη βαθμίδα. Είναι λοιπόν προφανές ότι θα πρέπει να περιμένει, για να πραγματοποιηθεί η πρόσθεση των δύο bit της προηγούμενης τάξης. Το ίδιο θα ισχύει και για τον αθροιστή των bit X_1 και Y_1 , ο οποίος θα πρέπει να περιμένει το κρατούμενο από τον αθροιστή των X_0 και Y_0 . Αν κάθε επιμέρους

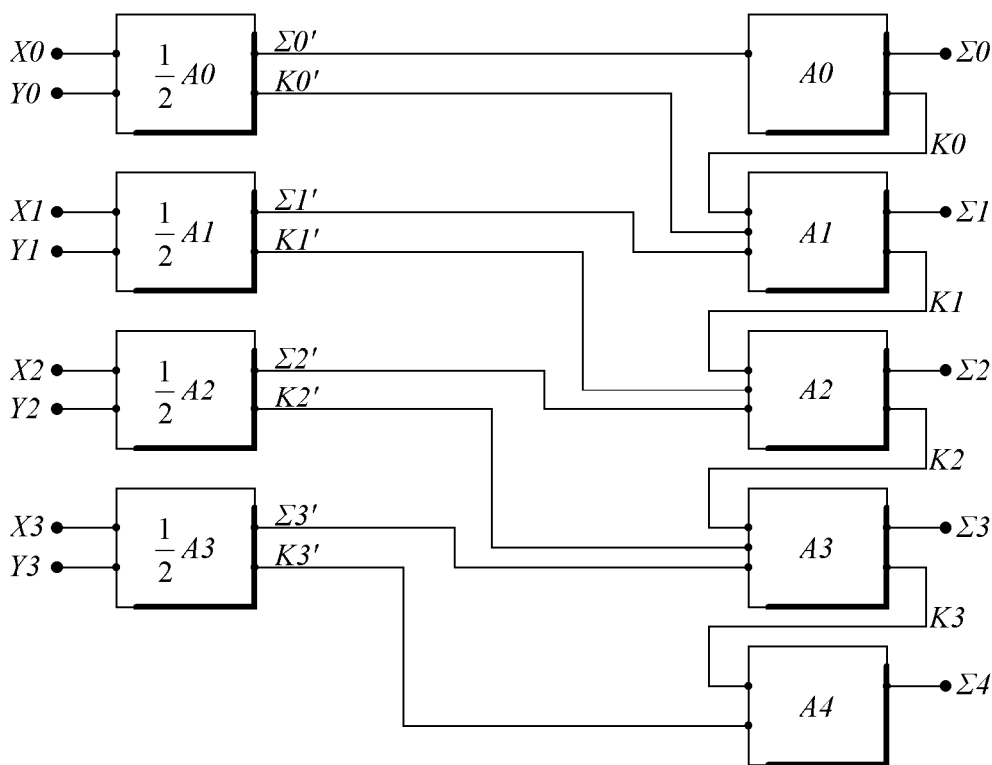
αθροιστής απαιτεί ένα χρόνο t , για να πραγματοποιήσει πρόσθεση δύο bit, τότε για να έχουμε το σωστό αποτέλεσμα στην έξοδο του κυκλώματος θα πρέπει να περιμένουμε ένα χρονικό διάστημα ίσο με $3 \cdot t$. Το πρόβλημα αυτό λύνεται με τη χρήση ενός συνθετότερου τύπου αθροιστή, με πρόβλεψη κρατούμενου, ο οποίος βάσει των τιμών των προηγούμενων bit, υπολογίζει πολύ γρήγορα τα επιμέρους κρατούμενα. Βεβαίως η πρόβλεψη κρατούμενου είναι αρκετά πολύπλοκη να πραγματοποιηθεί, γιατί απαιτεί μεγάλο αριθμό πυλών. Το τελευταίο κάνει δύσκολη την εφαρμογή αυτής της λύσης, ιδιαίτερα όταν οι προστιθέμενοι αριθμοί έχουν μεγάλο μήκος.

Μια λύση στο πρόβλημα αυτό δίνεται, αν υπολογισθεί το άθροισμα σε δύο φάσεις. Στην πρώτη φάση, αθροίζονται οι αριθμοί ψηφίο προς ψηφίο, παράγοντας τις ποσότητες Σ' και K' , όπου Σ' είναι το άθροισμα που προκύπτει χωρίς να ληφθούν υπόψη τα κρατούμενα και K' τα αποτελέσματα των επί μέρους κρατούμενων. Στη δεύτερη φάση αθροίζονται κατάλληλα τα Σ' και K' , παράγοντας το Σ . Το Σχήμα 4.18 παριστάνει ένα κύκλωμα αθροιστή τετρανήφικων δυαδικών αριθμών που εργάζεται με αυτή τη μέθοδο. Αρχικά οι δύο αριθμοί εφαρμόζονται ψηφίο προς ψηφίο σε ισάριθμους ημιαθροιστές σε παράλληλη μορφή. Στη συνέχεια, οι έξοδοι κάθε ημιαθροιστή στέλνονται στις εισόδους των πλήρων αθροιστών.

Τα κρατούμενα των ημιαθροιστών στέλνονται στις επόμενες βαθμίδες των αθροιστών μαζί με τα κρατούμενα των αθροιστών των προηγούμενων βαθμίδων. Οι εξισώσεις που περιγράφουν το κύκλωμα είναι οι:

$$\Sigma_i = (X_i + Y_i) \cdot \overline{(X_i \cdot Y_i)} \quad (4-14) \quad \text{και} \quad K_i = X_i \cdot Y_i \quad (4-15)$$

Οι εξισώσεις των πλήρων αθροιστών τριών εισόδων είναι οι ήδη γνωστές, με τη μόνη διαφορά ότι ο πίνακας αλήθειας, που τις καθορίζει, είναι πλέον απλοποιημένος, λόγω του ότι ορισμένοι συνδυασμοί αποκλείονται εκ των προτέρων. Πράγματι στις εξόδους των ημιαθροιστών είναι προφανές ότι είναι αδύνατη η συνύπαρξη παλμών Σ'_i και K'_i .



Σχήμα 4.18. Δομικό διάγραμμα δυαδικού αθροιστή τεσσάρων ψηφίων

Εξετάζοντας τον αθροιστή της πρώτης θέσης, προκύπτει ότι $K_0 = 0$ γιατί ο A_0 έχει μόνο μία είσοδο Σ_0 . Κατά συνέπεια, για να έχουμε από τον αθροιστή A_1 , $K_1 = 1$, πρέπει να έχουμε συγχρόνως $\Sigma'_1 = 1$ και $K'_0 = 1$. Επίσης, για να είναι το $K_2 = 1$, πρέπει τουλάχιστον σε δύο εισόδους του A_2 να εφαρμόζονται παλμοί. Δεδομένου ότι $\Sigma'_1 = 1$, θα είναι $K'_1 = 0$. Επομένως θα είναι $K_2 = 1$, μόνο εφόσον $\Sigma'_2 = 1$ και $K'_1 = 1$.

Συνεχίζοντας κατ' αυτόν τον τρόπο, συμπεραίνουμε ότι, εάν ο παλμός K_i είναι ίσος με «1», πρέπει υποχρεωτικά να είναι $\Sigma'_i = 1$ και επομένως $K'_i = 0$.

Δηλαδή είναι αδύνατο να είναι ταυτόχρονα «1» τα K_i και K'_i . Ο πίνακας αλήθειας, κατά συνέπεια, απλοποιείται και καταλήγει να έχει τη μορφή του πίνακα 4.13.

Σ'_i	K'_{i-1}	K_{i-1}	Σ_i	K_i
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1

Πίνακας 4.13. Άθροισμα και κρατούμενο δυαδικού αθροιστή τεσσάρων ψηφίων

Από τον πίνακα 4.13 προκύπτουν οι εξισώσεις:

$$K_i = \Sigma'_i \cdot (K'_{i-1} + K_{i-1}) \quad (4-16)$$

$$\Sigma_i = (\Sigma'_i + K'_{i-1} + K_{i-1}) \cdot [\Sigma'_i \cdot (K'_{i-1} + K_{i-1})] \quad (4-17)$$

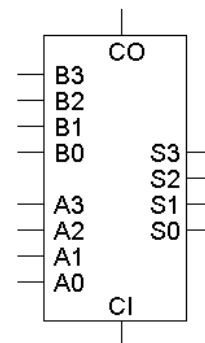
οι οποίες είναι αρκετά απλούστερες σε σύγκριση μ' αυτές που εξετάσαμε προηγουμένως.

4.9.1. Αθροιστής τετραψήφιων δυαδικών αριθμών με χρήση Ο.Κ.

Τέσσερις πλήρεις αθροιστές μαζί δίνουν έναν αθροιστή τετραψήφιων δυαδικών αριθμών. Σε έναν τέτοιο αθροιστή υπάρχουν συνολικά εννέα εισόδοι: Οι δυο αριθμοί των τεσσάρων δυαδικών ψηφίων, έστω A3A2A1A0 και B3B2B1B0 και το αρχικό κρατούμενο εισόδου, CI (Carry In).

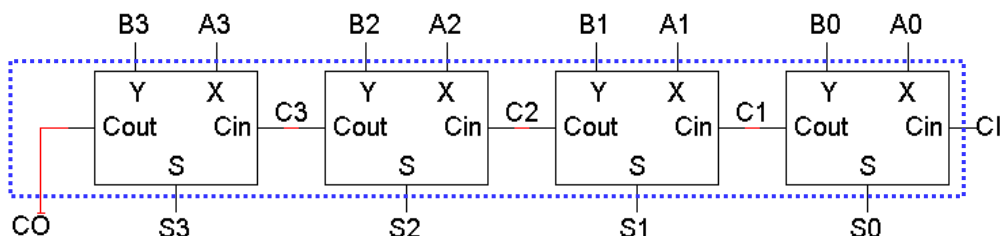
Οι πέντε εξόδοι είναι: Το άθροισμα, το οποίο είναι, επίσης, τετραψήφιο, έστω S3S2S1S0 και ένα κρατούμενο εξόδου, CO.

Φανταστείτε την περίπτωση όπου θα έπρεπε να σχεδιάσουμε έναν αθροιστή εννέα εισόδων χωρίς αυτή την ιεραρχική δομή - θα είχαμε έναν πίνακα αλήθειας με 512 γραμμές (2^9) και πέντε εξόδους!



Σχήμα 4.19. Δομικό διάγραμμα Ο.Κ. αθροιστή

Στο σχήμα 4.19 αποδίδεται το δομικό διάγραμμα του Ο.Κ. του αθροιστή δύο τετραψήφιων αριθμών. Στο Σχήμα 4.20 δίδεται το κύκλωμα του αθροιστή των 4- bits. (μη προσημασμένου)



Σχήμα 4.20. Αναλυτικό διάγραμμα αθροιστή τετραψήφιων αριθμών

Παράδειγμα – (πρόσθεση δύο τετραψήφιων δυαδικών αριθμών)

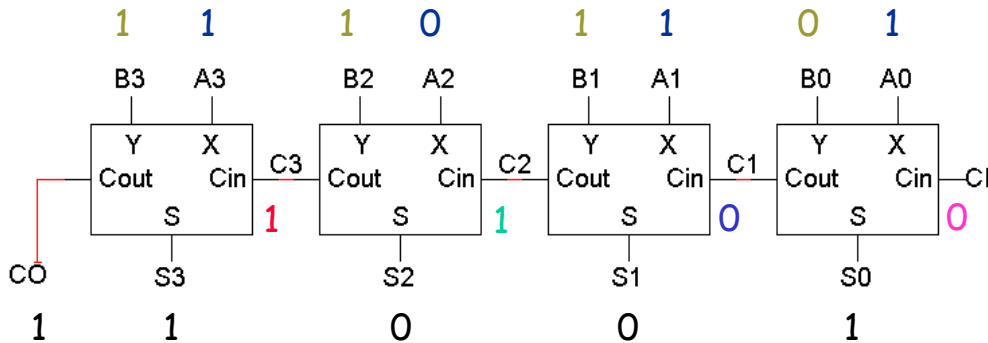
Ας τοποθετήσουμε στο κύκλωμα του προηγούμενου σχήματος 4.18: Έστω A=1011, και B=1110. (Σχήμα 4.21)

Βήμα 1: Τοποθετούμε σε όλες τις εισόδους τα δεδομένα και στο CI=0 (Αρχικοποίηση)

Βήμα 2: Το κύκλωμα παράγει το C1 και S0 ($1 + 0 + 0 = 01$)

- Βήμα 3:** Χρησιμοποιούμε το C1, για να βρούμε τα C2 και S1 $(1 + 1 + 0 = 10)$
Βήμα 4: Χρησιμοποιούμε το C2, για να υπολογίσουμε τα C3 και S2 $(0 + 1 + 1 = 10)$
Βήμα 5: Χρησιμοποιούμε το C3, για να υπολογίσουμε τα CO και S3 $(1 + 1 + 1 = 11)$

Το τελικό αποτέλεσμα είναι 11001.



Σχήμα 4.21. Παράδειγμα αθροιστή δύο τετραψήφιων αριθμών

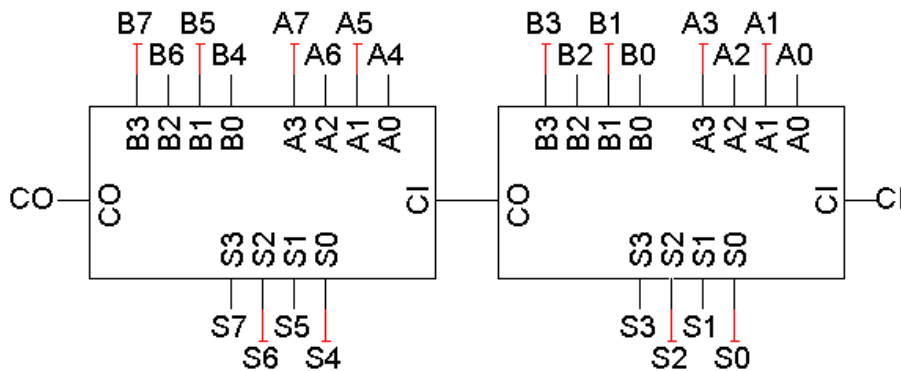
Υπερχείλιση

Στην περίπτωση αυτή, σημειώστε ότι η απάντηση (11001) έχει μήκος πέντε ψηφίων, αν και οι είσοδοί μας είχαν μόνο 4 ψηφία (1011 και 1110). Αυτό ονομάζεται *υπερχείλιση*.

Παρόλο που η απάντηση 11001 είναι σωστή, δεν μπορούμε να χρησιμοποιήσουμε το αποτέλεσμα αυτό σε κανένα επόμενο υπολογισμό με αθροιστή 4-bit. Στη μη προσημασμένη πρόσθεση, έχουμε υπερχείλιση, όταν το κρατούμενο εξόδου είναι 1.

Σχεδίαση Ιεραρχικού αθροιστή

Όταν προσθέτουμε δύο τετραψήφιους αριθμούς, το κρατούμενο εισόδου είναι πάντα 0. Αλλά τίθεται το ερώτημα γιατί ο αθροιστής των 4-bit πρέπει να έχει σαν είσοδο την CI (Είσοδος κρατουμένου). Ένας λόγος είναι, για να μπορέσουμε να τοποθετήσουμε δύο αθροιστές των 4-bit, ώστε να δημιουργήσουμε μεγαλύτερους αθροιστές. Στο σχήμα 4.22 απεικονίζεται ένας αθροιστής των 8-bit. Η είσοδος CI είναι, επίσης, χρήσιμη και στην αφαίρεση, όπως θα δούμε στη συνέχεια.



Σχήμα 4.22. Αθροιστής 8 δυαδικών ψηφίων

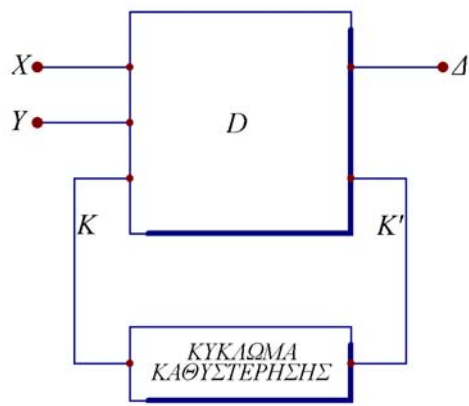
4.10. Σειριακός πλήρης αφαιρέτης (Serial full subtractor)

Για τη μελέτη του αφαιρέτη χρησιμοποιείται πίνακας αλήθειας, όπως και στον αθροιστή, μόνο που εδώ έχουμε έλλειμμα, και όχι περίσσειμα, σε περίπτωση ύπαρξης κρατουμένου. Ο πίνακας αλήθειας ενός σειριακού πλήρους αφαιρέτη είναι αυτός που φαίνεται στον πίνακα 4.14.

Από τον πίνακα 4.14 προκύπτουν οι παρακάτω εξισώσεις:

$$\Delta = \bar{X} \cdot \bar{Y} \cdot R + \bar{X} \cdot Y \cdot \bar{R} + X \cdot \bar{Y} \cdot \bar{R} + X \cdot Y \cdot R = \bar{X} \cdot (\bar{Y} \cdot R + Y \cdot \bar{R}) + X \cdot (\bar{Y} \cdot \bar{R} + Y \cdot R) = \bar{X} \cdot (Y + R) \cdot (\bar{Y} \cdot \bar{R}) + X \cdot [(\bar{Y} + R) + Y \cdot R] \quad (4-18)$$

Η πραγματοποίηση του κυκλώματος του σειριακού αφαιρέτη βασίζεται στην αρχή της επαναφοράς του κρατούμενου στην είσοδο του αφαιρέτη και στη διαδοχική λήψη του αποτελέσματος από την έξοδο. Στο Σχήμα 4.24 αποδίδεται το δομικό διάγραμμα σειριακού δυαδικού αφαιρέτη.



Σχήμα 4.24. Δομικό διάγραμμα σειριακού δυαδικού αφαιρέτη

4.10.1.Κυκλώματα αθροιστών και αφαιρέτών

Κύκλωμα αφαίρεσης

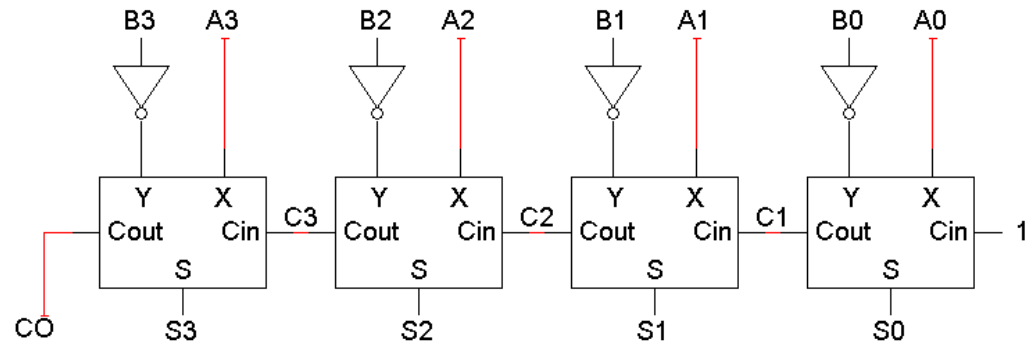
Θα μπορούσαμε να κατασκευάσουμε κατευθείαν ένα κύκλωμα αφαίρεσης με παρόμοιο τρόπο με αυτόν με τον οποίο κατασκευάσαμε μη προσημασμένους αθροιστές. Μας είναι ήδη γνωστή η διαδικασία μετατροπής της αφαίρεσης σε πρόσθεση, χρησιμοποιώντας το συμπλήρωμα ως προς δύο. Αλγεβρικά έχουμε, $A - B = A + (-B)$.

Έτσι, για να αφαιρέσουμε το B από το A, μπορούμε να προσθέσουμε τον αντίθετο του B στον A. Με αυτόν τον τρόπο μπορούμε να ξαναχρησιμοποιήσουμε το κύκλωμα του μη προσημασμένου αθροιστή, που έχουμε ήδη μελετήσει. (Σχήμα 4.33)

Κύκλωμα αφαίρεσης τετραψήφιων αριθμών με συμπλήρωμα ως προς δύο

Για να βρούμε το $A - B$ με έναν αθροιστή, πρέπει να συμπληρώσουμε κάθε bit του B. Κατόπιν θέτουμε στο κρατούμενο εισόδου του πρώτου αθροιστή το 1. Το τελικό αποτέλεσμα είναι $A + B' + 1$, όπου το $B' + 1$ είναι το συμπλήρωμα ως προς δύο, του B. (Σχήμα 4.25)

Λάβετε υπόψη σας πως τα A3, B3 και S3 εδώ, είναι στην πραγματικότητα τα bits προσήμων.



Σχήμα 4.25. Κύκλωμα αφαίρεσης 4-bit με συμπλήρωμα ως προς δύο

Σύγκριση των κυκλωμάτων του αθροιστή και του αφαιρέτη

Οι μόνες διαφορές μεταξύ των κυκλωμάτων αθροιστή και αφαιρέτη είναι ότι:

- Ο αφαιρέτης πρέπει να αντιστρέψει τα B3B2B1B0.
- Ο αφαιρέτης θέτει το αρχικό κρατούμενο σε 1, αντί για 0.

Συνεπώς δεν είναι δύσκολο να φτιάξουμε ένα κύκλωμα που θα εκτελεί συγχρόνως και πρόσθεση και αφαίρεση.

Κύκλωμα αθροιστή-αφαιρέτη

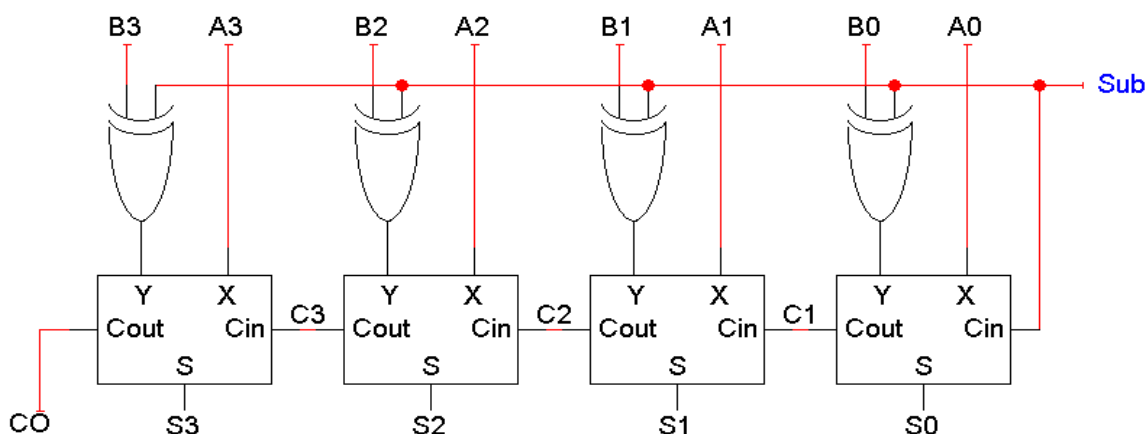
Οι πύλες XOR μας επιτρέπουν να συμπληρώσουμε επιλεκτικά την είσοδο B με βάση τις σχέσεις:

$$X \oplus 0 = X \quad X \oplus 1 = X'$$

Όταν η Sub = 0, οι έξοδοι των πυλών XOR B3' B2' B1' B0' και το κρατούμενο εισόδου είναι 0. Η έξοδος του αθροιστή θα είναι A + B + 0, ή A + B.

Όταν η Sub = 1, οι έξοδοι των πυλών XOR B3' B2' B1' B0' και το κρατούμενο εισόδου είναι 1. Έτσι η έξοδος του αθροιστή θα είναι μια αφαίρεση με συμπλήρωμα ως προς δύο, A - B.

Στο Σχήμα 4.26 που ακολουθεί απεικονίζεται το κύκλωμα του αθροιστή-αφαιρέτη.

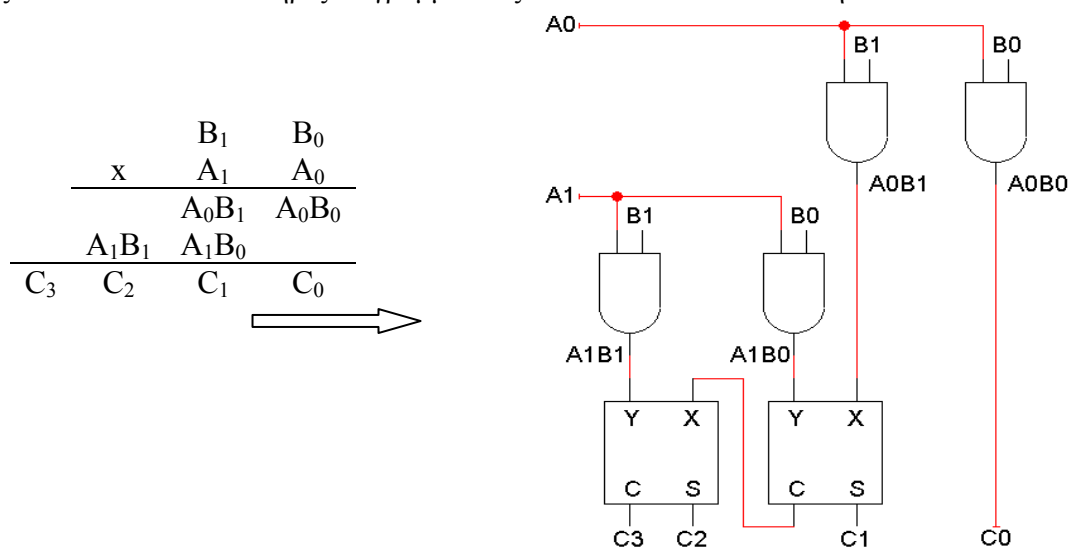


Σχήμα 4.26. Κύκλωμα αθροιστή- αφαιρέτη 4-bit

4.10.2.Κύκλωμα δυαδικού πολλαπλασιαστή

Οι πύλες AND παράγουν τα μερικά γινόμενα. Για ένα διψήφιο επί διψήφιο πολλαπλασιαστή, μπορούμε απλά να χρησιμοποιήσουμε δύο ημιαθροιστές, για να αθροίσουμε τα μερικά γινόμενα. Γενικά όμως θα χρησιμοποιήσουμε πλήρεις αθροιστές.

Τα C₃-C₀ του πίνακα είναι τα γινόμενα ή/και τα αθροίσματα των μερικών γινομένων, και όχι τα κρατούμενα! (Στο Σχήμα 4.27 απεικονίζεται ένας 2x2 δυαδικός πολλαπλασιαστής) και στο κύκλωμα του σχήματος 4.28 αποδίδεται το πλήρες διάγραμμα ενός δυαδικού πολλαπλασιαστή 4x4.

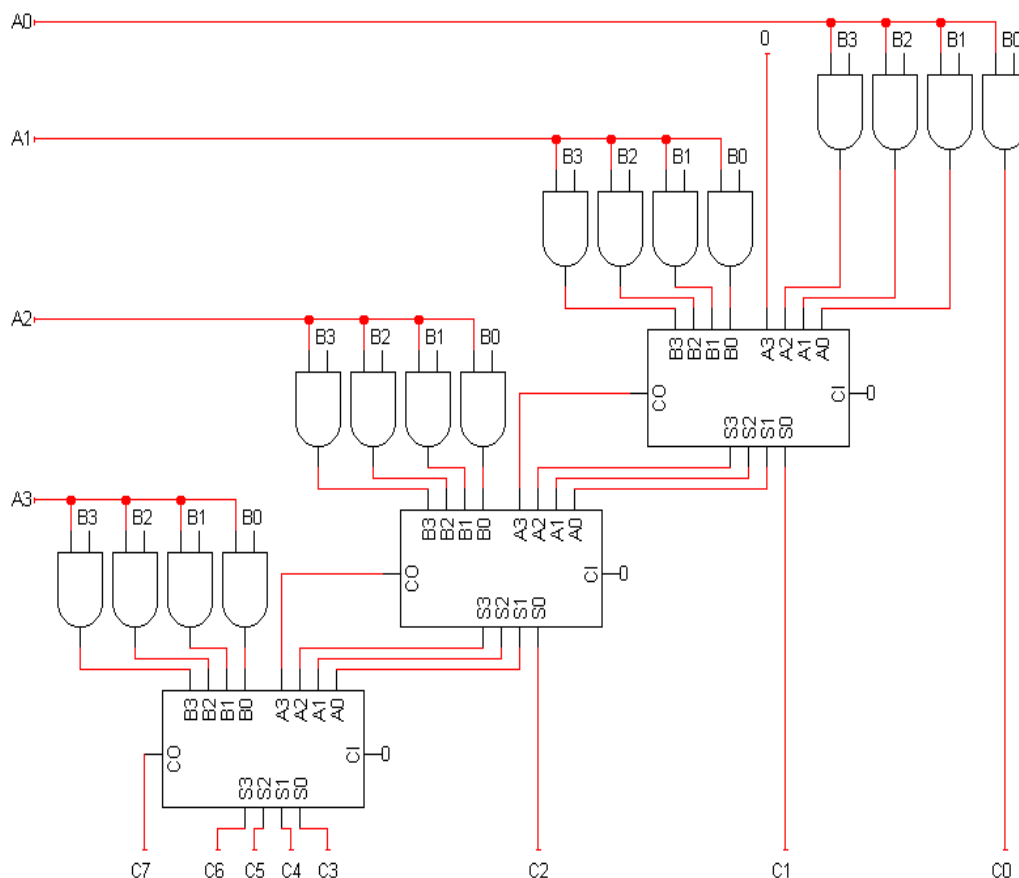


Σχήμα 4.27. Δυαδικός πολλαπλασιαστής 2x2

Ο πολλαπλασιαστής των τεσσάρων ψηφίων παράγει ένα αποτέλεσμα οκτώ ψηφίων. Θα μπορούσαμε να κρατήσουμε και τα οκτώ ψηφία του αποτελέσματος ή, αν θέλαμε αποτέλεσμα τεσσάρων ψηφίων, θα μπορούσαμε να αγνοήσουμε τα ψηφία C4-C7, και στην περίπτωση που το αποτέλεσμα είναι μεγαλύτερο από 4 ψηφία, να το θεωρήσουμε σαν κατάσταση υπερχείλισης.

Οι πολλαπλασιαστές είναι γενικά πολύπλοκα κυκλώματα. Έτσι, όταν πολλαπλασιάζουμε έναν αριθμό m -ψηφίων με έναν αριθμό n -ψηφίων:

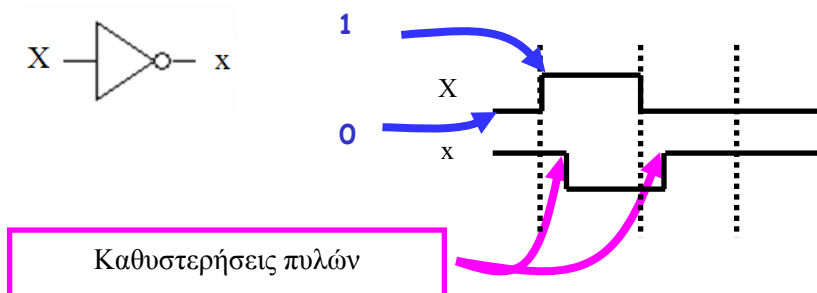
Υπάρχουν n μερικά γινόμενα, ένα για κάθε ψηφίο του πολλαπλασιαστή. Αυτό απαιτεί $n-1$ αθροιστές, ο καθένας των οποίων θα μπορεί να προσθέσει m ψηφία (το μέγεθος του πολλαπλασιαστέου). Το κύκλωμα για πολλαπλασιασμό 32-bit ή 64-bit θα ήταν τεράστιο!



Σχήμα 4.28. Αναλυτικό διάγραμμα δυαδικού πολλαπλασιαστή 4x4

4.11. Καθυστερήσεις πυλών

Κάθε πύλη χρειάζεται κάποιο χρόνο από τη στιγμή που θα δοθεί το σήμα στην είσοδο μέχρι να εμφανιστούν στην έξοδο τα σωστά αποτελέσματα. Αυτό το χρονικό διάστημα ονομάζεται *καθυστερήση πύλης*.



Σχήμα 4.29. Γραφική απεικόνιση των καθυστερήσεων των πυλών

Υπάρχουν πολλοί λεπτομερείς τρόποι υπολογισμού των καθυστερήσεων κάθε πύλης, που είναι όμως αρκετά πολύπλοκοι. Ας θεωρήσουμε πως αυτή η καθυστέρηση είναι σταθερή και ότι είναι η ίδια για όλες τις πύλες. Μπορούμε να χρησιμοποιήσουμε ένα διάγραμμα χρονισμού, για να δείξουμε γραφικά τις καθυστερήσεις των πυλών (Σχήμα 4.29).

Υπολογισμός των κρατούμενων εξόδου

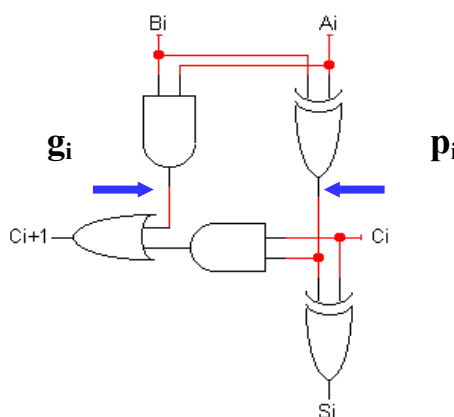
Μπορούμε, αντί να περιμένουμε τα κρατούμενα των εξόδων από όλες τις προηγούμενες βαθμίδες, να τα υπολογίσουμε απευθείας με ένα κύκλωμα δύο επιπέδων (Σχήμα 4.30), μειώνοντας έτσι την καθυστέρηση. Πρώτα ορίζουμε δύο συναρτήσεις:

Την «παράγουσα συνάρτηση» g_i που θα παράγει 1, όταν πρέπει να υπάρχει κρατούμενο εξόδου από τη θέση i (δηλ. όταν το A_i και το B_i είναι 1). $g_i = A_i B_i$

Τη συνάρτηση «διάδοσης» p_i που είναι αληθής, όταν, εάν υπάρχει ένα εισερχόμενο κρατούμενο, αυτό «διαδίδεται». (δηλ. όταν ή $A_i=1$ ή $B_i=1$, αλλά ποτέ και τα δύο μαζί) $p_i = A_i \oplus B_i$ Είναι προφανές ότι το αρχικό κρατούμενο είναι μηδέν.

Μπορούμε λοιπόν να ξαναγράψουμε τη συνάρτηση του κρατουμένου εξόδου ως: $c_{i+1} = g_i + p_i c_i$

A_i	B_i	C_i	C_{i+1}
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1



Σχήμα 4.30. Βασικό κύκλωμα δύο επιπέδων

Ας δούμε τις εξισώσεις των κρατουμένων εξόδου για συγκεκριμένα ψηφία, χρησιμοποιώντας την προηγούμενη γενική εξίσωση: $c_{i+1} = g_i + p_i c_i$. Έτσι έχουμε:

$$c_1 = g_0 + p_0 c_0$$

$$c_2 = g_1 + p_1 c_1 = g_1 + p_1 (g_0 + p_0 c_0) = g_1 + p_1 g_0 + p_1 p_0 c_0$$

$$c_3 = g_2 + p_2 c_2 = g_2 + p_2 (g_1 + p_1 g_0 + p_1 p_0 c_0) = g_2 + p_2 g_1 + p_2 p_1 g_0 + p_2 p_1 p_0 c_0$$

$$c_4 = g_3 + p_3 c_3 = g_3 + p_3 (g_2 + p_2 g_1 + p_2 p_1 g_0 + p_2 p_1 p_0 c_0) = g_3 + p_3 g_2 + p_3 p_2 g_1 + p_3 p_2 p_1 g_0 + p_3 p_2 p_1 p_0 c_0$$

4.12. Κυκλώματα σύγκρισης

Τα κυκλώματα σύγκρισης (Comparators) χρησιμεύουν για τη σύγκριση δυαδικών αριθμών ή αριθμών άλλου συστήματος αρίθμησης, που είναι κωδικοποιημένοι στο δυαδικό. Οι προς σύγκριση αριθμοί μπορούν να θεωρηθούν ότι μεταφέρουν, ή όχι, πληροφορία προσημίου, λαμβάνοντας υπόψη ότι το ψηφίο μεγαλύτερης τάξης παριστάνει το πρόσημο του αριθμού. Κατά συνέπεια, για τη σύγκριση μεταξύ ετερόσημων αριθμών είναι αρκετό να συγκριθούν μόνο τα πρόσημα, ενώ μεταξύ των ομόσημων να ληφθεί υπόψη η απόλυτη τιμή τους. Στη δεύτερη περίπτωση, εάν είναι και οι δύο θετικοί ο μεγαλύτερος βρίσκεται κανονικά, ενώ, εάν είναι και οι δύο αρνητικοί, ο μεγαλύτερος θα είναι αυτός που θα έχει τη μικρότερη απόλυτη τιμή. Οι προς σύγκριση αριθμοί έχουν πάντοτε τα ψηφία μικρότερης τάξης στο δεξιότερο μέρος τους. (Κοσσίδης, & Γιαννακόπουλος, 2006; Balabanian, & Carlson, 2007; Tocci, et al., 2010)

4.12.1. Σειριακός δυαδικός συγκριτής

Ο σειριακός δυαδικός συγκριτής (serial binary comparator) εκτελεί τη σε σειρά σύγκριση δύο αριθμών συγκρίνοντας τα δυαδικά ψηφία του ενός αριθμού με τα αντίστοιχα του άλλου, έως ότου φθάσουμε στο μεγαλύτερης τάξης ψηφίο. Κατά συνέπεια, η σύγκριση δύο ψηφίων θα έχει ένα από τα αποτελέσματα που φαίνονται στον πίνακα 4.15.

Ένας ηλεκτρονικός υπολογιστής είναι σε θέση να χρησιμοποιήσει κατά διάφορους τρόπους το αποτέλεσμα μιας σύγκρισης. Υπάρχουν υπολογιστές που, συγκρίνοντας δύο δυαδικά ψηφία, είναι ικανοί να δίνουν όλα τα δυνατά αποτελέσματα $>$, \geq , $=$, \leq , $<$, \neq , άλλοι που δίνουν μερικά από αυτά και τέλος άλλοι που είναι σε θέση να δώσουν σαν αποτέλεσμα μόνο την ένδειξη $=$ ή την ένδειξη \neq . Το βασικό κύκλωμα σύγκρισης σε κάθε συγκριτή είναι ο δυαδικός συγκριτής, που αποδίδεται στο Σχήμα 4.33. Το κύκλωμα αυτό προκύπτει εύκολα από τον πίνακα 4.16. (Με «1» παριστάνεται η ισότητα και με «0» η ανισότητα).

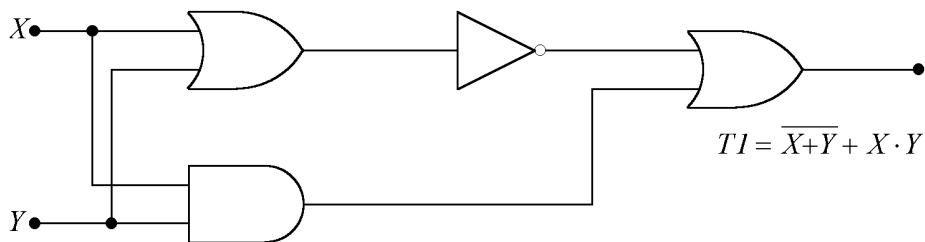
X	Y	Αποτέλεσμα
0	0	=
0	1	<
1	0	>
1	1	=

Πίνακας 4.15. Αποτελέσματα σύγκρισης δύο ψηφίων

X	Y	T ₁
0	0	1
0	1	0
1	0	0
1	1	1

Πίνακας 4.16. Πίνακας αλήθειας δυαδικού συγκριτή ισότητας

Από τον πίνακα προκύπτει ότι: $T_1 = \overline{X} \cdot \overline{Y} + X \cdot Y = \overline{X + Y} + X \cdot Y$ (XNOR)



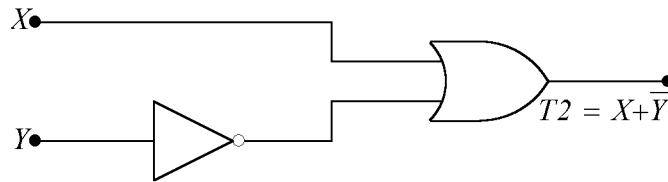
Σχήμα 4.33. Υλοποίηση δυαδικού συγκριτή ισότητας

Εάν ο δυαδικός συγκριτής έχει σχεδιαστεί, ώστε να ανταποκρίνεται στις περιπτώσεις **μεγαλύτερο ή ίσο (\geq) και μικρότερο ($<$)**, δίνοντας αποτέλεσμα «1» και «0» αντίστοιχα, θα έχουμε τον παρακάτω πίνακα 4.17.

X	Y	T ₂
0	0	1
0	1	0
1	0	1
1	1	1

Πίνακας 4.17. Πίνακας αλήθειας δυαδικού συγκριτή

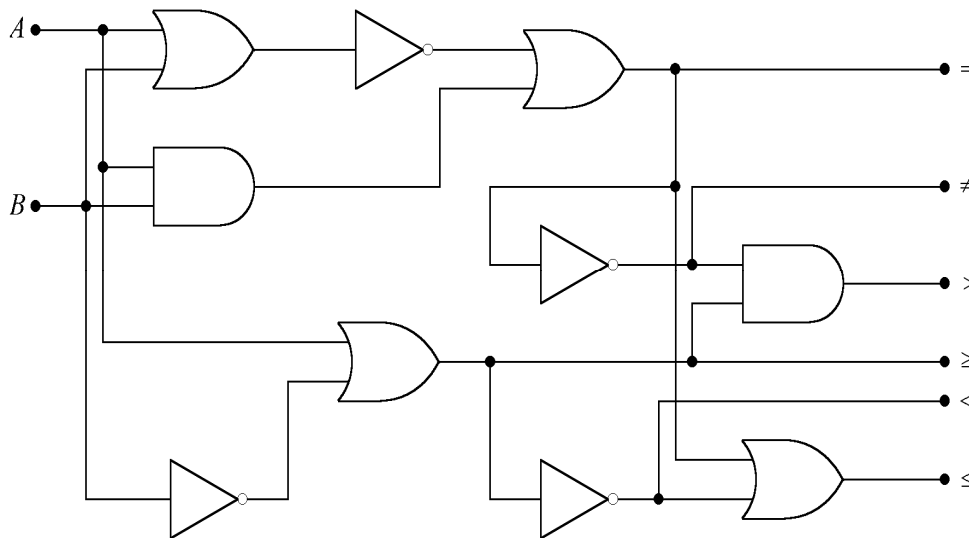
Η λογική συνάρτηση, που προκύπτει από τον ανωτέρω πίνακα αλήθειας, είναι η: $T_2 = \overline{X} \cdot \overline{Y} + X \cdot \overline{Y} + X \cdot Y = X + \overline{Y}$ και το αντίστοιχο κύκλωμα σύγκρισης δίνεται στο Σχήμα 4.34.



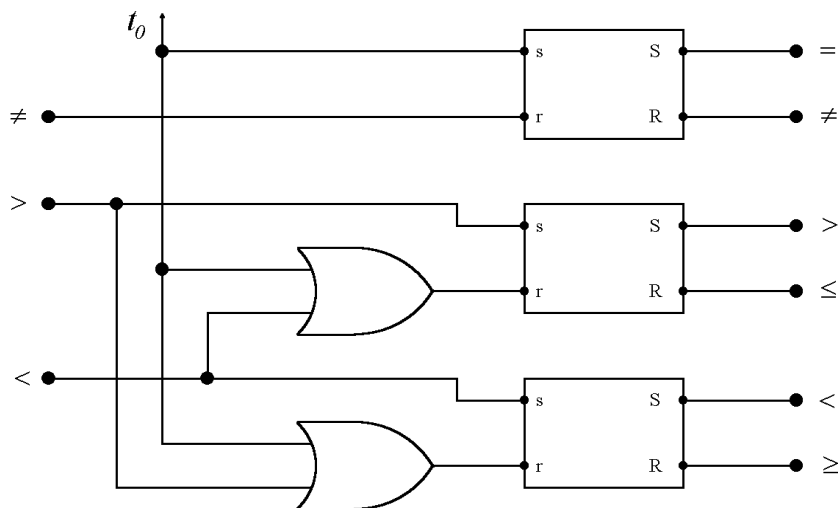
Σχήμα 4.34. Δυαδικός συγκριτής με πύλες

Συνενώνοντας τα δύο κυκλώματα, που υπολογίσαμε προηγουμένως, επιτυγχάνουμε ένα ενιαίο κύκλωμα (Σχήμα 4.35α) που είναι σε θέση να αναδείξει όλα τα δυνατά αποτελέσματα μιας σύγκρισης. Ο συγκριτής αυτός ανταποκρίνεται σε συγκρίσεις αριθμών του ενός ψηφίου. Σε περίπτωση αριθμών με περισσότερα ψηφία συγκρίνονται διαδοχικά όλα τα ψηφία, αρχίζοντας από αυτά της μικρότερης τάξης. Συνέπεια αυτού του τρόπου σύγκρισης είναι να έχουμε, πολλές φορές, λανθασμένο αποτέλεσμα, αφού οι έξοδοι του κυκλώματος θα διατηρούν το αποτέλεσμα της σύγκρισης των δύο τελευταίων ψηφίων, έχοντας «ξεχάσει» τα αποτελέσματα των προηγουμένων συγκρίσεων. Πράγματι, συγκρίνοντας τους δεκαδικούς αριθμούς 5240 και 6421, η ένδειξη που προκύπτει από τη σύγκριση των ψηφίων με μεγαλύτερο βάρος θα είναι το ίσον (=), αντίθετα προς την πραγματική ένδειξη που είναι το μικρότερο (<). Αυτό το σφάλμα παρακάμπτεται, καταχωρώντας την τελευταία ένδειξη ανισότητας, η οποία δεν πρέπει να μεταβάλλεται από τυχόν υπάρχουσες στη συνέχεια ισότητες. Η ανισότητα καταγράφεται με τη βοήθεια R-S *f-f*, όπως αυτών που φαίνονται στο Σχήμα 4.35β. (Τη λειτουργία των *f-f* θα μελετήσουμε στο έκτο κεφάλαιο).

Τα *f-f* καταγραφής αρχικά βρίσκονται όλα σε κατάσταση που να περιλαμβάνει την ισότητα (\geq , $=$, \leq). Αυτό επιτυγχάνεται εφαρμόζοντας έναν παλμό t_0 στην είσοδο των *f-f*, προτού αρχίσει η εφαρμογή των ψηφίων σύγκρισης στο κύκλωμα. Επομένως το *f-f* που παριστάνει το « \neq » ή το « \neq » παραμένει στην κατάσταση του « \neq », εφόσον εφαρμοσθεί παλμός ανισότητας. Κατ' αναλογία, για τις περιπτώσεις των επομένων *f-f* οι καταστάσεις « \leq » και « \geq » παραμένουν, έως ότου εφαρμοσθούν παλμοί « $>$ » ή « $<$ ».



Σχήμα 4.35α. Αναλυτικό διάγραμμα πλήρους σειριακού δυαδικού συγκριτή με πύλες

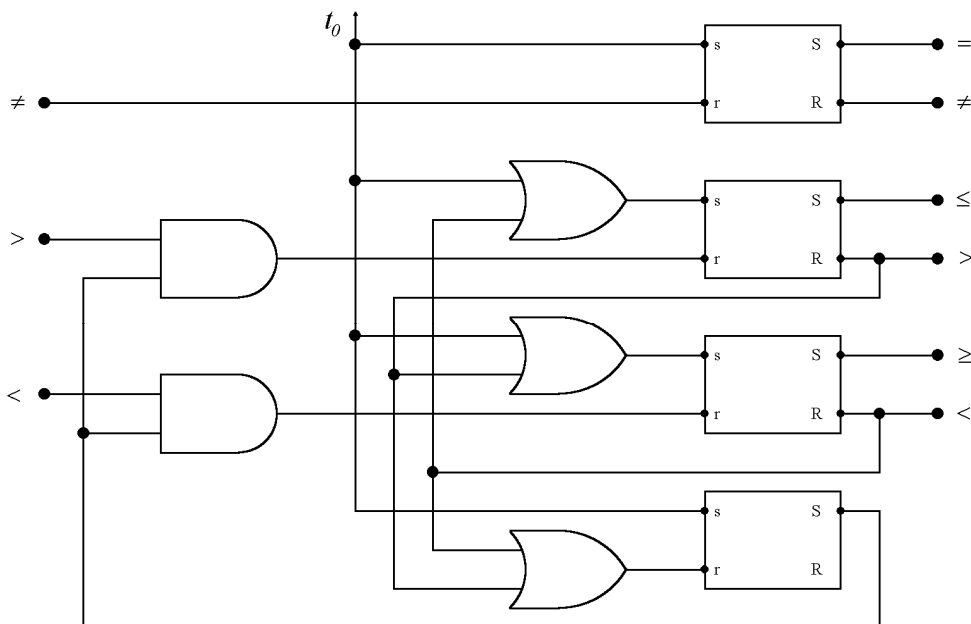


Σχήμα 4.35β. Αναλυτικό διάγραμμα πλήρους σειριακού δυαδικού συγκριτή

4.12.2. Παράλληλος δυαδικός συγκριτής

Μία απευθείας σύγκριση δύο αριθμών καταχωρημένων παράλληλα θα οδηγούσε σε πολύπλοκα κυκλώματα στην περίπτωση της σειριακής σύγκρισης δεδομένου ότι, εάν οι αριθμοί, που συγκρίνονται διαφέρουν ως προς το ψηφίο μικρότερης τάξης, το αποτέλεσμα εξαρτάται από τον διαδοχικό έλεγχο όλων των προηγούμενων ψηφίων, και απαιτούνται κυκλώματα που να συνδέουν ταυτόχρονα όλα τα υπάρχοντα ψηφία των δύο προς σύγκριση αριθμών. Λύση στο πρόβλημα αποτελεί η ανά ζεύγη σύγκριση των ψηφίων της ίδιας τάξης, με τη χρήση πυλών XNOR, όσων και το πλήθος των ψηφίων των αριθμών. Το αποτέλεσμα των εξόδων των πυλών οδηγείται σε μια πύλη AND της οποίας το αποτέλεσμα είναι «1» μόνο εάν όλα τα ψηφία των αριθμών έχουν την ίδια τιμή.

Μία άλλη λύση στο παραπάνω πρόβλημα επιτυγχάνεται, με τη σειριακή σύγκριση των ψηφίων των αριθμών, αρχίζοντας από τα ψηφία μεγαλύτερης τάξης. Ο συγκριτής αυτός ονομάζεται παράλληλος δυαδικός συγκριτής (Parallel binary comparator). Το κύκλωμά του αποδίδεται στο Σχήμα 4.36.



Σχήμα 4.36. Σειριακός δυαδικός συγκριτής αρχίζοντας από τα ψηφία μεγαλύτερης τάξης

4.12.3. Συγκριτής δύο δυαδικών ψηφίων

Έστω ότι θέλουμε να σχεδιάσουμε ένα κύκλωμα που θα συγκρίνει δύο αριθμούς των 2-bit, τον A και τον B. Το κύκλωμα πρέπει να έχει τρεις εξόδους:

- **G** («Greater») : θα είναι 1 μόνο όταν $A > B$ (μεγαλύτερο).
- **E** («Equal») : θα είναι 1 μόνο όταν $A = B$ (ίσο).
- **L** («Lesser») : θα είναι 1 μόνο όταν $A < B$ (μικρότερο).

Οι εισόδοι A και B θα είναι 00, 01, 10, ή 11 (0, 1, 2 ή 3 στο δεκαδικό). Για οποιαδήποτε είσοδο A και B, μόνο μια από τις τρεις εξόδους θα είναι 1.

Βήμα 1: Πόσες εισόδους και πόσες εξόδους θα έχουμε;

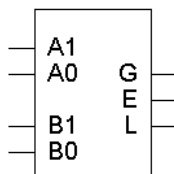
Δύο αριθμοί των 2-bit μας δίνουν συνολικά τέσσερις εισόδους.

- Αρχικά θα πρέπει να ονομάσουμε καθεμία από τις εισόδους.

Έστω ότι ο πρώτος αριθμός αποτελείται από τα ψηφία A1 και A0 από τα αριστερά προς τα δεξιά και ο δεύτερος αριθμός αποτελείται από τα ψηφία B1 και B0.

- Το πρόβλημα απαιτεί τρεις εξόδους: G, E και L.

Στο σχήμα 4.37 αποδίδεται το δομικό διάγραμμα στο οποίο απεικονίζονται σαφώς οι εισόδοι και οι εξόδοι. Αφού προσδιορίσουμε τις εξισώσεις, θα μπορέσουμε να σχεδιάσουμε τα στοιχεία του κυκλώματος, που περιέχονται μέσα στο δομικό στοιχείο.



Σχήμα 4.37. Δομικό διάγραμμα συγκριτή δύο δυαδικών ψηφίων

Βήμα 2: Προσδιορισμός της λειτουργίας

Για το πρόβλημα αυτό είναι ευκολότερο να αρχίσουμε με τον πίνακα αληθείας (πίνακας 4.18).

Με αυτόν τον τρόπο μπορούμε να δείξουμε σαφώς τις σχέσεις (>, =, <) μεταξύ των εισόδων.

Μια συνάρτηση τεσσάρων εισόδων έχει έναν πίνακα αληθείας δεκαέξι γραμμών. Συνήθως είναι πιο εύκολο να τοποθετήσουμε τις γραμμές του πίνακα αληθείας με δυαδική σειρά. Σε αυτή την περίπτωση, από 0000 ως 1111 για τα A1, A0, B1 και B0.

A1	A0	B1	B0	G	E	L
0	0	0	0	0	1	0
0	0	0	1	0	0	1
0	0	1	0	0	0	1
0	0	1	1	0	0	1
0	1	0	0	1	0	0
0	1	0	1	0	1	0
0	1	1	0	0	0	1
0	1	1	1	0	0	1
1	0	0	0	1	0	0
1	0	0	1	1	0	0
1	0	1	0	0	1	0
1	0	1	1	0	0	1
1	1	0	0	1	0	0
1	1	0	1	1	0	0
1	1	1	0	1	0	0
1	1	1	1	0	1	0

Πίνακας 4.18. Πίνακας αληθείας συγκριτή δύο δυαδικών ψηφίων

Για παράδειγμα, έστω ότι έχουμε να συγκρίνουμε το 01 με το 10. Το $01 < 10$ και έτσι η έκτη σειρά του πίνακα αληθείας, (που αντιστοιχεί στις εισόδους $A=01$ και $B=10$) μας δίνει την έξοδο $L=1$, και, συνεπώς, οι έξοδοι G και E είναι και οι δύο 0. Σκεπτόμενοι με το ίδιο τρόπο, συμπληρώνουμε τις εξόδους του πίνακα 3.20.

Βήμα 3: Απλουστευμένες Boolean εκφράσεις

Για την απλοποίηση θα χρησιμοποιήσουμε το χάρτη Karnaugh. Υπάρχουν τρεις συναρτήσεις (και για τις τρεις οι εισόδοι είναι κοινές $\overline{A1} A0 B1 B0$). Έτσι χρειαζόμαστε τρεις χάρτες Karnaugh.

		B1			
		0	0	0	0
		1	0	0	0
A1		1	1	0	1
		1	1	0	0
		B0			

$$G(A1, A0, B1, B0) = A1 A0 \overline{B0} + A0 \overline{B1} \overline{B0} + A1 \overline{B1}$$

		B1			
		1	0	0	0
		0	1	0	0
A1		0	0	1	0
		0	0	0	1
		B0			

$$E(A1, A0, B1, B0) = \overline{A1} \overline{A0} \overline{B1} \overline{B0} + \overline{A1} A0 \overline{B1} B0 + A1 A0 B1 B0 + A1 \overline{A0} B1 \overline{B0}$$

		B1			
		0	1	1	1
		0	0	1	1
A1		0	0	0	0
		0	0	1	0
		B0			

$$L(A1, A0, B1, B0) = \overline{A1} \overline{A0} B0 + \overline{A0} B1 B0 + \overline{A1} B1$$

Βήμα 4: Σχεδιασμός του κυκλώματος

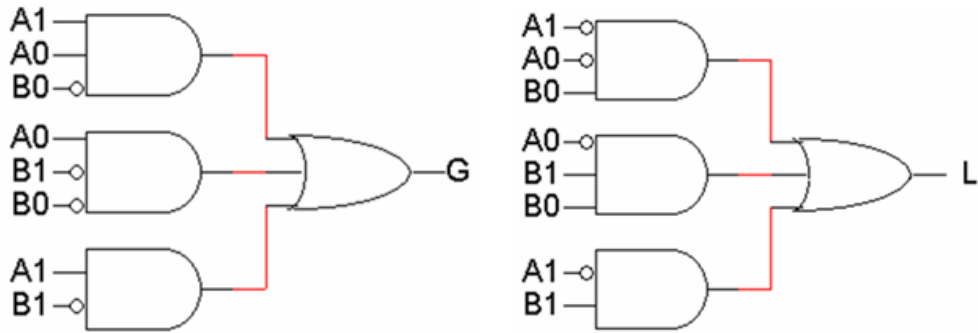
Με δεδομένες τις απλοποιημένες συναρτήσεις, όπως προέκυψαν μετά την απλοποίηση:

$$G = A1 A0 \overline{B0} + A0 \overline{B1} \overline{B0} + A1 \overline{B1}$$

$$E = \overline{A1} \overline{A0} \overline{B1} \overline{B0} + \overline{A1} A0 \overline{B1} B0 + A1 A0 B1 B0 + A1 \overline{A0} B1 \overline{B0}$$

$$L = \overline{A1} \overline{A0} B0 + \overline{A0} B1 B0 + \overline{A1} B1$$

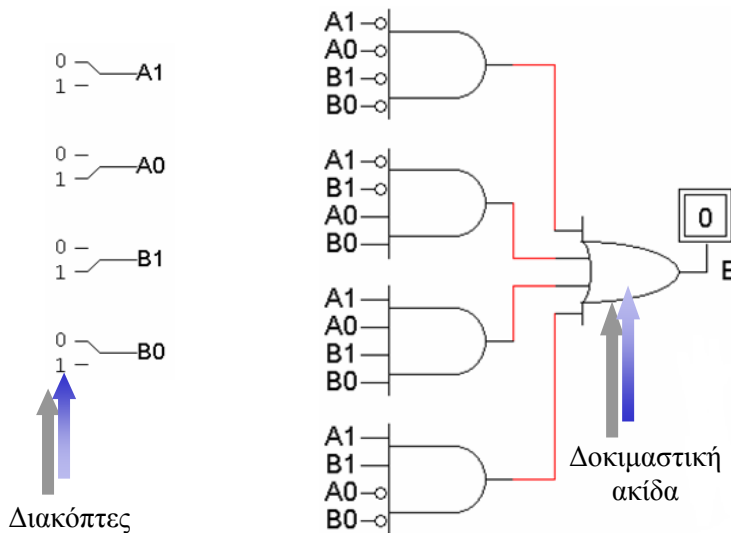
προχωρούμε στην υλοποίηση της κάθε συνάρτησης ξεχωριστά: (Σχήμα 4.38). Το κύκλωμα της συνάρτησης E αποδίδεται στο Σχήμα 4.39 με τη χρήση της δοκιμαστικής ακίδας καθώς και των διακοπών εισόδου.



Σχήμα 4.38. Υλοποίηση των συναρτήσεων G και L του συγκριτή

Έλεγχος του κυκλώματος με λογισμικό

Οι είσοδοι προφανώς προέρχονται από δυαδικούς διακόπτες. Για να δούμε και να ελέγξουμε τις εξόδους, χρησιμοποιούμε δυαδικές δοκιμαστικές ακίδες. Στο σχήμα που ακολουθεί αποδίδεται ο έλεγχος του κυκλώματος E σε περίπτωση στην οποία δεν ισχύει η ισότητα ($E=0$) (Σχήμα 4.39). Ο ίδιος έλεγχος στα κυκλώματα των G, L θα έδινε 1 στο κύκλωμα του L.



Σχήμα 4.39. Έλεγχος του κυκλώματος ισοδυναμίας «E» του συγκριτή

Για την αναπαράσταση των δεδομένων:

Χρησιμοποιήσαμε τρεις εξόδους, μία για κάθε πιθανό αποτέλεσμα της σύγκρισης των αριθμών, να είναι δηλαδή ο ένας αριθμός μεγαλύτερος, ίσος ή μικρότερος από τον άλλον. Αυτό ονομάζεται και κώδικας 1-από-3.

Πλεονεκτήματα και περιορισμοί των χαρτών Karnaugh

Τα κυκλώματά μας είναι υλοποιήσεις δύο επιπέδων, και είναι σχετικά εύκολο να τα σχεδιάσουμε και να τα παρακολουθήσουμε. Η προηγούμενη συνάρτηση $E(A1, A0, B1, B0)$ δεν μπορεί να απλοποιηθεί περαιτέρω με το χάρτη Karnaugh. Μπορούσαμε άραγε να καταφέρουμε κάτι καλύτερο, αν χρησιμοποιήσουμε την άλγεβρα Boole;

Επεκτασιμότητα

Χρησιμοποιήσαμε μια δυναμική προσέγγιση, καταγράφοντας όλες τις πιθανές εισόδους και εξόδους. Αυτό κάνει πιο δύσκολη την επέκταση του κυκλώματός μας για σύγκριση αριθμών με 3-bit, για παράδειγμα.

4.13. Κυκλώματα μετατροπής κωδίκων

Έστω ότι θέλουμε να μετατρέψουμε κώδικα BCD 8421 σε κώδικα BCD Excess 3 (XS3). (πίνακας 4.19) Για τη μετατροπή, εφαρμόζουμε τις αρχές που έχουμε αναφέρει. Αρχικά σχηματίζουμε τον πίνακα αλήθειας των προς μετατροπή κωδικών και, στη συνέχεια, απλοποιούμε με τους χάρτες Karnaugh. Τέλος σχεδιάζουμε το αντίστοιχο λογικό κύκλωμα. (Nashelsky, 1994; Κοσσιδάς, 1996)

A	B	C	D	X	Y	W	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

Πίνακας 4.19. Μετατροπή του BCD 8421 σε BCD XS3

Οι χάρτες Karnaugh και οι εξισώσεις που προκύπτουν είναι:

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

$$X = A + B \cdot D + B \cdot C$$

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

$$Y = \bar{B} \cdot C + \bar{B} \cdot D + B \cdot \bar{C} \cdot \bar{D}$$

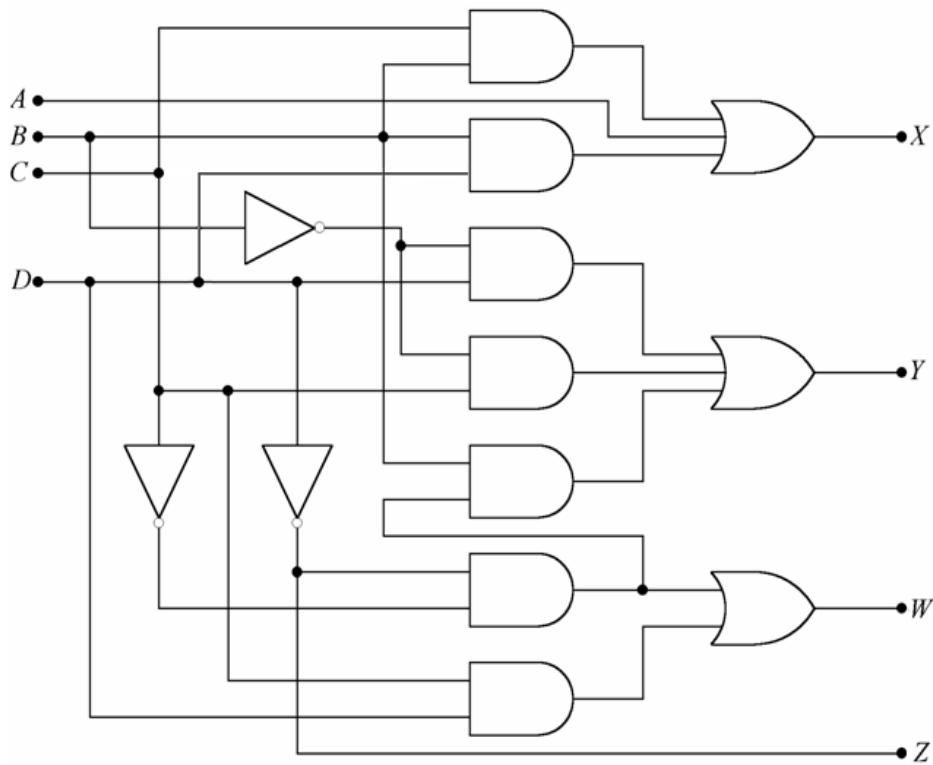
AB \ CD	00	01	11	10
00	1	1	X	1
01	0	0	X	0
11	1	1	X	X
10	0	0	X	X

$$W = \bar{C} \cdot \bar{D} + C \cdot D$$

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

$$Z = \bar{D}$$

Από τους παραπάνω χάρτες προκύπτει το κύκλωμα του σχήματος 4.40.



Σχήμα 4.40. Κύκλωμα μετατροπής του BCD 8421 σε BCD XS3

4.14. Έλεγχος των τιμών δυαδικών ψηφίων

A) Χειρισμός δυαδικών ψηφίων με την πύλη «OR»

Αλλαγή ενός bit (έστω του 4ου) από 0 σε «1». (Κοσσίδης, & Γιαννακόπουλος, 2006)

	0	1	1	0	1	1	1	0
OR	0	0	0	1	0	0	0	0
	0	1	1	1	1	1	1	0

Εκτελούμε την πράξη OR με λέξη που περιέχει στη θέση αυτή «1». Έτσι είμαστε βέβαιοι για την έξοδο του συγκεκριμένου ψηφίου.

B) Χειρισμός δυαδικών ψηφίων με την πύλη «AND»

Αλλαγή ενός bit (έστω του 4ου) από 1 σε «0»

	0	1	1	1	1	1	1	0
AND	1	1	1	0	1	1	1	1
	0	1	1	0	1	1	1	0

Εκτελούμε την πράξη AND με λέξη που περιέχει στη θέση αυτή «0». Έτσι, αντίστοιχα, ήμαστε βέβαιοι για την έξοδο του συγκεκριμένου ψηφίου.

Γ) Χειρισμός των bit με την πύλη «XOR»

Συμπλήρωμα (αντιστροφή) των bits (4 και 0) (μετρώ από δεξιά προς τα αριστερά)

	0	1	1	1	1	1	1	0
XOR	0	0	0	1	0	0	0	1
	0	1	1	0	1	1	1	1

Για να «συμπληρώσουμε» κάποιο δυαδικό ψηφίο, εκτελούμε την πράξη XOR με λέξη που περιέχει στις θέσεις που μας ενδιαφέρει «1». Έτσι είμαστε βέβαιοι ότι το τελικό αποτέλεσμα θα είναι το συμπληρωματικό του αρχικού, για τα συγκεκριμένα δυαδικά ψηφία.

Δ) Έλεγχος ισοδύναμων bytes με την πύλη «XOR»

	0	1	1	0	1	0	1	0
XOR	0	1	1	0	1	0	1	0
	0	0	0	0	0	0	0	0

Για να ελέγξουμε αν κάποια λέξη είναι η ζητούμενη, εκτελούμε την πράξη XOR με την ίδια τη λέξη, που είναι η ζητούμενη. Σε αυτή την περίπτωση, αν η λέξη είναι σωστή, το τελικό αποτέλεσμα θα είναι μηδέν.

Ε) Έλεγχος για μηδενικό Byte

Όμοια, για να ελέγξουμε αν κάποια λέξη είναι μηδενική, εκτελούμε την πράξη OR με το μηδέν. Σε αυτή την περίπτωση το τελικό αποτέλεσμα πρέπει να είναι μηδέν, αν η αρχική λέξη είναι επίσης μηδέν.

	0	0	0	0	0	0	0	0
OR	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0

4.15. Ασκήσεις – Ερωτήσεις

ΕΡΩΤΗΣΕΙΣ ΠΟΛΛΑΠΛΗΣ ΕΠΙΛΟΓΗΣ

- Για την υλοποίηση της έκφρασης $X = ABC\bar{D} + \bar{A}\bar{B}CD + ABC\bar{D}$ χρειάζεται μια πύλη OR και

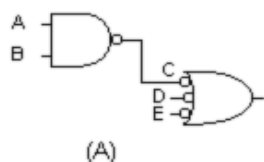
A. τρεις πύλες AND και τρεις inverter	Γ. τρεις πύλες AND
B. τρεις πύλες AND και τέσσερις inverter	Δ. μια πύλη AND
- Ένα κύκλωμα(AND-OR) 4 μεταβλητών παράγει στην έξοδο $Y=1$. Ποιος συνδυασμός εισόδων είναι ο σωστός;

A. $A = 0, B = 0, C = 0, D = 0$	B. $A = 0, B = 1, C = 1, D = 0$
Γ. $A = 1, B = 1, C = 0, D = 0$	Δ. $A = 1, B = 0, C = 0, D = 0$
- Πόσες πύλες AND απαιτούνται για την εφαρμογή της Boolean εξίσωσης $X = ABC\bar{C} + \bar{A}\bar{B}C + \bar{A}BC$;

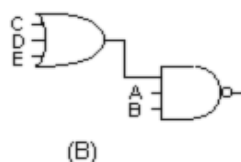
A. 1	B. 2	Γ. 3	Δ. 4
------	------	------	------
- Πόσες πύλες NOT απαιτούνται για την εφαρμογή της Boolean εξίσωσης $X = \bar{A}\bar{B}C + \bar{A}BC$;

A. 1	B. 2	Γ. 4	Δ. 5
------	------	------	------
- Με πόσες πύλες NAND μπορεί να παραχθεί ένας inverter;

A. 1	B. 2	Γ. 3	Δ. 4
------	------	------	------
- Υλοποιώντας την έκφραση $X = \overline{(A + B)(CDE)}$ και χρησιμοποιώντας λογική NAND, έχουμε:

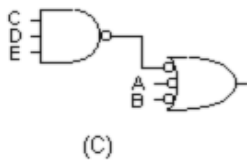


(A)

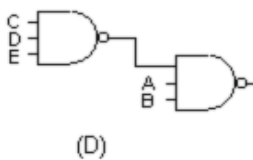


(B)

- A. (A)
B. (B)
Γ. (C)
Δ. (D)



(C)



(D)

- Μια πιθανή έξοδος για ένα κύκλωμα AND-OR-Invert έχοντας μία πύλη AND με εισόδους A, B και C και μία πύλη AND με εισόδους D και E είναι:

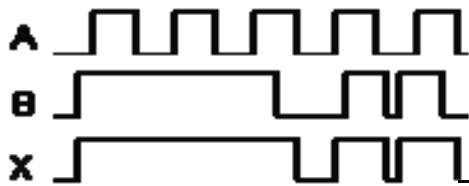
A. $X = \overline{ABC + DE}$	B. $X = \overline{\bar{A} + \bar{B} + \bar{C} + \bar{D} + \bar{E}}$
Γ. $X = \overline{(A + B + C)(D + E)}$	Δ. $X = (A + B + C)(D + E)$
- Η εφαρμογή της Boolean εξίσωσης $X = ABC + AB + \bar{A}C$ καταλήγει σε:

A. τρεις πύλες AND, μια πύλη OR
B. τρεις πύλες AND, μια πύλη NOT, μια πύλη OR
Γ. τρεις πύλες AND, μια πύλη NOT, τρεις πύλες OR
Δ. τρεις πύλες AND, τρεις πύλες OR
- Πόσες πύλες NOR δυο εισόδων χρειάζονται, για να παραχθεί μια πύλη NAND δυο εισόδων;

A. 1	B. 2	Γ. 3	Δ. 4
------	------	------	------
- Πόσες πύλες XNOR απαιτούνται για κύκλωμα συγκριτή των 8-bit;

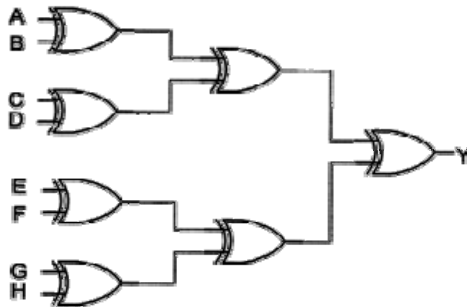
A. 4	B. 6	Γ. 8	Δ. 0
------	------	------	------
- Υλοποιώντας την έκφραση $X = \bar{A} + \bar{B} + CD$ χρησιμοποιώντας λογική NAND, έχουμε:

19. Οι ακόλουθες κυματομορφές αναπαριστούν μια:



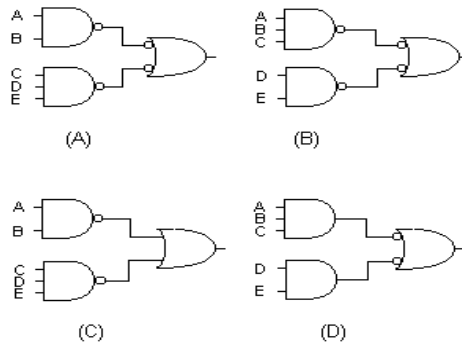
- A. πύλη AND δυο εισόδων
- B. πύλη OR δυο εισόδων
- Γ. πύλη XOR
- Δ. τίποτα από τα παραπάνω

20. Τα παρακάτω κύκλωμα αποτελείται από πύλες XOR 8 εισόδων και έχει έξοδο $Y=1$. Ποιος συνδυασμός εισόδων (με σειρά A-H) είναι σωστός;



- A. 10111100
- B. 10111000
- Γ. 11100111
- Δ. 00011101

21. Υλοποιώντας τη συνάρτηση $AB + CDE$ με τη χρήση λογικής NAND, έχουμε:



- A. (A)
- B. (B)
- Γ. (C)
- Δ. (D)

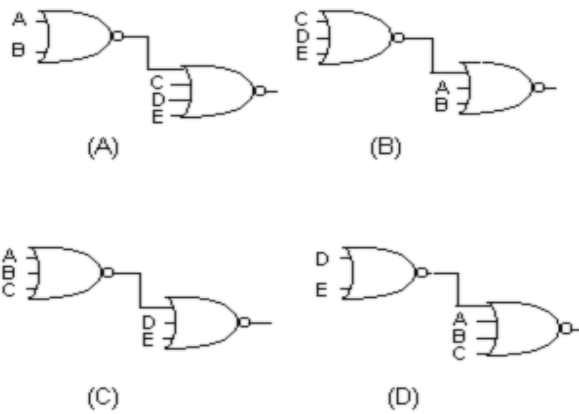
22. Ένας θετικός παλμός με $t_w=75\mu s$ εφαρμόζεται σε μία από τις εισόδους ενός κυκλώματος XOR. Ένας δεύτερος θετικός παλμός $t_w=15\mu s$ εφαρμόζεται στην άλλη είσοδο μετά από $20\mu s$ μετά τη προπορευόμενη ακμή του πρώτου παλμού. Ποια δήλωση περιγράφει την έξοδο σε σχέση με τις εισόδους;

- A. Η έξοδος της XOR είναι ένας παλμός $20\mu s$ ακολουθούμενος από έναν παλμό $40\mu s$, με ένα διαχωρισμό $15\mu s$ μεταξύ των παλμών.
- B. Η έξοδος της XOR είναι ένας παλμός $20\mu s$ ακολουθούμενος από έναν παλμό $15\mu s$, με ένα διαχωρισμό $40\mu s$ μεταξύ των παλμών.
- Γ. Η έξοδος της XOR είναι ένας παλμός $15\mu s$, ακολουθούμενος από έναν παλμό $40\mu s$
- Δ. Η έξοδος της XOR είναι ένας παλμός $20\mu s$ ακολουθούμενος από έναν παλμό $15\mu s$, ο οποίος ακολουθείται από έναν παλμό $40\mu s$.

23. Οι δυαδικοί αριθμοί $A = 1100$ και $B = 1001$ εφαρμόζονται στις εισόδους ενός συγκριτή. Ποια τα επίπεδα εξόδου (G, L, E);

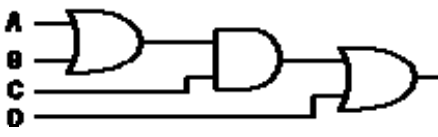
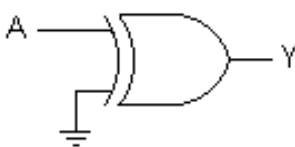
- A. $A > B = 1, A < B = 0, A = B = 1$
- B. $A > B = 0, A < B = 1, A = B = 0$
- Γ. $A > B = 1, A < B = 0, A = B = 0$
- Δ. $A > B = 0, A < B = 1, A = B = 1$

24. Υλοποιώντας την έκφραση $X = \overline{AB} + (C + D + E)$ με λογική NOR, έχουμε:



- A. (A)
- B. (B)
- Γ. (C)
- Δ. (D)

25. Να προσδιορίσετε τις τιμές των A, B, C και D που ικανοποιούν την σχέση: $X = \bar{A} + B + \bar{C} + D = 0$.
- | | |
|--------------------|--------------------|
| A. A=1 B=0 C=0 D=0 | B. A=1 B=0 C=1 D=0 |
| Γ. A=0 B=1 C=0 D=0 | Δ. A=1 B=0 C=1 D=1 |
26. Ένας SIPO 4-bit καταχωρητής ολίσθησης αρχικά περιέχει άσους. Το 0111 είναι σε αναμονή για να εισέλθει. Μετά από τέσσερις παλμούς ρολογιού περιέχει
- | | | | |
|---------|---------|---------|---------|
| A. 0000 | B. 1111 | Γ. 0111 | Δ. 1000 |
|---------|---------|---------|---------|
27. Με συχνότητα 200 KHz, οκτώ bits μπορούν να εγγραφούν σειριακά σε έναν καταχωρητή ολίσθησης σε:
- | | | | |
|--------------|---------------|----------------|----------|
| A. 4 μ s | B. 40 μ s | Γ. 400 μ s | Δ. 40 ms |
|--------------|---------------|----------------|----------|
28. Ένας πλήρης αθροιστής έχει $C_{in} = 0$. Ποιο είναι το άθροισμα sum (Σ) και το κρατούμενο carry (C_{out}) όταν το A = 1 και το B = 1;
- | | |
|------------------------------|------------------------------|
| A. $\Sigma = 0, C_{out} = 0$ | B. $\Sigma = 0, C_{out} = 1$ |
| Γ. $\Sigma = 1, C_{out} = 0$ | Δ. $\Sigma = 1, C_{out} = 1$ |
29. Εάν ένα σήμα διέρχεται μέσω μιας πύλης αναστέλλεται από την αποστολή ενός λογικού '0' σε μια από τις εισόδους της, και η έξοδος είναι λογικό '1' τότε η πύλη είναι:
- | | | | |
|--------|---------|--------|-------|
| A. AND | B. NAND | Γ. NOR | Δ. OR |
|--------|---------|--------|-------|
30. Η έξοδος μιας NOR πύλης είναι λογικό '1' όταν:
- A. Όλοι οι εισοδοί της είναι ίσες με λογικό '1'
 - B. Κάποιες από τις εισόδους της είναι ίσες με λογικό '1'
 - Γ. Κάποιες από τις εισόδους της είναι ίσες με λογικό '0'
 - Δ. Όλοι οι εισοδοί της είναι ίσες με λογικό '0'
31. Αν μια πύλη AND τριών εισόδων έχει 8 πιθανούς συνδυασμούς για τα δεδομένα στις εισόδους της, πόσοι από αυτούς τους συνδυασμούς έχουν σαν αποτέλεσμα στην έξοδο της πύλης το λογικό '1';
- | | | | |
|------|------|------|------|
| A. 1 | B. 2 | Γ. 7 | Δ. 8 |
|------|------|------|------|
32. Η λογική έκφραση μιας πύλης AND τριών εισόδων είναι:
- | | | | |
|-------------|--------------|----------------|---------------|
| A. $X = AB$ | B. $X = ABC$ | Γ. $X = A+B+C$ | Δ. $X = AB+C$ |
|-------------|--------------|----------------|---------------|
33. Οι πύλες NAND και NOR αναφέρονται ως «καθολικές» πύλες, επειδή και οι δύο:
- A. Μπορούν να βρεθούν σε όλα τα ψηφιακά κυκλώματα
 - B. Μπορούν να χρησιμοποιηθούν, ώστε να κατασκευαστούν όλοι οι άλλοι τύποι πυλών
 - Γ. Χρησιμοποιούνται σε όλες τις χώρες του κόσμου
 - Δ. Ήταν οι πρώτες πύλες που υλοποιήθηκαν
34. Μια πύλη NAND με άρνηση στις εισόδους της υλοποιεί την ίδια λειτουργία με μια _____ πύλη.
- | | | | |
|--------|-------|--------|---------|
| A. NOT | B. OR | Γ. NOR | Δ. NAND |
|--------|-------|--------|---------|
35. Μια πύλη NOR 2 εισόδων είναι ισοδύναμη με
- | | |
|-----------------------|---------------------------|
| A. Αρνητική πύλη OR | B. Αρνητική πύλη AND |
| Γ. Αρνητική πύλη NAND | Δ. Τίποτα από τα παραπάνω |

36. Ο όρος «fan-out» αναφέρεται
 Α. στην τάση
 Β. στο ρεύμα
 Γ. στα watts
 Δ. στις μονάδες φορτίου
37. Η λογική έκφραση μιας πύλης NOR είναι:
 Α. $X = A + B$
 Β. $X = \overline{A + B}$
 Γ. $X = A + B$
 Δ. $X = \overline{A + B}$
38. Δεδομένης μιας πύλης AND, ποιό από τα παρακάτω είναι αληθές;
 Α. Μια πύλη AND έχει 2 εισόδους και 1 έξοδο
 Β. Μια πύλη AND έχει 2 ή παραπάνω εισόδους και 2 εξόδους
 Γ. Εάν μια είσοδος μιας πύλης AND 2 εισόδων είναι λογικό «1», τότε η έξοδος αντιστοιχεί στην τιμή της άλλης εισόδου
 Δ. Σε μια πύλη AND 2 εισόδων υπάρχουν 8 πιθανοί συνδυασμοί στις εισόδους της
39. Να προσδιορίσετε την Boolean έκφραση για το λογικό κύκλωμα που ακολουθεί:

 Α. $X = CB + AC + CD$
 Β. $X = CB + AC + D$
 Γ. $X = C(B + A) + D$
 Δ. $X = C(B + A)\overline{D}$
40. Η έξοδος μιας πύλης AND με εισόδους A, B, C είναι το λογικό «1» όταν:
 Α. A=1, B=1, C=0
 Β. A=0, B=0, C=0
 Γ. A=1, B=1, C=1
 Δ. A=1, B=0, C=1
41. Η έξοδος του κυκλώματος που ακολουθεί θα είναι πάντα

 Α. 1
 Β. 0
 Γ. A
 Δ. \overline{A}
42. Η έξοδος μιας πύλης OR με εισόδους τα A, B, C είναι το λογικό «0» όταν:
 Α. A=0, B=0, C=0
 Β. A=0, B=1, C=1
 Γ. A=0, B=0, C=1
 Δ. Όλα τα παραπάνω
43. Εάν ένας 8-bit κυκλικός απαριθμητής έχει σαν αρχική κατάσταση την 10111110. Ποιά είναι η κατάσταση μετά τον τέταρτο παλμό ρολογιού;
 Α. 11101011 Β. 00010111 Γ. 11110000 Δ. 00000000
44. Ένας 4-bit καταχωρητής ολίσθησης που δέχεται 4 bit δεδομένων παράλληλα θα μετατοπιστεί προς τα _____ κατά _____ θέσεις για κάθε παλμό ρολογιού.
 Α. δεξιά, μία Β. δεξιά, δύο
45. Η έξοδος μιας XNOR πύλης είναι λογικό «1» όταν
 Α. οι εισοδοί είναι ίσες
 Β. μία είσοδος είναι λογικό '1' και η άλλη είσοδος είναι λογικό «0»
 Γ. οι εισοδοί είναι άνισοι
 Δ. τίποτα από τα παραπάνω
46. Ποιά από τις παρακάτω εξισώσεις περιγράφει καλύτερα τη λειτουργία μιας πύλης OR 4 εισόδων, όπου A=1, B=1, C=0, D=0
 Α. $1 + 1 + 0 + 0 = 01$
 Β. $1 + 1 + 0 + 0 = 1$
 Γ. $1 + 1 + 0 + 0 = 0$
 Δ. $1 + 1 + 0 + 0 = 00$

Αναφορές-Βιβλιογραφία

- Κοσσίδης, Α.Θ. (1996). *Σχεδίαση Ψηφιακών Κυκλωμάτων*, Εκδόσεις Μπένος
- Κοσσίδης, Α.Θ., Γιαννακόπουλος, Π. (2006). *Αριθμητικά Συστήματα και Ψηφιακά Κυκλώματα*, Εκδόσεις Νέων Τεχνολογιών, Αθήνα
- Balabanian, N., Carlson, B. (2007). *Digital Logic Design Principles*, John Wiley
- Balch, M. (2003). *Complete Digital Design*, Mc Graw Hill
- Butzen, P.F., Dal Bem, V., Reis, A. I., Ribas, R. P. (2010). *Transistor network restructuring against NBTI degradation*, *Microelectronics Reliability* 50(9):1298-1303. DOI: 10.1016/j.microrel.2010.07.140
- Floyd, Thomas L., (2006). *Digital Fundamentals*, 9th, Pearson International Edition
- Givone, D. (2002). *Digital Principles and Design*,. Mc Graw Hill
- Godse, A.P., Godse, D.A. (2010). *Digital Logic Design and Application*, Technical Publications Pune
- Hendrich, Norman (2006). *HADES Tutorial*, University of Hamburg, v.0.92.
- Holdsworth, Brian, Woods, Clive (2002). *Digital Logic Design*, 4th Edition, Newnes
- Katz, R. (2005). *Contemporary Logic Design*, 2/e, Prentice Hall
- Mano, M., Ciletti, M. (2014). *Ψηφιακή Σχεδίαση*, 5^η έκδοση, Παπασωτηρίου
- Maxfield, Clive, (2008). *Bebop to the Boolean Boogie, An Unconventional Guide to Electronics*, 3rd , Newnes
- Nashelsky, Louis (1994). *Introduction to Digital Technology*, 4th Ed., Prentice Hall
- Nelson, V., Nagle, H., Carroll, B., Irwin, J. (1995). *Digital Logic Circuit Analysis and Design*, Prentice-Hall
- Predko, Myke (2005). *Digital Electronics Demystified*, Mc Graw Hill
- Pritchard, N. (2015). *Fundamentals of Digital Electronics*, CreateSpace Independent Publishing Platform
- Roth, Charles, (2006). *Fundamentals of Logic Design*, Thomson-Nelson
- Tocci, R. J., Widmer, N. S, and Moss, Gr. L. (2010). *Digital Systems: Principles and Applications*, 11th, Boston, Addison-Wesley
- Wakerly, J. (2006). *Digital Design Principles and Practices*, 4/e, Prentice Hall