Κεφάλαιο 2

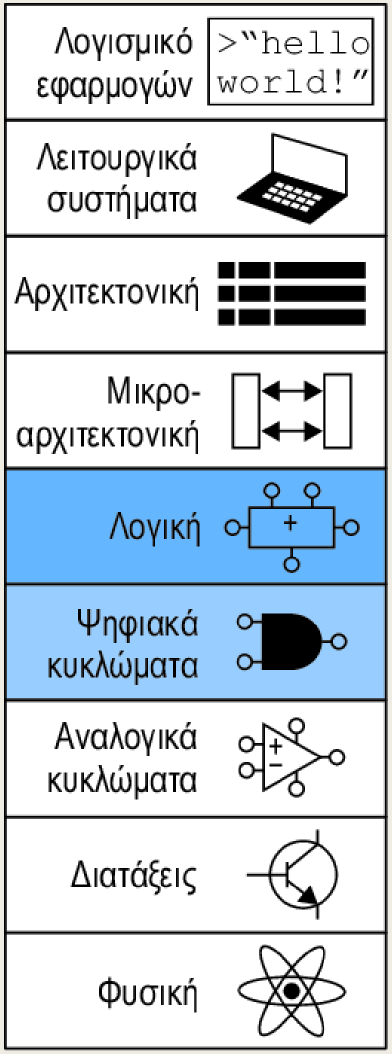
Σχεδίαση συνδυαστικής λογικής

Γιώργος Παπαδημητρίου, Αντώνης Πασχάλης,
Βασίλης Καρακώστας

DIGITAL SYSTEMS & COMPUTER ARCHITECTURE LABORATORY

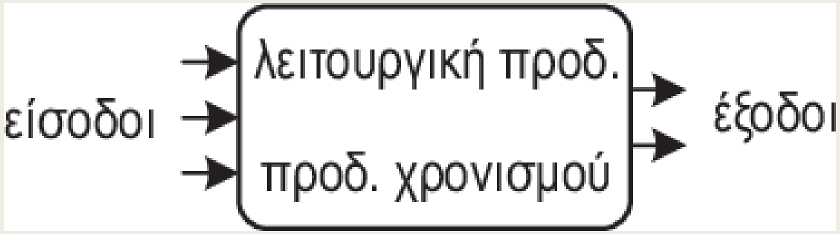
Περιεχόμενα κεφαλαίου 2

- Ψηφιακά κυκλώματα
- Εξισώσεις Boole
- Άλγεβρα Boole
- Σχηματικά διαγράμματα
- Πολυεπίπεδη συνδυαστική λογική
- Μη αποδεκτη τιμή X / Μετέωρη τιμή Z
- Χαρτες Karnaugh
- Κώδικες BCD και Gray
- Πολυπλέκτες / Αποκωδικοποιητές
- Χρονισμός συνδυαστικής λογικής



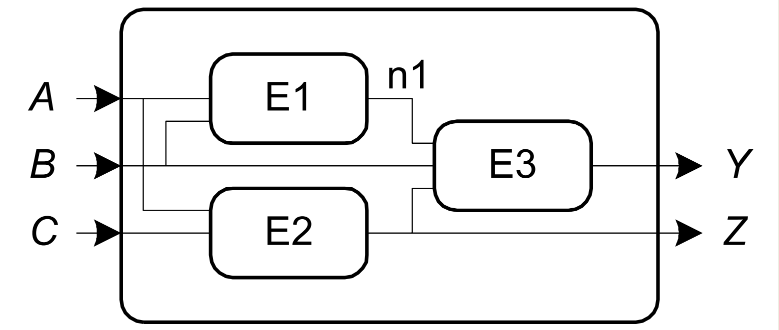
Ψηφιακά κυκλώματα

- Ένα **ψηφιακό κύκλωμα** είναι ένα δίκτυο στοιχείων που επεξεργάζεται μεταβλητές οι οποίες παίρνουν διακριτές τιμές
- Το κύκλωμα με βάση την αρχή της τμηματικότητας είναι ένα **μαύρο κουτί** με:
  - *ένα ή περισσότερα **τερματικά σημεία εισόδου** (input terminals) που παίρνουν διακριτές τιμές,*
  - *ένα ή περισσότερα **τερματικά σημεία εξόδου** (output terminals) που παίρνουν διακριτές τιμές,*
  - *μια **λειτουργική προδιαγραφή** (functional specification) που περιγράφει τη σχέση μεταξύ εισόδων και εξόδων,*
  - *μια **προδιαγραφή χρονισμού** (timing specification) που περιγράφει την καθυστέρηση με την οποία αποκρίνονται οι έξοδοι στη μεταβολή των εισόδων.*



Ψηφιακά κυκλώματα

- Ένα **ψηφιακό κύκλωμα** αποτελείται από **κόμβους** και **στοιχεία**
  - *Το **στοιχείο (element)** (E1, E2, E3) είναι και αυτό ψηφιακό κύκλωμα με εισόδους, εξόδους, και προδιαγραφή*
  - *Ο **κόμβος (node)** είναι ένα σύρμα, του οποίου η τάση μεταφέρει μια μεταβλητή που παίρνει διακριτές τιμές*
- Οι κόμβοι ταξινομούνται ως **κόμβοι εισόδου**, **κόμβοι εξόδου** ή **εσωτερικοί κόμβοι**
  - *Οι **κόμβοι εισόδου** (A, B, C) δέχονται τιμές από τον έξω κόσμο*
  - *Οι **κόμβοι εξόδου** (Y, Z) στέλνουν τιμές στον έξω κόσμο*
  - *Τα σύρματα που δεν είναι είσοδοι ή έξοδοι ονομάζονται **εσωτερικοί κόμβοι** (n1)*



Τύποι ψηφιακών κυκλωμάτων

- Τα ψηφιακά κυκλώματα ταξινομούνται σε **συνδυαστικά** (combinational) ή **ακολουθιακά** (sequential) κυκλώματα.
- Συνδυαστική Λογική
  - *Οι έξοδοι ενός συνδυαστικού κυκλώματος εξαρτώνται **μόνο από τις τρέχουσες τιμές** των εισόδων*
  - *Οι **λογικές πύλες** είναι τα πιο απλά συνδυαστικά κυκλώματα*
  - *Δεν έχουν **μνήμη** (ούτε καταστάσεις)*
- Ακολουθιακή Λογική
  - *Οι έξοδοι ενός ακολουθιακού κυκλώματος εξαρτώνται **και από τις τρέχουσες και από τις προηγούμενες τιμές** των εισόδων*
  - *Έχουν μνήμη (αποθηκεύουν καταστάσεις)*

Κανόνες συνδυαστικής σύνθεσης

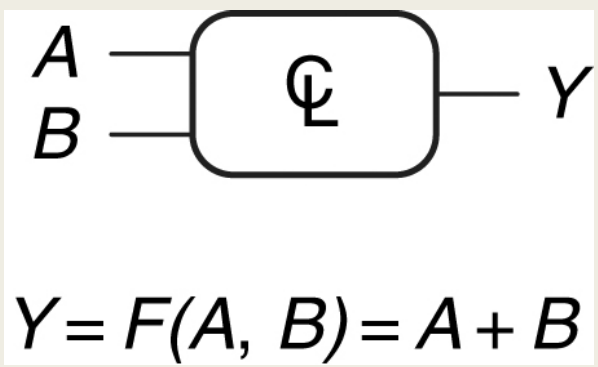
- Οι κανόνες της **συνδυαστικής σύνθεσης** (combinational composition) μας λένε πώς μπορούμε να κατασκευάσουμε ένα μεγάλο συνδυαστικό κύκλωμα από μικρότερα στοιχεία συνδυαστικών κυκλωμάτων (αρχή της ιεραρχίας)
  - *Το πιο μικρό στοιχείο είναι η λογική πύλη*
- Ένα ψηφιακό κύκλωμα είναι συνδυαστικό αν αποτελείται από αλληλοσυνδεόμενα στοιχεία κυκλωμάτων τέτοια ώστε:
  - *κάθε στοιχείο κυκλώματος είναι και το ίδιο συνδυαστικό·*
  - *κάθε κόμβος του κυκλώματος είτε είναι **είσοδος**, είτε συνδέεται σε **ακριβώς μία έξοδο**·*
  - *το κύκλωμα **δεν περιέχει κυκλικές διαδρομές**, δηλαδή κάθε διαδρομή μέσω των στοιχείων του κυκλώματος περνάει από κάθε κόμβο του το πολύ μία φορά.*

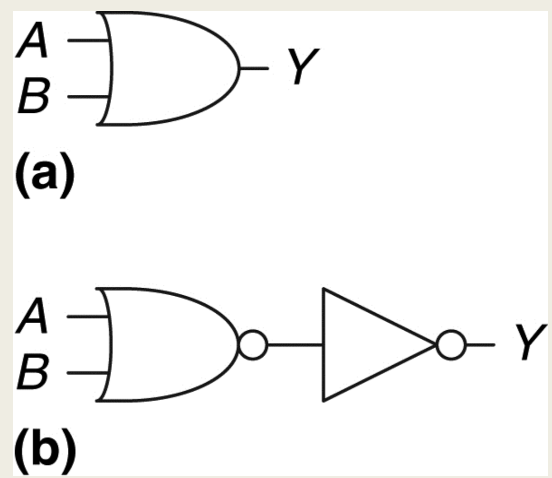
Κανόνες συνδυαστικής σύνθεσης

- Οι κανόνες της συνδυαστικής σύνθεσης μπορεί να είναι επαρκείς, αλλά δεν είναι αυστηρά υποχρεωτικοί
- Υπάρχουν ορισμένα κυκλώματα που, παρότι δεν υπακούν τους συγκεκριμένους κανόνες, εξακολουθούν να είναι συνδυαστικά
- Προϋπόθεση είναι όμως να ικανοποιούν τη συνθήκη ότι
  - ***οι έξοδοι εξαρτώνται μόνο από τις τρέχουσες τιμές των εισόδων***

Συνδυαστικά κυκλώματα

- Η **λειτουργική προδιαγραφή** ενός συνδυαστικού κυκλώματος εκφράζει **τις τιμές των εξόδων σε σχέση με τις τρέχουσες τιμές των εισόδων**
  - *Είναι ένας **πίνακας αλήθειας** ή μία **συνάρτηση (εξίσωση) Boole***
  - *Για παράδειγμα, η συνάρτηση F της **πύλης OR**: $Y = F(A, B) = A + B$*
- Μία συνάρτηση F μπορεί να έχει **πολλές διαφορετικές υλοποιήσεις**
  - *Για παράδειγμα, η συνάρτηση F της πύλης OR μπορεί να υλοποιηθεί*
    - (α) με την χρήση μιας πύλης OR
    - (β) με την χρήση μιας πύλης NOR και μίας πύλης NOT



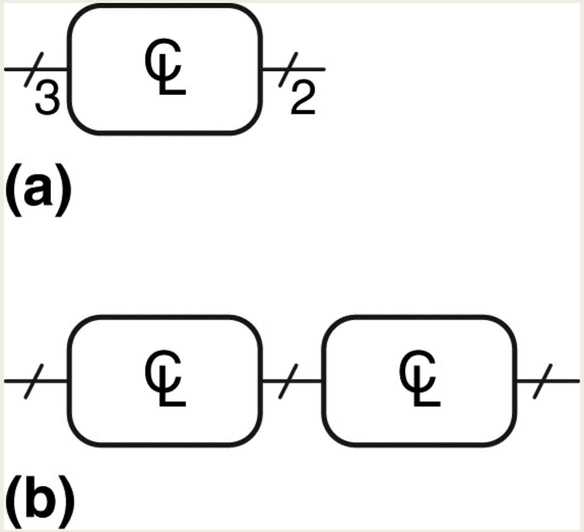


Συνδυαστικά κυκλώματα

- Ανάμεσα στις πολλές διαφορετικές υλοποιήσεις μίας συνάρτησης F, εμείς επιλέγουμε ποια υλοποίηση θα χρησιμοποιήσουμε ανάλογα με:
  - *Τα δομικά στοιχεία που έχουμε στη διάθεσή μας*
  - *Τους σχεδιαστικούς περιορισμούς που υπάρχουν, όπως:*
    - Επιφάνεια υλοποίησης
    - Ταχύτητα
    - Κατανάλωση ισχύος
    - Χρόνος σχεδίασης

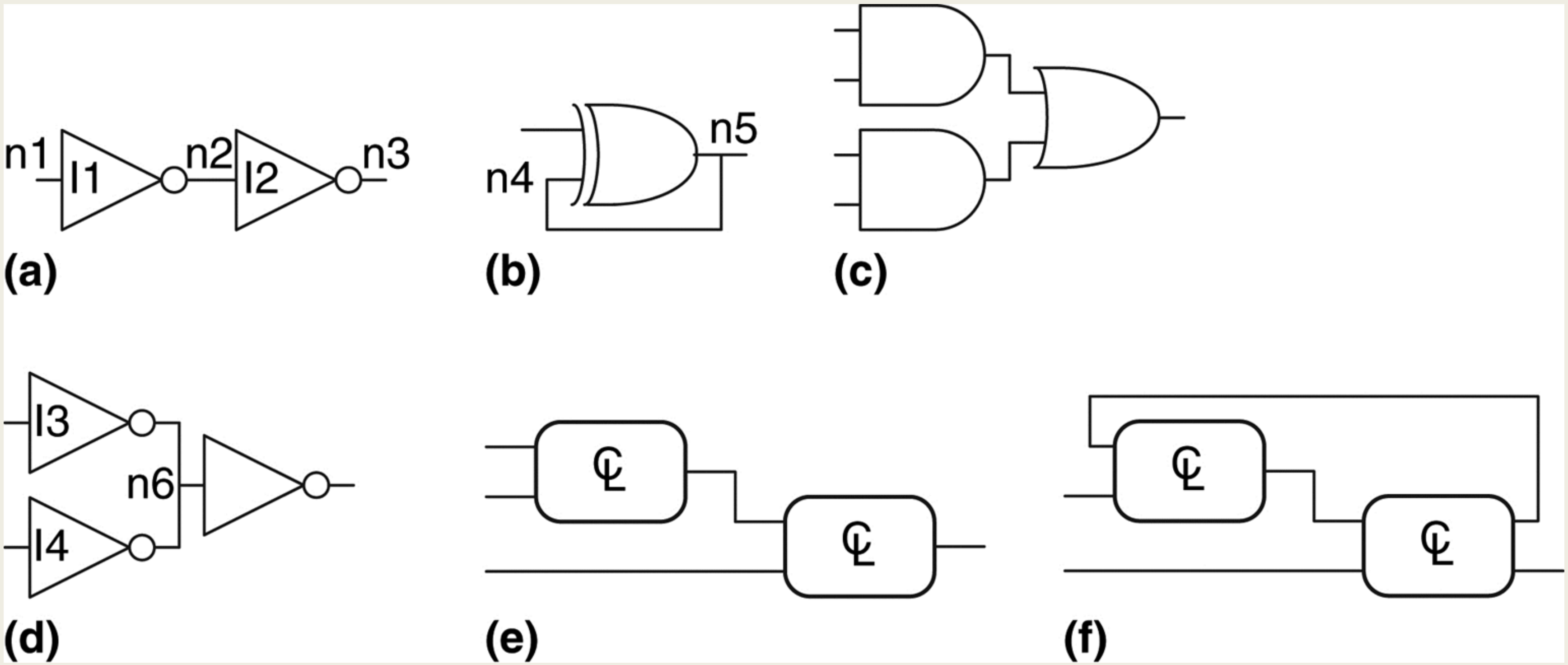
Δίαυλος (Bus)

- Σε μια προσπάθεια να απλοποιήσουμε τα διαγράμματα, συχνά χρησιμοποιούμε μία γραμμή με μια κάθετο (/) που την τέμνει, καθώς και έναν αριθμό δίπλα από αυτή για να συμβολίσουμε έναν **δίαυλο** (bus), δηλαδή μια δέσμη πολλών σημάτων
  - *Ο αριθμός καθορίζει το **πλήθος των σημάτων (των bit)** στον δίαυλο*
    - Αν το πλήθος των bit δεν παίζει ρόλο ή δεν είναι προφανές από τα συμφραζόμενα, η κάθετος ενδέχεται να μην συνοδεύεται από αριθμό



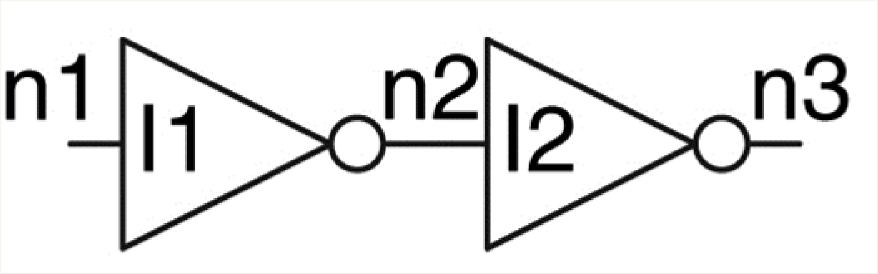
Παράδειγμα 2.1

- Ποια από τα παρακάτω κυκλώματα είναι συνδυαστικά?



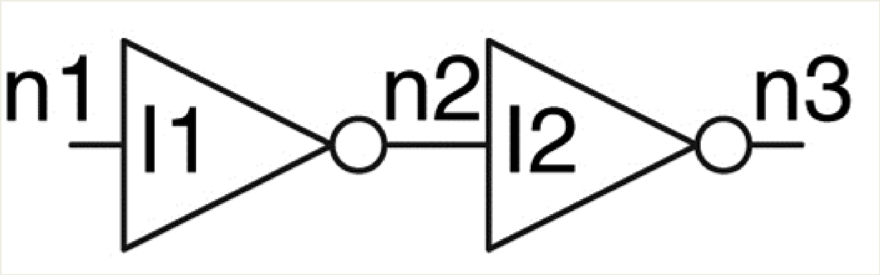
Παράδειγμα 2.1 (συνέχεια)

- Το κύκλωμα (α)



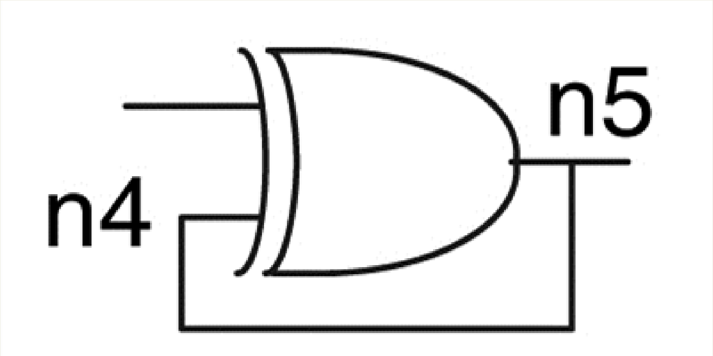
Παράδειγμα 2.1 (συνέχεια)

- Το κύκλωμα (α) **είναι συνδυαστικό**
  - *Έχει κατασκευαστεί από δύο στοιχεία συνδυαστικών κυκλωμάτων (αντιστροφείς I1 και I2)*
  - *Διαθέτει τρεις κόμβους: n1, n2 και n3*
  - *Ο κόμβος n1 είναι είσοδος του κυκλώματος και του αντιστροφέα I1*
  - *Ο κόμβος n2 είναι εσωτερικός κόμβος, ο οποίος αποτελεί την έξοδο του I1 και την είσοδο του I2*
  - *Ο κόμβος n3 είναι η έξοδος του κυκλώματος και του αντιστροφέα I2*



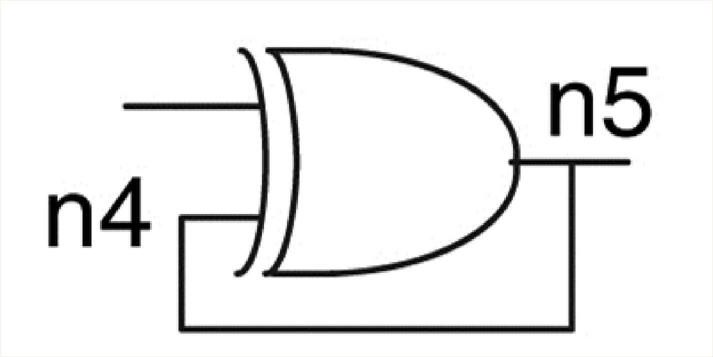
Παράδειγμα 2.1 (συνέχεια)

- Το κύκλωμα (β)



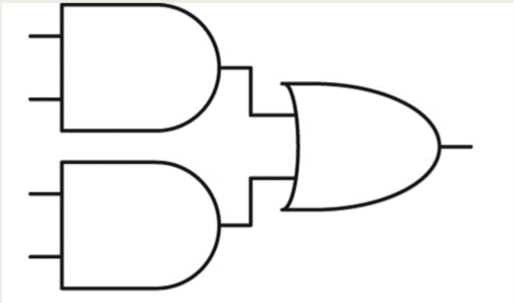
Παράδειγμα 2.1 (συνέχεια)

- Το κύκλωμα (β) **δεν είναι συνδυαστικό**
  - *Υπάρχει κυκλική διαδρομή: Η έξοδος της πύλης XOR ανατροφοδοτεί μια από τις εισόδους της*
  - *Επομένως, υπάρχει μια κυκλική διαδρομή που ξεκινάει από τον κόμβο n4, διέρχεται από την πύλη XOR και καταλήγει στον κόμβο n5, ο οποίος επιστρέφει στον κόμβο n4.*



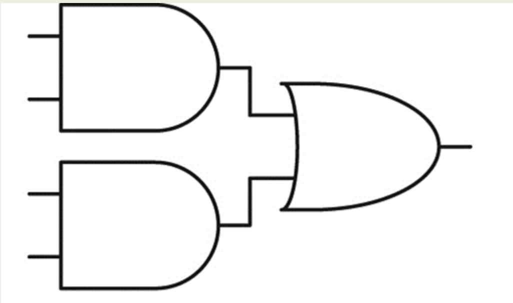
Παράδειγμα 2.1 (συνέχεια)

- Το κύκλωμα (γ)



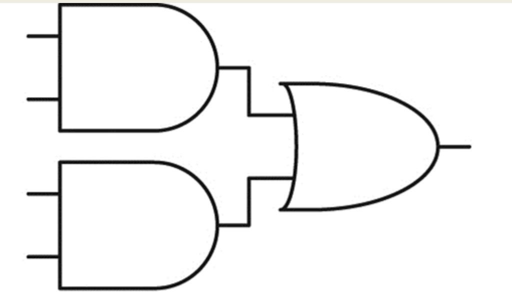
Παράδειγμα 2.1 (συνέχεια)

- Το κύκλωμα (γ) **είναι συνδυαστικό**

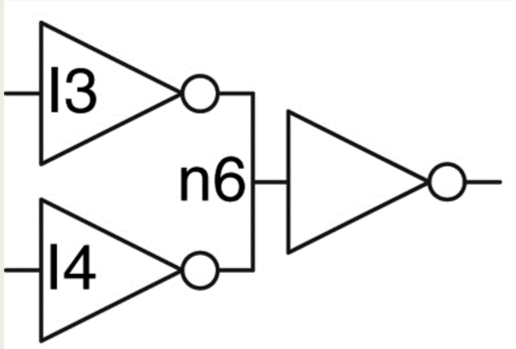


Παράδειγμα 2.1 (συνέχεια)

- Το κύκλωμα (γ) **είναι συνδυαστικό**

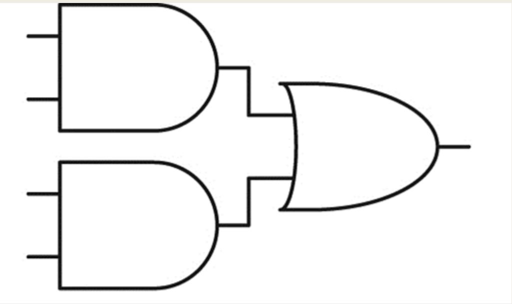


- Το κύκλωμα (δ)

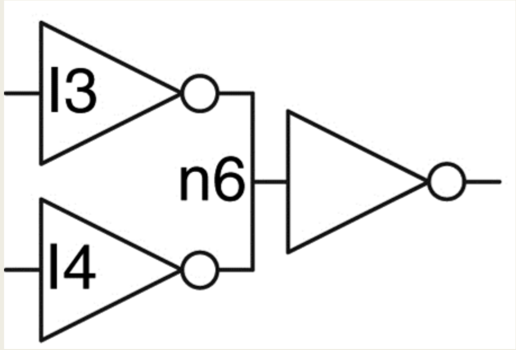


Παράδειγμα 2.1 (συνέχεια)

- Το κύκλωμα (γ) **είναι συνδυαστικό**

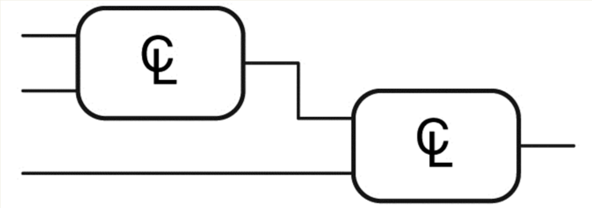


- Το κύκλωμα (δ) **δεν είναι συνδυαστικό**
  - *Ο κόμβος n6 συνδέεται στις εξόδους* ***και των δύο*** *αντιστροφέων I3 και I4*



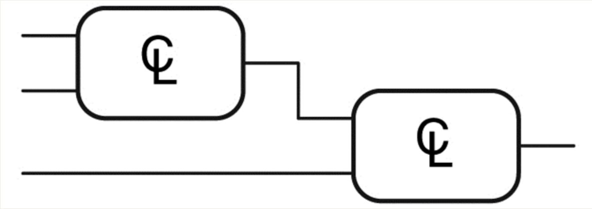
Παράδειγμα 2.1 (συνέχεια)

- Το κύκλωμα (ε)



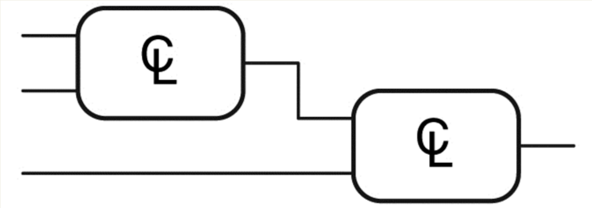
Παράδειγμα 2.1 (συνέχεια)

- Το κύκλωμα (ε) **είναι συνδυαστικό**
  - *αποτελεί παράδειγμα του πώς δύο συνδυαστικά κυκλώματα συνδέονται μεταξύ τους για να σχηματίσουν ένα μεγαλύτερο συνδυαστικό κύκλωμα*

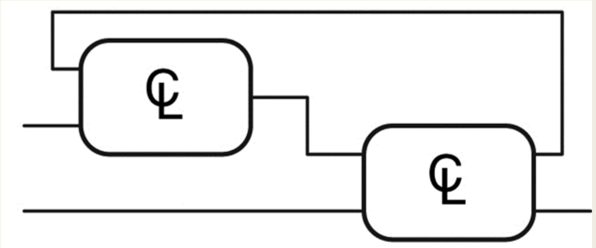


Παράδειγμα 2.1 (συνέχεια)

- Το κύκλωμα (ε) **είναι συνδυαστικό**
  - *αποτελεί παράδειγμα του πώς δύο συνδυαστικά κυκλώματα συνδέονται μεταξύ τους για να σχηματίσουν ένα μεγαλύτερο συνδυαστικό κύκλωμα*

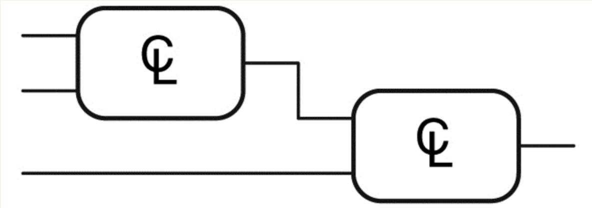


- Το κύκλωμα (στ)

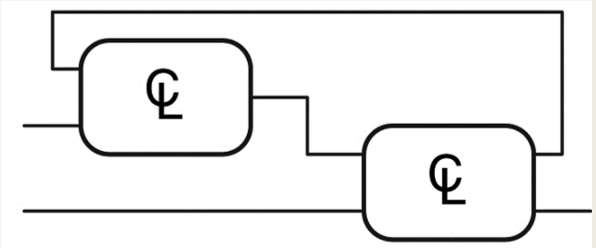


Παράδειγμα 2.1 (συνέχεια)

- Το κύκλωμα (ε) **είναι συνδυαστικό**
  - *αποτελεί παράδειγμα του πώς δύο συνδυαστικά κυκλώματα συνδέονται μεταξύ τους για να σχηματίσουν ένα μεγαλύτερο συνδυαστικό κύκλωμα*



- Το κύκλωμα (στ) **μπορεί να είναι ή να μην είναι συνδυαστικό**
  - *Περιέχει μια κυκλική διαδρομή που διέρχεται από τα δύο στοιχεία*
  - *Ανάλογα τις συναρτήσεις των δομικών στοιχείων, μπορεί να είναι ή να μην είναι συνδυαστικό κύκλωμα*



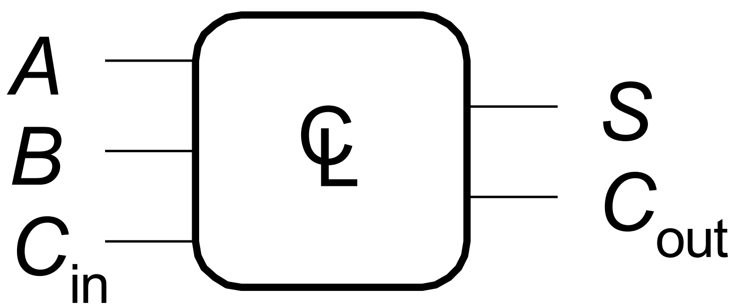
Περίπλοκα κυκλώματα

- Μεγάλα κυκλώματα, όπως οι μικροεπεξεργαστές, ενδέχεται να είναι ιδιαίτερα περίπλοκα
  - *σε τέτοιες περιπτώσεις χρησιμοποιούμε τις αρχές διαχείρισης της πολυπλοκότητας*
- Η θεώρηση ενός κυκλώματος ως μαύρου κουτιού με καλώς ορισμένη διασύνδεση και λειτουργία αποτελεί εφαρμογή της **«αφαίρεσης»** και της **τμηματικότητας**
- Η κατασκευή του κυκλώματος με χρήση μικρότερων στοιχείων κυκλωμάτων αποτελεί εφαρμογή της **ιεραρχίας**.
- Οι κανόνες της συνδυαστικής σύνθεσης συνιστούν εφαρμογή της **πειθαρχίας**

Εξισώσεις Boole

- Η **λειτουργική προδιαγραφή** ενός συνδυαστικού κυκλώματος συνήθως εκφράζεται με τη μορφή ενός **πίνακα αληθείας** ή μιας **εξίσωσης Boole**
- Οι **εξισώσεις Boole** περιέχουν **μεταβλητές** που παίρνουν είτε την τιμή **‘1’ (TRUE)** είτε την τιμή **‘0’ (FALSE)**, άρα είναι ιδανικές για την περιγραφή της ψηφιακής λογικής
- **Παράδειγμα: Πλήρης Αθροιστής (Full Adder)**

C_{in}	A	B	C_{out}	S
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1



$$S = A \oplus B \oplus C_{in}$$

$$C_{out} = AB + AC_{in} + BC_{in}$$

Ορολογία

- **Συμπλήρωμα** (complement) μιας μεταβλητής A είναι η αντίστροφή της, $\overline{\mathrm{A}}$ ($\overline{\mathrm{A}}$ = NOT A)
  - $\overline{\mathrm{A}}, \overline{\mathrm{B}}, \overline{\mathrm{C}}$
- **Λεκτικά** (literals) ονομάζονται η μεταβλητή ή το συμπλήρωμά της
  - A, $\overline{\mathrm{A}}$, B, $\overline{\mathrm{B}}$, C, $\overline{\mathrm{C}}$
- Αναφερόμαστε στο A ως την **αληθινή μορφή** της μεταβλητής A και στο $\overline{\mathrm{A}}$ ως τη **συμπληρωματική μορφή** της μεταβλητής A
  - *Εάν A = TRUE, τότε $\bar{A}$ = FALSE*
  - *Εάν A = FALSE, τότε $\bar{A}$ = TRUE*

Ορολογία

- **Λογικό γινόμενο (ή απλώς γινόμενο)** ή **όρος** (implicant) ονομάζεται το γινόμενο των λεκτικών (η λογική πράξη AND με ένα ή περισσότερα λεκτικά)
  - $AB\overline{C}$, $\overline{A}C$, BC
- **Ελαχιστόρος** (minterm) ονομάζεται το **γινόμενο** που περιλαμβάνει **όλες** τις μεταβλητές της εισόδου
  - $AB\overline{C}$, $A\overline{B}\overline{C}$, ABC
- **Λογικό άθροισμα (ή απλώς άθροισμα)** ονομάζεται το άθροισμα των λεκτικών (η λογική πράξη OR με ένα ή περισσότερα λεκτικά)
  - $A+B+\overline{C}$, $\overline{A}+C$, $B+C$
- **Μεγιστόρος** (maxterm) ονομάζεται το **άθροισμα** που περιλαμβάνει **όλες** τις μεταβλητές της εισόδου
  - $(A+\overline{B}+C)$, $(\overline{A}+B+\overline{C})$, $(\overline{A}+\overline{B}+C)$

Προτεραιότητα πράξεων

- Η σειρά των πράξεων **παίζει ρόλο** κατά την ερμηνεία εξισώσεων Boole
  - *Στις εξισώσεις Boole, η πράξη **NOT** έχει την υψηλότερη προτεραιότητα*
  - *Ακολουθεί η πράξη **AND***
  - *Τη χαμηλότερη προτεραιότητα έχει η πράξη **OR***
  - *Για παράδειγμα*

    $Y = A + BC$ *μεταφράζεται σε* $Y = A \text{ OR } (B \text{ AND } C)$
  - *Αλλαγή στην προτεραιότητα επιτυγχάνεται με τη χρήση παρενθέσεων*

    $Y = (A + B)C$ *μεταφράζεται σε* $Y = (A \text{ OR } B) \text{ AND } C$

Πίνακας αλήθειας – Ελαχιστόροι

- Για να βρούμε την εξίσωση Boole για ένα ψηφιακό κύκλωμα πρέπει πρώτα να σχεδίασουμε τον πίνακα αλήθειας του
- Ο πίνακας αληθείας ενός ψηφιακού κυκλώματος με *N* εισόδους περιέχει 2^N γραμμές, μία για κάθε πιθανό συνδυασμό τιμών των εισόδων του
  - *Κάθε γραμμή συσχετίζεται με ένα **δυαδικό αριθμό των N bit**, σε αύξουσα σειρά, που αντιστοιχεί με τιμές των μεταβλητών της εισόδου*
  - *Κάθε γραμμή συσχετίζεται με έναν **ελαχιστόρο** που είναι **'1' ή TRUE** για τις τιμές των μεταβλητών της εισόδου της συγκεκριμένης σειράς*
  - *Ο ελαχιστόρος ονομάζεται* m_i *(i=0. .., 2^N-1), όπου i είναι η δεκαδική τιμή του αντίστοιχου δυαδικού αριθμού των N bit της γραμμής*

A B	Ελαχιστόρος	Όνομα
0 0	$\overline{A}\overline{B}$	m_0
0 1	$\overline{A}B$	m_1
1 0	$A\overline{B}$	m_2
1 1	AB	m_3

Κανονική μορφή αθροίσματος γινομένων

- Μπορούμε να βρούμε την εξίσωση Boole από οποιονδήποτε πίνακα αληθείας **αθροίζοντας (πράξη OR) καθέναν από τους ελαχιστόρους (πράξη AND)** για τους οποίους η έξοδος έχει τιμή **'1' (TRUE)**
- Άρα, η εξίσωση Boole σε κανονική μορφή **αθροίσματος γινομένων** είναι **το άθροισμα (OR) των γινομένων (AND)**
  - *Γράφεται και με τον **συμβολισμό σίγμα**, ως άθροισμα ελαχιστόρων*

A	B	Y	Ελαχι-στόρος	Ονομασία ελαχιστόρου
0	0	0	$\overline{A}\ \overline{B}$	m_0
0	1	1	$\overline{A}\ B$	m_1
1	0	0	$A\ \overline{B}$	m_2
1	1	1	$A\ B$	m_3

$$\mathbf{Y = F(A, B)} = \overline{A}B + AB = \mathbf{\Sigma(m_1, m_3) = \Sigma(1, 3)}$$

Πίνακας αλήθειας – Μεγιστόροι

- Για να βρούμε την εξίσωση Boole για ένα ψηφιακό κύκλωμα πρέπει πρώτα να σχεδίασουμε τον πίνακα αληθείας του
- Ο πίνακας αληθείας ενός ψηφιακού κυκλώματος με *N* εισόδους περιέχει 2^N γραμμές, μία για κάθε πιθανό συνδυασμό τιμών των εισόδων του
  - *Κάθε γραμμή συσχετίζεται με ένα **δυαδικό αριθμό των N bit**, σε αύξουσα σειρά, που αντιστοιχεί με τιμές των μεταβλητών της εισόδου*
  - *Κάθε γραμμή συσχετίζεται με έναν **μεγιστόρο** που είναι ‘0’ ή FALSE για τις τιμές των μεταβλητών της εισόδου της συγκεκριμένης σειράς*
  - *Ο μεγιστόρος ονομάζεται M_i (i=0. .., 2^N-1), όπου i είναι η δεκαδική τιμή του αντίστοιχου δυαδικού αριθμού των N bit της γραμμής*

A B	Μεγιστόρος	Όνομα
0 0	$\mathbf{A}+\mathbf{B}$	M_0
0 1	$\mathbf{A}+\overline{\mathbf{B}}$	M_1
1 0	$\overline{\mathbf{A}}+\mathbf{B}$	M_2
1 1	$\overline{\mathbf{A}}+\overline{\mathbf{B}}$	M_3

Κανονική μορφή γινομένου αθροισμάτων

- Μπορούμε να βρούμε την εξίσωση Boole από οποιονδήποτε πίνακα αληθείας **πολλαπλασιάζοντας (πράξη AND) καθέναν από τους μεγιστόρους (πράξη OR)** για τους οποίους η έξοδος έχει τιμή **'0' (FALSE)**
- Άρα, η εξίσωση Boole σε κανονική μορφή **γινομένου αθροισμάτων** είναι **το γινόμενο (AND) των αθροισμάτων (OR)**
  - *Γράφεται και με τον **συμβολισμό πι**, ως γινόμενο μεγιστόρων*

A	B	Y	Μεγι-στόρος	Ονομασία μεγιστόρου
0	0	0	$A + B$	M_0
0	1	1	$A + \overline{B}$	M_1
1	0	0	$\overline{A} + B$	M_2
1	1	1	$\overline{A} + \overline{B}$	M_3

$$\mathbf{Y = F(A, B) = (}\boldsymbol{A + B}\mathbf{)(}\boldsymbol{\overline{A} + B}\mathbf{) = \Pi(M_0, M_2) = \Pi(0, 2)}$$

Παράδειγμα 2.3

- Ο Γιάννης έχει αποφασίσει να πάει για πικνίκ.
- Σκέφτεται ότι **δεν πρόκειται να περάσει καλά**
  - *αν βρέξει (R = 1) ή*
  - *αν υπάρχουν μυρμήγκια (A = 1)*
- Σχεδιάστε ένα κύκλωμα που θα παράγει ως έξοδο E την τιμή TRUE *(E=1)* ***μόνο*** **αν ο Γιάννης απολαμβάνει το πικνίκ του**

A	R	E
0	0	
0	1	
1	0	
1	1	

Παράδειγμα 2.3

- Ο Γιάννης έχει αποφασίσει να πάει για πικνίκ.
- Σκέφτεται ότι **δεν πρόκειται να περάσει καλά**
  - *αν βρέξει (R = 1) ή*
  - *αν υπάρχουν μυρμήγκια (A = 1)*
- Σχεδιάστε ένα κύκλωμα που θα παράγει ως έξοδο E την τιμή TRUE *(E=1)* ***μόνο** αν ο Γιάννης απολαμβάνει το πικνίκ του*

A	R	E
0	0	1
0	1	0
1	0	0
1	1	0

Παράδειγμα 2.3

- Κανονική μορφή αθροίσματος γινομένου

A	R	E	Ελαχιστόρος
0	0	1	
0	1	0	
1	0	0	
1	1	0	

- Κανονική μορφή γινομένου αθροισμάτων

A	R	E	Μεγιστόρος
0	0	1	
0	1	0	
1	0	0	
1	1	0	

Παράδειγμα 2.3

- Κανονική μορφή **αθροίσματος γινομένου**

A	R	E	Ελαχιστόρος
0	0	1	$\overline{A}\,\overline{R}$
0	1	0	$\overline{A}\,R$
1	0	0	$A\,\overline{R}$
1	1	0	$A\,R$

- Κανονική μορφή **γινομένου αθροισμάτων**

A	R	E	Μεγιστόρος
0	0	1	
0	1	0	
1	0	0	
1	1	0	

Παράδειγμα 2.3

- Κανονική μορφή **αθροίσματος γινομένου**

A	R	E	Ελαχιστόρος
0	0	1	$\bar{A}\,\bar{R}$
0	1	0	$\bar{A}\,R$
1	0	0	$A\,\bar{R}$
1	1	0	$A\,R$

$$\mathbf{E} = \bar{A}\,\bar{R} = \Sigma(0)$$

- Κανονική μορφή **γινομένου αθροισμάτων**

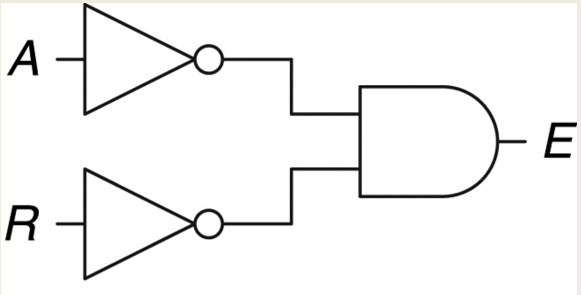
A	R	E	Μεγιστόρος
0	0	1	
0	1	0	
1	0	0	
1	1	0	

Παράδειγμα 2.3

- Κανονική μορφή **αθροίσματος γινομένου**

A	R	E	Ελαχιστόρος
0	0	1	$\bar{A}\,\bar{R}$
0	1	0	$\bar{A}\,R$
1	0	0	$A\,\bar{R}$
1	1	0	$A\,R$

$$\mathbf{E} = \bar{A}\,\bar{R} = \Sigma(0)$$



- Κανονική μορφή **γινομένου αθροισμάτων**

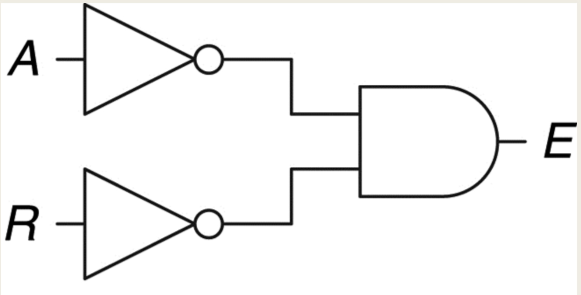
A	R	E	Μεγιστόρος
0	0	1	
0	1	0	
1	0	0	
1	1	0	

Παράδειγμα 2.3

- Κανονική μορφή **αθροίσματος γινομένου**

A	R	E	Ελαχιστόρος
0	0	1	$\overline{A}\,\overline{R}$
0	1	0	$\overline{A}\,R$
1	0	0	$A\,\overline{R}$
1	1	0	$A\,R$

$$\mathbf{E} = \overline{\boldsymbol{A}}\,\overline{\boldsymbol{R}} = \boldsymbol{\Sigma(0)}$$



- Κανονική μορφή **γινομένου αθροισμάτων**

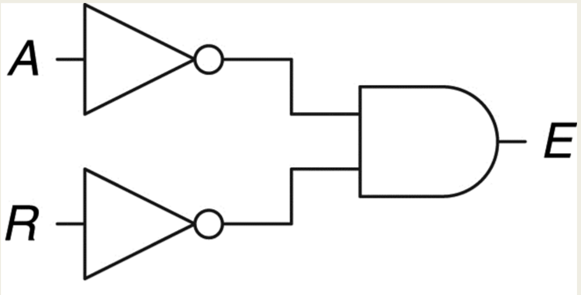
A	R	E	Μεγιστόρος
0	0	1	$A + R$
0	1	0	$A + \overline{R}$
1	0	0	$\overline{A} + R$
1	1	0	$\overline{A} + \overline{R}$

Παράδειγμα 2.3

- Κανονική μορφή **αθροίσματος γινομένου**

A	R	E	Ελαχιστόρος
0	0	1	$\bar{A}\,\bar{R}$
0	1	0	$\bar{A}\,R$
1	0	0	$A\,\bar{R}$
1	1	0	$A\,R$

$$\mathbf{E} = \bar{A}\,\bar{R} = \Sigma(0)$$



- Κανονική μορφή **γινομένου αθροισμάτων**

A	R	E	Μεγιστόρος
0	0	1	$A + R$
0	1	0	$A + \bar{R}$
1	0	0	$\bar{A} + R$
1	1	0	$\bar{A} + \bar{R}$

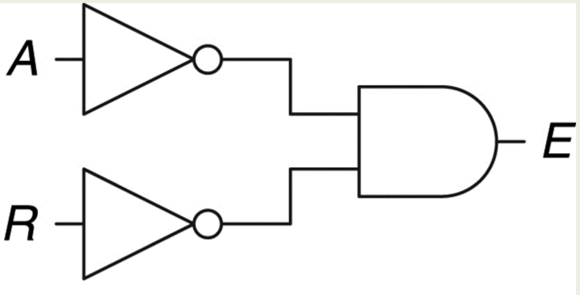
$$\mathbf{E} = (A + \bar{R})(\bar{A} + R)(\bar{A} + \bar{R}) = \Pi(1,2,3) = \overline{A + R}$$

Παράδειγμα 2.3

- Κανονική μορφή αθροίσματος γινομένου

A	R	E	Ελαχιστόρος
0	0	1	$\bar{A}\,\bar{R}$
0	1	0	$\bar{A}\,R$
1	0	0	$A\,\bar{R}$
1	1	0	$A\,R$

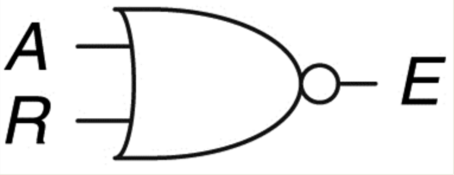
$$\mathbf{E} = \bar{A}\,\bar{R} = \Sigma(0)$$



- Κανονική μορφή γινομένου αθροισμάτων

A	R	E	Μεγιστόρος
0	0	1	$A + R$
0	1	0	$A + \bar{R}$
1	0	0	$\bar{A} + R$
1	1	0	$\bar{A} + \bar{R}$

$$\mathbf{E} = (\mathbf{A} + \bar{R})(\bar{A} + \mathbf{R})(\bar{A} + \bar{R}) = \Pi(1,2,3) = \overline{\mathbf{A} + R}$$



Επιλεγμένες ασκήσεις

- Άσκηση 2.33
  Ο Γιάννης Μπιτάκης απολαμβάνει το πικνίκ του όταν η μέρα είναι ηλιόλουστη και δεν υπάρχουν μυρμήγκια. Το απολαμβάνει επίσης όταν βλέπει ένα κολιμπρί την ημέρα του πικνίκ, καθώς επίσης όταν, την ημέρα του πικνίκ, υπάρχουν μυρμήγκια και πασχαλίτσες.
  - *Γράψτε μια εξίσωση Boole για την απόλαυσή του (enjoyment, E) σε σχέση με:*
    - τον ήλιο (sun, S),
    - τα μυρμήγκια (ants, A),
    - τα κολιμπρί (hummingbirds, H), και
    - τις πασχαλίτσες (ladybugs, L)

$$E = S\overline{A} + AL + H$$

Επιλεγμένες ασκήσεις

- **Άσκηση 2.2**
  Γράψτε μια εξίσωση Boole σε κανονική μορφή αθροίσματος γινομένων για τον πίνακα αληθείας (β) της Εικόνας 2.81
  - *Χρησιμοποιήστε τους ελαχιστόρους*
- **Άσκηση 2.4**
  Γράψτε μια εξίσωση Boole σε κανονική μορφή γινομένου αθροισμάτων για τον πίνακα αληθείας (β) της Εικόνας 2.81
  - *Χρησιμοποιήστε τους μεγιστόρους*

(b)

A	B	C	Y
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	0

$$Y = \overline{A}\,\overline{B}C + \overline{A}B\overline{C} + \overline{A}BC + A\overline{B}\,\overline{C} + AB\overline{C}$$

$$Y = (A + B + C)(\overline{A} + B + \overline{C})(\overline{A} + \overline{B} + \overline{C})$$

Άλγεβρα Boole

- Η κανονική μορφή αθροίσματος γινομένων ή γινομένου αθροισμάτων συνήθως **δεν δίνει την απλούστερη εξίσωση**
- Χρησιμοποιώντας την άλγεβρα Boole μπορούμε να **ελαχιστοποιήσουμε** τις εξισώσεις Boole
- Η άλγεβρα Boole βασίζεται σε ένα σύνολο αξιωμάτων και θεωρημάτων
  - *Τα **αξιώματα είναι αναπόδεικτα** με την έννοια ότι ένας ορισμός δεν μπορεί να αποδειχθεί*
  - *Τα **θεωρήματα αποδεικνύονται με βάση τα αξιώματα***
- Τα αξιώματα και τα θεωρήματα της άλγεβρας Boole ακολουθούν την **αρχή του *δυϊσμού*** *(δεν ισχύει στη συνήθη άλγεβρα)*
  - ***Αν, σε μία λογική έκφραση, τα σύμβολα 0 και 1 και οι τελεστές • (AND) και + (OR) εναλλαχθούν, η λογική έκφραση θα εξακολουθεί να είναι ορθή***

Αξιώματα άλγεβρας Boole

	Αξίωμα	Ονομασία
A1	$B = 0$ αν $B \neq 1$	Δυαδικό πεδίο
A2	$0 = \overline{1}$	NOT
A3	$0 \bullet 0 = 0$	AND
A4	$1 \bullet 1 = 1$	AND
A5	$0 \bullet 1 = 1 \bullet 0 = 0$	AND

Αξιώματα άλγεβρας Boole

	Αξίωμα	Ονομασία
A1	$B = 0$ αν $B \neq 1$	Δυαδικό πεδίο
A2	$0 = \overline{1}$	NOT
A3	$0 \bullet 0 = 0$	AND
A4	$1 \bullet 1 = 1$	AND
A5	$0 \bullet 1 = 1 \bullet 0 = 0$	AND

Δυϊσμός: Αντιστρέψτε: $\bullet$ με $+$

0 με 1

Αξιώματα άλγεβρας Boole

	Αξίωμα	Δυϊκό αξίωμα	Ονομασία
A1	$B = 0$ αν $B \neq 1$	$B = 1$ αν $B \neq 0$	Δυαδικό πεδίο
A2	$0 = \overline{1}$	$1 = \overline{0}$	NOT
A3	$0 \bullet 0 = 0$	$1 + 1 = 1$	AND/OR
A4	$1 \bullet 1 = 1$	$0 + 0 = 0$	AND/OR
A5	$0 \bullet 1 = 1 \bullet 0 = 0$	$1 + 0 = 0 + 1 = 1$	AND/OR

Δυϊσμός: Αντιστρέψτε: • με +

0 με 1

Αξιώματα άλγεβρας Boole

	Αξίωμα	Δυϊκό αξίωμα	Ονομασία
A1	$B = 0$ αν $B \neq 1$	$B = 1$ αν $B \neq 0$	Δυαδικό πεδίο

- Το αξίωμα A1 ορίζει ότι μια μεταβλητή Boole *B* έχει τιμή 0 αν δεν έχει τιμή 1
- Το δυϊκό αξίωμα, ορίζει ότι η μεταβλητή έχει τιμή 1 αν δεν έχει τιμή 0
- Και τα δύο μαζί, μας λένε ότι εργαζόμαστε σε ένα πεδίο Boole ή **δυαδικό πεδίο με μηδενικά (0) και άσους (1)**

Αξιώματα άλγεβρας Boole

	Αξίωμα	Δυϊκό αξίωμα	Ονομασία
A2	$0 = \overline{1}$	$1 = \overline{0}$	NOT

- Τα αξιώματα A2 και A2′ ορίζουν την **πράξη NOT**

	Αξίωμα	Δυϊκό αξίωμα	Ονομασία
A3	$0 \bullet 0 = 0$	$1 + 1 = 1$	AND/OR
A4	$1 \bullet 1 = 1$	$0 + 0 = 0$	AND/OR
A5	$0 \bullet 1 = 1 \bullet 0 = 0$	$1 + 0 = 0 + 1 = 1$	AND/OR

- Τα αξιώματα A3 έως A5 ορίζουν την **πράξη AND**
- Τα δυϊκά τους, ορίζουν την **πράξη OR**

Θεωρήματα μίας μεταβλητής

	Θεώρημα	Ονομασία
T1	$B \bullet 1 = B$	Ουδέτερο στοιχείο
T2	$B \bullet 0 = 0$	Κυρίαρχο στοιχείο
T3	$B \bullet B = B$	Ταυτοδυναμία
T4	$\overline{\overline{B}} = B$	Διπλό συμπλήρωμα
T5	$B \bullet \overline{B} = 0$	Συμπληρώματα

Θεωρήματα μίας μεταβλητής

	Θεώρημα	Ονομασία
T1	$B \bullet 1 = B$	Ουδέτερο στοιχείο
T2	$B \bullet 0 = 0$	Κυρίαρχο στοιχείο
T3	$B \bullet B = B$	Ταυτοδυναμία
T4	$\overline{\overline{B}} = B$	Διπλό συμπλήρωμα
T5	$B \bullet \overline{B} = 0$	Συμπληρώματα

Δυϊσμός: Αντιστρέψτε: $\bullet$ με $+$
0 με 1

Θεωρήματα μίας μεταβλητής

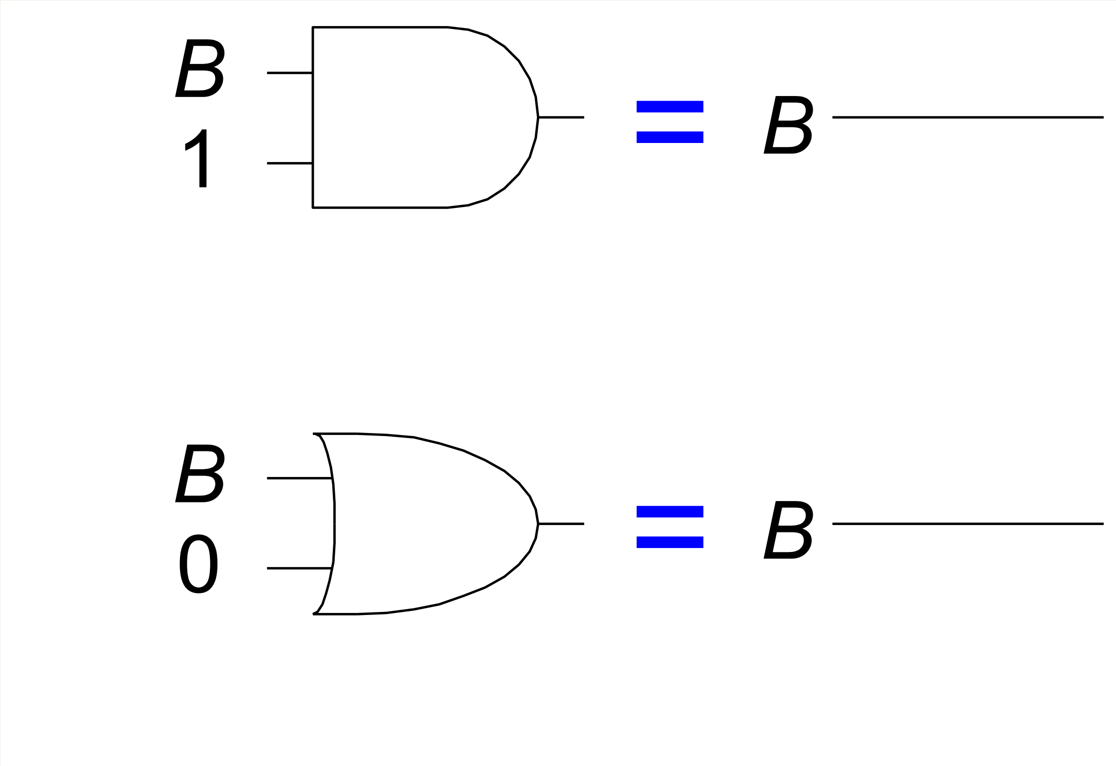
	Θεώρημα	Δυϊκό Θεώρημα	Ονομασία
T1	$B \bullet 1 = B$	$B + 0 = B$	Ουδέτερο στοιχείο
T2	$B \bullet 0 = 0$	$B + 1 = 1$	Κυρίαρχο στοιχείο
T3	$B \bullet B = B$	$B + B = B$	Ταυτοδυναμία
T4	$\bar{\bar{B}} = B$		Διπλό συμπλήρωμα
T5	$B \bullet \bar{B} = 0$	$B + \bar{B} = 1$	Συμπληρώματα

Δυϊσμός: Αντιστρέψτε: $\bullet$ με $+$

0 με 1

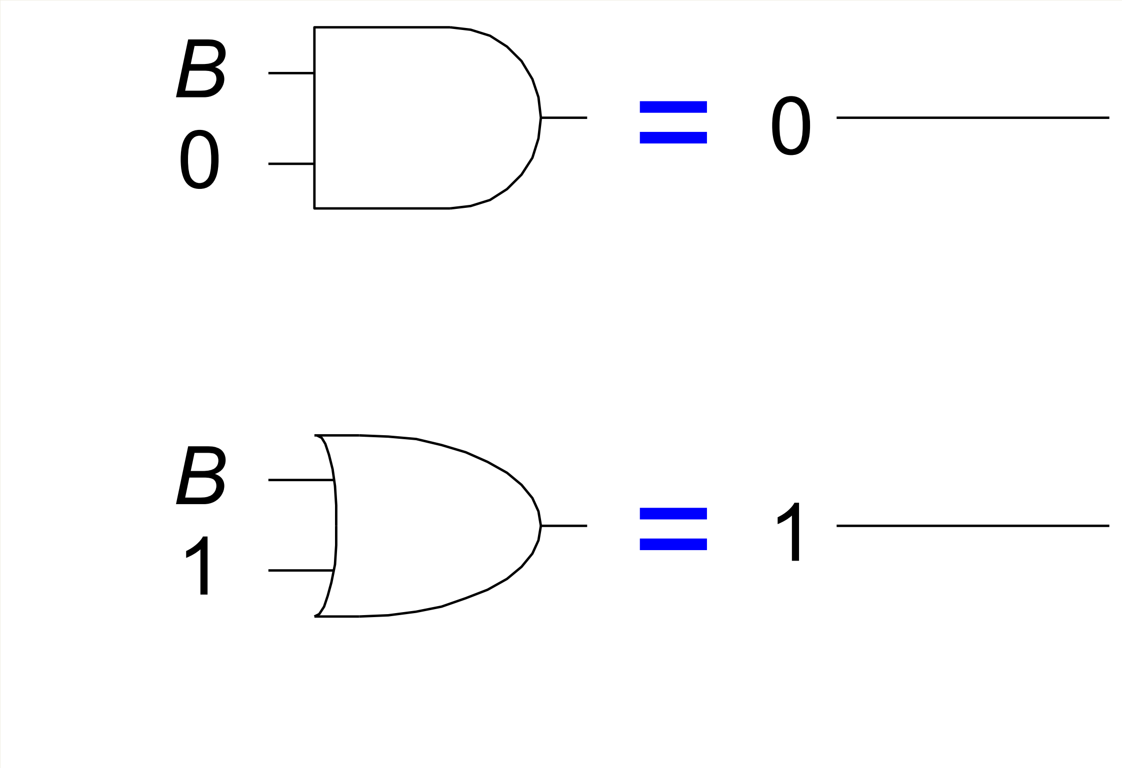
T1: Ουδέτερο στοιχείο

- B • 1 = B (Το 1 είναι ουδέτερο στοιχείο στην πράξη AND)
- B + 0 = B (Το 0 είναι ουδέτερο στοιχείο στην πράξη OR)



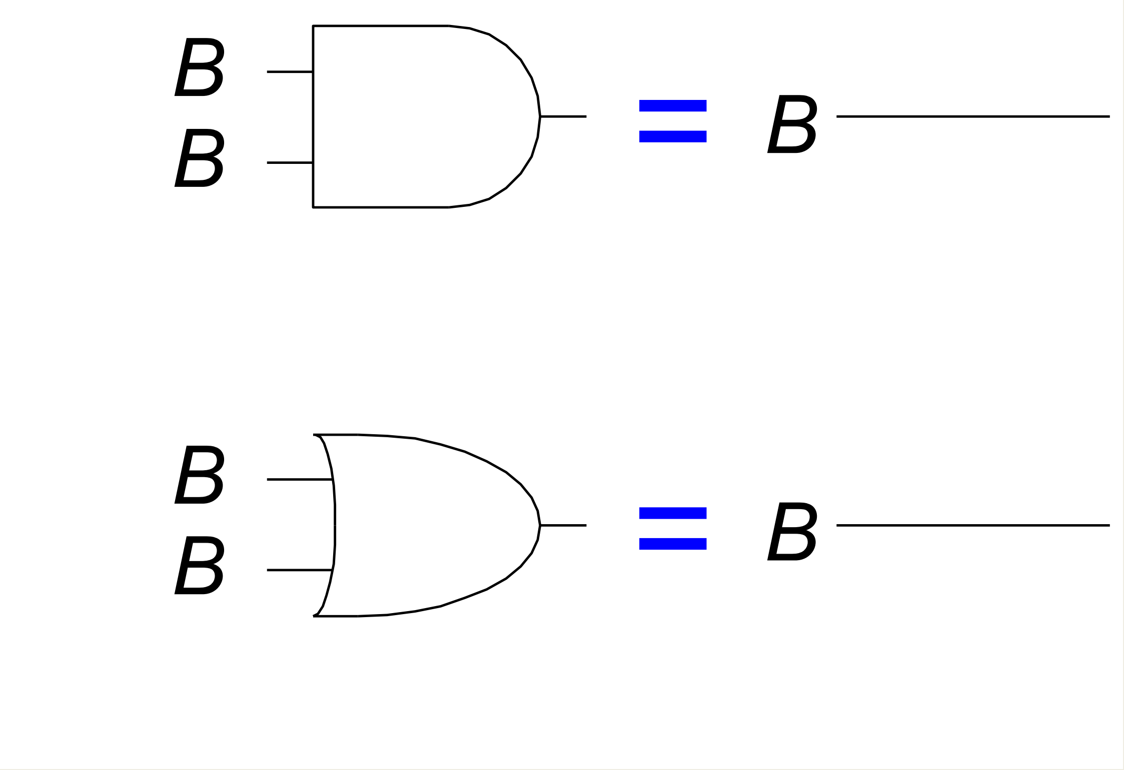
T2: Κυρίαρχο στοιχείο

- B • 0 = 0 (Το 0 είναι κυρίαρχο στοιχείο στην πράξη AND)
- B + 1 = 1 (Το 1 είναι κυρίαρχο στοιχείο στην πράξη OR)
  - *(δεν ισχύει στη συνήθη άλγεβρα)*



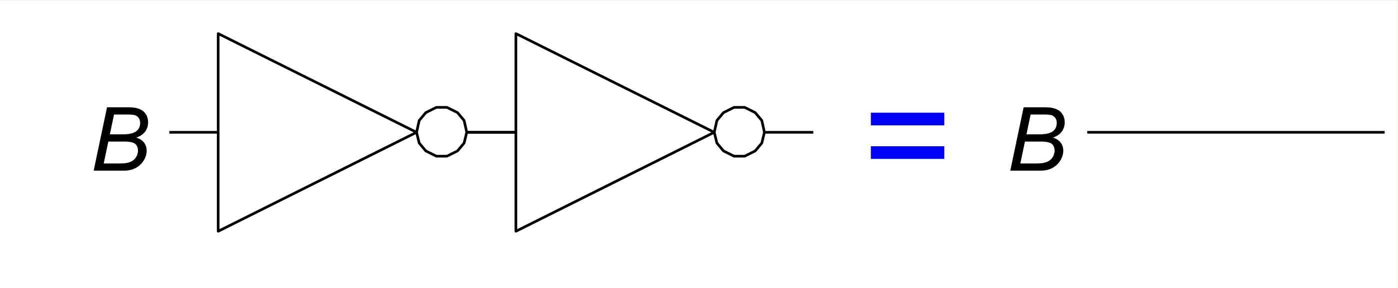
T3: Ταυτοδυναμία (Idempotency)

- $B \bullet B = B$ (δεν ισχύει στη συνήθη άλγεβρα)
- $B + B = B$ (δεν ισχύει στη συνήθη άλγεβρα)



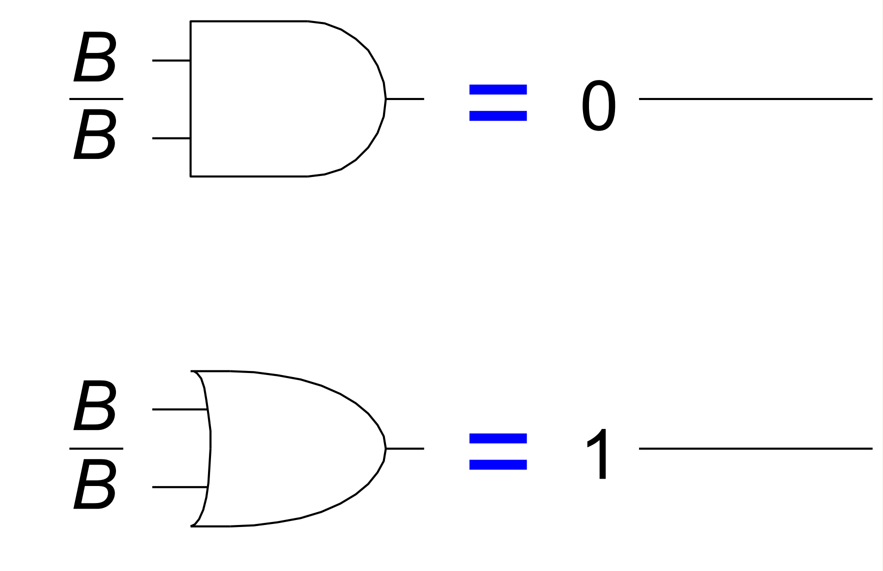
T4: Διπλό συμπλήρωμα

- $\bar{\bar{\mathbf{B}}} = \mathbf{B}$ **(δεν ισχύει στη συνήθη άλγεβρα)**



T5: Θεώρημα των συμπληρωμάτων (Complement Theorem)

- Το **θεώρημα των συμπληρωμάτων** (Complement Theorem), ορίζει ότι
  - *η πράξη AND μιας μεταβλητής με το συμπλήρωμά της δίνει αποτέλεσμα 0* $(\mathbf{B} \cdot \overline{\mathbf{B}} = \mathbf{0})$
  - *η πράξη OR μιας μεταβλητής με το συμπλήρωμά της δίνει αποτέλεσμα 1* $(\mathbf{B} + \overline{\mathbf{B}} = \mathbf{1})$



Θεωρήματα μίας μεταβλητής

	Θεώρημα	Δυϊκό Θεώρημα	Ονομασία
T1	$B \bullet 1 = B$	$B + 0 = B$	Ουδέτερο στοιχείο
T2	$B \bullet 0 = 0$	$B + 1 = 1$	Κυρίαρχο στοιχείο
T3	$B \bullet B = B$	$B + B = B$	Ταυτοδυναμία
T4	$\overline{\overline{B}} = B$		Διπλό συμπλήρωμα
T5	$B \bullet \overline{B} = 0$	$B + \overline{B} = 1$	Συμπληρώματα

Δυϊσμός: Αντιστρέψτε: $\bullet$ με $+$

0 με 1

Θεωρήματα πολλών μεταβλητών

	Θεώρημα	Δυϊκό Θεώρημα	Ονομασία
T6	$B \bullet C = C \bullet B$	$B + C = C + B$	Αντιμεταθετικότητα
T7	$(B \bullet C) \bullet D = B \bullet (C \bullet D)$	$(B + C) + D = B + (C + D)$	Προσεταιριστικότητα
T8	$(B \bullet C) + (B \bullet D) = B \bullet (C + D)$	$(B + C) \bullet (B + D) = B + (C \bullet D)$	Επιμεριστικότητα
T9	$B \bullet (B + C) = B$	$B + (B \bullet C) = B$	Κάλυψη
T10	$(B \bullet C) + (B \bullet \overline{C}) = B$	$(B + C) \bullet (B + \overline{C}) = B$	Συνδυασμός
T11	$(B \bullet C) + (\overline{B} \bullet D) + (C \bullet D) =$ $B \bullet C + \overline{B} \bullet D$	$(B + C) \bullet (\overline{B} + D) \bullet (C + D) =$ $(B + C) \bullet (\overline{B} + D)$	Ομοφωνία

Δυϊσμός: Αντιστρέψτε: $\bullet$ με $+$

0 με 1

Θεωρήματα πολλών μεταβλητών

	Θεώρημα	Ονομασία
T6	$B \bullet C = C \bullet B$	Αντιμεταθετικότητα
T7	$(B \bullet C) \bullet D = B \bullet (C \bullet D)$	Προσεταιριστικότητα
T8	$(B \bullet C) + (B \bullet D) = B \bullet (C + D)$	Επιμεριστικότητα
T9	$B \bullet (B + C) = B$	Κάλυψη
T10	$(B \bullet C) + (B \bullet \overline{C}) = B$	Συνδυασμός
T11	$(B \bullet C) + (\overline{B} \bullet D) + (C \bullet D) =$ $B \bullet C + \overline{B} \bullet D$	Ομοφωνία

Πώς αποδεικνύονται?

Μέθοδοι απόδειξης θεωρημάτων Boole

- **Μέθοδος 1:** Τέλεια επαγωγή (perfect induction) η οποία στηρίζεται στον **πίνακα αλήθειας**
  - *Να δειχτεί δηλαδή ότι το θεώρημα ισχύει **για όλες τις πιθανές τιμές** των μεταβλητών*
- **Μέθοδος 2:** Χρησιμοποιώντας **άλλα θεωρήματα ή αξιώματα** με σκοπό να απλοποιήσουμε την εξίσωση
  - *Δηλαδή μορφοποιούμε το ένα μέλος της εξίσωσης έτσι ώστε να είναι **πανομοιότυπο με το άλλο***

Απόδειξη με τέλεια επαγωγή

	Θεώρημα	Ονομασία
T6	$B \bullet C = C \bullet B$	Αντιμεταθετικότητα

B	C	BC	CB
0	0		
0	1		
1	0		
1	1		

Απόδειξη με τέλεια επαγωγή

	Θεώρημα	Ονομασία
T6	B • C = C • B	Αντιμεταθετικότητα

B	C	BC	CB
0	0	0	0
0	1	0	0
1	0	0	0
1	1	1	1

Θεωρήματα πολλών μεταβλητών

	Θεώρημα	Ονομασία
T6	$B \bullet C = C \bullet B$	Αντιμεταθετικότητα
T7	$(B \bullet C) \bullet D = B \bullet (C \bullet D)$	Προσεταιριστικότητα
T8	$(B \bullet C) + (B \bullet D) = B \bullet (C + D)$	Επιμεριστικότητα
T9	$B \bullet (B + C) = B$	Κάλυψη
T10	$(B \bullet C) + (B \bullet \overline{C}) = B$	Συνδυασμός
T11	$(B \bullet C) + (\overline{B} \bullet D) + (C \bullet D) =$ $B \bullet C + \overline{B} \bullet D$	Ομοφωνία

T9: Κάλυψη

	Θεώρημα	Ονομασία
T9	$B \bullet (B + C) = B$	Κάλυψη

Να αποδειχθεί με:

- **Μέθοδος 1:** Τέλεια επαγωγή
- **Μέθοδος 2:** Χρησιμοποιώντας άλλα θεωρήματα ή αξιώματα

T9: Κάλυψη

	Θεώρημα	Ονομασία
T9	$B \bullet (B + C) = B$	Κάλυψη

Μέθοδος 1: Τέλεια επαγωγή

B	C	(B + C)	B(B + C)
0	0		
0	1		
1	0		
1	1		

T9: Κάλυψη

	Θεώρημα	Ονομασία
T9	$B \bullet (B + C) = B$	Κάλυψη

Μέθοδος 1: Τέλεια επαγωγή

B	C	(B + C)	B(B + C)
0	0	0	
0	1	1	
1	0	1	
1	1	1	

T9: Κάλυψη

	Θεώρημα	Ονομασία
T9	$B \bullet (B + C) = B$	Κάλυψη

Μέθοδος 1: Τέλεια επαγωγή

B	C	(B + C)	B(B + C)
0	0	0	0
0	1	1	0
1	0	1	1
1	1	1	1

T9: Κάλυψη

	Θεώρημα	Ονομασία
T9	B • (B + C) = B	Κάλυψη

Μέθοδος 1: Τέλεια επαγωγή

B	C	(B + C)	B(B + C)
0	0	0	0
0	1	1	0
1	0	1	1
1	1	1	1

T9: Κάλυψη

	Θεώρημα	Ονομασία
T9	$B \bullet (B + C) = B$	Κάλυψη

Μέθοδος 2: Χρησιμοποιώντας άλλα θεωρήματα ή αξιώματα

$B \bullet (B+C) \quad =$

T9: Κάλυψη

	Θεώρημα	Ονομασία
T9	$B \bullet (B + C) = B$	Κάλυψη

Μέθοδος 2: Χρησιμοποιώντας άλλα θεωρήματα ή αξιώματα

$B \bullet (B+C) \quad = B \bullet B + B \bullet C$ **T8: Επιμεριστικότητα**

T9: Κάλυψη

	Θεώρημα	Ονομασία
T9	$B \bullet (B + C) = B$	Κάλυψη

Μέθοδος 2: Χρησιμοποιώντας άλλα θεωρήματα ή αξιώματα

$B \bullet (B+C)$	$= B \bullet B + B \bullet C$	T8: Επιμεριστικότητα
	$= B + B \bullet C$	T3: Ταυτοδυναμία

T9: Κάλυψη

	Θεώρημα	Ονομασία
T9	$B \bullet (B + C) = B$	Κάλυψη

Μέθοδος 2: Χρησιμοποιώντας άλλα θεωρήματα ή αξιώματα

$B \bullet (B+C)$	$= B \bullet B + B \bullet C$	T8: Επιμεριστικότητα
	$= B + B \bullet C$	T3: Ταυτοδυναμία
	$= B \bullet 1 + B \bullet C$	T1: Ουδέτερο στοιχείο

T9: Κάλυψη

	Θεώρημα	Ονομασία
T9	$B \bullet (B + C) = B$	Κάλυψη

Μέθοδος 2: Χρησιμοποιώντας άλλα θεωρήματα ή αξιώματα

$B \bullet (B+C)$	$= B \bullet B + B \bullet C$	T8: Επιμεριστικότητα
	$= B + B \bullet C$	T3: Ταυτοδυναμία
	$= B \bullet 1 + B \bullet C$	T1: Ουδέτερο στοιχείο
	$= B \bullet (1 + C)$	T8: Επιμεριστικότητα

T9: Κάλυψη

	Θεώρημα	Ονομασία
T9	$B \bullet (B + C) = B$	Κάλυψη

Μέθοδος 2: Χρησιμοποιώντας άλλα θεωρήματα ή αξιώματα

$B \bullet (B+C)$	$= B \bullet B + B \bullet C$	T8: Επιμεριστικότητα
	$= B + B \bullet C$	T3: Ταυτοδυναμία
	$= B \bullet 1 + B \bullet C$	T1: Ουδέτερο στοιχείο
	$= B \bullet (1 + C)$	T8: Επιμεριστικότητα
	$= B \bullet (1)$	T2: Κυρίαρχο στοιχείο

T9: Κάλυψη

	Θεώρημα	Ονομασία
T9	$B \bullet (B + C) = B$	Κάλυψη

Μέθοδος 2: Χρησιμοποιώντας άλλα θεωρήματα ή αξιώματα

$B \bullet (B+C)$	$= B \bullet B + B \bullet C$	T8: Επιμεριστικότητα
	$= B + B \bullet C$	T3: Ταυτοδυναμία
	$= B \bullet 1 + B \bullet C$	T1: Ουδέτερο στοιχείο
	$= B \bullet (1 + C)$	T8: Επιμεριστικότητα
	$= B \bullet (1)$	T2: Κυρίαρχο στοιχείο
	$= B$	T1: Ουδέτερο στοιχείο

T10: Συνδυασμός

	Θεώρημα	Ονομασία
T10	$(B \bullet C) + (B \bullet \overline{C}) = B$	Συνδυασμός

Μέθοδος 2: Χρησιμοποιώντας άλλα θεωρήματα ή αξιώματα

$$(B \bullet C) + (B \bullet \overline{C}) \quad =$$

T10: Συνδυασμός

	Θεώρημα	Ονομασία
T10	$(B \bullet C) + (B \bullet \overline{C}) = B$	Συνδυασμός

Μέθοδος 2: Χρησιμοποιώντας άλλα θεωρήματα ή αξιώματα

$(B \bullet C)+(B \bullet \overline{C}) \quad = B \bullet (C+\overline{C})$ **T8: Επιμεριστικότητα**

T10: Συνδυασμός

	Θεώρημα	Ονομασία
T10	$(B \bullet C) + (B \bullet \overline{C}) = B$	Συνδυασμός

Μέθοδος 2: Χρησιμοποιώντας άλλα θεωρήματα ή αξιώματα

$(B \bullet C)+(B \bullet \overline{C})$	$= B \bullet (C+\overline{C})$	T8: Επιμεριστικότητα
	$= B \bullet (1)$	T5': Συμπλήρωμα

T10: Συνδυασμός

	Θεώρημα	Ονομασία
T10	$(B \bullet C) + (B \bullet \overline{C}) = B$	Συνδυασμός

Μέθοδος 2: Χρησιμοποιώντας άλλα θεωρήματα ή αξιώματα

$(B \bullet C)+(B \bullet \overline{C})$	$= B \bullet (C+\overline{C})$	**T8: Επιμεριστικότητα**
	$= B \bullet (1)$	**T5’: Συμπλήρωμα**
	$= B$	**T1: Ουδέτερο στοιχείο**

T11: Ομοφωνία

	Θεώρημα	Ονομασία
T11	$(B \bullet C) + (\overline{B} \bullet D) + (C \bullet D) =$ $B \bullet C + \overline{B} \bullet D$	Ομοφωνία

Αποδείξτε το χρησιμοποιώντας (1) την τέλεια επαγωγή και (2) άλλα θεωρήματα ή αξιώματα

T11: Ομοφωνία

	Θεώρημα	Ονομασία
T11	$(B \bullet C) + (\overline{B} \bullet D) + (C \bullet D) = B \bullet C + \overline{B} \bullet D$	Ομοφωνία

Μέθοδος 1: τέλεια επαγωγή (Παράδειγμα 2.5)

B	C	D	$B \bullet C$	$\overline{B} \bullet D$	$C \bullet D$	$(B \bullet C) + (\overline{B} \bullet D) + (C \bullet D)$	$B \bullet C + \overline{B} \bullet D$
0	0	0	0	0	0		
0	0	1	0	1	0		
0	1	0	0	0	0		
0	1	1	0	1	1		
1	0	0	0	0	0		
1	0	1	0	0	0		
1	1	0	1	0	0		
1	1	1	1	0	1		

T11: Ομοφωνία

	Θεώρημα	Ονομασία
T11	$(B \bullet C) + (\overline{B} \bullet D) + (C \bullet D) =$ $B \bullet C + \overline{B} \bullet D$	Ομοφωνία

Μέθοδος 1: τέλεια επαγωγή (Παράδειγμα 2.5)

B	C	D	$B \bullet C$	$\overline{B} \bullet D$	$C \bullet D$	$(B \bullet C) + (\overline{B} \bullet D) + (C \bullet D)$	$B \bullet C + \overline{B} \bullet D$
0	0	0	0	0	0	0	
0	0	1	0	1	0	1	
0	1	0	0	0	0	0	
0	1	1	0	1	1	1	
1	0	0	0	0	0	0	
1	0	1	0	0	0	0	
1	1	0	1	0	0	1	
1	1	1	1	0	1	1	

T11: Ομοφωνία

	Θεώρημα	Ονομασία
T11	$(B \bullet C) + (\overline{B} \bullet D) + (C \bullet D) =$ $B \bullet C + \overline{B} \bullet D$	Ομοφωνία

Μέθοδος 1: τέλεια επαγωγή (Παράδειγμα 2.5)

B	C	D	$B \bullet C$	$\overline{B} \bullet D$	$C \bullet D$	$(B \bullet C) + (\overline{B} \bullet D) + (C \bullet D)$	$B \bullet C + \overline{B} \bullet D$
0	0	0	0	0	0	0	0
0	0	1	0	1	0	1	1
0	1	0	0	0	0	0	0
0	1	1	0	1	1	1	1
1	0	0	0	0	0	0	0
1	0	1	0	0	0	0	0
1	1	0	1	0	0	1	1
1	1	1	1	0	1	1	1

T11: Ομοφωνία

	Θεώρημα	Ονομασία
T11	$(B \bullet C) + (\overline{B} \bullet D) + (C \bullet D) =$ $B \bullet C + \overline{B} \bullet D$	Ομοφωνία

Μέθοδος 1: τέλεια επαγωγή (Παράδειγμα 2.5)

B	C	D	$B \bullet C$	$\overline{B} \bullet D$	$C \bullet D$	$(B \bullet C) + (\overline{B} \bullet D) + (C \bullet D)$	$B \bullet C + \overline{B} \bullet D$
0	0	0	0	0	0	0	0
0	0	1	0	1	0	1	1
0	1	0	0	0	0	0	0
0	1	1	0	1	1	1	1
1	0	0	0	0	0	0	0
1	0	1	0	0	0	0	0
1	1	0	1	0	0	1	1
1	1	1	1	0	1	1	1

T11: Ομοφωνία

	Θεώρημα	Ονομασία
T11	$(B \bullet C) + (\overline{B} \bullet D) + (C \bullet D) =$ $B \bullet C + \overline{B} \bullet D$	Ομοφωνία

Μέθοδος 2: άλλα θεωρήματα ή αξιώματα

$B \bullet C + \overline{B} \bullet D + C \bullet D$

$= B \bullet C + \overline{B} \bullet D + C \bullet D \bullet 1$ T1: Ουδέτερο στοιχείο

$= B \bullet C + \overline{B} \bullet D + C \bullet D \bullet (B + \overline{B})$ T5': Συμπληρώματα

$= B \bullet C + \overline{B} \bullet D + C \bullet D \bullet B + C \bullet D \bullet \overline{B}$ T8: Επιμεριστικότητα

$= B \bullet C + \overline{B} \bullet D + B \bullet C \bullet D + \overline{B} \bullet C \bullet D$ T6: Αντιμεταθετικότητα

$= B \bullet C + \overline{B} \bullet D$ T9': Κάλυψη

Εφαρμογή των θεωρημάτων

	Θεώρημα	Δυϊκό Θεώρημα	Ονομασία
T6	$B \bullet C = C \bullet B$	$B + C = C + B$	Αντιμεταθετικότητα
T7	$(B \bullet C) \bullet D = B \bullet (C \bullet D)$	$(B + C) + D = B + (C + D)$	Προσεταιριστικότητα
T8	$(B \bullet C) + (B \bullet D) = B \bullet (C + D)$	$(B + C) \bullet (B + D) = B + (C \bullet D)$	Επιμεριστικότητα
T9	$B \bullet (B + C) = B$	$B + (B \bullet C) = B$	Κάλυψη
T10	$(B \bullet C) + (B \bullet \overline{C}) = B$	$(B + C) \bullet (B + \overline{C}) = B$	Συνδυασμός
T11	$(B \bullet C) + (\overline{B} \bullet D) + (C \bullet D) = B \bullet C + \overline{B} \bullet D$	$(B + C) \bullet (\overline{B} + D) \bullet (C + D) = (B + C) \bullet (\overline{B} + D)$	Ομοφωνία

Δυϊσμός: Αντιστρέψτε: $\bullet$ με $+$

0 με 1

Ανεξαρτησία στη σειρά των εισόδων

	Θεώρημα	Δυϊκό Θεώρημα	Ονομασία
T6	$B \bullet C = C \bullet B$	$B + C = C + B$	Αντιμεταθετικότητα
T7	$(B \bullet C) \bullet D = B \bullet (C \bullet D)$	$(B + C) + D = B + (C + D)$	Προσεταιριστικότητα

- Λόγω της αντιμεταθετικότητας, η σειρά των εισόδων σε μια πύλη AND ή OR δεν επηρεάζει την τιμή της εξόδου της.
- Λόγω της προσεταιριστικότητας, οι εκάστοτε ομαδοποιήσεις των εισόδων δεν επηρεάζουν την τιμή της εξόδου
  - *Παράδειγμα: Σε ένα δένδρο από πύλες AND ή πύλες OR δύο εισόδων για υλοποιήσεις πράξεων πολλών bit (π.χ. 8 bit) δεν έχει σημασία η σειρά των εισόδων στις πύλες δύο εισόδων*
    - Στα 8 bit, το δένδρο θα έχει 7 πύλες κατανεμημένες σε 3 επίπεδα: στο $1^{ο}$ επίπεδο είναι 4 πύλες, στο $2^{ο}$ επίπεδο 2 πύλες και στο $3^{ο}$ επίπεδο 1 πύλη

Ελαχιστοποίηση εξισώσεων Boole

	Θεώρημα	Δυϊκό Θεώρημα	Ονομασία
T8	(B • C) + (B • D) = B • (C + D)	(B + C) • (B + D) = B + (C • D)	Επιμεριστικότητα

- Ελαχιστοποίηση (ή απλοποίηση) εξισώσεων Boole σημαίνει η εύρεση εκείνης της εξίσωσης που έχει:
  - *Τους λιγότερους όρους*
  - *Κάθε όρος έχει τα λιγότερα λεκτικά (είναι πρώτος όρος)*
- Η ελαχιστοποίηση των εξισώσεων Boole βασίζεται κυρίως στα θεωρήματα της επιμεριστικότητας, των συμπληρωμάτων, του ουδέτερου στοιχείου και του κυρίαρχου στοιχείου
  - *Οι εξισώσεις Boole σε μορφή αθροίσματος γινομένων ελαχιστοποιούνται με το T8 (που ισχύει στη συνήθη άλγεβρα)*
  - *Οι εξισώσεις Boole σε μορφή γινομένου αθροισμάτων ελαχιστοποιούνται με το T8’ (που δεν ισχύει στη συνήθη άλγεβρα)*
    - Έχουμε μία δυσκολία στην εφαρμογή του T8’ και για αυτό το λόγο συνήθως χρησιμοποιούμε τα **αθροίσματα γινομένων**, αντί των γινομένων αθροισμάτων

Ελαχιστοποίηση εξισώσεων Boole

	Θεώρημα	Δυϊκό Θεώρημα	Ονομασία
T8	$(B \bullet C) + (B \bullet D) = B \bullet (C + D)$	$(B + C) \bullet (B + D) = B + (C \bullet D)$	Επιμεριστικότητα
T9	$B \bullet (B + C) = B$	$B + (B \bullet C) = B$	Κάλυψη
T10	$(B \bullet C) + (B \bullet \overline{C}) = B$	$(B + C) \bullet (B + \overline{C}) = B$	Συνδυασμός
T11	$(B \bullet C) + (\overline{B} \bullet D) + (C \bullet D) = B \bullet C + \overline{B} \bullet D$	$(B + C) \bullet (\overline{B} + D) \bullet (C + D) = (B + C) \bullet (\overline{B} + D)$	Ομοφωνία

- Τα θεωρήματα της **κάλυψης** (T9), του **συνδυασμού** (T10), και της **ομοφωνίας** (T11), μας επιτρέπουν να **εξαλείφουμε πλεονάζουσες μεταβλητές**
  - *Βασίζονται κυρίως στα θεωρήματα της επιμεριστικότητας, των συμπληρωμάτων, του ουδέτερου στοιχείου και του κυρίαρχου στοιχείου*

Ελαχιστοποίηση εξισώσεων Boole

- Αναφερόμαστε σε εξισώσεις Boole που είναι σε **μορφή αθροίσματος γινομένων**
  - *Έστω P οποιοδήποτε γινόμενο λεκτικών*
  - *Βασίζομαστε κυρίως στα θεωρήματα της επιμεριστικότητας, των συμπληρωμάτων, του ουδέτερου στοιχείου και του κυρίαρχου στοιχείου*
  - *Χρησιμοποιούμε και τα ακόλουθα θεωρήματα:*
    - Γενίκευση του θεωρήματος της ταυτοδυναμίας T3’
      - $\boldsymbol{P = P + P}$
    - Γενίκευση του θεωρήματος της κάλυψης T9’:
      - $\boldsymbol{P \cdot A + A = P \cdot A + A \cdot 1 = A \cdot (P + 1) = A \cdot 1 = A}$
    - Γενίκευση του θεωρήματος του συνδυασμού T10:
      - $\boldsymbol{P \cdot A + P \cdot \bar{A} = P \cdot (A + \bar{A}) = P \cdot 1 = P}$
    - Χρήση του θεωρήματος της επιμεριστικότητας T8’:
      - $\boldsymbol{P \cdot A + \bar{A} = (P + \bar{A}) \cdot (A + \bar{A}) = (P + \bar{A}) \cdot 1 = P + \bar{A}}$
      - $\boldsymbol{P \cdot \bar{A} + A = (P + A) \cdot (A + \bar{A}) = (P + A) \cdot 1 = P + A}$

Παράδειγμα 2.6

- Να ελαχιστοποιήσετε την εξίσωση Boole: $\overline{\mathbf{A}}\overline{\mathbf{B}}\overline{\mathbf{C}} + \boldsymbol{A}\overline{\mathbf{B}}\overline{\mathbf{C}} + \boldsymbol{A}\overline{\mathbf{B}}\boldsymbol{C}$
- Λύση 1 (ημιτελής) Λύση 2 (πλήρης)

Λύση 1	Λύση 2	
$\overline{\mathbf{A}}\overline{\mathbf{B}}\overline{\mathbf{C}} + \boldsymbol{A}\overline{\mathbf{B}}\overline{\mathbf{C}} + \boldsymbol{A}\overline{\mathbf{B}}\boldsymbol{C}$	$\overline{\mathbf{A}}\overline{\mathbf{B}}\overline{\mathbf{C}} + \boldsymbol{A}\overline{\mathbf{B}}\overline{\mathbf{C}} + \boldsymbol{A}\overline{\mathbf{B}}\overline{\mathbf{C}} + \boldsymbol{A}\overline{\mathbf{B}}\boldsymbol{C}$	(T3’)
$\overline{\mathbf{B}}\overline{\mathbf{C}}(\overline{\mathbf{A}} + \boldsymbol{A}) + \boldsymbol{A}\overline{\mathbf{B}}\boldsymbol{C}$	$\overline{\mathbf{B}}\overline{\mathbf{C}}(\overline{\mathbf{A}} + \boldsymbol{A}) + \mathbf{A}\overline{\mathbf{B}}(\overline{\boldsymbol{C}} + \boldsymbol{C})$	(T8)
$\overline{\mathbf{B}}\overline{\mathbf{C}}(\mathbf{1}) + \boldsymbol{A}\overline{\mathbf{B}}\boldsymbol{C}$	$\overline{\mathbf{B}}\overline{\mathbf{C}}(\mathbf{1}) + \boldsymbol{A}\overline{\mathbf{B}}(\mathbf{1})$	(T5’)
$\overline{\mathbf{B}}\overline{\mathbf{C}} + \boldsymbol{A}\overline{\mathbf{B}}\boldsymbol{C}$	$\overline{\mathbf{B}}\overline{\mathbf{C}} + \boldsymbol{A}\overline{\mathbf{B}}$	(T1)

- Η χρήση ενός **κοινού ελαχιστόρου** σύμφωνα με το θεώρημα της ταυτοδυναμίας (όπως ο $\boldsymbol{A}\overline{\mathbf{B}}\overline{\mathbf{C}}$) δύο ή περισσότερες φορές κατά τη διαδικασία της ελαχιστοποίησης οδηγεί σε πιο ελαχιστοποιημένη εξίσωση Boole σε **μορφή αθροίσματος γινομένων** (λύση 2)
- Η λύση 2 με χρήση του T8 οδηγεί σε πιο ελαχιστοποιημένη εξίσωση Boole σε **μορφή γινομένου αθροισμάτων** (λύση 3)

$\overline{\mathbf{B}}\,(\boldsymbol{A} + \overline{\mathbf{C}})$

Επιλεγμένες ασκήσεις

A	B	C	Y
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	0

(b)

- Άσκηση 2.6
  Ελαχιστοποιήστε την εξίσωση Boole σε κανονική μορφή αθροίσματος γινομένων της Άσκησης 2.2 – (β) (έχει 2 λύσεις σε μορφή αθροίσματος γινομένων)

$$Y = A\overline{C} + \overline{A}C + B\overline{C}$$

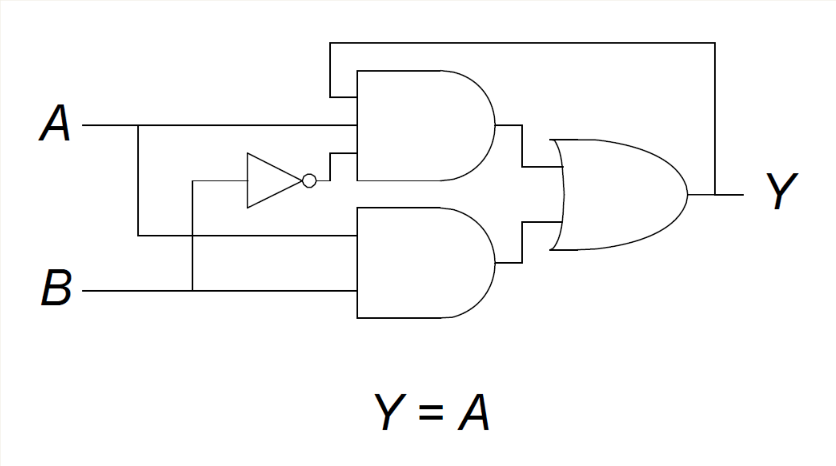
$$Y = A\overline{C} + \overline{A}C + \overline{A}B$$

  - *Ξεκινήστε από την εξίσωση Boole σε κανονική μορφή αθροίσματος γινομένων που προκύπτει από τον Πίνακα Αλήθειας*

$$Y = \overline{A}\overline{B}C + \overline{A}B\overline{C} + \overline{A}BC + A\overline{B}\overline{C} + AB\overline{C}$$

Επιλεγμένες ασκήσεις

- **Άσκηση 2.20**
  Παραθέστε ένα παράδειγμα κυκλώματος το οποίο θα περιέχει μια κυκλική διαδρομή, αλλά θα εξακολουθεί μολαταύτα να είναι συνδυαστικό



Το θεώρημα De Morgan

	Θεώρημα	Ονομασία
T12	$\overline{B_0 \bullet B_1 \bullet B_2 \ldots} = \overline{B_0} + \overline{B_1} + \overline{B_2} \ldots$	DeMorgan

	Δυϊκό Θεώρημα	Ονομασία
T12’	$\overline{B_0 + B_1 + B_2 \ldots} = \overline{B_0} \bullet \overline{B_1} \bullet \overline{B_2} \ldots$	DeMorgan

Το **συμπλήρωμα** του γινομένου

είναι το

άθροισμα των **συμπληρωμάτων**

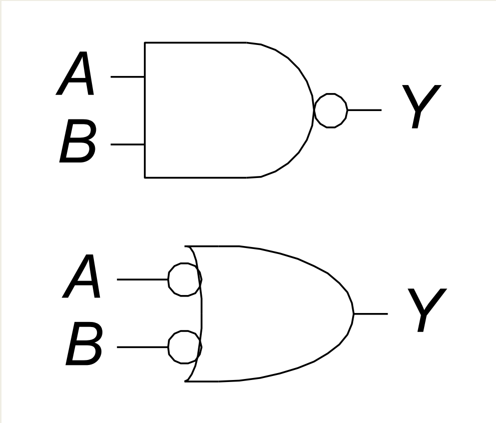
Δυϊκό: Το **συμπλήρωμα** των αθροισμάτων

είναι το

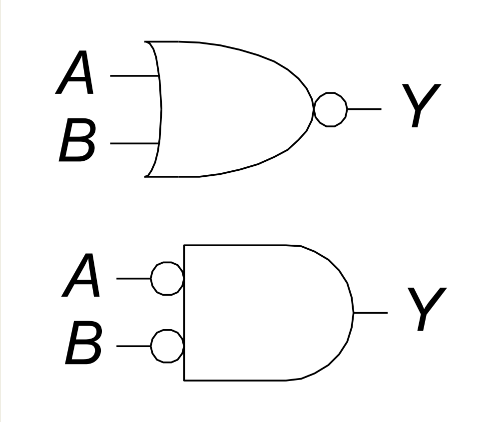
γινόμενο των **συμπληρωμάτων**

Το θεώρημα De Morgan

- $Y = \overline{AB} = \overline{A} + \overline{B}$
  - *μια **πύλη NAND** είναι **ισοδύναμη** με μια πύλη **OR με αντεστραμμένες εισόδους***

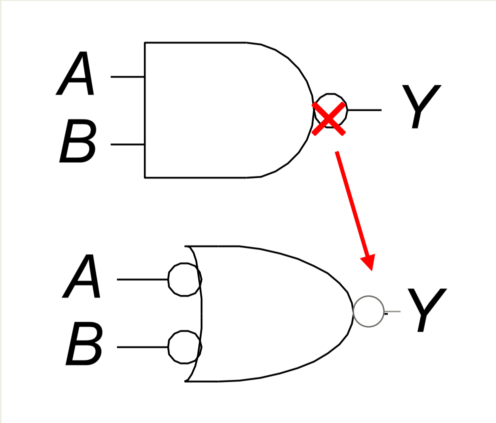


- $Y = \overline{A + B} = \overline{A} \bullet \overline{B}$
  - *μια **πύλη NOR** είναι **ισοδύναμη** με μια πύλη **AND με αντεστραμμένες εισόδους***

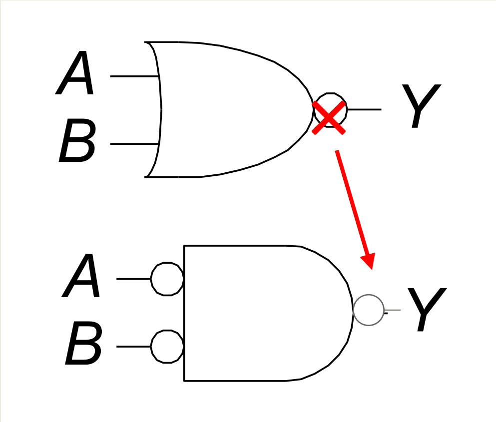


Το θεώρημα De Morgan

- $Y = \overline{\overline{AB}} = \overline{\overline{A} + \overline{B}} = AB$
  - *μια **πύλη AND** είναι **ισοδύναμη** με μια **πύλη NOR με αντεστραμμένες εισόδους***



- $Y = \overline{\overline{A + B}} = \overline{\overline{A} \bullet \overline{B}} = A + B$
  - *μια **πύλη OR** είναι **ισοδύναμη** με μια πύλη **NAND με αντεστραμμένες εισόδους***



Καθολικότητα πυλών

- Ερώτηση 2.4
  Μια πύλη ή ένα σύνολο πυλών είναι **καθολική(-ό)** αν μπορεί να χρησιμοποιηθεί για την υλοποίηση οποιασδήποτε συνάρτησης Boole
  - *Για παράδειγμα, **το σύνολο πυλών AND, OR, NOT είναι καθολικό***

  (α) Είναι μια πύλη AND καθολική από μόνη της; ΟΧΙ

  (β) Είναι το σύνολο πυλών OR και NOT καθολικό; ΝΑΙ

  (γ) Είναι μια πύλη NAND καθολική από μόνη της; ΝΑΙ

  Γιατί ΝΑΙ ή γιατί ΟΧΙ;

  Εξετάζουμε εάν από το διαθέσιμο σύνολο πυλών μπορούμε να δημιουργήσουμε τις πύλες AND, OR, NOT

  Σημείωση: Μία πύλη NAND ή μία πύλη NOR με βραχυκυκλωμένες εισόδους υλοποιεί μία πύλη NOT

Το θεώρημα De Morgan

- Ποια είναι η εξίσωση Boole σε μορφή αθροίσματος γινομένων;
  - *Εφαρμόζουμε το θεώρημα De Morgan όσες φορές χρειαστεί*

Παράδειγμα 1

$$Y = \overline{(A+\overline{BD})\overline{C}}$$

$$= \overline{(A+\overline{BD})} + \overline{\overline{C}}$$

$$= (\overline{A} \bullet \overline{\overline{(BD)}}) + C$$

$$= (\overline{A} \bullet (BD)) + C$$

$$= \overline{A}BD + C$$

Παράδειγμα 2

$$Y = \overline{(\overline{AC}E+\overline{D}) + B}$$

$$= \overline{(\overline{AC}E+\overline{D})} \bullet \overline{B}$$

$$= (\overline{\overline{AC}E} \bullet \overline{\overline{D}}) \bullet \overline{B}$$

$$= ((\overline{\overline{AC}}+\overline{E}) \bullet D) \bullet \overline{B}$$

$$= ((AC+\overline{E}) \bullet D) \bullet \overline{B}$$

$$= (ACD + D\overline{E}) \bullet \overline{B}$$

$$= A\overline{B}CD + \overline{B}D\overline{E}$$

Επιλεγμένες ασκήσεις

- Άσκηση 2.13
  Ελαχιστοποιήστε την παρακάτω εξίσωση Boole χρησιμοποιώντας θεωρήματα της άλγεβρας Boole
  - *Να ονοματίσετε τα θεωρήματα που χρησιμοποιείτε*

$$Y = \overline{A}\,\overline{B} + \overline{A}B\overline{C} + (\overline{A + \overline{C}})$$

  - *Λύση*

$$Y = \overline{A}$$

Οφέλη ελαχιστοποίησης

- Αφού η τελική εξίσωση είναι λογικά ισοδύναμη με την αρχική γιατί είναι τόσο σημαντική η διαδικασία της ελαχιστοποίησης;
  - *Η ελαχιστοποίηση μειώνει το πλήθος των πυλών που χρησιμοποιούνται για τη φυσική υλοποίηση της εξίσωσης (συνάρτησης) Boole, με αποτέλεσμα η τελική υλοποίηση να έχει:*
    - πιο μικρό μέγεθος και κόστος,
    - μικρότερη κατανάλωση ισχύος, και
    - πιθανώς αυξημένη ταχύτητα.

Συμπληρωματική συνάρτηση $\overline{Y}$

- Για κάθε **συνάρτηση Y** ορίζεται η **συμπληρωματική της συνάρτηση $\overline{Y}$**
  - *Σε κάθε σειρά του πίνακα αλήθειας που η **συνάρτηση Y** έχει την **τιμή 1**, η **συμπληρωματική συνάρτηση $\overline{Y}$** έχει την **τιμή 0***
  - *Σε κάθε σειρά του πίνακα αλήθειας που η **συνάρτηση Y** έχει την **τιμή 0**, η **συμπληρωματική συνάρτηση $\overline{Y}$** έχει την **τιμή 1***
- Παράδειγμα

A	B	Y	$\overline{Y}$
0	0	0	1
0	1	0	1
1	0	1	0
1	1	1	0

Κανονική μορφή αθροίσματος γινομένων

- Μπορούμε να βρούμε την εξίσωση Boole από οποιονδήποτε πίνακα αληθείας **αθροίζοντας (πράξη OR) καθέναν από τους ελαχιστόρους (πράξη AND)** για τους οποίους η έξοδος έχει τιμή **‘1’ (TRUE)**

A	B	Y	$\overline{Y}$	Ελαχιστόρος
0	0	0	1	$\overline{A}\ \overline{B}$
0	1	0	1	$\overline{A}\ B$
1	0	1	0	$A\ \overline{B}$
1	1	1	0	$A\ B$

$$\mathbf{Y = F(A, B) = A\overline{B} + AB = \Sigma(m_2, m_3) = \Sigma(2, 3)}$$

$$\mathbf{\overline{Y} = F(A, B) = \overline{A}\overline{B} + \overline{A}B = \Sigma(m_0, m_1) = \Sigma(0, 1)}$$

Εύρεση της μορφής γινομένου αθροισμάτων*

- Χρησιμοποιώντας το θεώρημα De Morgan, βρείτε την κανονική μορφή γινομένου αθροισμάτων της **συνάρτησης $\boldsymbol{Y}$** από τη μορφή αθροίσματος γινομένων της **συμπληρωματικής της συνάρτησης $\boldsymbol{\overline{Y}}$**
  - *Εφαρμόζουμε δύο φορές το θεώρημα De Morgan*

$$\boldsymbol{\overline{Y} = \overline{A}\overline{B} + \overline{A}B}$$

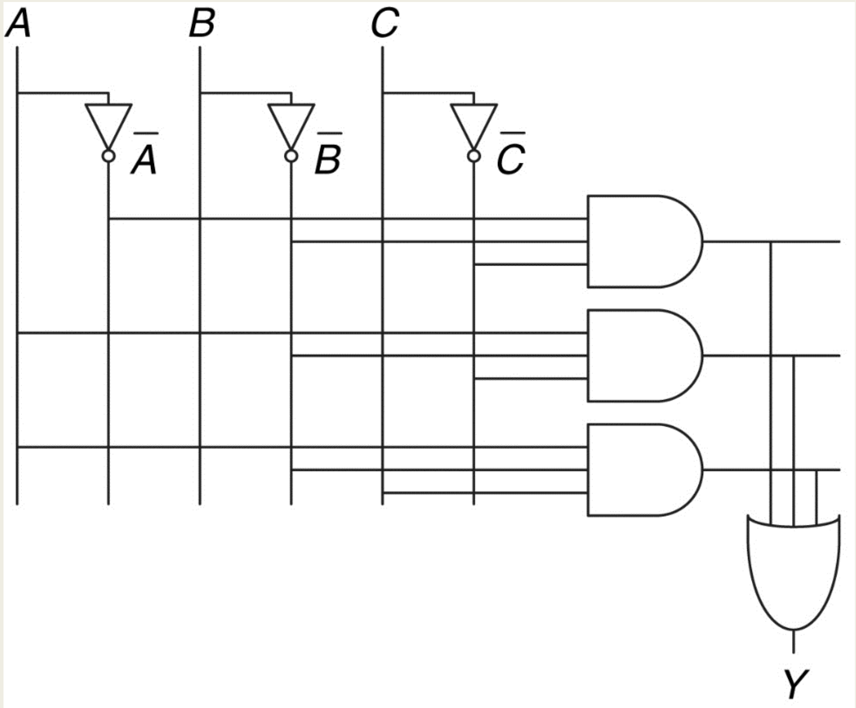
$$\boldsymbol{\overline{\overline{Y}} = Y = \overline{\overline{A}\overline{B} + \overline{A}B} = (\overline{\overline{A}\overline{B}})(\overline{\overline{A}B}) = (A+B)(A+\overline{B})}$$

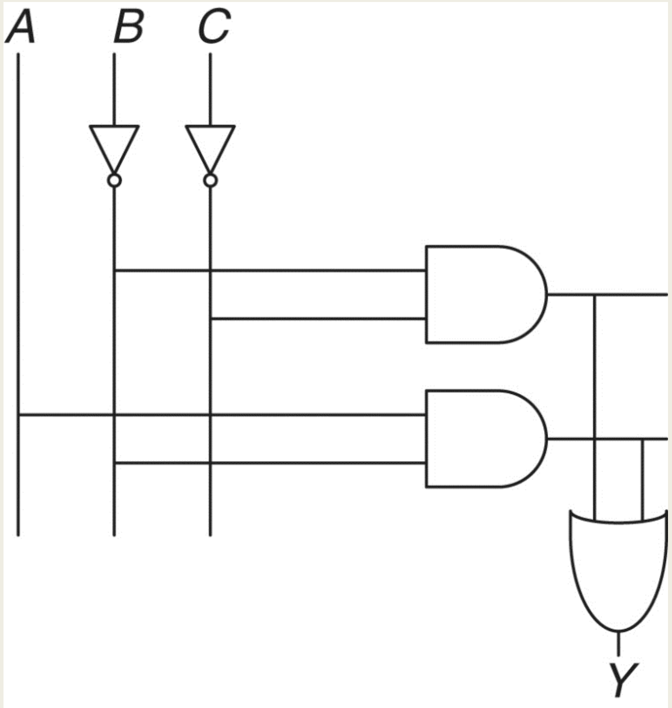
  - *Ισχύει ο εξής κανόνας:*
    - Η συμπληρωματική μίας συνάρτησης προκύπτει, εάν στη συνάρτηση:
      - ***οι μεταβλητές αλλάξουν από $\boldsymbol{X}$ σε $\boldsymbol{\overline{X}}$ και από $\boldsymbol{\overline{X}}$ σε $\boldsymbol{X}$***
      - ***οι τελεστές • (AND) και + (OR) εναλλαχθούν χωρίς να αλλάξει η ιεραρχία των πράξεων***

*Παράδειγμα 2.4

Σχηματικά διαγράμματα

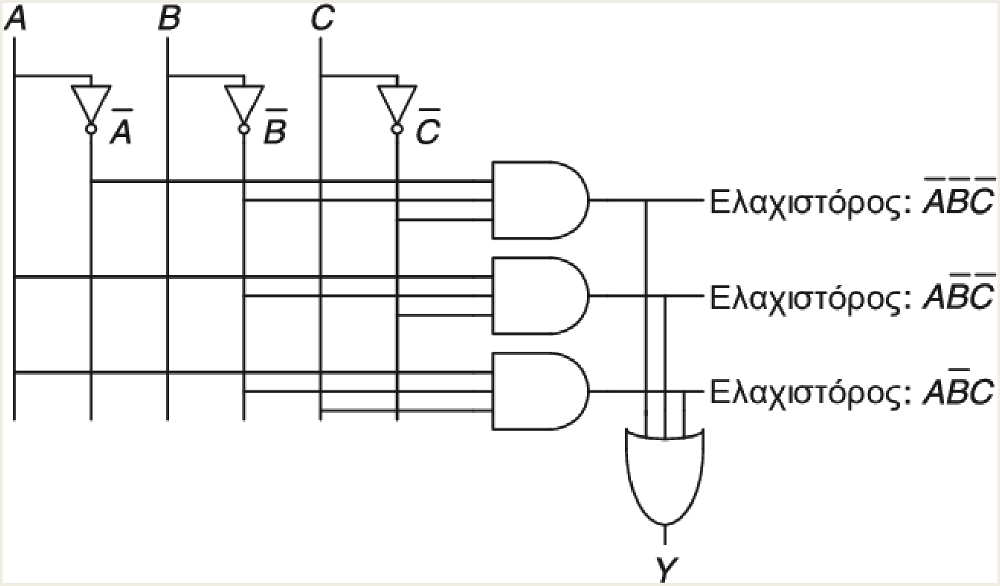
- Το **σχηματικό διάγραμμα** είναι ένα διάγραμμα του ψηφιακού κυκλώματος στο οποίο φαίνονται τα στοιχεία (π.χ. πύλες) και τα σύρματα που τα συνδέουν
  - *Παράδειγμα:* $\boldsymbol{Y = \overline{A}\overline{B}\overline{C} + A\overline{B}\overline{C} + A\overline{B}C}$ ***(κανονική μορφή)***
  - $\boldsymbol{Y = \overline{B}\overline{C} + A\overline{B}}$ ***(ελαχιστοποιημένη – πρότυπη μορφή)***
    - Υλοποίηση με λιγότερες πύλες και πιο γρήγορη





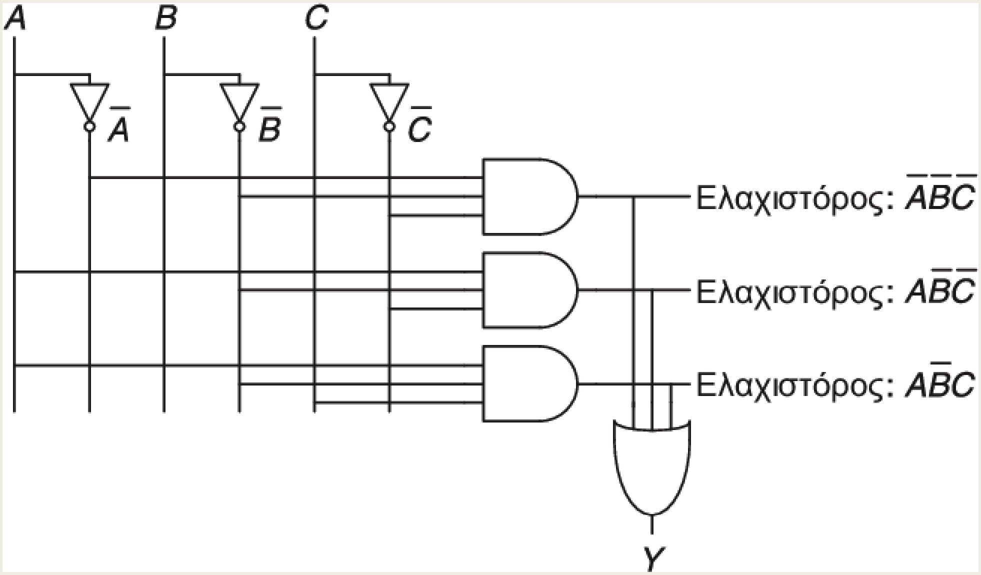
Σχηματικά διαγράμματα: κανόνες

- Οι είσοδοι βρίσκονται στην αριστερή (ή την πάνω) πλευρά
- Οι έξοδοι βρίσκονται στη δεξιά (ή την κάτω) πλευρά
- Οποτεδήποτε είναι εφικτό, οι πύλες θα πρέπει να έχουν κατεύθυνση **από τα αριστερά προς τα δεξιά**
- Είναι καλύτερο να χρησιμοποιούνται **σύρματα σε ευθεία** αντί για σύρματα με πολλές γωνίες.



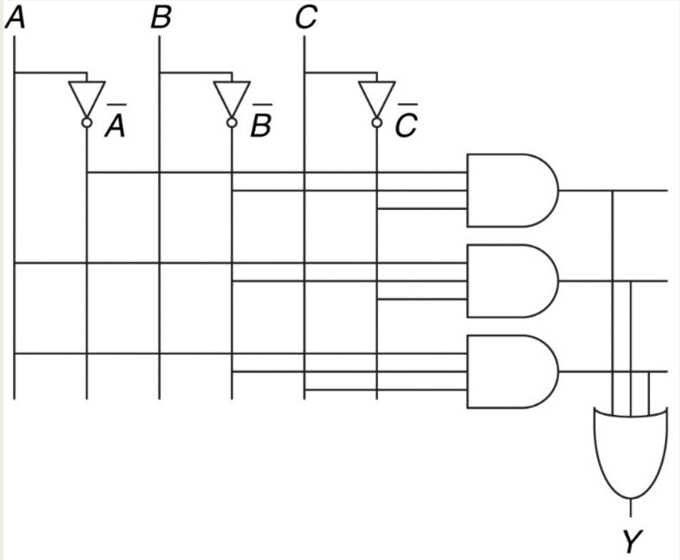
Σχηματικά διαγράμματα: κανόνες

- Τα σύρματα συνδέονται πάντα σε **επαφές Τ**
- Μια **τελεία** στο σημείο τομής συρμάτων υποδεικνύει ότι **υπάρχει σύνδεση μεταξύ των συρμάτων**
- Σύρματα τα οποία τέμνονται **χωρίς τελεία** δεν συνδέονται μεταξύ τους



Σχηματικά διαγράμματα: κανόνες

- Τα βήματα για την σχεδίαση σχηματικών διαγραμμάτων για εξισώσεις Boole σε μορφή αθροίσματος γινομένων:
  - *Πρώτα σχεδιάζουμε στήλες για τις εισόδους*
  - *Τοποθετούμε αντιστροφείς σε γειτονικές στήλες για να παρέχουμε τις συμπληρωματικές εισόδους, αν είναι απαραίτητο*
  - *Σχεδιάζουμε γραμμές με πύλες AND για καθέναν από τους ελαχιστόρους*
  - *Έπειτα, για κάθε έξοδο, σχεδιάζουμε μια πύλη OR η οποία συνδέεται με τις πύλες AND (τους ελαχιστόρους) που σχετίζονται με τη συγκεκριμένη έξοδο*



- Αυτό το στυλ ονομάζεται ***προγραμματιζόμενη λογική διάταξη*** (programmable logic array, PLA) επειδή οι αντιστροφείς, οι πύλες AND και οι πύλες OR διατάσσονται με συστηματικό τρόπο

Παράδειγμα 2.7 Κυκλώματα με πολλές εξόδους

- **Ο Πρύτανης**, ο **Πρόεδρος ενός Τμήματος**, ένας **βοηθός διδασκαλίας** και ο **υπεύθυνος κοινωνικών εκδηλώσεων** ενός Πανεπιστημίου χρησιμοποιούν ενίοτε το κεντρικό αμφιθέατρο του ιδρύματος με σκοπό την οργάνωση εκδηλώσεων
- Αυτό έχει σαν αποτέλεσμα μερικές φορές, παραπάνω από ένα άτομα να έχουν κάνει κράτηση του κεντρικού αμφιθέατρου την ίδια στιγμή
- Το σύστημα έχει τέσσερις εισόδους A_3 - A_0 και τέσσερις εξόδους Y_3 - Y_0
- Κάθε χρήστης ενεργοποιεί την είσοδό του όταν αιτείται τη χρήση του αμφιθεάτρου
  – ***A_3 = 1 (Πρύτανης), A_2 = 1 (Πρόεδρος ενός Τμήματος), A_1 = 1 (βοηθός διδασκαλίας) και A_0 = 1 (υπεύθυνος εκδηλώσεων)***
- Το σύστημα ενεργοποιεί το πολύ μία έξοδο, παραχωρώντας το αμφιθέατρο στον χρήστη με την υψηλότερη προτεραιότητα
  – ***Y_3 = 1 (Πρύτανης), Y_2 = 1 (Πρόεδρος ενός Τμήματος), Y_1 = 1 (βοηθός διδασκαλίας) και Y_0 = 1 (υπεύθυνος εκδηλώσεων)***
- Συμπληρώστε τον πίνακα αληθείας και γράψτε τις εξισώσεις Boole για το σύστημα. Σχεδιάστε ένα κύκλωμα γι’ αυτή τη συνάρτηση

Παράδειγμα 2.7

Η συγκεκριμένη συνάρτηση ονομάζεται *κύκλωμα προτεραιότητας* (priority circuit) με τέσσερις εισόδους

A_3	A_2	A_1	A_0	Y_3	Y_2	Y_1	Y_0
0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	1
0	0	1	0	0	0	1	0
0	0	1	1	0	0	1	0
0	1	0	0	0	1	0	0
0	1	0	1	0	1	0	0
0	1	1	0	0	1	0	0
0	1	1	1	0	1	0	0
1	0	0	0	1	0	0	0
1	0	0	1	1	0	0	0
1	0	1	0	1	0	0	0
1	0	1	1	1	0	0	0
1	1	0	0	1	0	0	0
1	1	0	1	1	0	0	0
1	1	1	0	1	0	0	0
1	1	1	1	1	0	0	0

Παράδειγμα 2.7: κύκλωμα προτεραιότητας

A_3	A_2	A_1	A_0	Y_3	Y_2	Y_1	Y_0
0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	1
0	0	1	0	0	0	1	0
0	0	1	1	0	0	1	0
0	1	0	0	0	1	0	0
0	1	0	1	0	1	0	0
0	1	1	0	0	1	0	0
0	1	1	1	0	1	0	0
1	0	0	0	1	0	0	0
1	0	0	1	1	0	0	0
1	0	1	0	1	0	0	0
1	0	1	1	1	0	0	0
1	1	0	0	1	0	0	0
1	1	0	1	1	0	0	0
1	1	1	0	1	0	0	0
1	1	1	1	1	0	0	0

- Το Y_3 έχει την **μεγαλύτερη προτεραιότητα**, άρα όταν αυτό είναι TRUE τότε οι τιμές των υπολοίπων μεταβλητών εισόδου είναι αδιάφορες
- Το Y_2 έχει τιμή TRUE **αν είναι ενεργοποιημένο το A_2** και δεν είναι ενεργοποιημένο το A_3. Οι τιμές των A_1 και A_0 είναι αδιάφορες.
- Το Y_1 έχει τιμή TRUE **αν είναι ενεργοποιημένο το A_1** και δεν είναι ενεργοποιημένη καμία από τις εισόδους A_3 και A_2. Η τιμή του A_0 είναι αδιάφορη.
- Το Y_0 έχει τιμή TRUE **αν είναι ενεργοποιημένο το A_0** και δεν είναι ενεργοποιημένη κάποια άλλη είσοδος

Παράδειγμα 2.7: κύκλωμα προτεραιότητας

A_3	A_2	A_1	A_0	Y_3	Y_2	Y_1	Y_0
0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	1
0	0	1	0	0	0	1	0
0	0	1	1	0	0	1	0
0	1	0	0	0	1	0	0
0	1	0	1	0	1	0	0
0	1	1	0	0	1	0	0
0	1	1	1	0	1	0	0
1	0	0	0	1	0	0	0
1	0	0	1	1	0	0	0
1	0	1	0	1	0	0	0
1	0	1	1	1	0	0	0
1	1	0	0	1	0	0	0
1	1	0	1	1	0	0	0
1	1	1	0	1	0	0	0
1	1	1	1	1	0	0	0

$$Y_3 = A_3$$

$$Y_2 = \overline{A_3}\,A_2$$

$$Y_1 = \overline{A_3}\,\overline{A_2}\,A_1$$

$$Y_0 = \overline{A_3}\,\overline{A_2}\,\overline{A_1}\,A_0$$

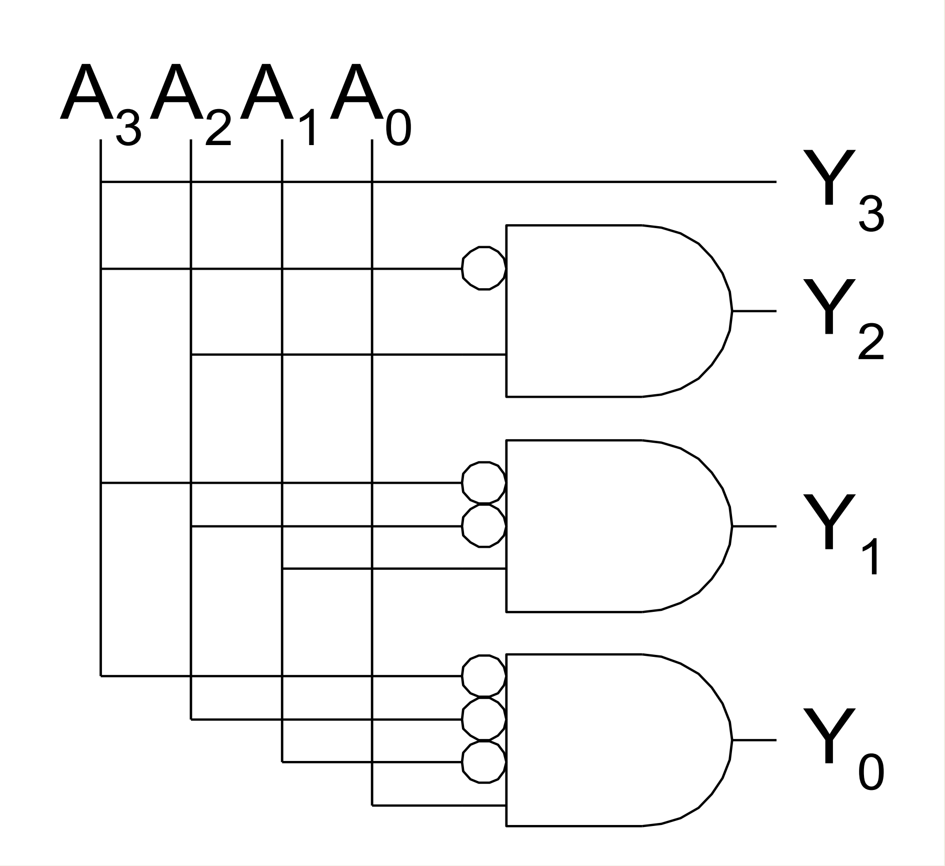
Παράδειγμα 2.7: κύκλωμα προτεραιότητας

$$Y_3 = A_3$$

$$Y_2 = \overline{A_3}\,A_2$$

$$Y_1 = \overline{A_3}\,\overline{A_2}\,A_1$$

$$Y_0 = \overline{A_3}\,\overline{A_2}\,\overline{A_1}\,A_0$$



Χρήση του «Χ» στους πίνακες αλήθειας

- Σε έναν πίνακα αλήθειας η χρήση του **συμβόλου Χ** σημαίνει ότι υπάρχουν **αδιάφορες τιμές**.
  - *Το Χ μπορεί να είναι είτε 0, είτε 1 (όχι και τα δύο ταυτόχρονα)*
- Το Χ στις εισόδους απλοποιεί τον πίνακα αλήθειας (μειώνει τις γραμμές)

A_3	A_2	A_1	A_0	Y_3	Y_2	Y_1	Y_0
0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	1
0	0	1	X	0	0	1	0
0	1	X	X	0	1	0	0
1	X	X	X	1	0	0	0

- Το Y_3 έχει την **μεγαλύτερη προτεραιότητα**, άρα όταν αυτό είναι TRUE τότε οι τιμές των υπολοίπων μεταβλητών εισόδου είναι αδιάφορες
- Το Y_2 έχει τιμή TRUE **αν είναι ενεργοποιημένο το A_2** και δεν είναι ενεργοποιημένο το A_3. Οι τιμές των A_1 και A_0 είναι αδιάφορες.
- Το Y_1 έχει τιμή TRUE **αν είναι ενεργοποιημένο το A_1** και δεν είναι ενεργοποιημένη καμία από τις εισόδους A_3 και A_2. Η τιμή του A_0 είναι αδιάφορη.
- Το Y_0 έχει τιμή TRUE **αν είναι ενεργοποιημένο το A_0** και δεν είναι ενεργοποιημένη κάποια άλλη είσοδος

Κωδικοποιητής προτεραιότητας

- Άσκηση 2.36
  Ένας ***κωδικοποιητής προτεραιότητας*** (priority encoder) διαθέτει **2^N εισόδους**, **διατεταγμένες από το 2^N-1 έως το 0**
  - *Παράγει μια **δυαδική έξοδο των N bit** που υποδεικνύει το περισσότερο σημαντικό bit της εισόδου το οποίο έχει την τιμή ‘1’ (TRUE), ή την έξοδο μηδέν αν καμία από τις εισόδους δεν έχει την τιμή ‘1’ (TRUE)*
  - *Παράγει επίσης το **σήμα εγκυρότητας VALID** (validation) το οποίο έχει την τιμή ‘1’ (TRUE) αν τουλάχιστον μία από τις εισόδους έχει την τιμή ‘1’ (TRUE)*
  - *Σχεδιάστε τον πίνακα αληθείας ενός κωδικοποιητή προτεραιότητας οκτώ εισόδων, με εισόδους $A_{7:0}$ και εξόδους $Y_{2:0}$ και το σήμα VALID*
  - *Παρέχετε μια ελαχιστοποιημένη εξίσωση Boole για κάθε έξοδο*

Κωδικοποιητής προτεραιότητας

- Πίνακας αληθείας ενός κωδικοποιητή προτεραιότητας οκτώ εισόδων, με εισόδους $A_{7:0}$ και εξόδους $Y_{2:0}$ και VALID

A_7	A_6	A_5	A_4	A_3	A_2	A_1	A_0	Y_2	Y_1	Y_0	*VALID*
0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	1	0	0	0	1
0	0	0	0	0	0	1	X	0	0	1	1
0	0	0	0	0	1	X	X	0	1	0	1
0	0	0	0	1	X	X	X	0	1	1	1
0	0	0	1	X	X	X	X	1	0	0	1
0	0	1	X	X	X	X	X	1	0	1	1
0	1	X	X	X	X	X	X	1	1	0	1
1	X	X	X	X	X	X	X	1	1	1	1

- Το **σήμα VALID** στην έξοδο ενός κυκλώματος δηλώνει ότι η έξοδος δεδομένων αυτού του κυκλώματος έχει έγκυρα δεδομένα
  - *Η έξοδος δεδομένων $Y_{2:0}$ είναι έγκυρη όταν υποδεικνύει το MSB της εισόδου $A_{7:0}$ το οποίο έχει την τιμή 1*

Κωδικοποιητής προτεραιότητας

- Παρέχετε μια ελαχιστοποιημένη εξίσωση Boole για κάθε έξοδο

$$Y_2 = A_7 + A_6 + A_5 + A_4$$

$$Y_1 = A_7 + A_6 + \overline{A_5}\,\overline{A_4}A_3 + \overline{A_5}\,\overline{A_4}A_2$$

$$Y_0 = A_7 + \overline{A_6}A_5 + \overline{A_6}\,\overline{A_4}A_3 + \overline{A_6}\,\overline{A_4}\,\overline{A_2}A_1$$

- Για παράδειγμα στην έξοδο Y_0 ο όρος $\overline{A_6}\,\overline{A_4}\,A_3$ εξασφαλίζει ότι το Y_0 έχει την τιμή 1 όταν $\mathrm{A}_6 = 0$, $\mathrm{A}_4 = 0$ και $\mathrm{A}_3 = 1$
  - *Σημειώστε ότι το Y_0 έχει την τιμή 0 όταν $A_6 = 1$ ή $A_4 = 1$ ενώ το Y_0 έχει την τιμή 1 όταν $A_7 = 1$ ή $A_5 = 1$ και για αυτό τον λόγο δεν εμφανίζονται τα A_7 και A_5 στον συγκεκριμένο όρο*

$$VALID = A_7 + A_6 + A_5 + A_4 + A_3 + A_2 + A_1 + A_0$$

Πολυεπίπεδη Συνδυαστική Λογική

- Η λογική είτε σε μορφή αθροίσματος γινομένων είτε σε μορφή γινομένου αθροισμάτων ονομάζεται **λογική δύο επιπέδων** (two-level logic) επειδή αποτελείται από λεκτικά συνδεδεμένα σε **ένα επίπεδο πυλών AND** το οποίο συνδέεται με **ένα επίπεδο πυλών OR**
  - *οι αντιστροφείς στην είσοδο αγνοούνται στον υπολογισμό των επιπέδων (ενσωματώνονται στις πύλες)*
- Συχνά οι σχεδιαστές κατασκευάζουν κυκλώματα με περισσότερα από δύο επίπεδα λογικών πυλών
- Αυτά τα πολυεπίπεδα συνδυαστικά κυκλώματα στοχεύουν στο να χρησιμοποιούν **λιγότερο υλικό** από τα αντίστοιχα κυκλώματα δύο επιπέδων (να έχουν μικρότερο κόστος σε υλικό)
  - *Σε πολλές περιπτώσεις **μειώνεται η ταχύτητα** του κυκλώματος*

Πολυεπίπεδη Συνδυαστική Λογική

- **Εύρεση κοινών μεταβλητών**
- Παράδειγμα: Η διεπίπεδη απλοποιημένη συνάρτηση f
  - **$f = ADF+AEF+BDF+BEF+CDF+CEF+G$**
- Απλοποιείται με την εφαρμογή της μεθόδου της **εύρεσης κοινών μεταβλητών**
  - *Εφαρμογή του θεωρήματος της επιμεριστικότητας (T8) όσες φορές απαιτείται*

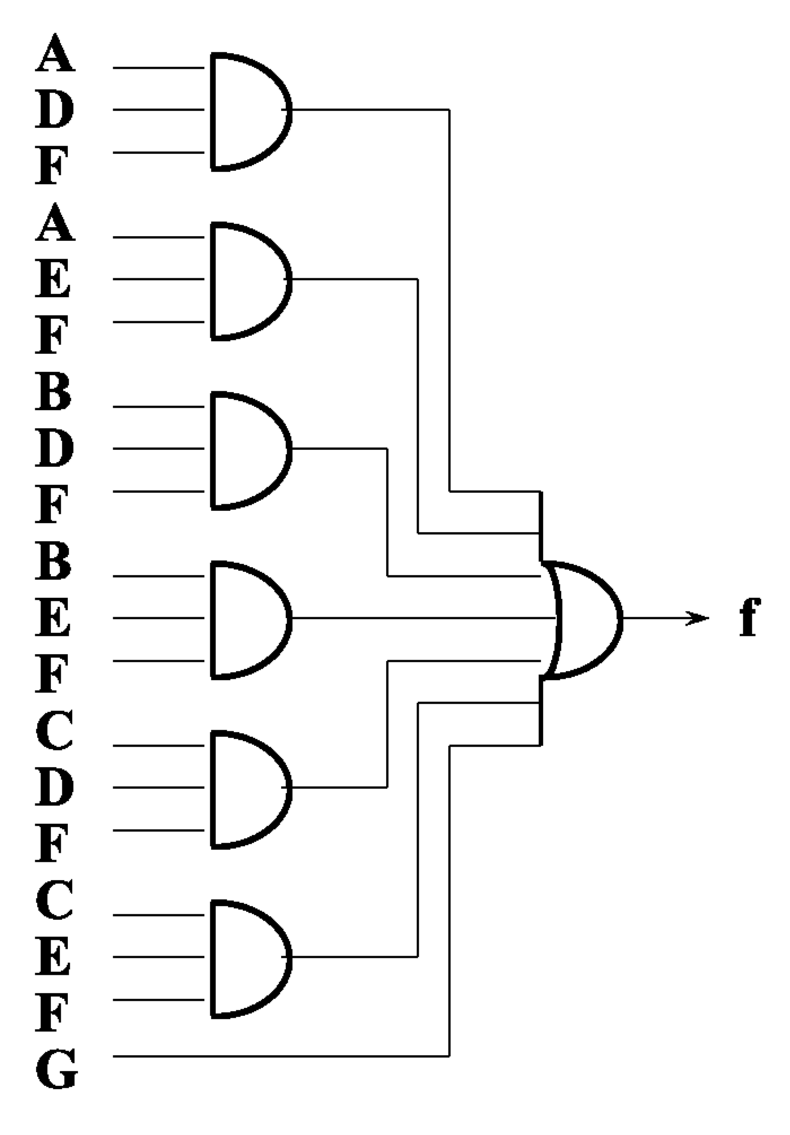
$$f = (AD+AE+BD+BE+CD+CE)F+G$$

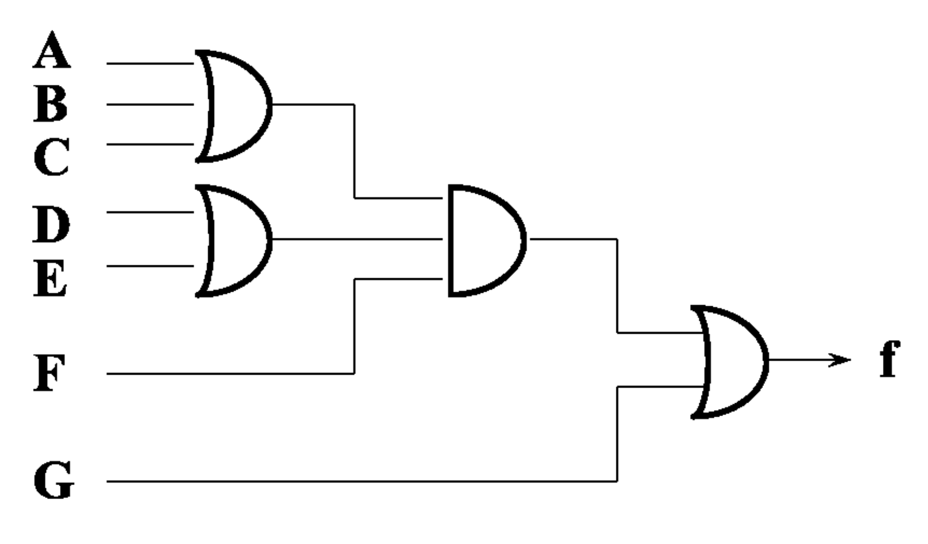
$$= [(A+B+C)D+(A+B+C)E]F+G$$

$$= (A+B+C)(D+E)F+G$$

Πολυεπίπεδη Συνδυαστική Λογική

- Εύρεση κοινών μεταβλητών





- Στη συγκεκριμένη περίπτωση η πολυεπίπεδη υλοποίηση της συνάρτησης f **μικρότερου κόστους σε υλικό** είναι και **πιο γρήγορη**
  - *Η πύλη OR των 7 εισόδων υλοποιείται σε δύο επίπεδα με τρεις πύλες OR των 3 εισόδων*

Μείωση υλικού με τη χρήση πυλών XOR

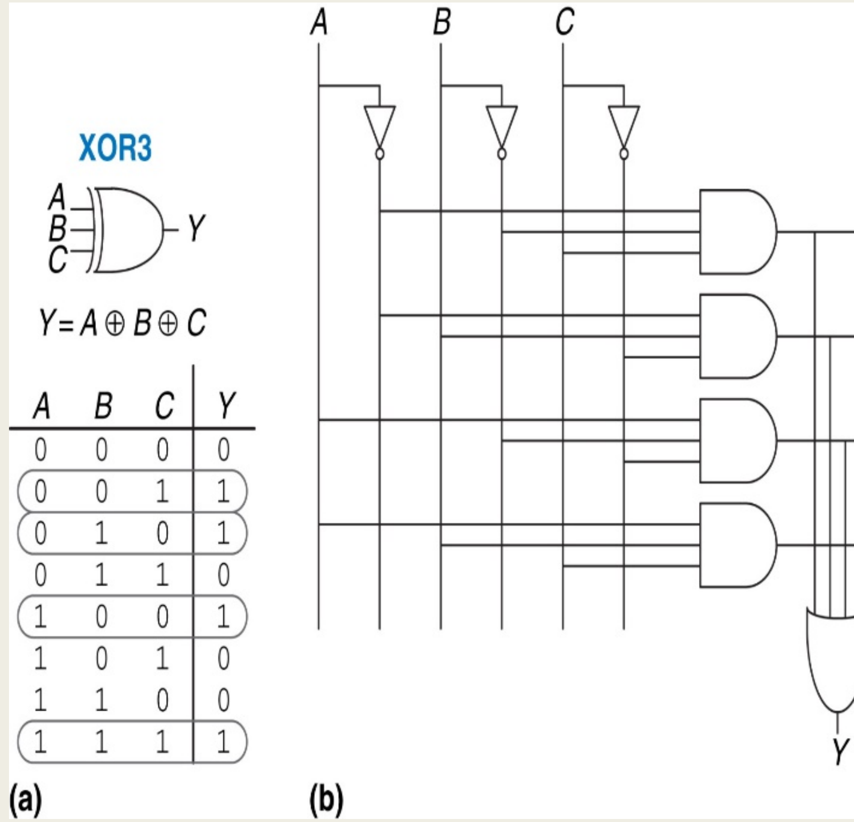
- Παράδειγμα: **Πύλη XOR με 3 εισόδους**
- Η εξίσωση Boole στην απλοποιημένη μορφή αθροίσματος γινομένων δύο επιπέδων είναι:

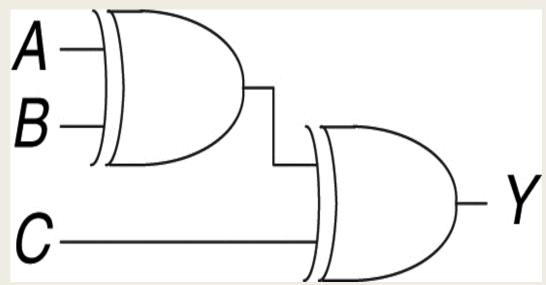
$$\Upsilon = \overline{A}\,\overline{B}C + \overline{A}B\overline{C} + A\overline{B}\,\overline{C} + ABC$$

- Εάν προχωρήσουμε την ελαχιστο-ποίηση, ώστε να χρησιμοποιήσουμε την πύλη XOR έχουμε:

$$\begin{aligned}\Upsilon &= \overline{A}\,(\overline{B}C + B\overline{C}) + A\,(\overline{B}\,\overline{C} + BC) \\ &= \overline{A}\,(B \oplus C) + A\,(\overline{B \oplus C}) \\ &= A \oplus B \oplus C = (A \oplus B) \oplus C\end{aligned}$$

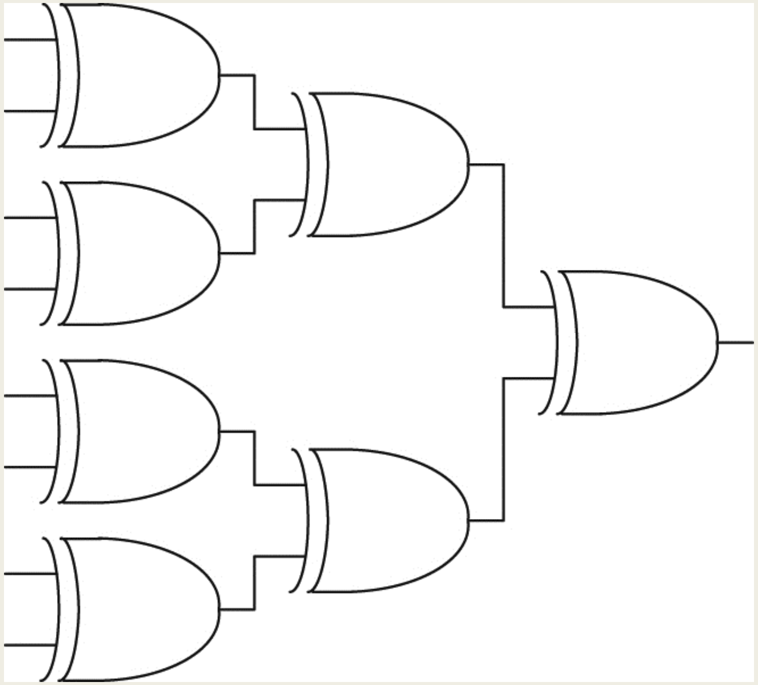
- Άρα μπορεί να υλοποιηθεί με δύο πύλες XOR των 2 εισόδων





Μείωση υλικού με τη χρήση πυλών XOR

- Παράδειγμα: Πύλη XOR με 8 εισόδους
- Η εξίσωση Boole στην απλοποιημένη μορφή αθροίσματος γινομένων δύο επιπέδων θα απαιτούσε:
  - *128 πύλες AND των 8 εισόδων*
  - *1 πύλη OR των 128 εισόδων!*
- Μπορεί όμως να υλοποιηθεί με μόνο:
  - *7 πύλες XOR των 2 εισόδων σε 3 επίπεδα*
    - στο 1ο επίπεδο είναι 4 πύλες, στο 2ο επίπεδο 2 πύλες και στο 3ο επίπεδο 1 πύλη



Κώδικες ανίχνευσης λαθών

- Σε ένα ψηφιακό σύστημα ορίζουμε ως **λάθος**, την αλλαγή τιμής σε ένα ή περισσότερα ψηφία πληροφορίας (0 → 1 ή 1 → 0)
  - *Παράδειγμα απλού λάθους (αλλαγή τιμής σε ένα ψηφίο)*
    - 1010 → 1011, 1010 → 1000
- Για να ανιχνεύσουμε την ύπαρξη λαθών κωδικοποιούμε την πληροφορία που παράγουμε, μεταδίδουμε ή αποθηκεύουμε σε έναν **κώδικα ανίχνευσης λαθών**, ο οποίος αποτελείται από **κωδικές λέξεις** και **μη-κωδικές λέξεις**
- Η ύπαρξη λάθους ανιχνεύεται με την **εμφάνιση μίας μη-κωδικής λέξης** κατά τον έλεγχο της κωδικοποιημένης πληροφορίας
- Ως **απόσταση** μεταξύ δύο κωδικών λέξεων ορίζεται το πλήθος των ψηφίων στα οποία αυτές διαφέρουν
- Ένας κώδικας ανίχνευσης λαθών ανιχνεύει όλα τα **απλά λάθη**, εάν η ελάχιστη απόσταση μεταξύ όλων των πιθανών ζευγών κωδικών λέξεων είναι 2

Κώδικες άρτιας ή περιττής ισοτιμίας

- Προσθέτουμε στην πληροφορία των *N* bit ένα επιπλέον ψηφίο, **το ψηφίο ισοτιμίας P (parity bit)** ώστε το πλήθος των 1 να είναι άρτιο (ή περιττό) σε όλες τις κωδικές λέξεις του κώδικα άρτιας (ή περιττής) ισοτιμίας
- Κωδικές λέξεις του κώδικα άρτιας (ή περιττής) ισοτιμίας

Άρτιας ισοτιμίας

XYZ	P_{even}
000	0
001	1
010	1
100	1
011	0
101	0
110	0
111	1

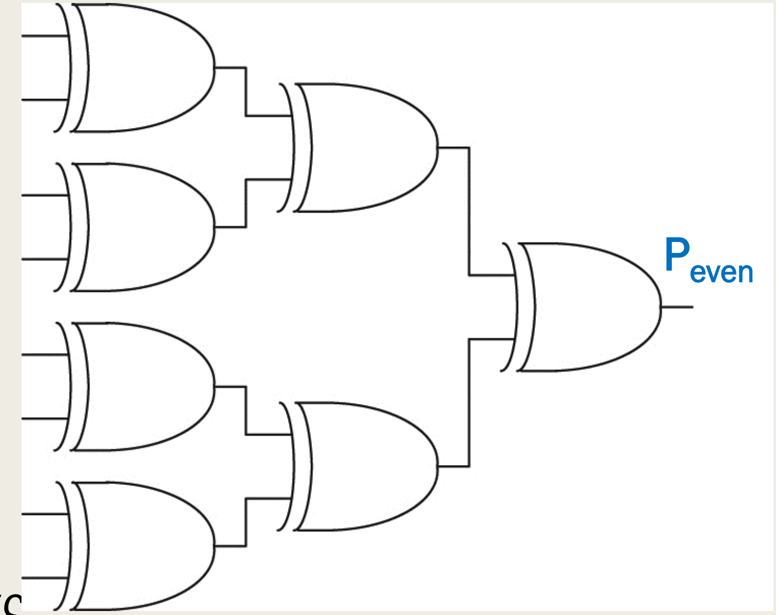
Περιττής ισοτιμίας

XYZ	P_{odd}
000	1
001	0
010	0
100	0
011	1
101	1
110	1
111	0

Οι μη κωδικές λέξεις του κώδικα άρτιας (ή περιττής) ισοτιμίας είναι οι κωδικές λέξεις του κώδικα περιττής (ή άρτιας) ισοτιμίας

Κώδικες άρτιας ή περιττής ισοτιμίας

- Ο κώδικας ισοτιμίας είναι ο πιο απλός κώδικας ανίχνευσης λαθών και ανιχνεύει **απλά λάθη ή ένα περιττό πλήθος λαθών**
  - *Είναι κώδικας απόστασης 2*
- Το κύκλωμα που παράγει το ψηφίο ισοτιμίας για μία **πληροφορία των *N* bit** είναι:
  - *μία **πύλη XOR με N εισόδους για τον κώδικα άρτιας ισοτιμίας** ή*
  - *μία **πύλη XNOR με N εισόδους για τον κώδικα περιττής ισοτιμίας***



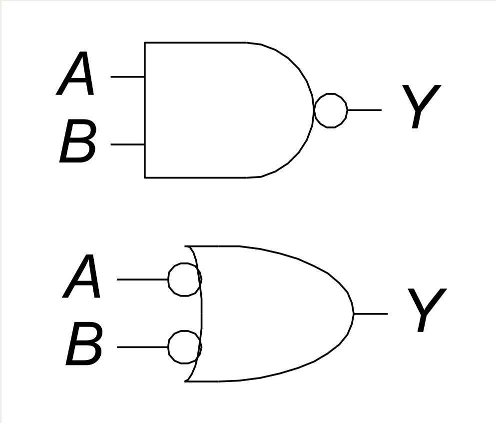
- Στην πράξη η κωδικοποίηση στον κώδικα ισοτιμίας γίνεται συνήθως **ανά byte (8 bit)**
  - *Στο σχήμα φαίνεται η πύλη XOR με 8 εισόδους που παράγει το ψηφίο άρτιας ισοτιμίας P_{even} για ένα byte πληροφορίας*

Ώθηση φυσαλίδας

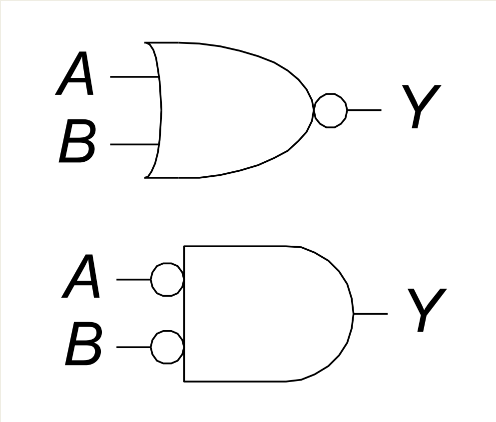
- Οπως προαναφέρθηκε για τα κυκλώματα CMOS προτιμώνται οι πύλες NAND και NOR αντί των πυλών AND και OR.
- Όμως, η ανάγνωση της εξίσωσης με απλή εξέταση για ένα πολυεπίπεδο κύκλωμα με πύλες NAND και NOR ενδέχεται να αποδειχθεί αρκετά δύσκολη.
- Η **ώθηση φυσαλίδων** αποτελεί έναν χρήσιμο τρόπο για να ξανασχεδιάζουμε αυτά τα κυκλώματα έτσι ώστε οι φυσαλίδες να αλληλοακυρώνονται και να καθίσταται πιο εύκολος ο προσδιορισμός της συνάρτησης
- Βασίζεται στο **θεώρημα De Morgan**

Το θεώρημα De Morgan

- $Y = \overline{AB} = \overline{A} + \overline{B}$
  - *μια **πύλη NAND** είναι **ισοδύναμη** με μια πύλη **OR με αντεστραμμένες εισόδους***

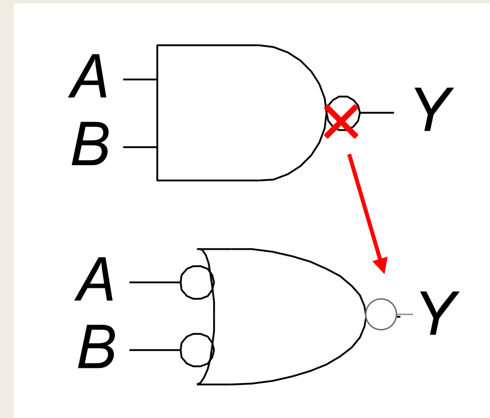


- $Y = \overline{A + B} = \overline{A} \bullet \overline{B}$
  - *μια **πύλη NOR** είναι **ισοδύναμη** με μια πύλη **AND με αντεστραμμένες εισόδους***

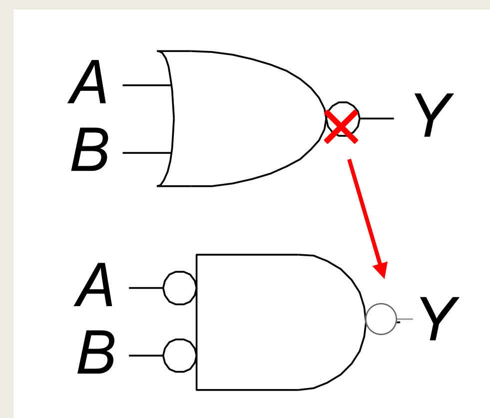


Το θεώρημα De Morgan

- $Y = \overline{\overline{AB}} = \overline{\overline{A} + \overline{B}} = AB$
  - *μια **πύλη AND** είναι **ισοδύναμη** με μια **πύλη NOR με αντεστραμμένες εισόδους***

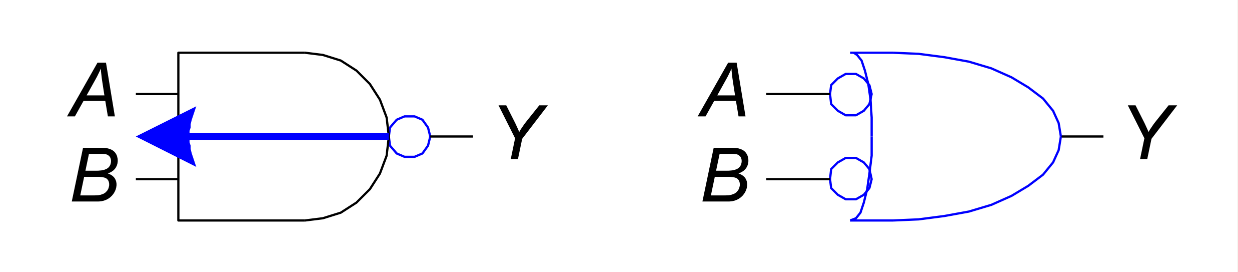


- $Y = \overline{\overline{A + B}} = \overline{\overline{A} \bullet \overline{B}} = A + B$
  - *μια **πύλη OR** είναι **ισοδύναμη** με μια πύλη **NAND με αντεστραμμένες εισόδους***

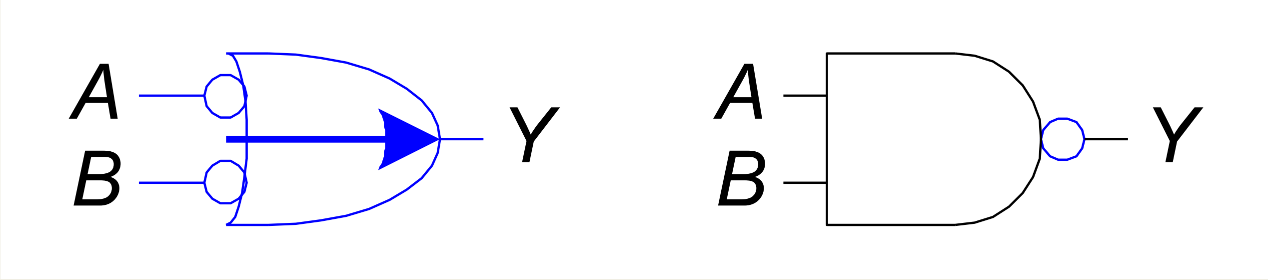


Ώθηση φυσαλίδας

- Ώθηση φυσαλίδων **προς τα πίσω**
  - *αλλάζει το σώμα της πύλης (από AND/OR σε OR/AND)*
  - *φυσαλίδες σε όλες τις εισόδους της πύλης*

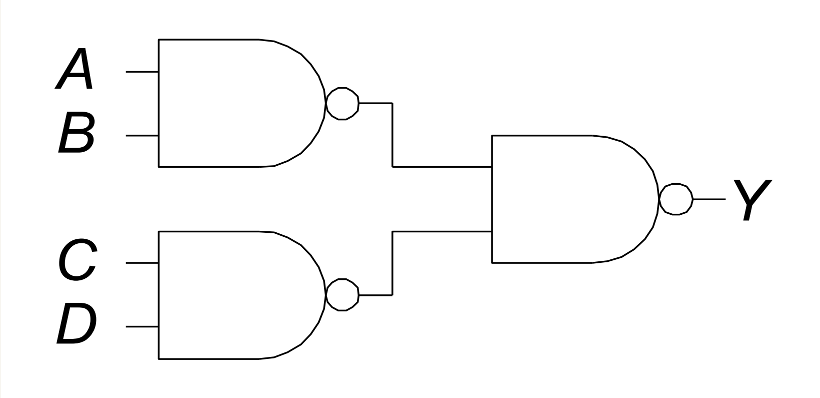


- Ώθηση φυσαλίδων **προς τα μπροστά**
  - *αλλάζει το σώμα της πύλης (από AND/OR σε OR/AND)*
  - *μια φυσαλίδα στην έξοδο της πύλης*



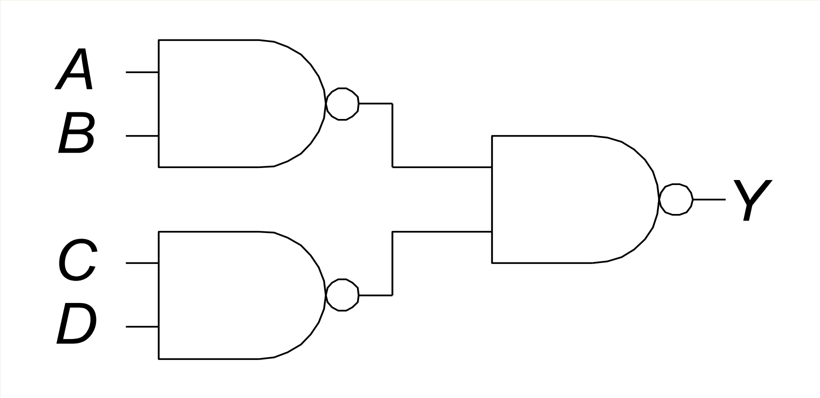
Ώθηση φυσαλίδας

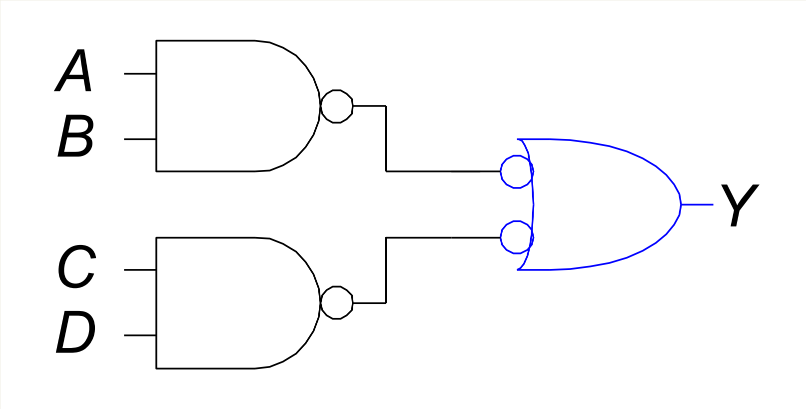
- Ποια είναι η εξίσωση Boole του παρακάτω κυκλώματος?
  - *Οι πύλες NAND και NOR χωρίς τη χρήση των ισοδυνάμων τους δεν βοηθούν στο να βρούμε την εξίσωση*
  - *Απαιτείται η εφαρμογή της ώθησης φυσαλίδας*



Ώθηση φυσαλίδας

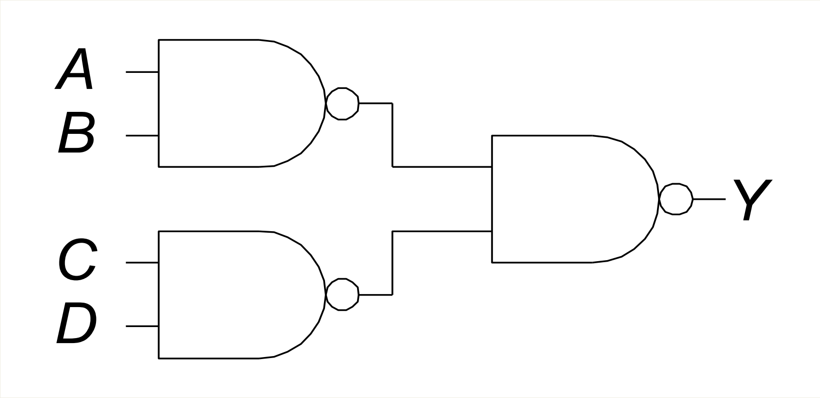
- Ποια είναι η εξίσωση Boole του παρακάτω κυκλώματος?
  - *Οι πύλες NAND και NOR χωρίς τη χρήση των ισοδυνάμων τους δεν βοηθούν στο να βρούμε την εξίσωση*
  - *Απαιτείται η εφαρμογή της ώθησης φυσαλίδας*
  - *Ξεκινάμε από την έξοδο με πορεία προς τα πίσω για να φύγει η αντιστροφή*

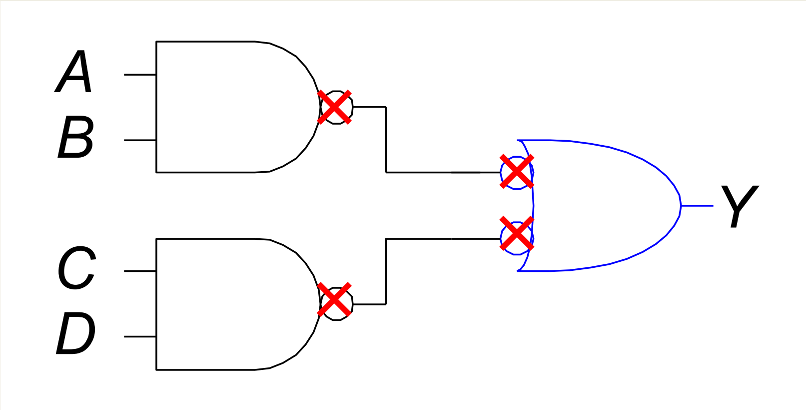




Ώθηση φυσαλίδας

- Ποια είναι η εξίσωση Boole του παρακάτω κυκλώματος?
  - *Οι πύλες NAND και NOR χωρίς τη χρήση των ισοδυνάμων τους δεν βοηθούν στο να βρούμε την εξίσωση*
  - *Απαιτείται η εφαρμογή της ώθησης φυσαλίδας*
  - *Ξεκινάμε από την έξοδο με πορεία προς τα πίσω για να φύγει η αντιστροφή*
  - *Δύο φυσαλίδες στην ίδια διασύνδεση αλληλοακυρώνονται*





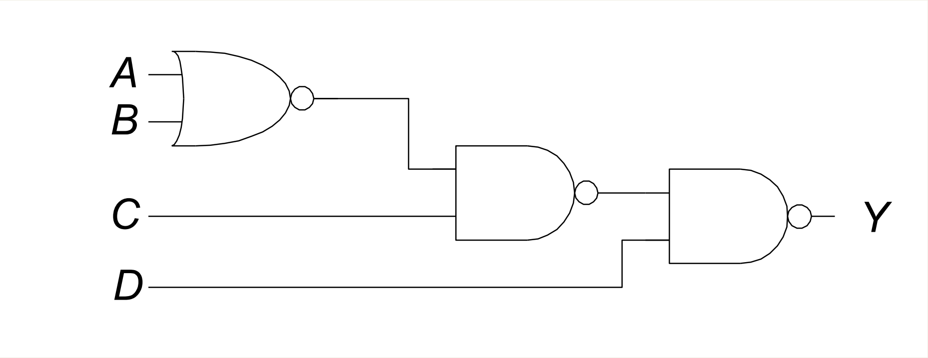
- *Άρα η εξίσωση Boole είναι:* $Y = AB + CD$

Ώθηση φυσαλίδας: κανόνες

- Ξεκινάμε **από την έξοδο** του κυκλώματος, και δουλεύουμε **με κατεύθυνση προς τις εισόδους**
- **«Σπρώχνουμε»** τυχόν φυσαλίδες που υπάρχουν στην τελική έξοδο **προς τα πίσω**, δηλαδή προς τις εισόδους
  - *έτσι μπορούμε να «διαβάσουμε» την εξίσωση σε σχέση με την αληθινή έξοδο ($\mathbf{Y}$), και όχι σε σχέση με το συμπλήρωμά της ($\overline{\mathbf{Y}}$)*
  - *Δουλεύοντας προς τα πίσω, σχεδιάζουμε κάθε πύλη σε τέτοια μορφή ώστε σε μία διασύνδεση **οι φυσαλίδες να αλληλοακυρώνονται***
    ή να μην υπάρχουν
  - *Αν η τρέχουσα πύλη διαθέτει **φυσαλίδα σε κάποια είσοδο**, σχεδιάστε την **προηγούμενη πύλη με φυσαλίδα στην έξοδο***
  - *Αν η τρέχουσα πύλη **δεν έχει φυσαλίδες στις εισόδους**, σχεδιάστε την **προηγούμενη πύλη χωρίς φυσαλίδα στην έξοδο***
- Στο τέλος, κάποιες τερματικές είσοδοι διαθέτουν φυσαλίδες
  - *Αν η τερματική είσοδος **διαθέτει φυσαλίδα** τότε **αντιστρέφεται***

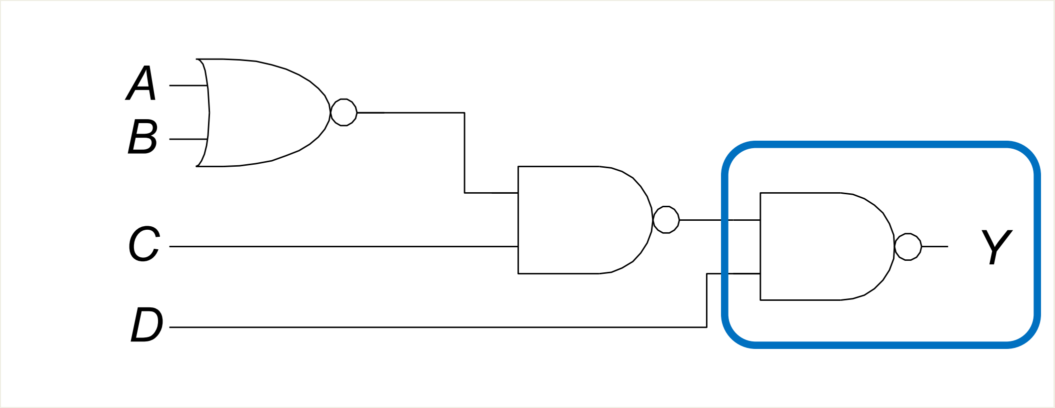
Ώθηση φυσαλίδας: παράδειγμα

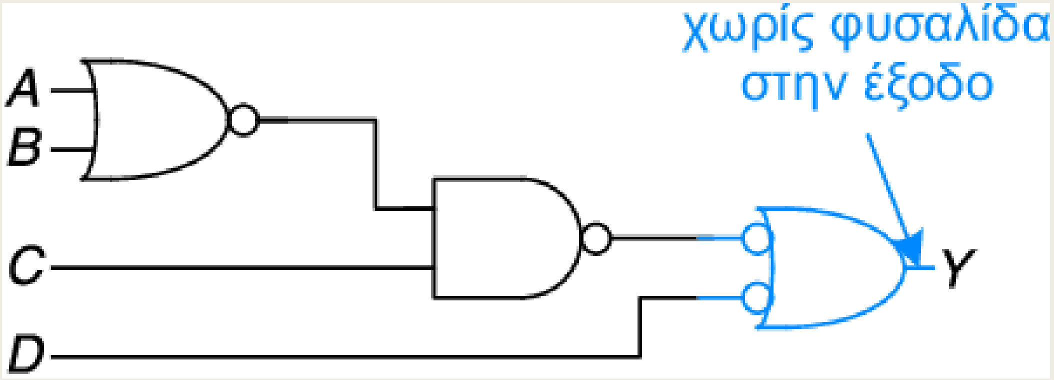
- Ξεκινάμε **από την έξοδο** του κυκλώματος, και δουλεύουμε με κατεύθυνση προς τις εισόδους



Ώθηση φυσαλίδας: παράδειγμα

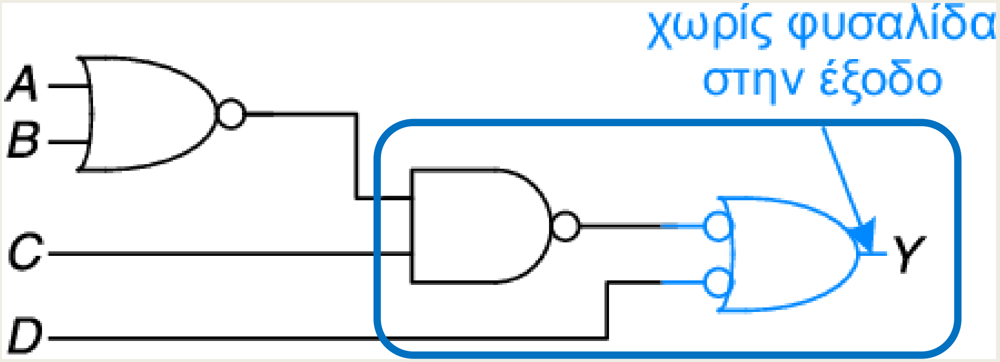
- **«Σπρώχνουμε»** τυχόν φυσαλίδες που υπάρχουν στην τελική έξοδο **προς τα πίσω**, δηλαδή προς τις εισόδους, ώστε να βρούμε τη συνάρτηση Y και όχι τη συμπληρωματική της

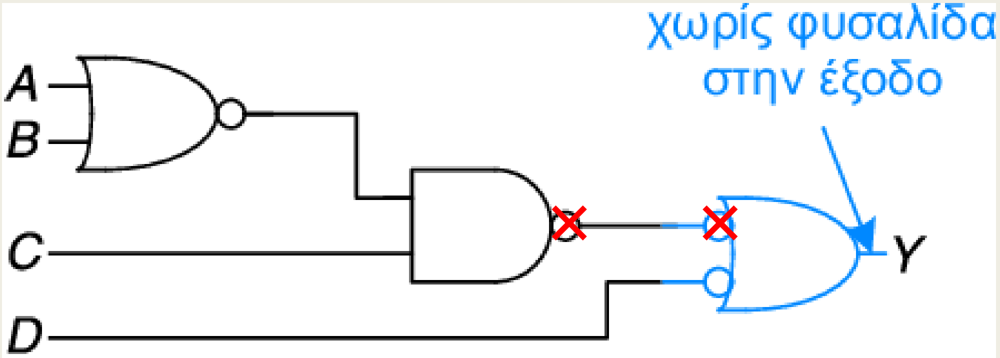




Ώθηση φυσαλίδας: παράδειγμα

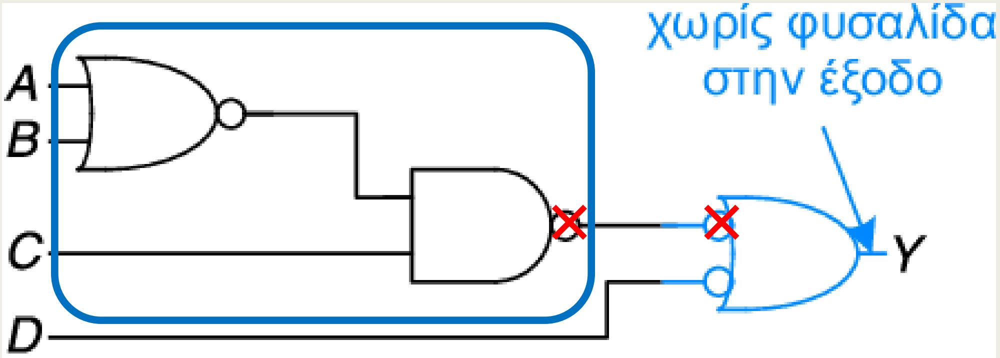
- Αν η τρέχουσα πύλη διαθέτει **φυσαλίδα σε κάποια είσοδο**, σχεδιάστε την **προηγούμενη πύλη με φυσαλίδα στην έξοδο**
  - *Υπάρχει τέτοια πύλη – δεν κάνουμε τίποτα*
  - *οι φυσαλίδες αλληλοακυρώνονται*

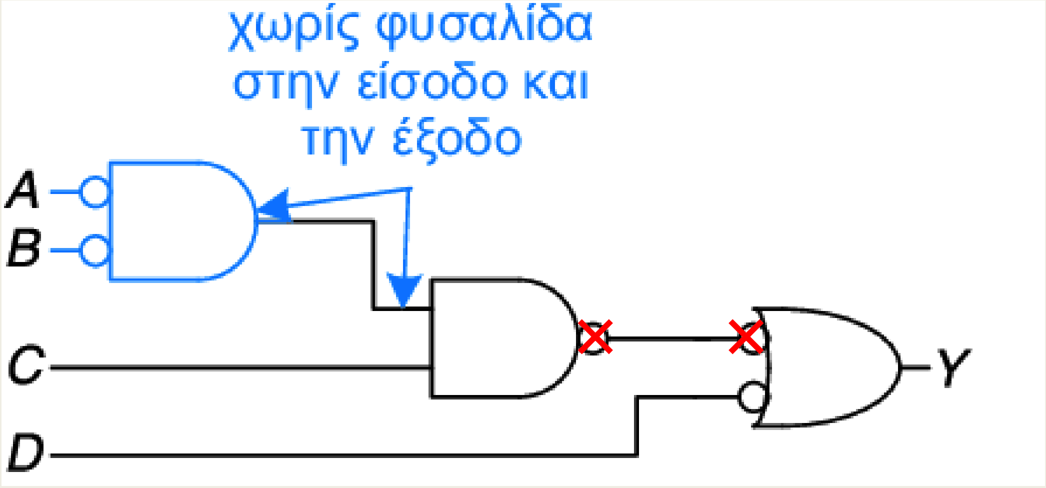




Ώθηση φυσαλίδας: παράδειγμα

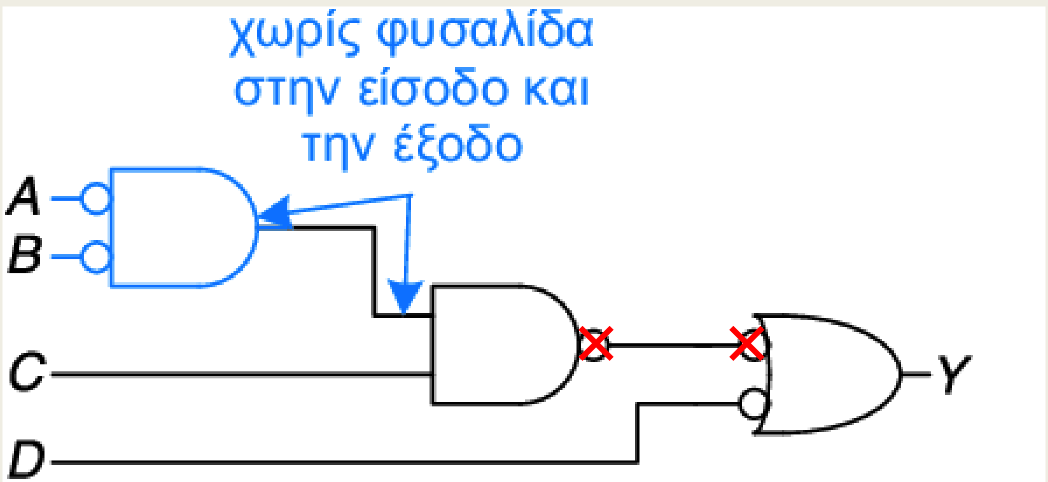
- Αν η τρέχουσα πύλη **δεν έχει φυσαλίδες στις εισόδους**, σχεδιάστε την προηγούμενη πύλη **χωρίς φυσαλίδα στην έξοδο**





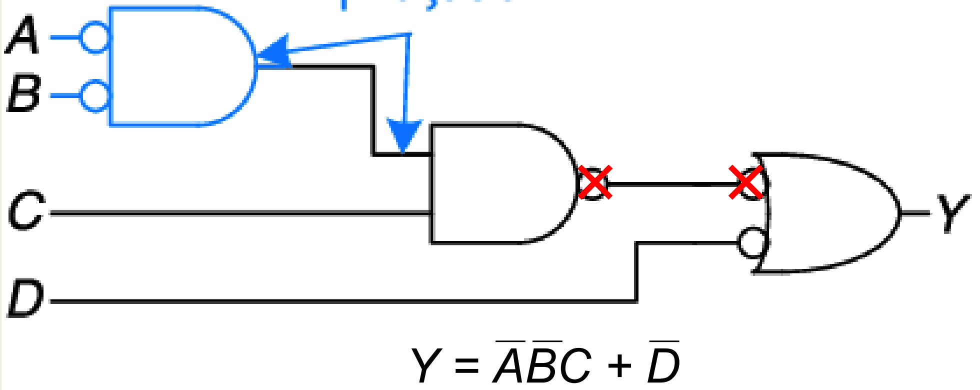
Ώθηση φυσαλίδας: παράδειγμα

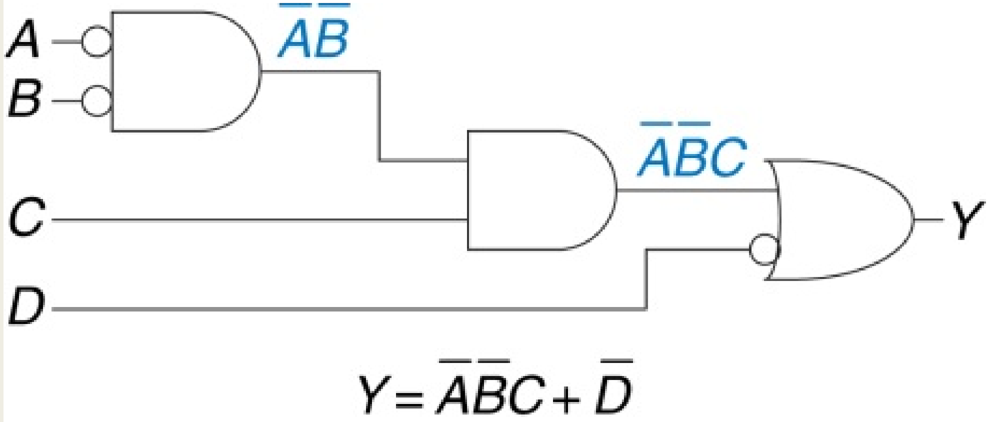
- Στο τέλος, κάποιες τερματικές είσοδοι διαθέτουν φυσαλίδες
  - *Αν η τερματική είσοδος **διαθέτει φυσαλίδα** τότε **αντιστρέφεται***
  - *Στο παράδειγμα ισχύει για τις εισόδους **A, B** και **D***



$$Y = \overline{A}\,\overline{B}C + \overline{D}$$

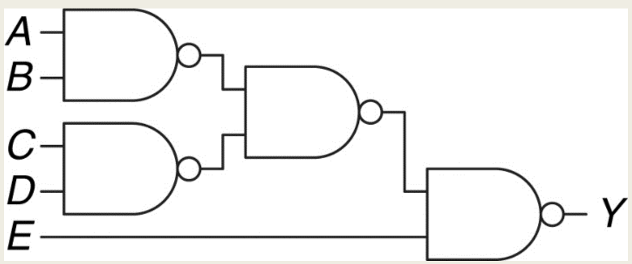
Ώθηση φυσαλίδας: ισοδύναμα κυκλώματα



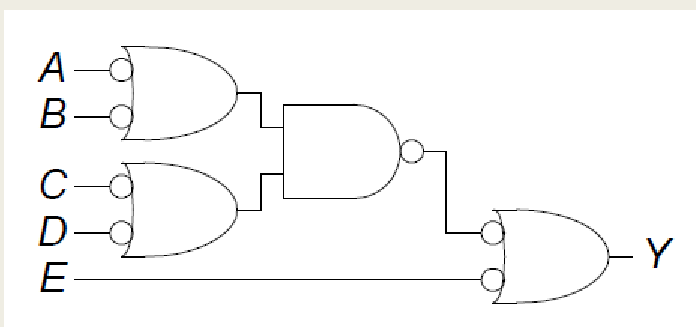


Επιλεγμένες ασκήσεις

- **Άσκηση 2.26**
  Χρησιμοποιώντας πύλες ισοδύναμες κατά De Morgan και μεθόδους «ώθησης» φυσαλίδων, ξανασχεδιάστε το κύκλωμα της εικόνας 2.83 έτσι ώστε να μπορείτε να βρείτε τις εξισώσεις Boole με απλή οπτική εξέταση. Γράψτε τις εξισώσεις Boole.

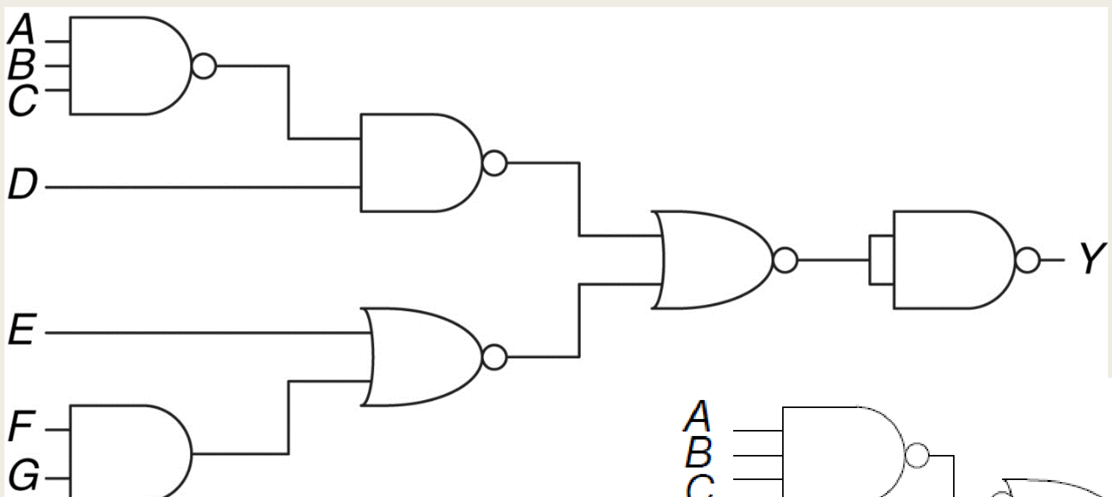


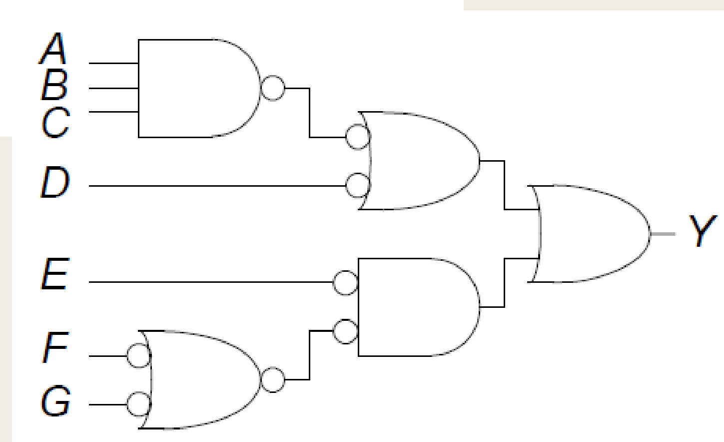
$$Y = (\overline{A} + \overline{B})(\overline{C} + \overline{D}) + \overline{E}$$



Επιλεγμένες ασκήσεις

- **Άσκηση 2.27**
  Χρησιμοποιώντας πύλες ισοδύναμες κατά De Morgan και μεθόδους «ώθησης» φυσαλίδων, ξανασχεδιάστε το κύκλωμα της εικόνας 2.84 έτσι ώστε να μπορείτε να βρείτε τις εξισώσεις Boole με απλή οπτική εξέταση. Γράψτε τις εξισώσεις Boole.

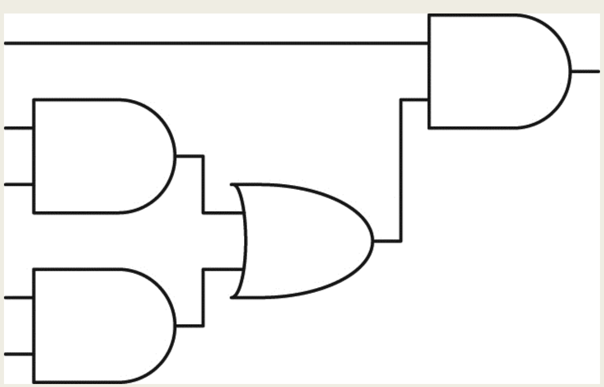


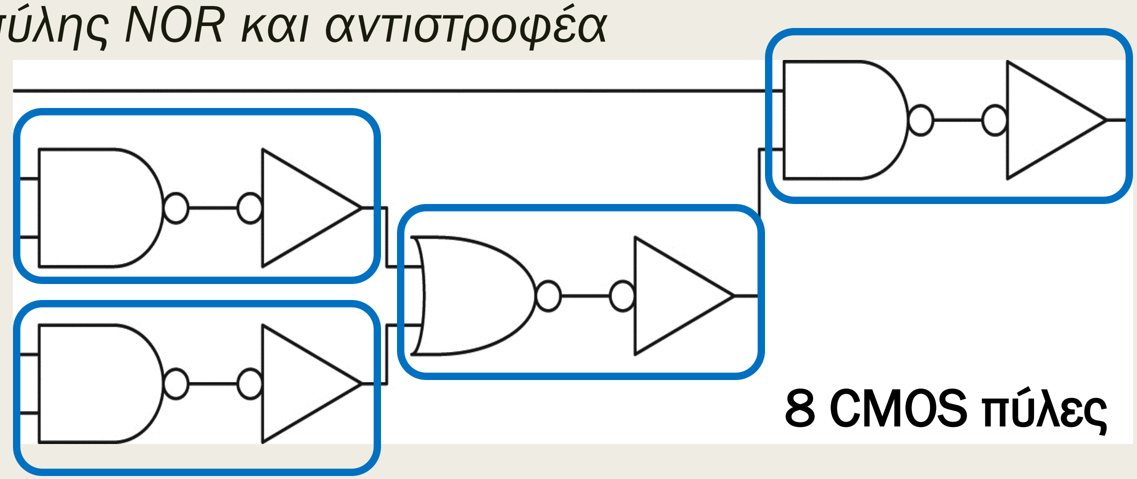


$$Y = ABC + \overline{D} + (\overline{F} + \overline{G})\overline{E}$$
$$= ABC + \overline{D} + \overline{E}\,\overline{F} + \overline{E}\,\overline{G}$$

Ώθηση φυσαλίδας: λογική CMOS*

- Οι περισσότεροι σχεδιαστές σκέφτονται με γνώμονα τις πύλες AND και OR, αλλά στη CMOS τεχνολογία προτιμώνται **οι πύλες NAND, οι πύλες NOR και οι αντιστροφείς.**
- Χρησιμοποιήστε την «ώθηση» φυσαλίδων για να μετατρέψετε το κύκλωμα σε πύλες NAND, πύλες NOR και αντιστροφείς
- Μέθοδος 1 (μη αποδοτική):
  - *Απλή αντικατάσταση κάθε πύλης AND με ένα ζεύγος πύλης NAND και αντιστροφέα, και κάθε πύλης OR με ένα ζεύγος πύλης NOR και αντιστροφέα*

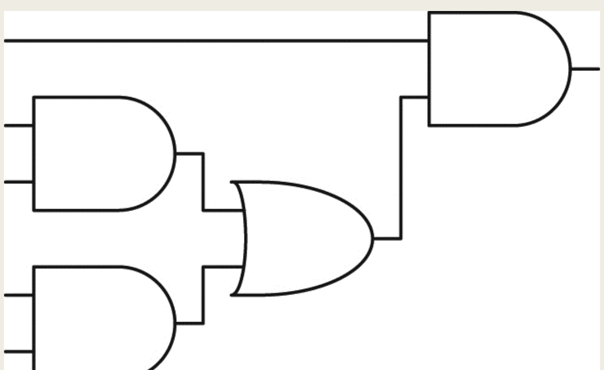


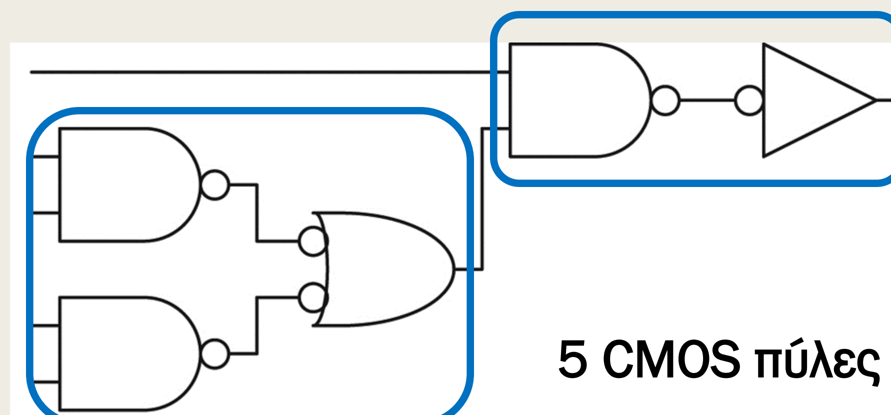


*Παράδειγμα 2.8

Ώθηση φυσαλίδας: λογική CMOS*

- Χρησιμοποιήστε την «ώθηση» φυσαλίδων για να μετατρέψετε το κύκλωμα σε πύλες NAND, πύλες NOR και αντιστροφείς
- Μέθοδος 2 (αποδοτική):
  - *Προσθέτουμε ζεύγη φυσαλίδων στην έξοδο μιας πύλης και την είσοδο την επόμενης πύλης χωρίς να μεταβληθεί η συνάρτηση*
    - Για NAND υλοποίηση μεταξύ πυλών AND (έξοδος) και OR (είσοδος)
    - Για NOR υλοποίηση μεταξύ πυλών OR (έξοδος) και AND (είσοδος)
  - *Προσθέτουμε και αντιστροφείς, όπου απαιτείται*
    - Όταν οι πύλες στην έξοδο και την είσοδο είναι ίδιες (AND ή OR)
    - Στην έξοδο για μετατροπή από AND σε NAND ή από OR σε NOR

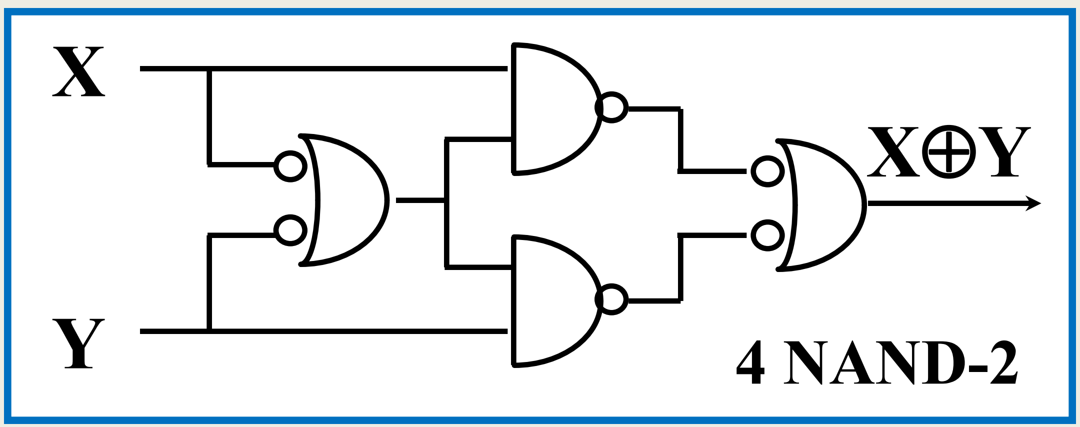




*Παράδειγμα 2.8

Υλοποίηση XOR με NAND

- Εξισώσεις Boole των XOR και XNOR
  - $X \oplus Y = \overline{X}Y + X\overline{Y}$
  - $\overline{X \oplus Y} = \overline{X}\,\overline{Y} + XY$
- Υλοποίηση πύλης XOR με 4 πύλες NAND δύο εισόδων σε 3 επίπεδα
  - *Σχηματικό διάγραμμα*



  - *Αλγεβρική απόδειξη*

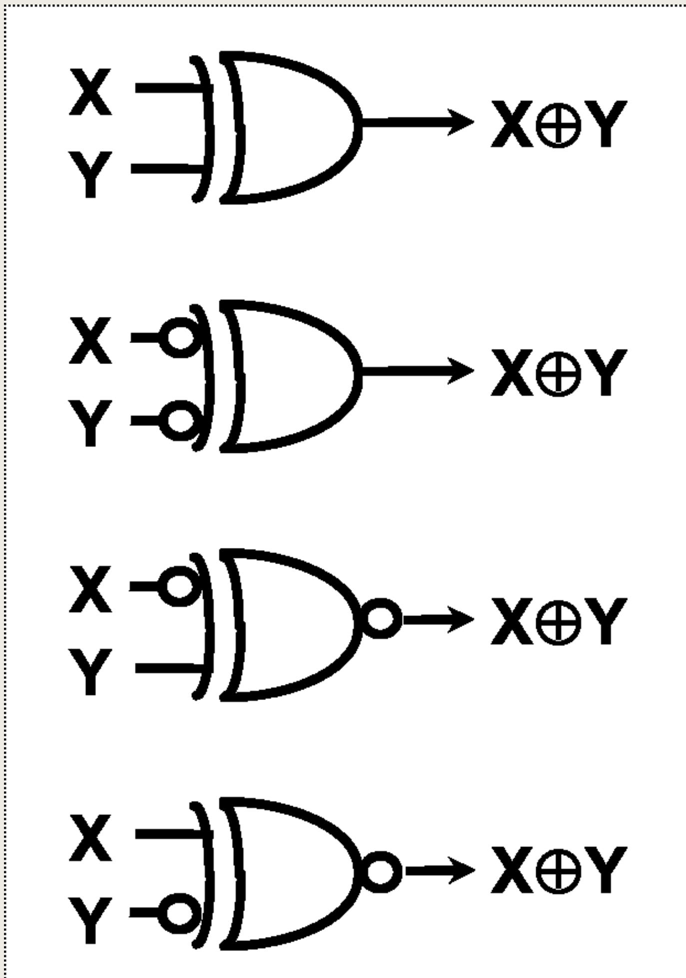
$$X(\overline{X} + \overline{Y}) + Y(\overline{X} + \overline{Y}) = X\,\overline{X} + X\,\overline{Y} + Y\,\overline{X} + Y\,\overline{Y} = \overline{X}\,Y + X\,\overline{Y} = X \oplus Y$$

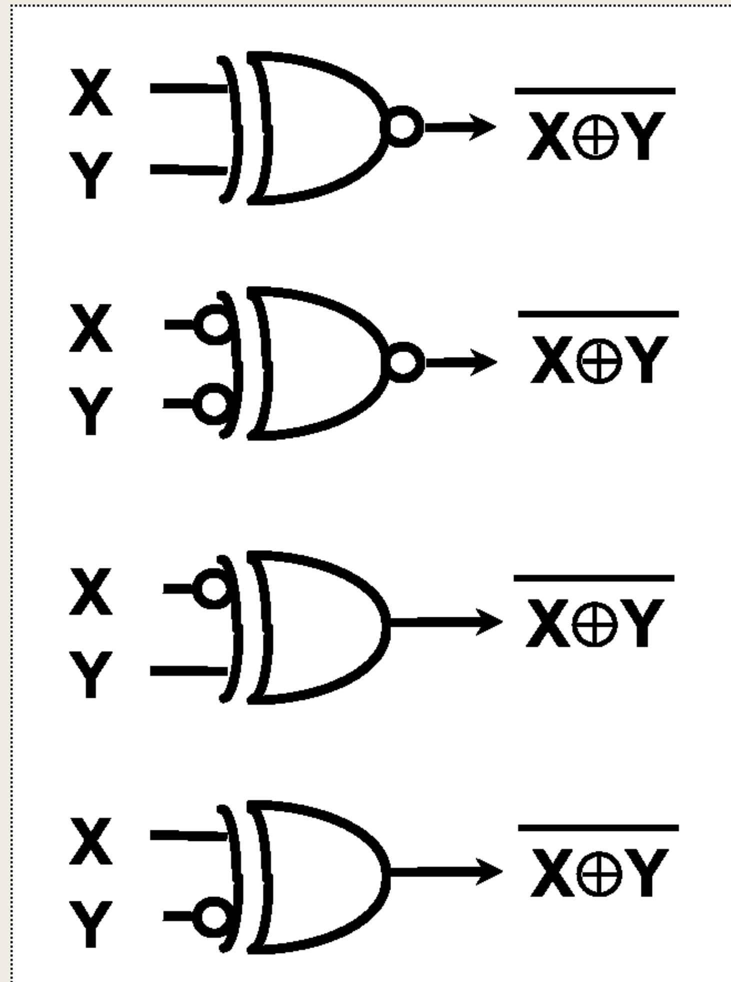
Αλγεβρικές ισότητες για XOR/XNOR

- $X \oplus Y = Y \oplus X$
- $(X \oplus Y) \oplus Z = X \oplus (Y \oplus Z) = X \oplus Y \oplus Z$
- $X \oplus 0 = X$
- $X \oplus 1 = \overline{X}$
- $X \oplus X = 0$
- $X \oplus \overline{X} = 1$
- $\overline{X} \oplus \overline{Y} = X \oplus Y$
- $\overline{X} \oplus Y = \overline{X \oplus Y}$
- $X \oplus \overline{Y} = \overline{X \oplus Y}$

Συμβολισμοί πυλών XOR και XNOR

- Ισοδύναμοι συμβολισμοί XOR και XNOR (για ώθηση φυσαλίδων)



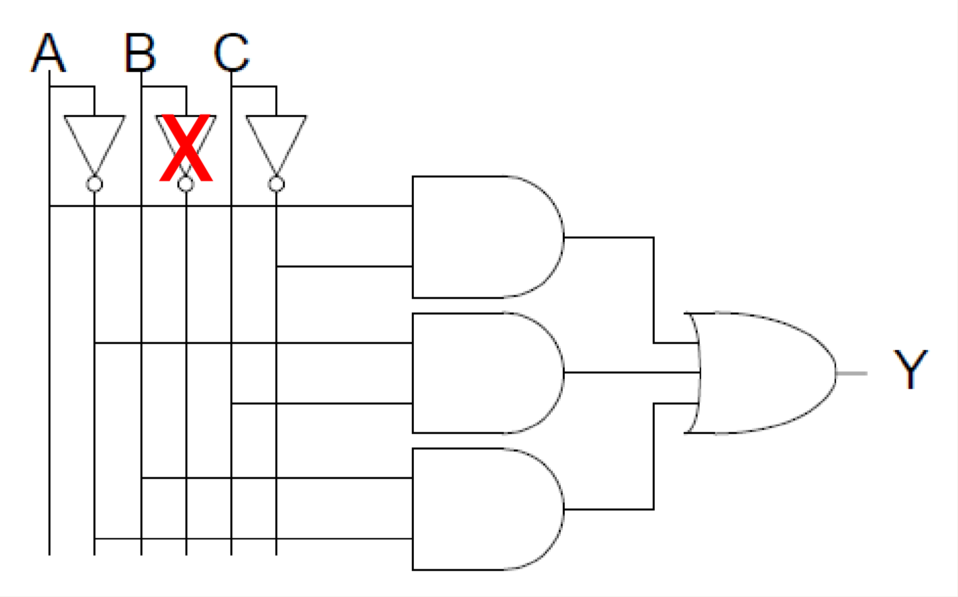


Επιλεγμένες ασκήσεις

- Ασκήσεις 2.10
  Με βάση την απλοποιημένη εξίσωση Boole της Άσκησης 2.6 – (β)

  $$Y = A\overline{C} + \overline{A}C + \overline{A}B$$

  να σχεδιάσετε με πύλες NOT στις εισόδους, όπου απαιτείται:
  - *Την υλοποίηση AND-OR δύο επιπέδων*

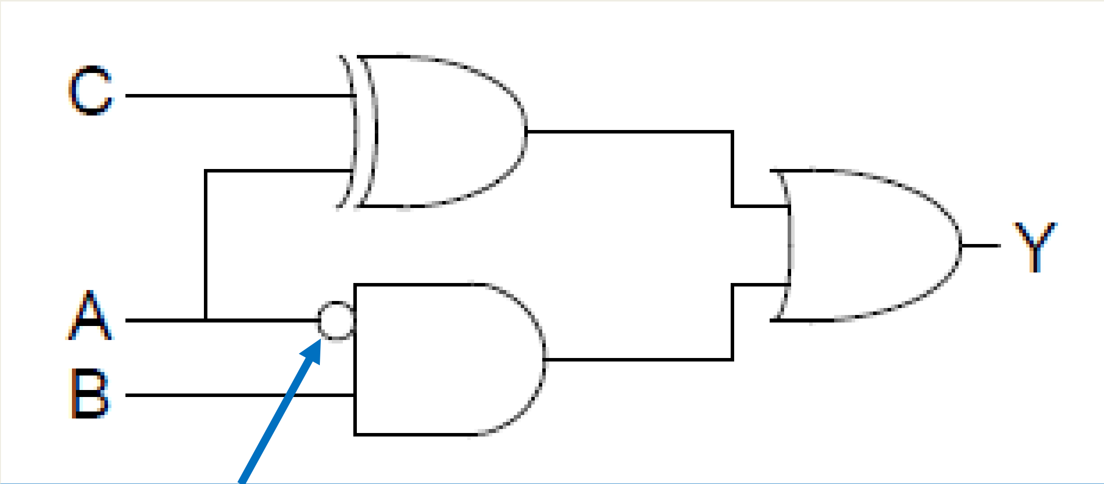


Επιλεγμένες ασκήσεις

- Ασκήσεις 2.8
  Με βάση την απλοποιημένη εξίσωση Boole της Άσκησης 2.6 – (β)

$$Y = A\overline{C} + \overline{A}C + \overline{A}B$$

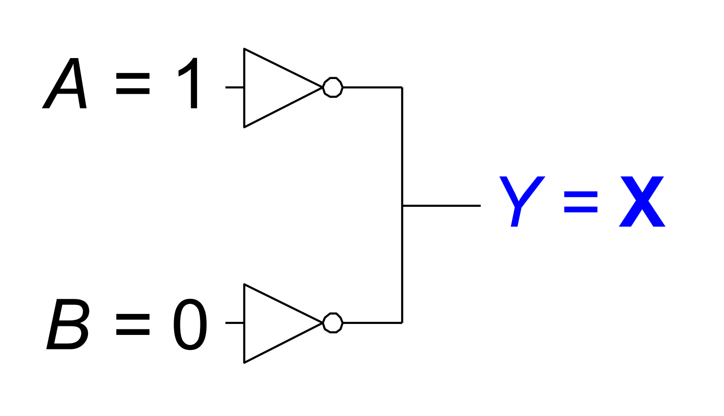
  να σχεδιάσετε με πύλες NOT στις εισόδους, όπου απαιτείται:
  - *Την υλοποίηση που χρησιμοποιεί και πύλη XOR για μείωση υλικού*



Ενσωμάτωση της πύλης NOT στην είσοδο της πύλης AND

Μη αποδεκτή τιμή: X

- Η άλγεβρα Boole περιορίζεται στη χρήση των τιμών 0 και 1
- Ωστόσο, τα κυκλώματα στον πραγματικό κόσμο μπορούν επίσης να έχουν **μη αποδεκτές** τιμές
- Το σύμβολο **X** υποδεικνύει ότι ο κόμβος του κυκλώματος έχει **άγνωστη ή μη αποδεκτή τιμή**
- Αυτό συνήθως συμβαίνει αν οδηγείται ταυτόχρονα και στην τιμή 0 και στην τιμή 1
  - *Αυτή η κατάσταση στον κόμβο, που ονομάζεται* ***ανταγωνισμός****, θεωρείται εσφαλμένη σχεδίαση και πρέπει να αποφεύγεται*



Μη αποδεκτή τιμή: Χ

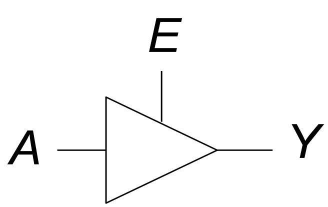
- Η πραγματική τάση σε έναν κόμβο με ανταγωνισμό μπορεί να έχει κάποια **τιμή μεταξύ 0 και V_{DD}**
  - *Συχνά, αλλά όχι πάντα, έχει τιμή που εμπίπτει στην **απαγορευμένη ζώνη***
- Ο **ανταγωνισμός** μπορεί επίσης να προκαλέσει την κατανάλωση **μεγάλων ποσοτήτων ισχύος** μεταξύ των αντιμαχόμενων πυλών, με αποτέλεσμα το κύκλωμα να **υπερθερμανθεί** και ίσως να **καταστραφεί**
- Οι τιμές Χ χρησιμοποιούνται από τους προσομοιωτές κυκλωμάτων για να υποδεικνύουν **αρχικές τιμές που δεν έχουν καθοριστεί**
- Προσοχή:
  - *Τα **Χ** χρησιμοποιούνται για να δείξουν και τους **αδιάφορους όρους στους πίνακες αλήθειας** αλλά και τον **ανταγωνισμό στους κόμβους***

Μετέωρη τιμή: Z

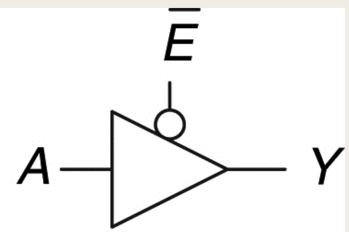
- Το σύμβολο Z υποδεικνύει ότι ένας κόμβος δεν οδηγείται ούτε σε τάση HIGH ούτε σε τάση LOW. Τότε λέμε ότι κόμβος είναι «**μετέωρος**», έχει **υψηλή αντίσταση**, ή **υψηλό Z**
- Ενας μετέωρος κόμβος μπορεί να έχει τιμή 0, τιμή 1, ή κάποια τάση στο ενδιάμεσο, ανάλογα με το ιστορικό της λειτουργίας του κυκλώματος
- Η ύπαρξη ενός μετέωρου κόμβου δεν σημαίνει πάντα ότι υπάρχει σφάλμα στο κύκλωμα
  - *για παράδειγμα, όταν υπάρχει κάποιο άλλο στοιχείο που οδηγεί τον ίδιο κόμβο στην τιμή 0 ή την τιμή 1*
- Τρόποι δημιουργίας ενός μετέωρου κόμβου:
  - *όταν ξεχνάμε να συνδέσουμε μια τάση σε μια είσοδο ενός κυκλώματος*
  - *όταν υποθέτουμε ότι μια μη συνδεδεμένη είσοδος είναι ίδια με μία είσοδο που έχει την τιμή 0 (λάθος)*

Απομονωτής τριών καταστάσεων

- Ο **απομονωτής τριών καταστάσεων** (tristate buffer), διαθέτει τρεις πιθανές καταστάσεις εξόδου: HIGH (1), LOW (0), και μετέωρη (Z).
- Ο απομονωτής αριστερά έχει μια είσοδο *A*, την έξοδο *Y* και ένα σήμα **enable** (έγκρισης) *E* ***ενεργό στο HIGH (active high)***
- Ο απομονωτής δεξιά έχει μια είσοδο *A*, την έξοδο *Y* και ένα σήμα **enable** (έγκρισης) *E* ***ενεργό στο LOW (active low)***



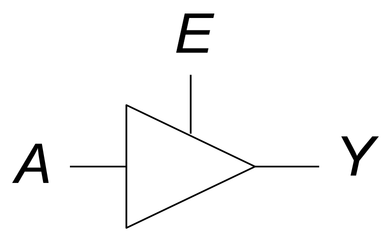
E	A	Y
0	0	Z
0	1	Z
1	0	0
1	1	1



$\bar{E}$	A	Y
0	0	0
0	1	1
1	0	Z
1	1	Z

Απομονωτής τριών καταστάσεων

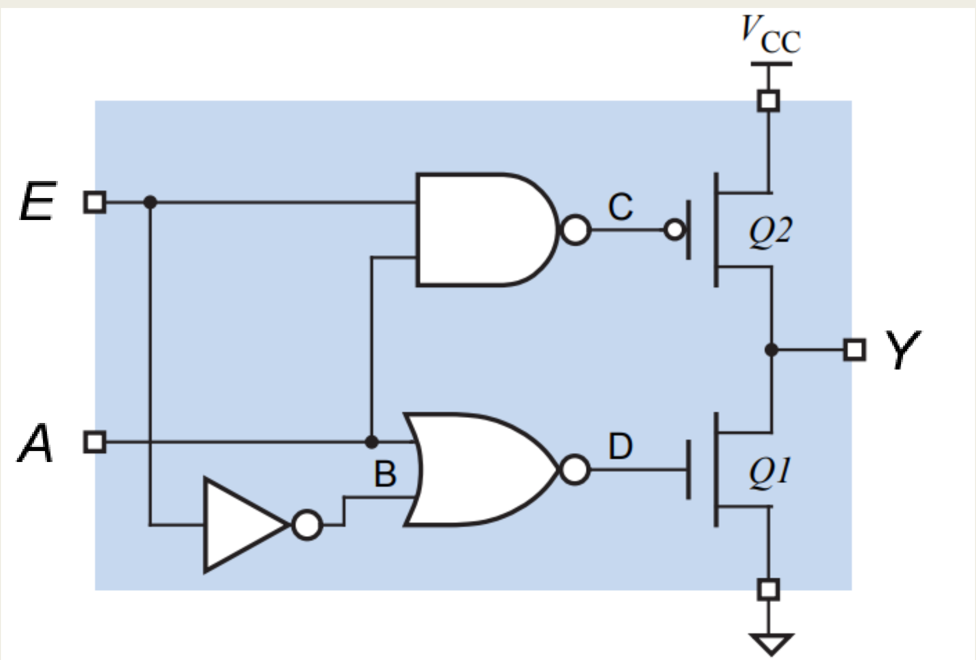
- Στον απομονωτή τριών καταστάσεων (tri-state buffer) όταν το **σήμα enable έχει τιμή ‘1’ (TRUE)**, επιτρέπεται η μεταφορά τιμής από την είσοδο στην έξοδο, οπότε η έξοδος έχει την τιμή της εισόδου
  - *ο απομονωτής τριών καταστάσεων συμπεριφέρεται **ως απλώς απομονωτής***
- Όταν το **σήμα enable έχει τιμή ‘0’ (FALSE)**, δεν επιτρέπεται η μεταφορά τιμής από την είσοδο στην έξοδο, οπότε η έξοδος έχει υψηλή αντίσταση ή είναι μετέωρη (Z)
  - *ο απομονωτής τριών καταστάσεων συμπεριφέρεται **ως πυκνωτής***

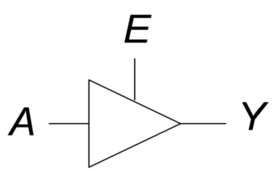


E	A	Y
0	0	Z
0	1	Z
1	0	0
1	1	1

Απομονωτής τριών καταστάσεων

- Απλουστευμένο διάγραμμα κυκλώματος για έναν **απομονωτή τριών καταστάσεων** (tri-state buffer) της λογικής οικογένειας CMOS
  - *Οι εσωτερικές πύλες NAND, NOR και ο αντιστροφέας απεικονίζονται με τα σύμβολά τους και όχι με τρανζίστορ*
- Όταν το **σήμα enable έχει τιμή ‘0’**, τα δύο τρανζίστορ της εξόδου δεν άγουν και η έξοδος βρίσκεται στη μετέωρη κατάσταση Z
- Όταν το **σήμα enable έχει τιμή ‘1’**, η έξοδος καθορίζεται από την είσοδο A



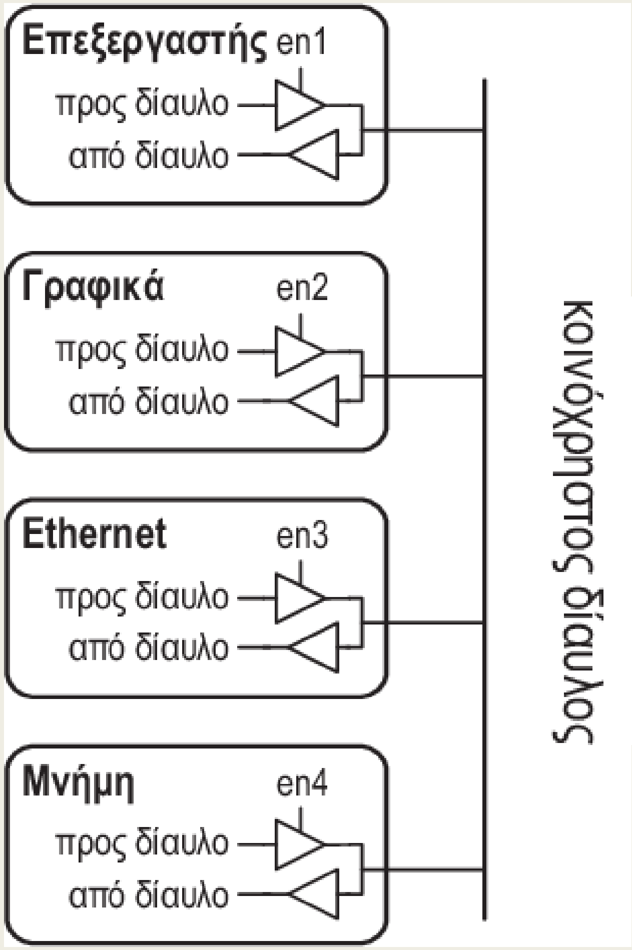


E	A	Y
0	0	Z
0	1	Z
1	0	0
1	1	1

Διάγραμμα από το βιβλίο «Ψηφιακή Σχεδίαση: Αρχές και Πρακτικές» του John F. Wakerly

Παράδειγμα χρήσης του απομονωτή

- Οι απομονωτές τριών καταστάσεων συνήθως χρησιμοποιούνται σε διαύλους (buses) που συνδέουν πολλά τσιπ
- Κάθε τσιπ μπορεί να συνδέεται σε έναν κοινόχρηστο δίαυλο μνήμης χρησιμοποιώντας απομονωτές τριών καταστάσεων
- Μόνο ένα τσιπ τη φορά επιτρέπεται να ενεργοποιεί το σήμα enable για να οδηγεί μια τιμή στον δίαυλο.
- Τα υπόλοιπα τσιπ πρέπει να παράγουν μετέωρους εξόδους, έτσι ώστε να μην προκαλούν ανταγωνισμό με το τσιπ που «συνομιλεί» με τη μνήμη



Χάρτες Karnaugh (K-maps)

- Οι **χάρτες Karnaugh** (Karnaugh maps, K-maps) αποτελούν μια οπτικοποιημένη (γραφική) μέθοδο για την **ελαχιστοποίηση εξισώσεων Boole σε κανονική μορφή αθροίσματος γινομένων μέχρι 4 μεταβλητές (εισόδους)**
- Η **λειτουργική προδιαγραφή** ενός συνδυαστικού κυκλώματος εκφράζεται και με τη μορφή του χάρτη Karnaugh
- Τους επινόησε το 1953 ο **Maurice Karnaugh**

A	B	C	Y
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	0

Y: C \ AB	00	01	11	10
0	1	0	0	0
1	1	0	0	0

Χάρτες Karnaugh των 3 εισόδων

- Η πάνω γραμμή του χάρτη περιέχει τις τέσσερις πιθανές τιμές για τις εισόδους *A* και *B*. *Οι τιμές των εισόδων A και B είναι:*
  - *00, 01, 11, 10 (κώδικας Gray)* *και* *όχι 00, 01, 10, 11* *ώστε δύο γειτονικά τετράγωνα να διαφέρουν στην τιμή μίας μόνο εισόδου*
- Η αριστερή στήλη περιέχει τις δύο πιθανές τιμές για την είσοδο C

A	*B*	*C*	*Y*
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	0

Y — *C* \ *AB*	00	01	11	10
0	1	0	0	0
1	1	0	0	0

Χάρτες Karnaugh των 3 εισόδων

- Κάθε τετράγωνο του χάρτη αντιστοιχεί σε μια γραμμή του πίνακα αληθείας και περιέχει την τιμή της εξόδου Y για τη συγκεκριμένη γραμμή, ανάλογα με την τιμή των εισόδων
- Κάθε δύο γειτονικά τετράγωνα διαφέρουν στην τιμή μίας μόνο εισόδου. Οι άλλες δύο είσοδοι έχουν την ίδια τιμή

A	*B*	*C*	*Y*
0	0	0	**1**
0	0	1	**1**
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	0

Y: *C* \ *AB*	00	01	11	10
0	**1**	0	0	0
1	**1**	0	0	0

Χάρτες Karnaugh των 3 εισόδων

- Ακριβώς όπως ισχύει για τις γραμμές ενός πίνακα αληθείας, έτσι και κάθε τετράγωνο ενός χάρτη Karnaugh αναπαριστά έναν **ελαχιστόρο**
- Κάθε δύο γειτονικά τετράγωνα έχουν τα ίδια λεκτικά εκτός από ένα, το οποίο εμφανίζεται με την αληθινή μορφή του στο ένα τετράγωνο και με τη συμπληρωματική μορφή του στο άλλο
  - *Για παράδειγμα στην πρώτη στήλη τα τετράγωνα που αναπαριστούν τους ελαχιστόρους* $\overline{A}\overline{B}\overline{C}$ *και* $\overline{A}\overline{B}C$ *είναι γειτονικά και διαφέρουν μόνο ως προς τη μεταβλητή* **C**

Y: C \ AB	00	01	11	10
0	1	0	0	0
1	1	0	0	0

Y: C \ AB	00	01	11	10
0	$\overline{A}\overline{B}\overline{C}$	$\overline{A}B\overline{C}$	$AB\overline{C}$	$A\overline{B}\overline{C}$
1	$\overline{A}\overline{B}C$	$\overline{A}BC$	ABC	$A\overline{B}C$

Αναδίπλωση του χάρτη Karnaugh

- Ένα άλλο χαρακτηριστικό του χάρτη Karnaugh είναι ότι **«αναδιπλώνεται»**
- Τα τετράγωνα που βρίσκονται **εντελώς δεξιά** είναι ουσιαστικά **γειτονικά** με τα τετράγωνα που βρίσκονται **εντελώς αριστερά**, με την έννοια ότι διαφέρουν μόνο ως προς μία μεταβλητή
  - *Για παράδειγμα στην πρώτη σειρά τα τετράγωνα που αναπαριστούν τους ελαχιστόρους* $\bar{A}\bar{B}\bar{C}$ *και* $A\bar{B}\bar{C}$ *είναι γειτονικά και διαφέρουν μόνο ως προς τη μεταβλητή* A

Y

C \ AB	00	01	11	10
0	1	0	0	0
1	1	0	0	0

Y

C \ AB	00	01	11	10
0	$\bar{A}\bar{B}\bar{C}$	$\bar{A}B\bar{C}$	$AB\bar{C}$	$A\bar{B}\bar{C}$
1	$\bar{A}\bar{B}C$	$\bar{A}BC$	ABC	$A\bar{B}C$

Ελαχιστοποίηση εξισώσεων Boole

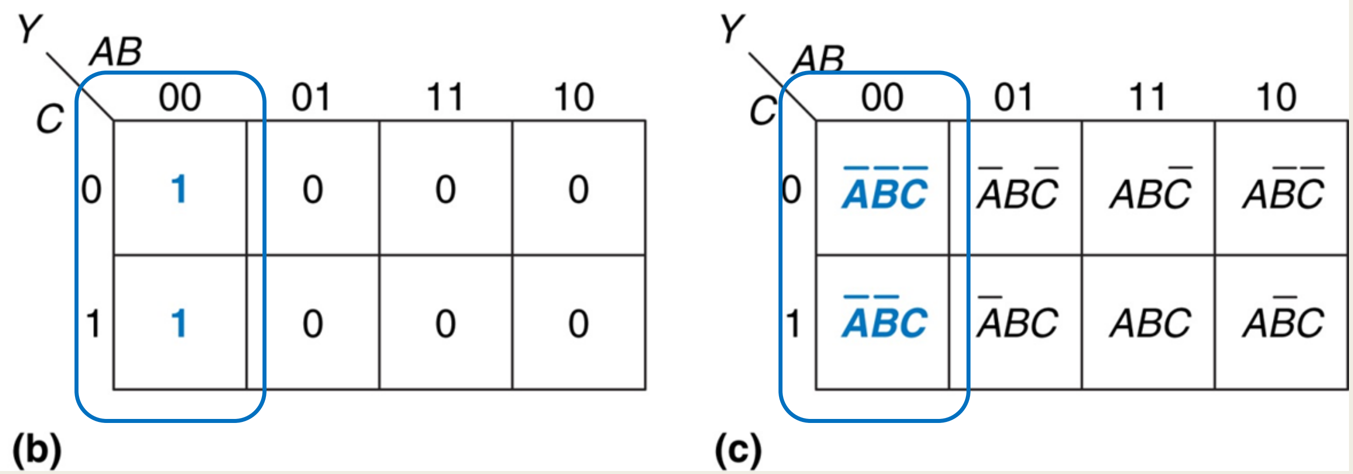
- Η ελαχιστοποίηση λογικών εξισώσεων Boole σε κανονική μορφή αθροίσματος γινομένων περιλαμβάνει τον **συνδυασμό δύο ελαχιστόρων** για τους οποίους η έξοδος έχει την τιμή 1, οι οποίοι περιέχουν ένα **γινόμενο λεκτικών P**, καθώς και την **αληθινή και τη συμπληρωματική μορφή κάποιας μεταβλητής, έστω A**
- *Όταν συνδυαστούν* οι 2 ελαχιστόροι εξαλείφεται η μεταβλητή *A*, *ενώ παραμένει το* **γινόμενο λεκτικών P**
  - $\boldsymbol{P \cdot A + \mathrm{P} \cdot \overline{\mathrm{A}} = P \cdot (A + \overline{\mathrm{A}}) = P \cdot 1 = P}$

Χάρτες Karnaugh: Ελαχιστοποίηση

- Η εύρεση των εξισώσεων από τον χάρτη Karnaugh είναι ακριβώς ισοδύναμη με την εύρεση των εξισώσεων σε μορφή αθροίσματος γινομένων απευθείας από τον πίνακα αληθείας.
- Στον χάρτη Karnaugh του σχήματος υπάρχουν μόνο δύο ελαχιστόροι στην εξίσωση: $\overline{A}\,\overline{B}\,\overline{C}$ και $\overline{A}\,\overline{B}C$ για τους οποίους η έξοδος έχει την τιμή 1
- $Y = \overline{A}\,\overline{B}\,\overline{C} + \overline{A}\,\overline{B}C$

A	*B*	*C*	*Y*
0	0	0	**1**
0	0	1	**1**
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	0

(a)



Χάρτες Karnaugh: Ελαχιστοποίηση

- Αν ελαχιστοποιήσουμε αυτήν την εξίσωση με τη βοήθεια της άλγεβρας Boole έχουμε:

$$\boldsymbol{\Upsilon = \overline{A}\,\overline{B}\,\overline{C} + \overline{A}\,\overline{B}C = \overline{A}\,\overline{B}(\overline{C} + C) = \overline{A}\,\overline{B}(1) = \overline{A}\,\overline{B}}$$

- Οι χάρτες Karnaugh μας βοηθούν να κάνουμε αυτή ακριβώς την ελαχιστοποίηση γραφικά, **κυκλώνοντας τους άσους σε δύο γειτονικά τετράγωνα που έχουν την τιμή 1**
  - *Για κάθε κύκλο, γράφουμε τον αντίστοιχο όρο στην εξίσωση*
  - *Μεταβλητές των οποίων η αληθινή και η συμπληρωματική μορφή περιέχονται και οι δύο στον κύκλο εξαιρούνται από τον όρο*
  - *Δηλαδή στον όρο κρατούμε τις μεταβλητές που είναι κοινές και στα δύο γειτονικά τετράγωνα. Επομένως:*

$$\boldsymbol{\Upsilon = \overline{A}\,\overline{B}}$$

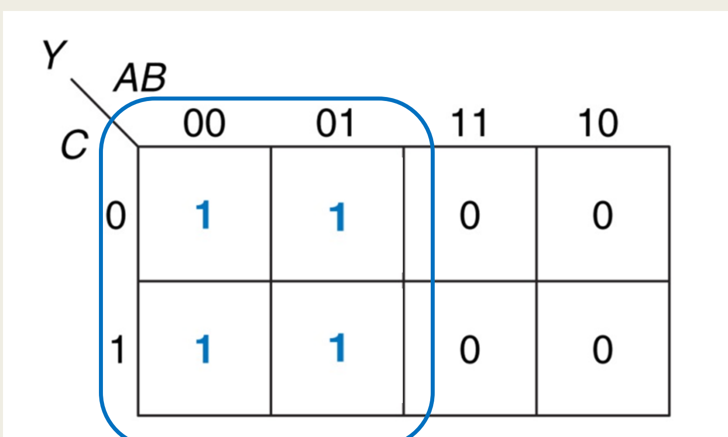
Y: C \ AB	00	01	11	10
0	1	0	0	0
1	1	0	0	0

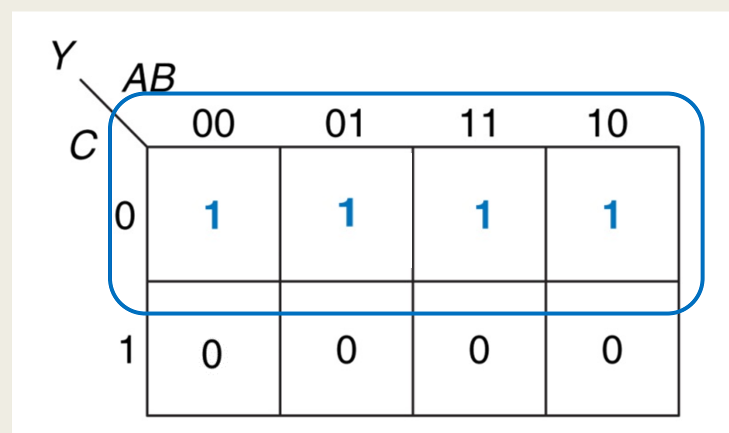
Ελαχιστοποίηση εξισώσεων Boole

- Η ελαχιστοποίηση λογικών εξισώσεων Boole σε κανονική μορφή αθροίσματος γινομένων **γενικεύεται**, ώστε να περιλαμβάνει τον **συνδυασμό τεσσάρων ελαχιστόρων** για τους οποίους η έξοδος έχει την τιμή 1, οι οποίοι περιέχουν ένα **γινόμενο λεκτικών P**, καθώς και έναν από τους τέσσερεις πιθανούς συνδυασμούς της **αληθινής και της συμπληρωματικής μορφής κάποιων δύο μεταβλητών, έστω A και B. ήτοι $\overline{\mathrm{A}}\overline{\mathrm{B}}, \overline{\mathrm{A}}\mathrm{B}, \mathrm{A}\overline{\mathrm{B}}, \mathrm{AB}$**
- *Όταν συνδυαστούν* οι 4 ελαχιστόροι εξαλείφονται οι μεταβλητές *A και B, ενώ παραμένει το* **γινόμενο λεκτικών P**
  - $$\begin{aligned}&\boldsymbol{P\cdot \overline{A}\overline{B}+P\cdot \overline{A}B+P\cdot A\overline{B}+P\cdot AB=}\\&\boldsymbol{P\cdot(\overline{A}\overline{B}+\overline{A}B+A\overline{B}+AB)=}\\&\boldsymbol{P\cdot(\overline{A}(\overline{B}+B)+A(\overline{B}+B))=}\\&\boldsymbol{P\cdot(\overline{A}(1)+A(1))=P\cdot(\overline{A}+A)=P\cdot 1=P}\end{aligned}$$
- Αντίστοιχη και η περίπτωση των 8 συνδυάσιμων ελαχιστόρων, όπου εξαλείφονται 3 μεταβλητές

Χάρτες Karnaugh: Ελαχιστοποίηση

- Στον χάρτη Karnaugh του σχήματος **αριστερά** υπάρχουν 4 ελαχιστόροι στην εξίσωση: $\overline{A}\,\overline{B}\,\overline{C}$, $\overline{A}\,\overline{B}C$, $\overline{A}B\overline{C}$, και $\overline{A}BC$ για τους οποίους η έξοδος έχει την τιμή 1
- $Y = \overline{A}\,\overline{B}\,\overline{C} + \overline{A}\,\overline{B}C + \overline{A}B\overline{C} + \overline{A}BC$
- Στον χάρτη Karnaugh του σχήματος **δεξιά** υπάρχουν 4 ελαχιστόροι στην εξίσωση: $\overline{A}\,\overline{B}\,\overline{C}$, $\overline{A}B\overline{C}$, $AB\overline{C}$ και $A\overline{B}\,\overline{C}$ για τους οποίους η έξοδος έχει την τιμή 1
- $Y = \overline{A}\,\overline{B}\,\overline{C} + \overline{A}B\overline{C} + AB\overline{C} + A\overline{B}\,\overline{C}$





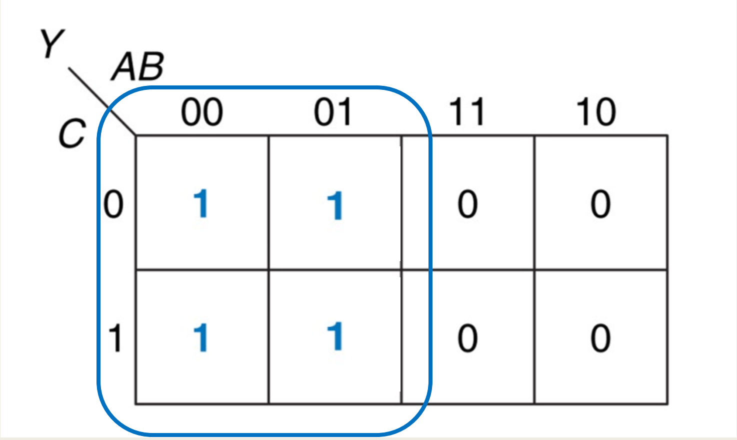
Χάρτες Karnaugh: Ελαχιστοποίηση

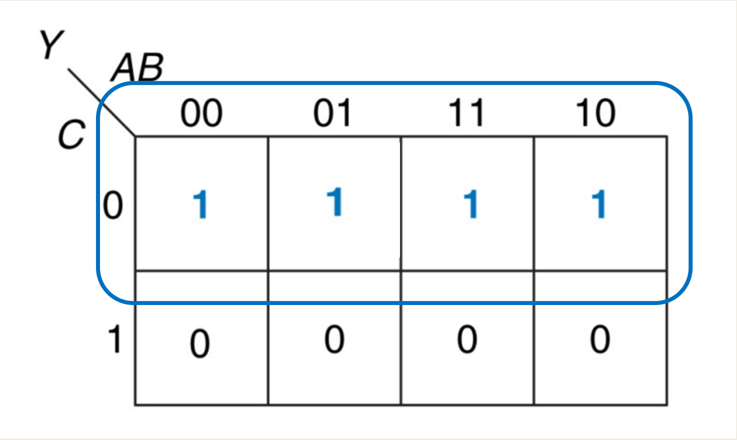
- Αν ελαχιστοποιήσουμε αυτές τις εξισώσεις με την άλγεβρα Boole έχουμε:

$\boldsymbol{Y = \overline{A}\,\overline{B}\,\overline{C} + \overline{A}\,\overline{B}C + \overline{A}B\overline{C} + \overline{A}BC = \overline{A} \cdot (\overline{B}\,\overline{C} + \overline{B}C + B\overline{C} + BC) = \overline{A}}$ (αριστερά)

$\boldsymbol{Y = \overline{A}\,\overline{B}\,\overline{C} + \overline{A}B\overline{C} + AB\overline{C} + A\overline{B}\,\overline{C} = \overline{C} \cdot (\overline{A}\,\overline{B} + \overline{A}B + A\overline{B} + AB) = \overline{C}}$ (δεξιά)

- Οι χάρτες Karnaugh μας βοηθούν να κάνουμε την ελαχιστοποίηση γραφικά, **κυκλώνοντας τους άσους σε 4 γειτονικά τετράγωνα που έχουν την τιμή 1**
  - *Τα 4 γειτονικά τετράγωνα σχηματίζουν ένα τετράγωνο (αριστερά) ή απαρτίζουν μία σειρά (δεξιά) (ή μία στήλη) του χάρτη Karnaugh*
  - *Για κάθε κύκλο, γράφουμε τον αντίστοιχο όρο στην εξίσωση*
  - *Στον όρο κρατούμε τις μεταβλητές που είναι κοινές και στα 4 γειτονικά τετράγωνα. Επομένως:* $\boldsymbol{Y = \overline{A}}$ *(αριστερά) και* $\boldsymbol{Y = \overline{C}}$ *(δεξιά)*





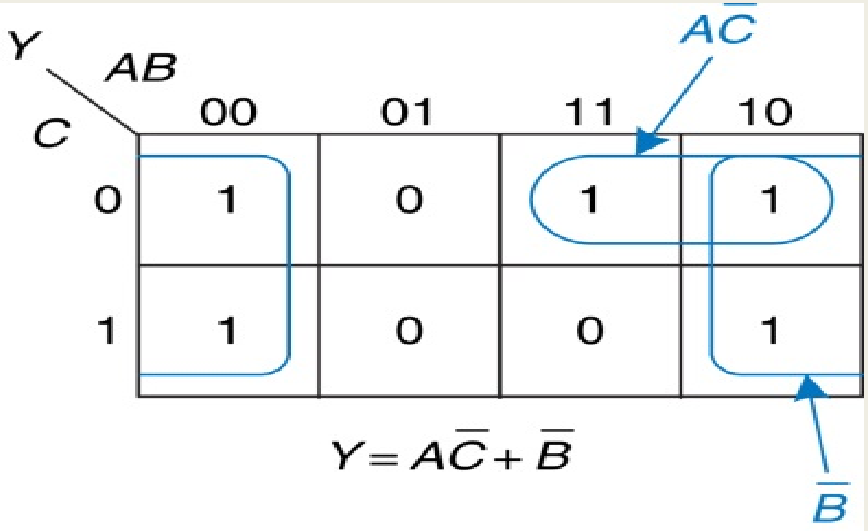
Διαδικασία ελαχιστοποίησης με K-map

- Χρησιμοποιήστε το **ελάχιστο πλήθος κύκλων** που είναι απαραίτητο για την κάλυψη όλων των τετραγώνων με τιμή 1
- Όλα τα τετράγωνα μέσα σε κάθε κύκλο πρέπει να περιέχουν την τιμή 1
- Κάθε κύκλος πρέπει να καλύπτει ένα ορθογώνιο τμήμα το οποίο να περιέχει ένα **πλήθος γειτονικών τετραγώνων που αποτελεί δύναμη του δύο** (δηλαδή 1, 2, 4 ή 8) προς κάθε κατεύθυνση
- Κάθε κύκλος πρέπει να έχει **το μεγαλύτερο δυνατό μέγεθος** έτσι ώστε να αναπαριστά **πρώτους όρους**
- Ένας κύκλος μπορεί να **«αναδιπλώνεται»** γύρω από τα άκρα ενός χάρτη Karnaugh.
- Ένα τετράγωνο με τιμή 1 στον χάρτη Karnaugh μπορεί να **κυκλωθεί πολλές φορές**, αν αυτό έχει ως αποτέλεσμα ότι θα χρησιμοποιηθούν **λιγότεροι κύκλοι**

Παράδειγμα 2.9

- Ελαχιστοποίηση μίας συνάρτησης τριών μεταβλητών με χρήση K-map
- Έστω ότι έχουμε τη συνάρτηση $Y = F(A, B, C)$ του K-map που φαίνεται κάτω αριστερά. Ελαχιστοποιήστε την εξίσωση χρησιμοποιώντας τον χάρτη.
  - *Κυκλώνουμε τα τετράγωνα με τιμή 1 στον χάρτη Karnaugh χρησιμοποιώντας όσο το δυνατόν λιγότερους κύκλους*
  - *Κάθε κύκλος στον χάρτη αναπαριστά έναν πρώτο όρο, και η διάσταση κάθε κύκλου αποτελεί δύναμη του δύο*

Y: C \ AB	00	01	11	10
0	1	0	1	1
1	1	0	0	1



Χάρτες Karnaugh των 4 εισόδων

- Η πάνω γραμμή του χάρτη περιέχει τις τέσσερις πιθανές τιμές για τις εισόδους *A* και *B*. *Οι τιμές των εισόδων A* και *B είναι:*
  - ***00, 01, 11, 10 (κώδικας Gray)***
- Η αριστερή στήλη του χάρτη περιέχει τις τέσσερις πιθανές τιμές για τις εισόδους *C* και *D*. *Οι τιμές των εισόδων C* και *D επίσης είναι:*
  - ***00, 01, 11, 10 (κώδικας Gray)***

A	B	C	D	Y
0	0	0	0	1
0	0	0	1	0
0	0	1	0	1
0	0	1	1	1
0	1	0	0	0
0	1	0	1	1
0	1	1	0	1
0	1	1	1	1
1	0	0	0	1
1	0	0	1	1
1	0	1	0	1
1	0	1	1	0
1	1	0	0	0
1	1	0	1	0
1	1	1	0	0
1	1	1	1	0

Στήλη AB=00

Στήλη AB=01

Στήλη AB=10

Στήλη AB=11

Y

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

Χάρτες Karnaugh των 4 εισόδων

- Κάθε τετράγωνο με τιμή 1 αντιστοιχεί σε έναν ελαχιστόρο με 4 λεκτικά. Είναι πρώτος όρος, εάν όλα τα γειτονικά του τετράγωνα έχουν τιμή 0, (όπως για παράδειγμα η πύλη XOR των 4 εισόδων)
- Κάθε 2 γειτονικά τετράγωνα με τιμή 1 αντιστοιχούν σε έναν όρο με 3 λεκτικά
- Κάθε 4 γειτονικά τετράγωνα με τιμή 1 αντιστοιχούν σε έναν όρο με 2 λεκτικά
- Κάθε 8 γειτονικά τετράγωνα με τιμή 1 αντιστοιχούν σε έναν όρο με 1 λεκτικό
- Κάθε 16 γειτονικά τετράγωνα με τιμή 1 αντιστοιχούν στη συνάρτηση F = 1

Y

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

$$Y = \overline{A}C + \overline{A}BD + A\overline{B}\,\overline{C} + \overline{B}\,\overline{D}$$

Δυαδικοί κώδικες για δεκαδικούς αριθμούς

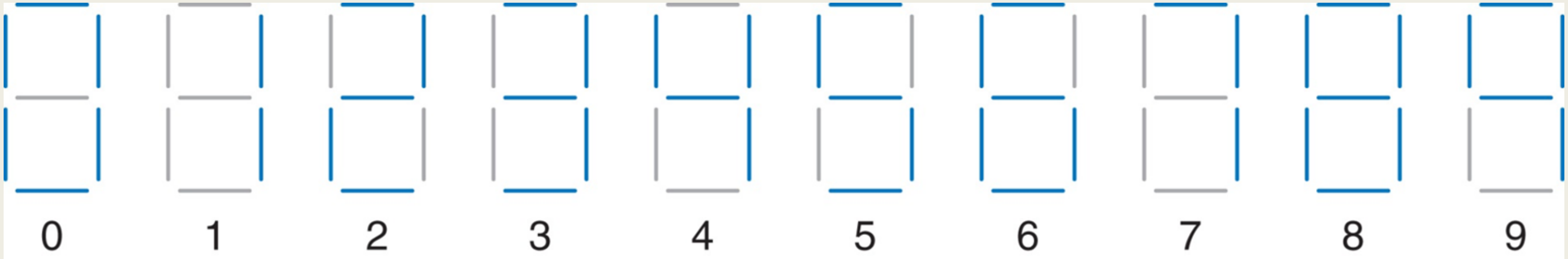
- Ο **κώδικας BCD (Binary-Coded-Decimal)** έχει 4 δυαδικά ψηφία και χρησιμοποιείται για την κωδικοποίηση των μονοψήφιων δεκαδικών αριθμών από το 0 μέχρι το 9.
  - *Τα βάρη στον κώδικα BCD είναι 8-4-2-1, όπως των δυαδικών αριθμών*
  - *Στην πράξη χρησιμοποιείται όπου χειριζόμαστε δεκαδικούς αριθμούς,*
    - Στους αποκωδικοποιητές απεικόνισης επτά τμημάτων
    - Στην αναπαράσταση του διψήφιου δεκαδικού κλασματικού μέρους με ένα byte στις χρηματοοικονομικές εφαρμογές
- Παράδειγμα: Το 37_{10} γράφεται ως 00110111_{BCD}

Παράδειγμα 2.10

- **Αποκωδικοποιητής απεικόνισης επτά τμημάτων**
- Δέχεται ένα δίαυλο δεδομένων $D_{3:0}$ των 4 bit στον κώδικά BCD
- Παράγει ένα δίαυλο σημάτων ελέγχου $S_a – S_g$ των 7 bit που ελέγχουν τα **επτά τμήματα *a* έως *g*** *που σχηματίζουν ένα* ***δεκαδικό ψηφίο (0 – 9)***
- *Κάθε τμήμα υλοποιείται με ένα LED. Εάν το αντίστοιχο σήμα ελέγχου έχει την τιμή 1 το LED ανάβει, αλλιώς είναι σβηστό*
- Συμπληρώστε τον πίνακα αληθείας
- Χρησιμοποιήστε χάρτες Karnaugh προκειμένου να βρείτε τις ελαχιστο-ποιημένες εξισώσεις Boole για τις εξόδους S_a και S_b .
- Υποθέστε ότι όλα τα τμήματα παραμένουν «σβηστά» για τις μη αποδεκτές τιμές εισόδου (10–15).

Παράδειγμα 2.10

- **Αποκωδικοποιητής απεικόνισης επτά τμημάτων**
- Σχηματισμός ενός δεκαδικού αριθμού με επτά τμήματα



Παράδειγμα 2.10

- **Αποκωδικοποιητής απεικόνισης επτά τμημάτων**
  - Πίνακας αληθείας

$D_{3:0}$	S_a	S_b	S_c	S_d	S_e	S_f	S_g
0000	1	1	1	1	1	1	0
0001	0	1	1	0	0	0	0
0010	1	1	0	1	1	0	1
0011	1	1	1	1	0	0	1
0100	0	1	1	0	0	1	1
0101	1	0	1	1	0	1	1
0110	1	0	1	1	1	1	1
0111	1	1	1	0	0	0	0
1000	1	1	1	1	1	1	1
1001	1	1	1	0	0	1	1
άλλες	0	0	0	0	0	0	0

Παράδειγμα 2.10

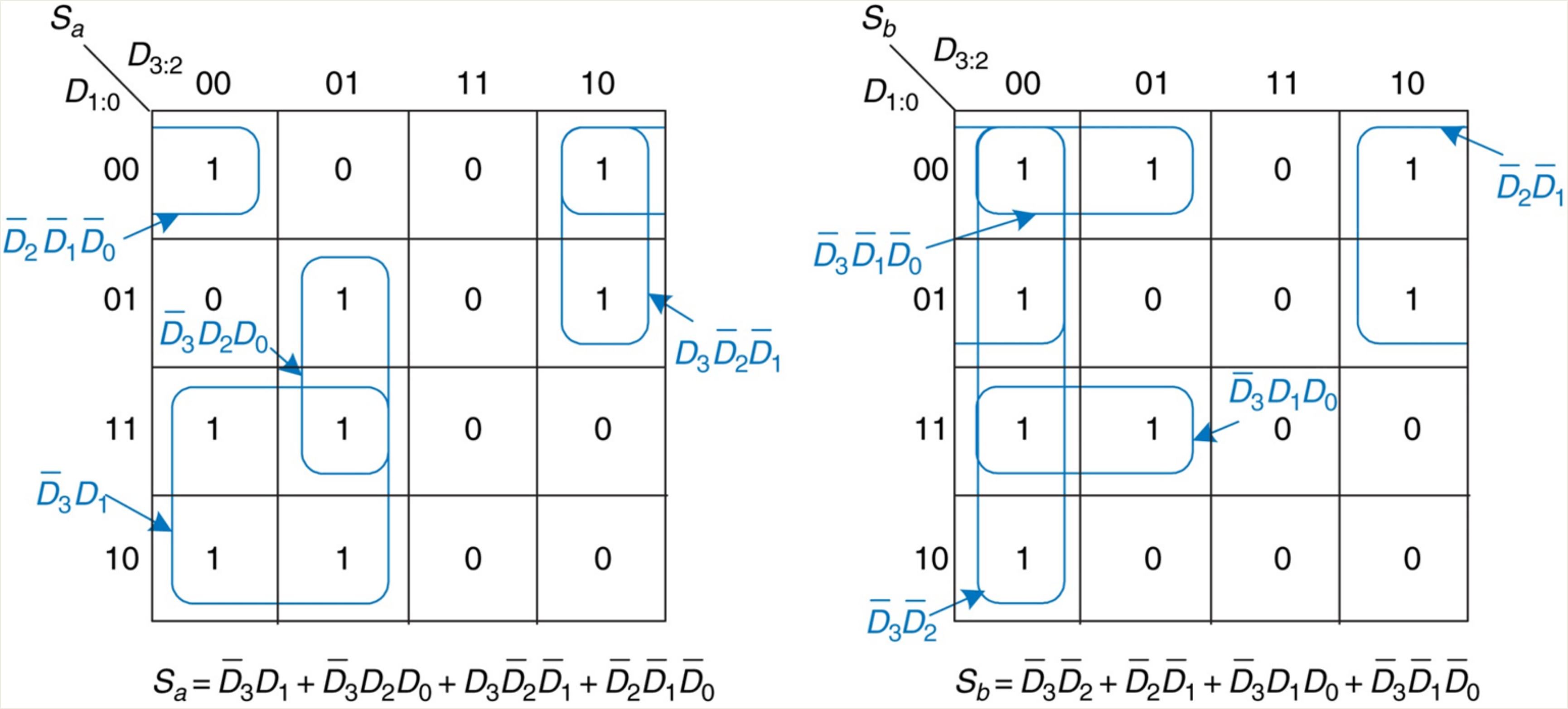
- **Αποκωδικοποιητής απεικόνισης επτά τμημάτων**
- Χάρτες Karnaugh για τις εξόδους S_a και S_b

S_a: $D_{1:0}$ \ $D_{3:2}$	00	01	11	10
00	1	0	0	1
01	0	1	0	1
11	1	1	0	0
10	1	1	0	0

S_b: $D_{1:0}$ \ $D_{3:2}$	00	01	11	10
00	1	1	0	1
01	1	0	0	1
11	1	1	0	0
10	1	0	0	0

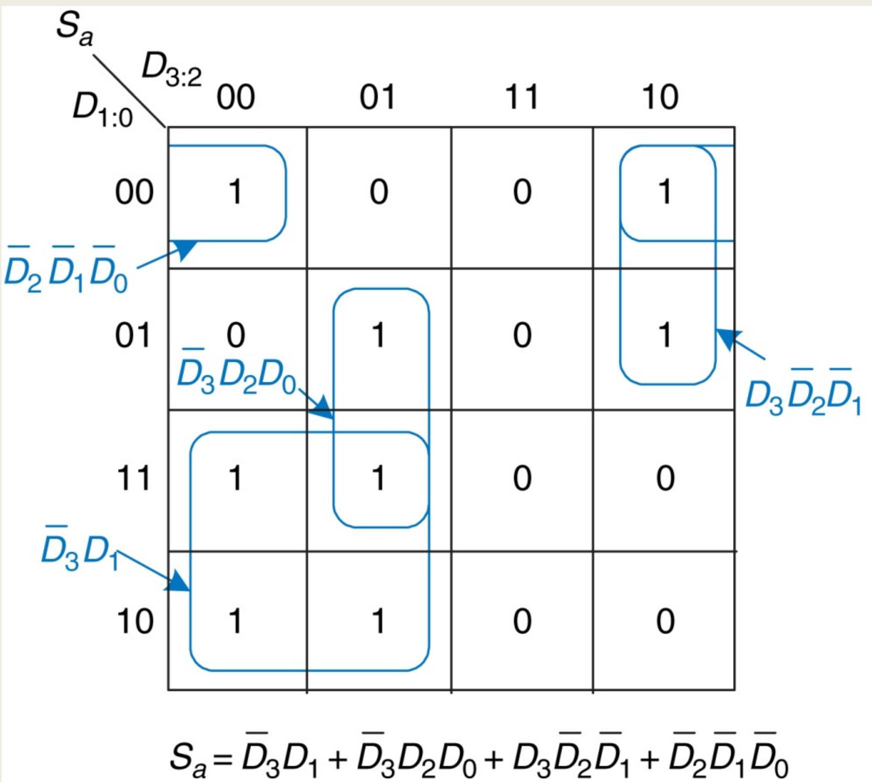
Παράδειγμα 2.10

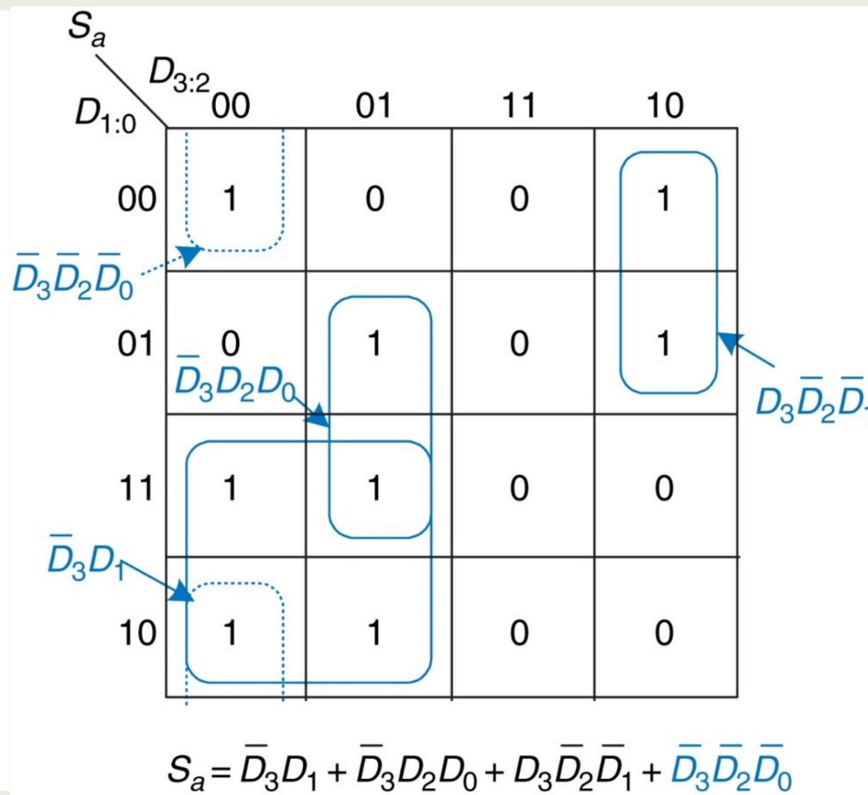
- ## Αποκωδικοποιητής απεικόνισης επτά τμημάτων
- Απλοποιημένες εξισώσεις Boole για τις εξόδους S_a και S_b



Παράδειγμα 2.10

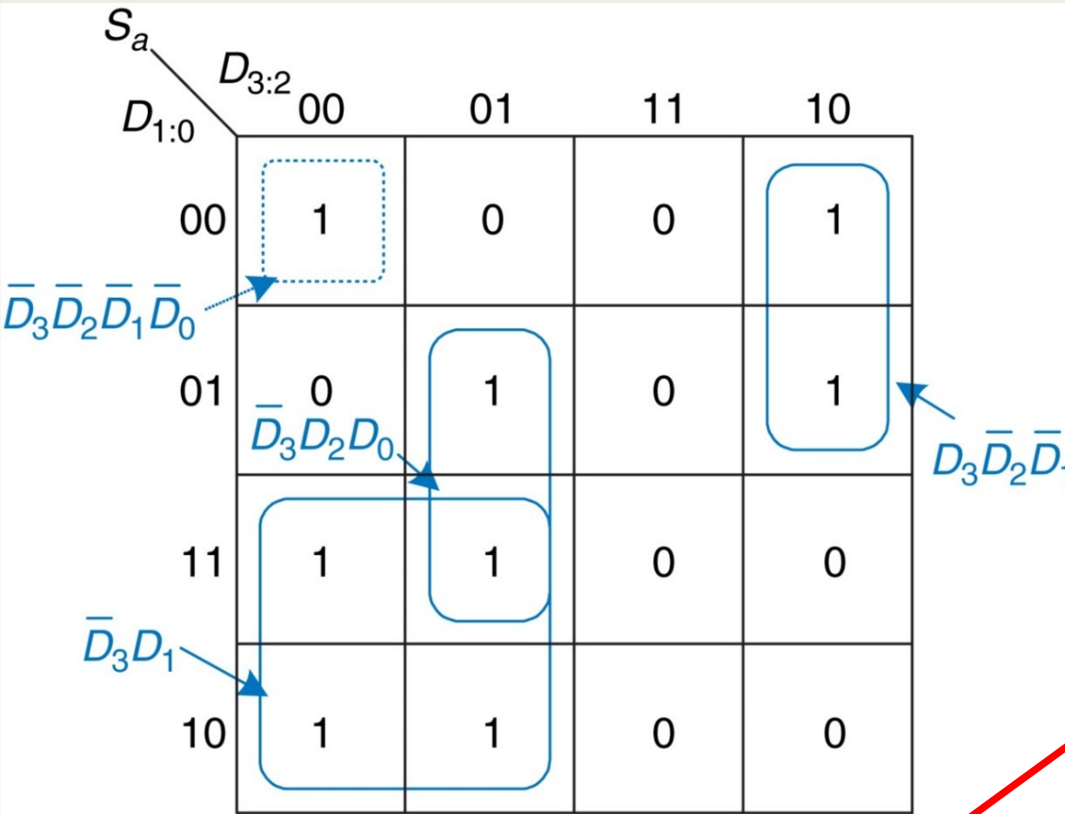
- **Αποκωδικοποιητής απεικόνισης επτά τμημάτων**
- Απλοποιημένη εξίσωση Boole για την έξοδο S_a (εναλλακτική λύση)





Παράδειγμα 2.10

- **Αποκωδικοποιητής απεικόνισης επτά τμημάτων**
- Απλοποιημένη εξίσωση Boole για την έξοδο S_a (εσφαλμένη λύση)



$$S_a = \overline{D}_3D_1 + \overline{D}_3D_2D_0 + D_3\overline{D}_2\overline{D}_1 + \boxed{\overline{D}_3\overline{D}_2\overline{D}_1\overline{D}_0}$$

Δεν ελήφθη υπόψη η αναδίπλωση του χάρτη Karnaugh: τα ακραία είναι γειτονικά

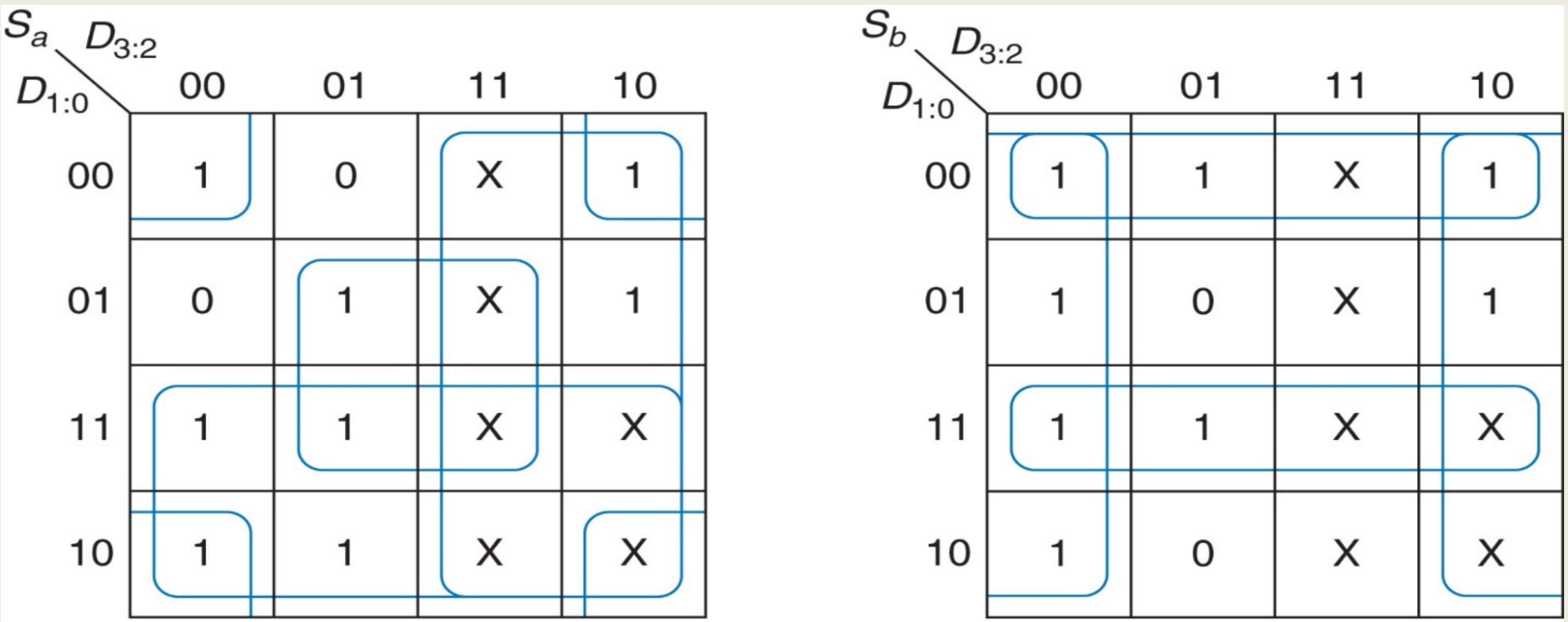
Δεν είναι πρώτος όρος

Αδιάφορες τιμές στην έξοδο

- Οι αδιάφορες τιμές (σύμβολο X) στις εισόδους σε έναν πίνακα αλήθειας χρησιμοποιούνται για να μειώσουμε το πλήθος των γραμμών του πίνακα, όταν κάποιες μεταβλητές δεν επηρεάζουν την έξοδο
- Τέτοιες αδιάφορες τιμές μπορούν επίσης να εμφανίζονται και σε εξόδους ενός πίνακα αληθείας όταν η τιμή της εξόδου δεν παίζει ρόλο ή όταν ο αντίστοιχος συνδυασμός εισόδων δεν εμφανίζεται σε κανονική λειτουργία του κυκλώματος
- Σε έναν χάρτη Karnaugh, τα σύμβολα X μας επιτρέπουν να προχωρήσουμε σε περαιτέρω ελαχιστοποίηση των λογικών εξισώσεων
- **Μπορούμε να κυκλώνουμε τα X αν αυτό μας βοηθάει να καλύπτουμε τα τετράγωνα με τιμή 1 με λιγότερους κύκλους ή κύκλους μεγαλύτερου μεγέθους, αλλά δεν χρειάζεται να τα κυκλώνουμε όταν αυτό δεν χρησιμεύει σε κάτι τέτοιο**
  - *Οι κυκλωμένες αδιάφορες τιμές εκλαμβάνονται ως 1, ενώ οι μη κυκλωμένες εκλαμβάνονται ως 0.*
  - *Η χρήση αδιάφορων τιμών ελαχιστοποιεί σημαντικά τη λογική*

Παράδειγμα 2.11

- **Αποκωδικοποιητής απεικόνισης επτά τμημάτων**
- Απλοποιημένες εξισώσεις Boole για τις εξόδους S_a και S_b λαμβάνοντας υπόψη ότι οι τιμές των εξόδων είναι αδιάφορες για τις εισόδους που δεν ανήκουν στον κώδικα BCD (1010 – 1111)



$S_a = D_3 + D_2D_0 + \overline{D}_2\overline{D}_0 + D_1$ $\qquad$ $S_b = \overline{D}_2 + D_1D_0 + \overline{D}_1\overline{D}_0$

$S_a = \overline{D}_3D_1 + \overline{D}_3D_2D_0 + D_3\overline{D}_2\overline{D}_1 + \overline{D}_2\overline{D}_1\overline{D}_0$ $\qquad$ $S_b = \overline{D}_3\overline{D}_2 + \overline{D}_2\overline{D}_1 + \overline{D}_3D_1D_0 + \overline{D}_3\overline{D}_1\overline{D}_0$

Επιλεγμένες ασκήσεις

- **Άσκηση 2.34**
  Ολοκληρώστε τη σχεδίαση των **τμημάτων S_c έως S_g** για τον αποκωδικοποιητή απεικόνισης επτά τμημάτων
  - *Βρείτε εξισώσεις Boole για τις εξόδους Sc έως Sg, υποθέτοντας ότι οι είσοδοι με τιμή μεγαλύτερη από 9 είναι «αδιάφορες».*
- Σχεδιάστε μια ευλόγως απλή υλοποίηση του αποκωδικοποιητή απεικόνισης επτά τμημάτων σε επίπεδο πυλών για όλες τις εξόδους του
  - *Όπου είναι απαραίτητο, επιτρέπεται η κοινή χρήση πυλών από πολλές εξόδους*

Μείωση υλικού με τη χρήση πυλών XOR

- **Παραδείγματα χαρτών Karnaugh που υλοποιούνται και με τη χρήση πυλών XOR των δύο εισόδων:**
  - *Βρείτε αρχικά την* ***εξίσωση Boole*** *στην απλοποιημένη μορφή* ***αθροίσματος γινομένων δύο επιπέδων***
  - *Στη συνέχεια προχωρήστε την ελαχιστοποίηση, ώστε να χρησιμοποιήσετε την* ***πύλη XOR/XNOR δύο εισόδων***
  - *Τέλος, σχεδιάστε το* ***σχηματικό διάγραμμα***

- Παράδειγμα: **Πύλη XOR με 3 εισόδους**
  - *Εξίσωση Boole:*

$$Y = \bar{A}\bar{B}C + \bar{A}B\bar{C} + A\bar{B}\bar{C} + ABC$$

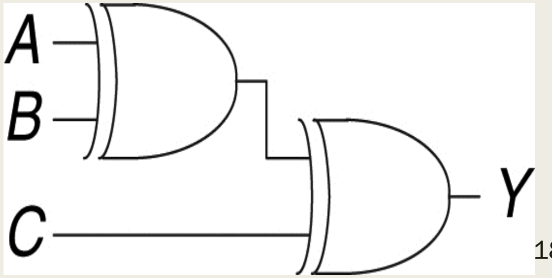
  - *Διαδικασία ελαχιστοποίησης με XOR:*

$$Y = \bar{A}\,(\bar{B}C + B\bar{C}) + A\,(\bar{B}\bar{C} + BC)$$
$$= \bar{A}\,(B \oplus C) + A\,(\overline{B \oplus C})$$
$$= A \oplus B \oplus C = (A \oplus B) \oplus C$$

  - *Υλοποίηση με 2 πύλες XOR των 2 εισόδων*

Domino

C \ AB	00	01	11	10
0	0	1	0	1
1	1	0	1	0



Μείωση υλικού με τη χρήση πυλών XOR

- Παραδείγματα χαρτών Karnaugh που υλοποιούνται και με τη χρήση πυλών XOR των δύο εισόδων. Να βρείτε τις πιο ελαχιστοποιημένες εξισώσεις

$Y = \overline{A \oplus B \oplus C \oplus D}$

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

(α)

$Y = \overline{B \oplus D}$

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

(β)

$Y = B \oplus (CD)$

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

(γ)

$Y = B \oplus C \oplus D$

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

(δ)

$Y = \overline{A}\overline{(B \oplus C)}$

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

(ε)

$Y = (AB) \oplus (CD)$

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

(στ)

Επιλεγμένες ερωτήσεις

- **Ερώτηση 2.2**
  Σχεδιάστε ένα κύκλωμα το οποίο να προσδιορίζει αν ένας δεδομένος μήνας έχει 31 μέρες
  - *Ο μήνας καθορίζεται από μια είσοδο $A_{3:0}$ των 4 bit*
    - Για παράδειγμα, αν η είσοδος είναι 0001, ο μήνας είναι ο Ιανουάριος, και αν η είσοδος είναι 1100, ο μήνας είναι ο Δεκέμβριος.
  - *Η έξοδος Y του κυκλώματος πρέπει να έχει την τιμή '1' HIGH μόνο όταν ο μήνας που καθορίζεται από την είσοδο έχει 31 μέρες*

Γράψτε την ελαχιστοποιημένη εξίσωση και σχεδιάστε το διάγραμμα του κυκλώματος χρησιμοποιώντας τον ελάχιστο δυνατό αριθμό πυλών

Month	A_3	A_2	A_1	A_0	Y
Jan	0	0	0	1	1
Feb	0	0	1	0	0
Mar	0	0	1	1	1
Apr	0	1	0	0	0
May	0	1	0	1	1
Jun	0	1	1	0	0
Jul	0	1	1	1	1
Aug	1	0	0	0	1
Sep	1	0	0	1	0
Oct	1	0	1	0	1
Nov	1	0	1	1	0
Dec	1	1	0	0	1

Y

$A_{1:0}$ \ $A_{3:2}$	00	01	11	10
00	X	0	1	1
01	1	1	X	0
11	1	1	X	0
10	0	0	X	1

$$Y = \overline{A}_3 A_0 + A_3 \overline{A}_0 = A_3 \oplus A_0$$

Κώδικας Gray

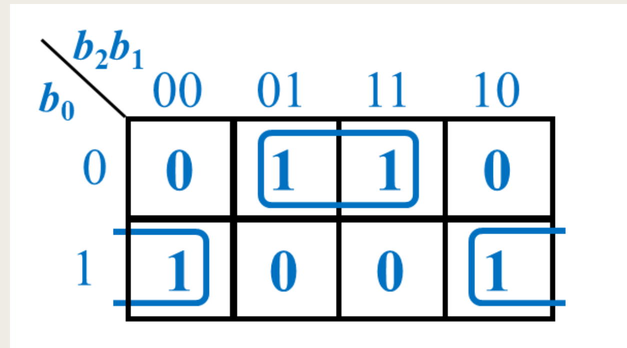
- Ο **κώδικας Gray των 3 bit (g_2, g_1, g_0)** παράγεται από τον **δυαδικό κώδικα των 3 bit (b_2, b_1, b_0)** ως εξής:
  - ***Εάν $b_1 = b_0$, τότε $g_0 = 0$, αλλιώς $g_0 = 1$.***
  - ***Εάν $b_2 = b_1$, τότε $g_1 = 0$, αλλιώς $g_1 = 1$.***
  - ***Για το περισσότερο σημαντικό ψηφίο $g_2 = b_2$***
- Η ίδια διαδικασία παραγωγής του κώδικα Gray από το δυαδικό κώδικα ακολουθείται ανεξάρτητα από το πλήθος των ψηφίων.
- Το σημαντικό χαρακτηριστικό του κώδικα Gray είναι ότι δύο διαδοχικές κωδικές λέξεις του διαφέρουν μόνο κατά ένα ψηφίο
  - *Χρησιμοποιείται στους ADCs, για να αποφευχθούν ενδιάμεσες τιμές, όπου τα προκύπτοντα ψηφιακά δεδομένα αυξάνονται ή μειώνονται κατά 1.*
  - *Χρησιμοποιείται στην κωδικοποίηση καταστάσεων των ακολουθιακών κυκλωμάτων για **μείωση κατανάλωσης ισχύος***

Κύκλωμα μετατροπής από τον δυαδικό στον Gray

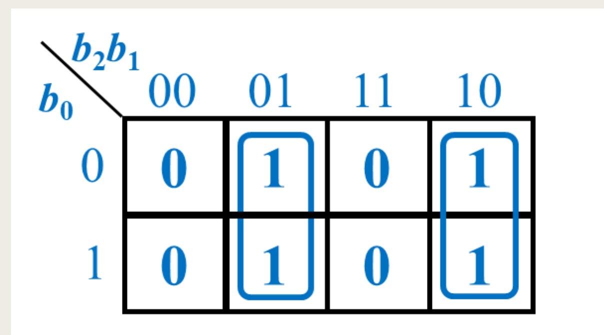
- Πίνακας αληθείας

$b_2b_1b_0$	$g_2g_1g_0$
000	000
001	001
010	011
011	010
100	110
101	111
110	101
111	100

- Χάρτες Karnaugh και εξισώσεις Boole



  - $g_0 = b_1 \oplus b_0$



  - $g_1 = b_2 \oplus b_1$
  - $g_2 = b_2$

Λογικοί συνθετητές

- Η άλγεβρα Boole και οι χάρτες Karnaugh αποτελούν δύο μεθόδους ελαχιστοποίησης λογικών εξισώσεων.
- Μια σύγχρονη πρακτική των σχεδιαστών ψηφιακών συστημάτων είναι να χρησιμοποιούν προγράμματα υπολογιστή, τα οποία ονομάζονται λογικοί συνθετητές (logic synthesizers), για να παράγουν ελαχιστοποιημένα κυκλώματα από μια περιγραφή της λογικής συνάρτησης.
- Για σύνθετα προβλήματα σχεδίασης, οι λογικοί συνθετητές είναι πολύ πιο αποδοτικοί από τους ανθρώπους.

Δομικά στοιχεία συνδυαστικής λογικής

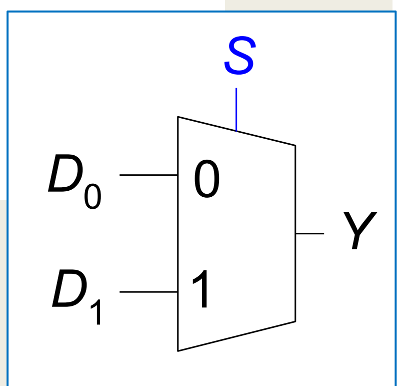
- Η συνδυαστική λογική συχνά ομαδοποιείται σε μεγαλύτερα δομικά στοιχεία για την κατασκευή πιο σύνθετων συστημάτων
- Αυτή είναι μια εφαρμογή της αρχής της «αφαίρεσης»
  - *αποκρύπτονται οι περιττές λεπτομέρειες σε επίπεδο πυλών ώστε να δοθεί έμφαση στη συνάρτηση του δομικού στοιχείου*
- Τέτοια κυκλώματα που έχουμε ήδη αναφέρει είναι:
  - *ο **πλήρης αθροιστής***
  - *το **κύκλωμα και ο κωδικοποιητής προτεραιότητας***
  - *ο **αποκωδικοποιητής απεικόνισης επτά τμημάτων***
- Στη συνέχεια θα επικεντρωθούμε:
  - *στους **πολυπλέκτες***
  - *στους **δυαδικούς αποκωδικοποιητές**, και*
  - *στους **αποπλέκτες***

Πολυπλέκτης 2 σε 1

- Ο **πολυπλέκτης 2 σε 1** επιλέγει τη μία από τις δύο εισόδους δεδομένων D_0 και D_1 με βάση την τιμή του σήματος **S** (select)
  - *αν* $S = 0$, *τότε* $Y = D_0$
  - *αλλιώς, αν* $S = 1$, *τότε* $Y = D_1$

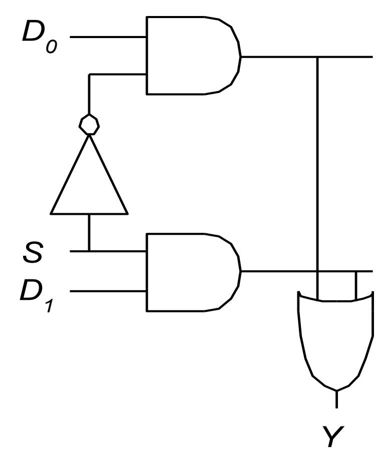
S	D_1	D_0	Y
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1

S	Y
0	D_0
1	D_1



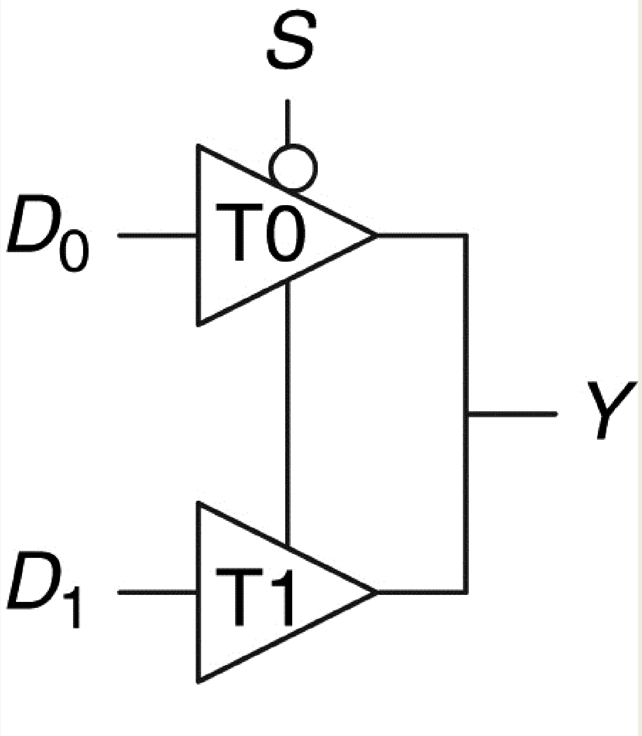
S \ D_0D_1	00	01	11	10
0	0	0	1	1
1	0	1	1	0

$$Y = D_0\overline{S} + D_1S$$



Πολυπλέκτης 2 σε 1

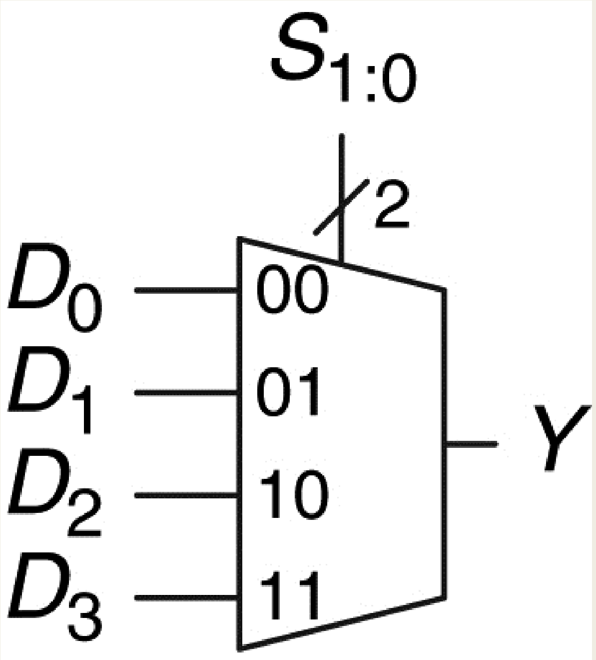
- Εναλλακτικά, πολυπλέκτες μπορούν να κατασκευαστούν από **απομονωτές τριών καταστάσεων**
- Τα σήματα έγκρισης (enable) τριών καταστάσεων είναι διατεταγμένα έτσι ώστε, σε όλες τις περιπτώσεις, να είναι ενεργός ακριβώς ένας απομονωτής τριών καταστάσεων
- Όταν ισχύει $S = 0$, είναι ενεργοποιημένος ο απομονωτής T0, γεγονός που επιτρέπει στην τιμή του D_0 να μεταφερθεί στην έξοδο Y
- Όταν ισχύει $S = 1$, είναι ενεργοποιημένος ο απομονωτής T1, γεγονός που επιτρέπει στην τιμή του D_1 να μεταφερθεί στην έξοδο Y



$$Y = D_0\overline{S} + D_1 S$$

Πολυπλέκτης 4 σε 1

- Ο **πολυπλέκτης 4 σε 1** επιλέγει τη μία από τις τέσσερεις εισόδους δεδομένων D_0, D_1, D_2 και D_3 με βάση την τιμή του διαύλου επιλογής $S_{1:0}$
  - *αν* $S_{1:0} = 00$, *τότε* $Y = D_0$
  - *αν* $S_{1:0} = 01$, *τότε* $Y = D_1$
  - *αν* $S_{1:0} = 10$, *τότε* $Y = D_2$
  - *αν* $S_{1:0} = 11$, *τότε* $Y = D_3$

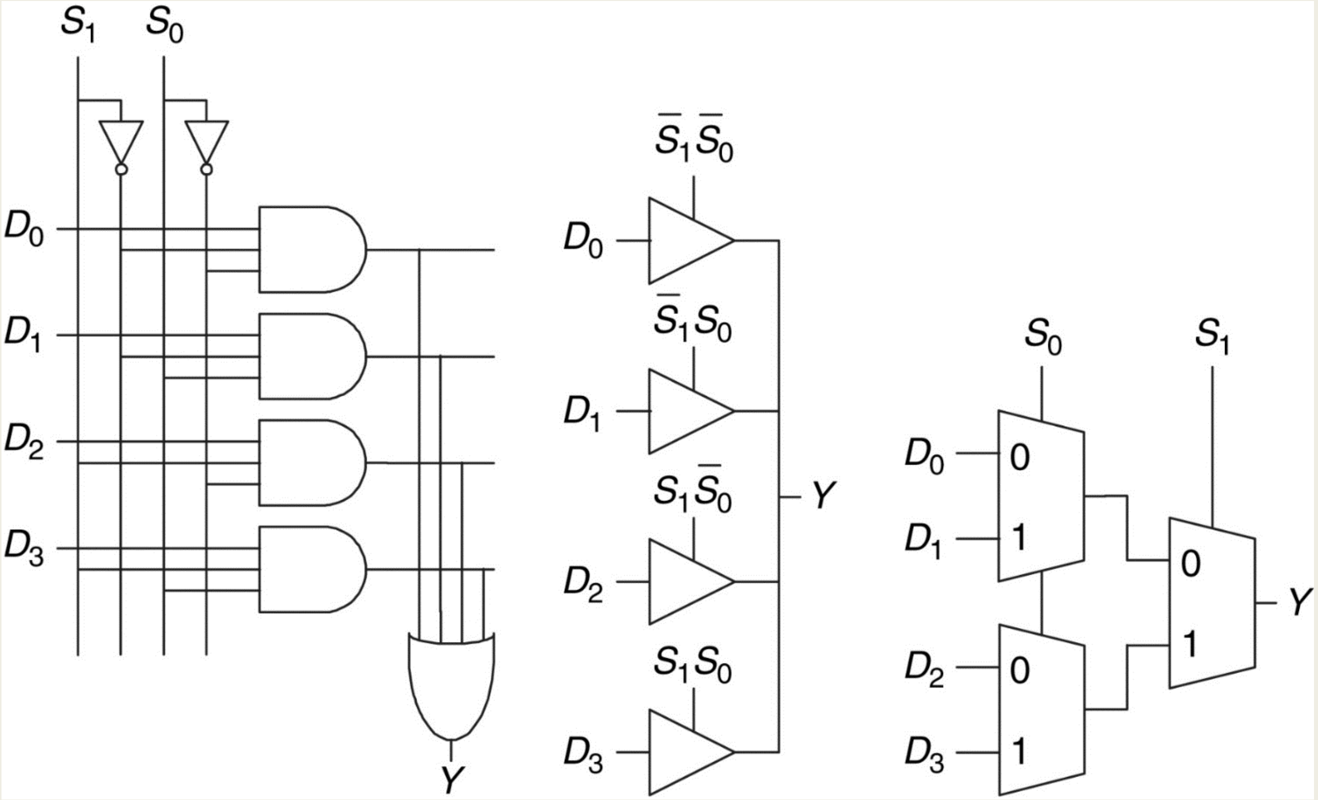


- Εξίσωση Boole του **πολυπλέκτη 4 σε 1**:

$$Y = \bar{S}_1\bar{S}_0 D_0 + \bar{S}_1 S_0 D_1 + S1\bar{S}_0 D_2 + S1S0D_3$$

Πολυπλέκτης 4 σε 1

- Μπορούμε να κατασκευάσουμε τον πολυπλέκτη 4 σε 1 χρησιμοποιώντας (α) λογική αθροίσματος γινομένων, (β) απομονωτές τριών καταστάσεων ή (γ) δένδρο πολυπλεκτών 2 σε 1



Έχουμε τη δυνατότητα να κατασκευάσουμε μεγαλύτερους πολυπλέκτες, επεκτείνοντας αυτές τις τρεις μεθόδους. Ενας πολυπλέκτης $N = 2^n$ σε 1 απαιτεί $\log_2 N = n$ σήματα επιλογής (select).

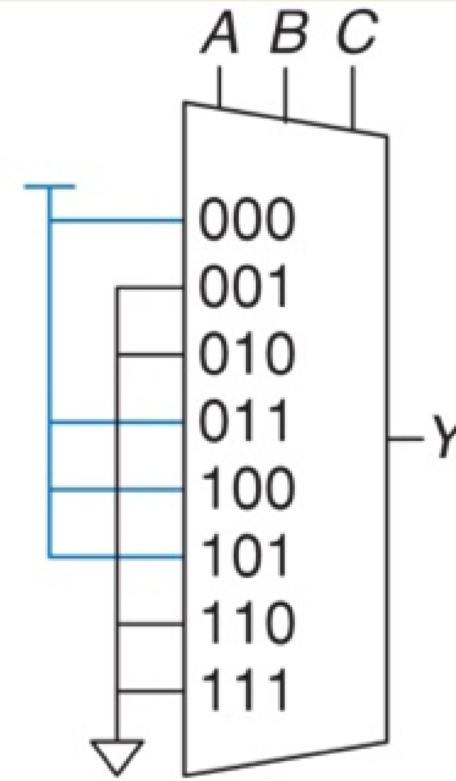
Υλοποίηση συνδυαστικής λογικής

- **Παράδειγμα 4.12**
- Υλοποιήστε την συνάρτηση

  $$Y = A\overline{B} + \overline{B}\,\overline{C} + \overline{A}BC$$

  *με την χρήση μονάχα ενός **πολυπλέκτη 8 σε 1***

A	B	C	Y
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	0



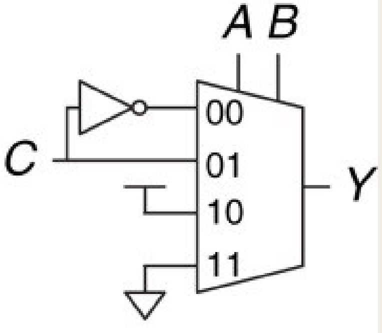
- Ο πολυπλέκτης χρησιμεύει ως **πίνακας αναζήτησης (Look-Up Table – LUT)**, όπου **κάθε γραμμή του πίνακα αληθείας (κάθε ελαχιστόρος) αντιστοιχεί σε μια είσοδο του πολυπλέκτη**
  - *Οι είσοδοι του πολυπλέκτη ταυτίζονται με την έξοδο Y του πίνακα*

Υλοποίηση συνδυαστικής λογικής

- **Παράδειγμα 4.12 (μείωση του πολυπλέκτη στο μισό)**
- Υλοποιήστε την συνάρτηση $\boldsymbol{Y = A\overline{B} + \overline{B}\,\overline{C} + \overline{A}BC}$
  - *Με την χρήση μονάχα ενός* ***πολυπλέκτη 4 σε 1*** *και μίας* ***πύλης NOT***
- Μειώνουμε το μέγεθος του πίνακα αληθείας στο μισό, από 8 γραμμές σε 4 γραμμές επιτρέποντας στην έξοδο να εξαρτάται από την τιμή του *C*
  - *Για κάθε δύο γραμμές του πίνακα αλήθειας εξετάζουμε την τιμή της εξόδου Y με βάση την τιμή της εισόδου C (του LSB)*
  - *Υπάρχουν 4 πιθανές περιπτώσεις:* $\mathbf{Y = C, Y = \overline{C}, Y = 1}$ **και** $\mathbf{Y = 0}$
- Υλοποιούμε τον πολυπλέκτη με βάση τον μειωμένο πίνακα αλήθειας

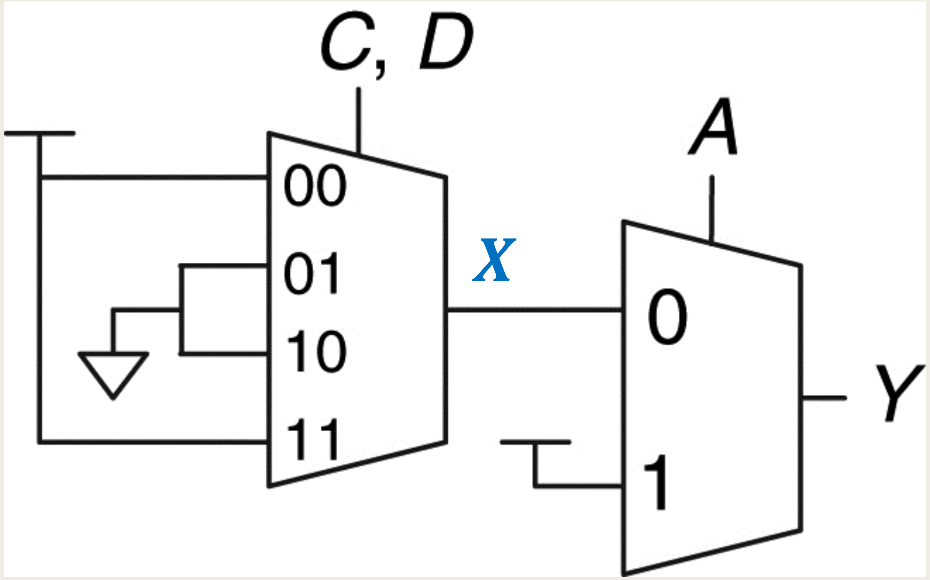
A	B	C	Y
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	0

A	B	Y
0	0	$\overline{C}$
0	1	C
1	0	1
1	1	0



Επιλεγμένες ασκήσεις

- **Άσκηση 2.39**
  Γράψτε μια ελαχιστοποιημένη εξίσωση Boole για τη συνάρτηση που εκτελείται από το κύκλωμα της Εικόνας 2.87



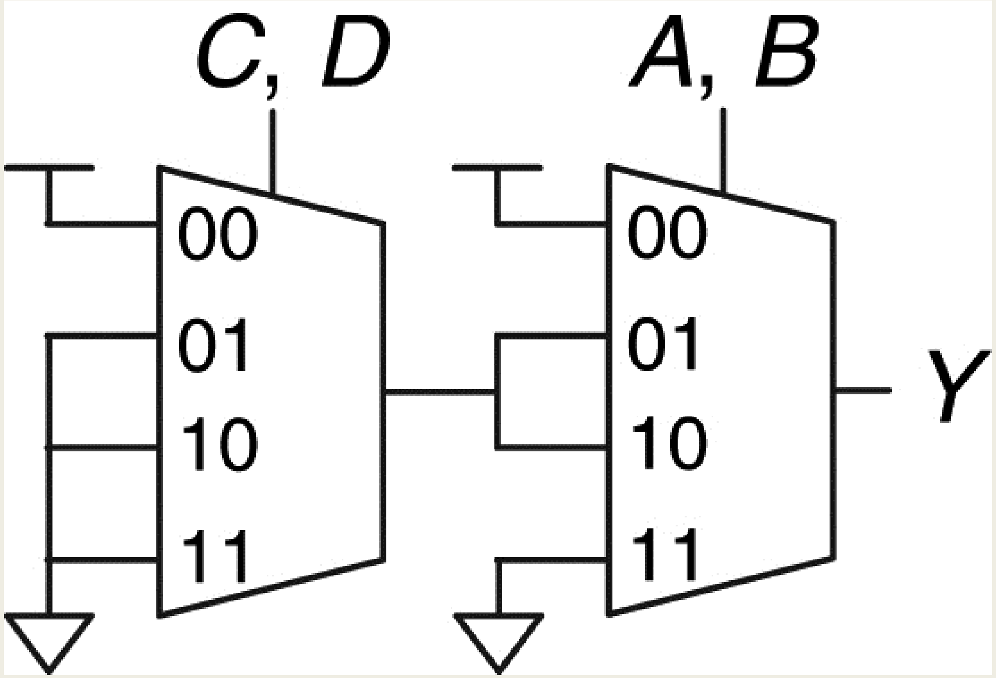
$$\mathbf{X} = \bar{\mathbf{C}}\bar{\mathbf{D}}(1) + \bar{\mathbf{C}}\mathbf{D}(0) + \mathbf{C}\bar{\mathbf{D}}(0) + \mathbf{C}\mathbf{D}(1) = \bar{\mathbf{C}}\bar{\mathbf{D}} + \mathbf{C}\mathbf{D} = \overline{\mathbf{C} \oplus \mathbf{D}}$$

$$\mathbf{Y} = \bar{A}\,(\overline{\mathbf{C} \oplus \mathbf{D}}) + A(1) = \bar{A}(\overline{\mathbf{C} \oplus \mathbf{D}}) + A = (\bar{A} + A)(\overline{\mathbf{C} \oplus \mathbf{D}} + A)$$

$$= (1)(\overline{\mathbf{C} \oplus \mathbf{D}} + A) = \overline{\mathbf{C} \oplus \mathbf{D}} + A = \bar{C}\bar{D} + CD + A$$

Επιλεγμένες ασκήσεις

- **Άσκηση 2.40**
  Γράψτε μια ελαχιστοποιημένη εξίσωση Boole για τη συνάρτηση που εκτελείται από το κύκλωμα της Εικόνας 2.88



$$Y = \overline{C}\,\overline{D}(A \oplus B) + \overline{A}\,\overline{B} = \overline{A}\,\overline{C}\,\overline{D} + \overline{B}\,\overline{C}\,\overline{D} + \overline{A}\,\overline{B}$$

Επιλεγμένες ασκήσεις

- Άσκηση 2.41
  Υλοποιήστε τη συνάρτηση της Εικόνας 2.81(β) χρησιμοποιώντας

  (α) έναν πολυπλέκτη 8 σε 1·

  (β) έναν πολυπλέκτη 4 σε 1 και έναν αντιστροφέα·

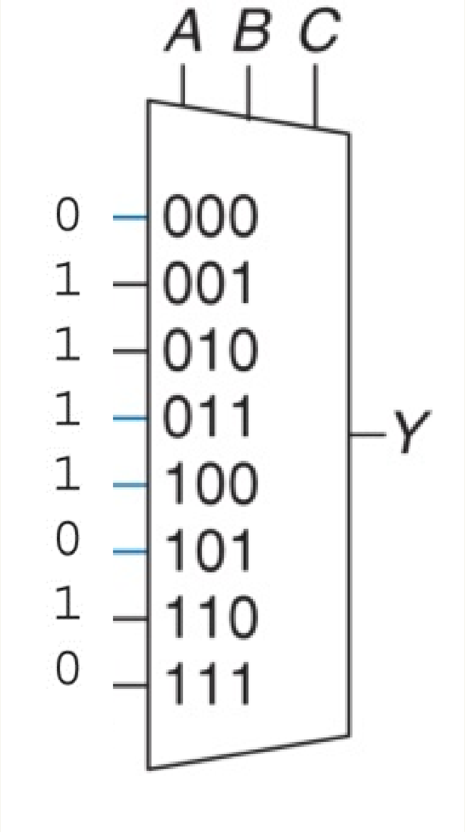
  (γ) έναν πολυπλέκτη 2 σε 1 και δύο άλλες λογικές πύλες

A	B	C	Y
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	0

Επιλεγμένες ασκήσεις

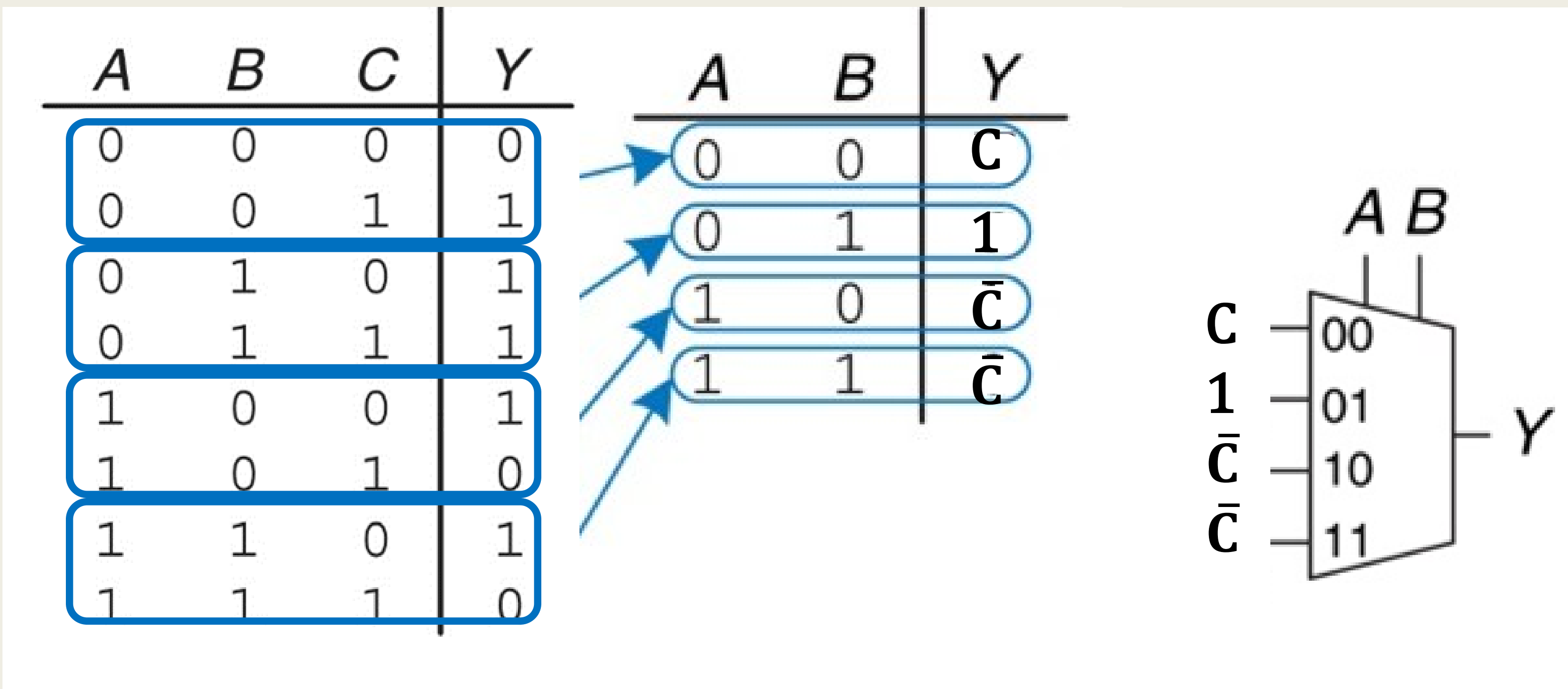
- **Άσκηση 2.41**
  Υλοποιήστε τη συνάρτηση της Εικόνας 2.81(β) χρησιμοποιώντας (α) έναν πολυπλέκτη 8 σε 1·

A	B	C	Y
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	0



Επιλεγμένες ασκήσεις

- **Άσκηση 2.41**
  Υλοποιήστε τη συνάρτηση της Εικόνας 2.81(β) χρησιμοποιώντας (β) έναν πολυπλέκτη 4 σε 1 και έναν αντιστροφέα

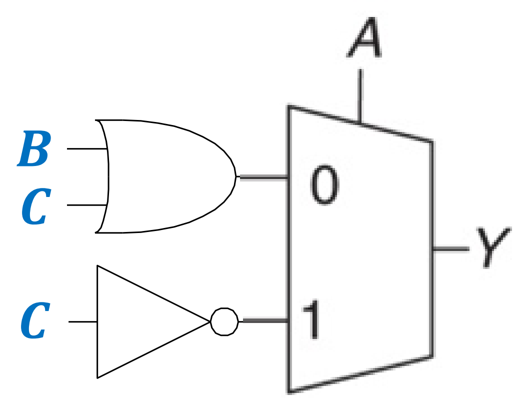


Επιλεγμένες ασκήσεις

- **Άσκηση 2.41**
  Υλοποιήστε τη συνάρτηση της Εικόνας 2.81(β) χρησιμοποιώντας (γ) έναν πολυπλέκτη 2 σε 1 και δύο άλλες λογικές πύλες
  - *Μειώνουμε το μέγεθος του πίνακα αληθείας από 8 γραμμές σε 2 γραμμές επιτρέποντας στην έξοδο να εξαρτάται από τα B και C*
  - *Για κάθε 4 γραμμές του πίνακα αλήθειας βρίσκουμε τη συνάρτηση δύο μεταβλητών B και C που αντιστοιχεί στην έξοδο Y*
    - Υπάρχουν 16 πιθανές συναρτήσεις δύο μεταβλητών
  - *Υλοποιούμε τις δύο προκύπτουσες συναρτήσεις με λογικές πύλες και συνδέουμε τις εξόδους τους στις αντίστοιχες εισόδους του πολυπλέκτη 2 σε 1 με βάση τον μειωμένο πίνακα αλήθειας*

A	B	C	Y
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	0

A	Y
0	$B + C$
1	$\overline{C}$



Δυαδικός αποκωδικοποιητής *N* σε 2^N

- Ένας δυαδικός αποκωδικοποιητής (binary decoder) *N* σε 2^N
  - *δέχεται* ***δυαδικούς αριθμούς των N bit*** ***ως εισόδους*** A_m, $(m = 0,..,N-1)$
  - *και παράγει* 2^N ***εξόδους*** Y_n, $(n = 0,.., 2^N-1)$
  - *Ενεργοποιεί ακριβώς μία από τις εξόδους του* Y_n*, εκείνη που έχει ως δείκτη n τη δεκαδική τιμή του δυαδικού αριθμού στην είσοδο*
- Οι έξοδοι αποκαλούνται **έξοδοι «μοναδικού σημαντικού» (one-hot)**, ή «μονόθερμες», επειδή ακριβώς μία από αυτές είναι «θερμή» (HIGH) σε οποιαδήποτε δεδομένη στιγμή

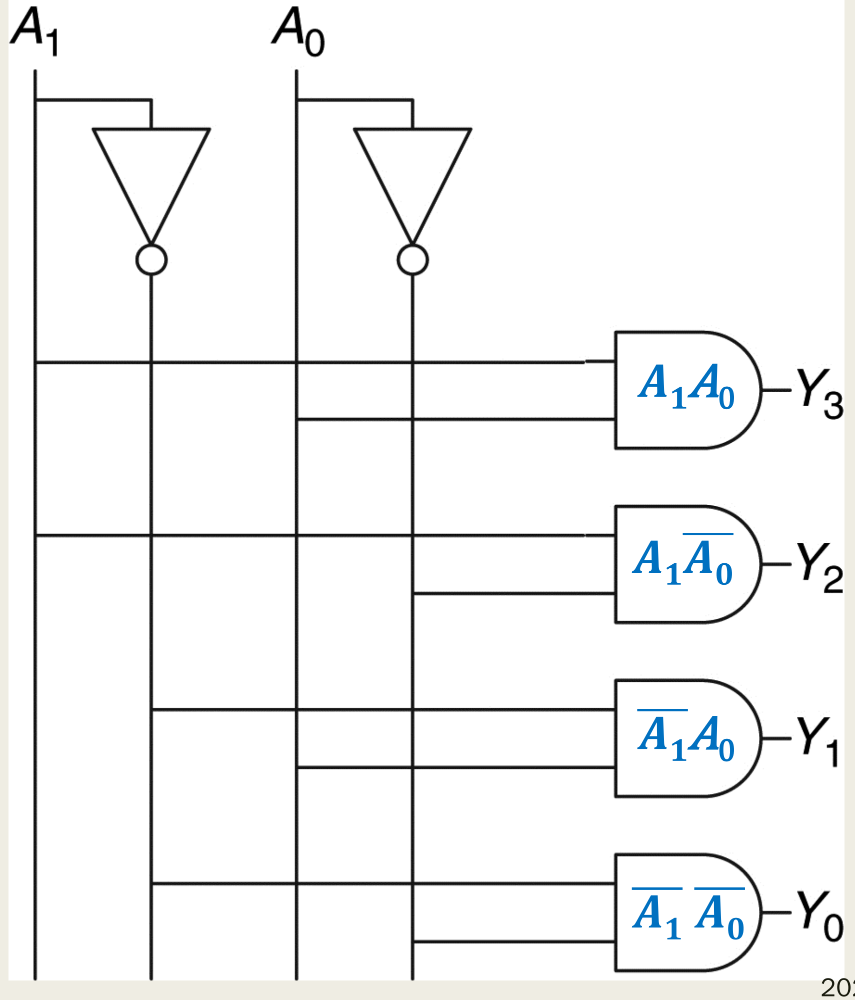
Παράδειγμα: Δυαδικός αποκωδικοποιητής 2 σε 4

A_1	A_0	Y_3	Y_2	Y_1	Y_0	n
0	0	0	0	0	1	0
0	1	0	0	1	0	1
1	0	0	1	0	0	2
1	1	1	0	0	0	3

Υλοποίηση του δυαδικού αποκωδικοποιητή

- Υλοποιείται με **πύλες AND**
- Κάθε πύλη AND εξαρτάται είτε από την αληθινή είτε από τη συμπληρωματική μορφή κάθε εισόδου A_m, $(m = 0,..,N\text{-}1)$
- Ένας **αποκωδικοποιητής *N* σε 2^N** μπορεί να κατασκευαστεί από **2^N πύλες AND των *N* εισόδων**
  - *που δέχονται τους πιθανούς συνδυασμούς των αληθινών ή συμπληρωματικών εισόδων*
- Κάθε **έξοδος Y_n $(n = 0,..., 2^N\text{-}1)$** αναπαριστά τον **ελαχιστόρο m_n** (π.χ., $Y_3 = m_3 = A_0 A_1$)

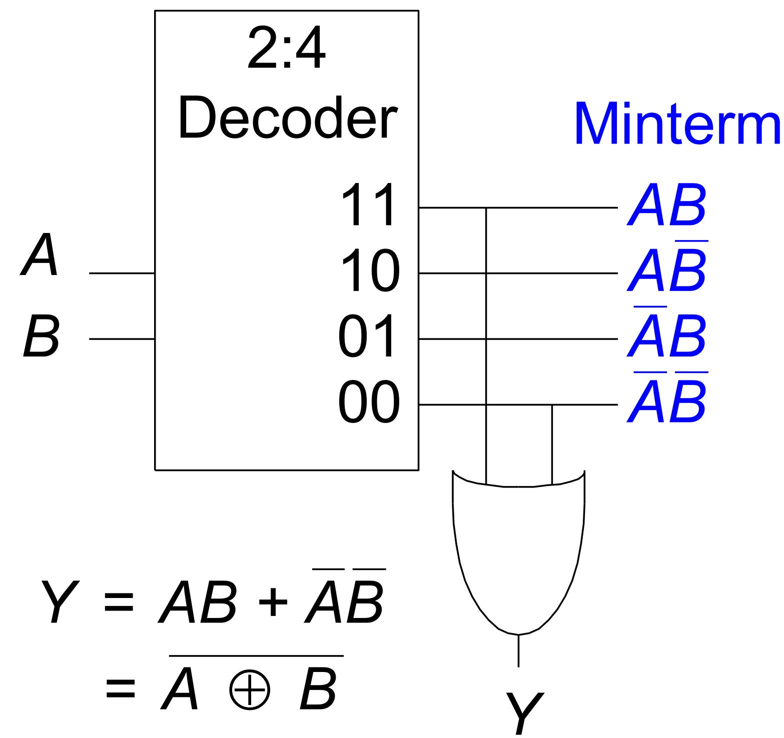
Παράδειγμα: Δυαδικός αποκωδικοποιητής *2* σε *4*



Υλοποίηση συνδυαστικής λογικής

- Οι δυαδικοί αποκωδικοποιητές μπορούν να συνδυαστούν με **πύλες OR** για την υλοποίηση συνδυαστικής λογικής
- Επειδή κάθε έξοδος Y_n ενός αποκωδικοποιητή $(n = 0,\ldots, 2^N-1)$ αναπαριστά τον **ελαχιστόρο m_n**, η συνάρτηση υλοποιείται ως **η πράξη OR όλων των ελαχιστόρων της συνάρτησης**
  - ***που έχουν την τιμή 1*** *στον πίνακα αληθείας*
- Η ίδια ιδέα εφαρμόζεται και στην υλοποίηση συνδυαστικής λογικής με **διατάξεις μνήμης**

Παράδειγμα: Υλοποίηση της πύλης XNOR δύο εισόδων με δυαδικό αποκωδικοποιητή 2 σε 4

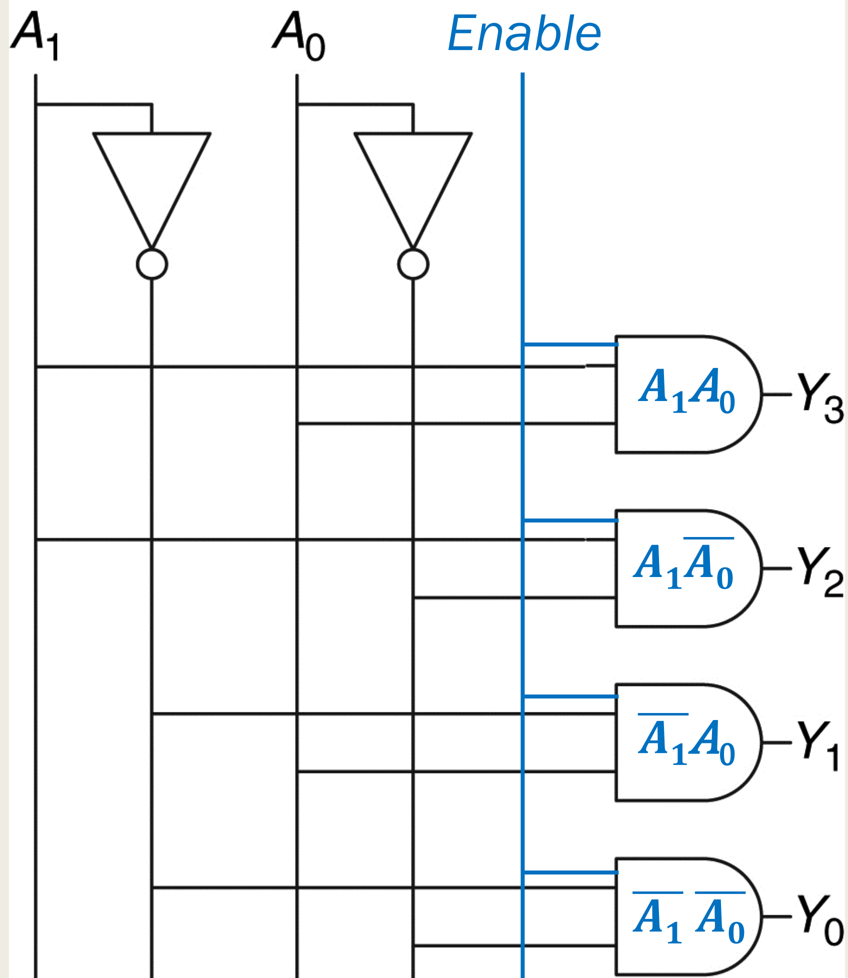


$$Y = AB + \bar{A}\bar{B} = \overline{A \oplus B}$$

Δυαδικός αποκωδικοποιητής με έγκριση

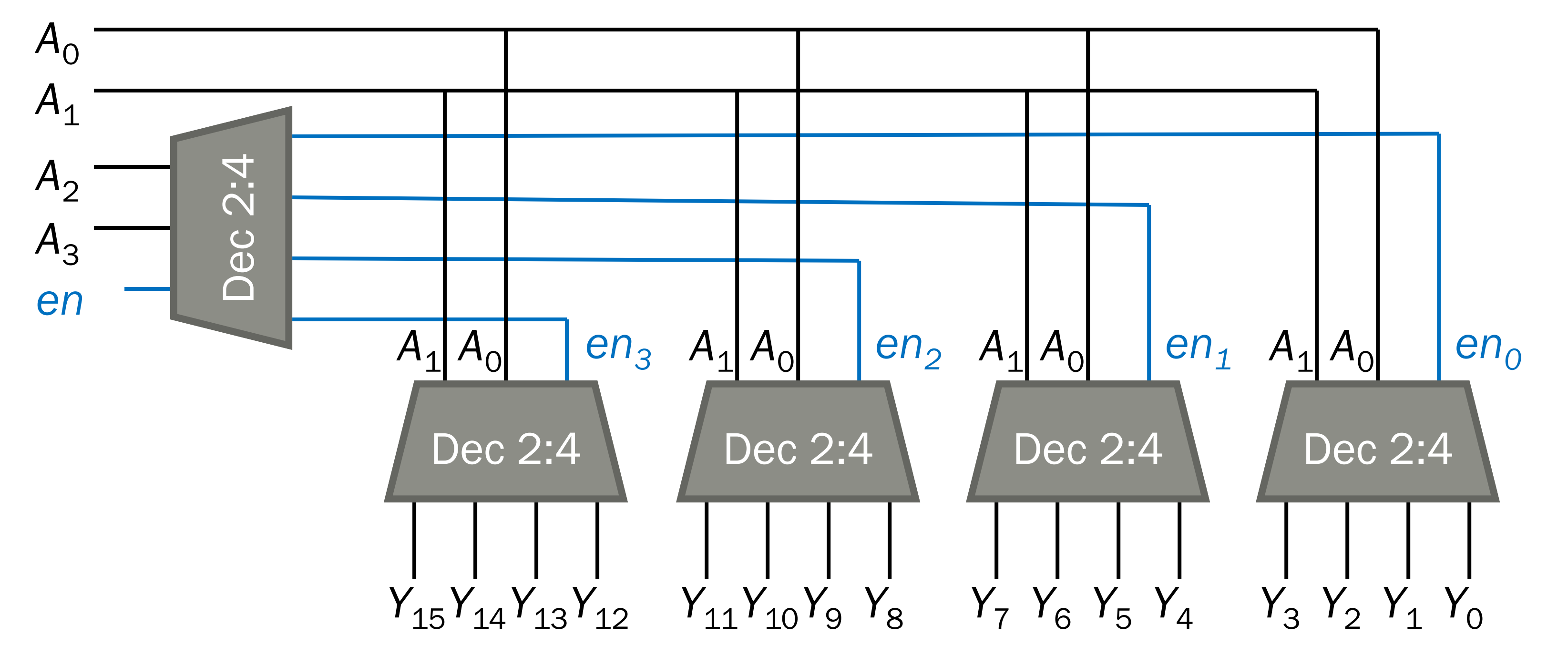
- Το **σήμα enable (έγκρισης)** είναι μία επιπλέον είσοδος του δυαδικού αποκωδικοποιητή
  - *όταν το **σήμα enable έχει τιμή '1' (TRUE) ο αποκωδικοποιητής λειτουργεί κανονικά***
  - *όταν το **σήμα enable έχει τιμή '0' (FALSE) ο αποκωδικοποιητής έχει όλες τις εξόδους μηδενισμένες***
- Ένας **αποκωδικοποιητής N σε 2^N** μπορεί να κατασκευαστεί από **2^N πύλες AND των $N+1$ εισόδων**
  - *που δέχονται τους πιθανούς συνδυασμούς των αληθινών ή συμπληρωματικών εισόδων και το σήμα enable*

Παράδειγμα: Δυαδικός αποκωδικοποιητής 2 σε 4 με enable



Δυαδικός αποκωδικοποιητής με έγκριση

- Το σήμα enable (έγκρισης) είναι μία επιπλέον είσοδος του δυαδικού αποκωδικοποιητή που χρησιμοποιείται όταν κατασκευάζουμε μεγαλύτερους αποκωδικοποιητές από πολλούς μικρότερους
  - *Η χρήση του σήματος enable για επέκταση του μεγέθους ενός κυκλώματος δεν περιορίζεται μόνο στους αποκωδικοποιητές, αλλά εφαρμόζεται και σε άλλα κυκλώματα (μετρητές, μνήμες, κ.α.)*
- Παράδειγμα: **Κατασκευή δυαδικού αποκωδικοποιητή 4 σε 16 με έγκριση χρησιμοποιώντας 5 αποκωδικοποιητές 2 σε 4 με enable**

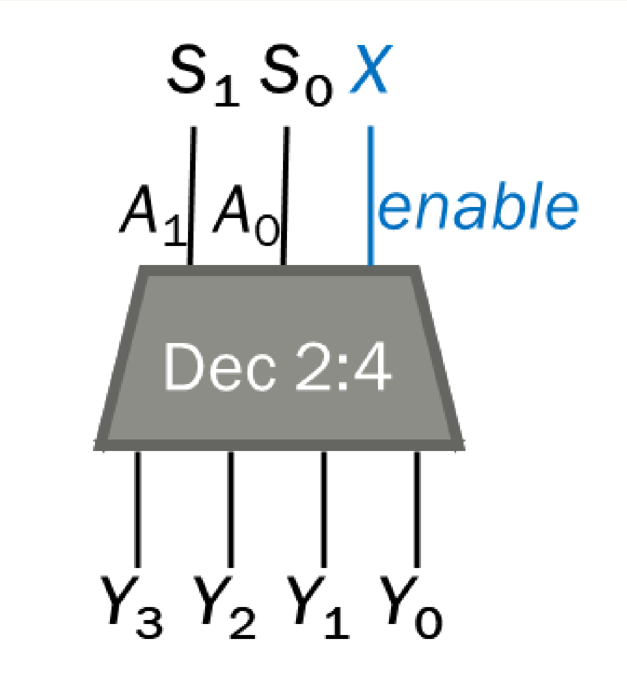


Αποπλέκτης 1 σε 2^N

- Ο **αποπλέκτης (demultiplexer) 1 σε 2^N** συνδέει τη μοναδική **είσοδο δεδομένων *X*** με μία από τις **2^N εξόδους δεδομένων** με βάση την τιμή των ***N* σημάτων επιλογής S (select)**
- Ο **δυαδικός αποκωδικοποιητής *N* σε 2^N με έγκριση** χρησιμοποιείται και ως **αποπλέκτης (demultiplexer) 1 σε 2^N** *ως εξής:*
  - *Το **σήμα enable** του αποκωδικοποιητή είναι η **είσοδος δεδομένων X** του αποπλέκτη*
  - *Οι **N είσοδοι δεδομένων** του αποκωδικοποιητή είναι τα **N σήματα επιλογής** του αποπλέκτη*
  - *Οι **2^N εξοδοι «μοναδικού σημαντικού»** του αποκωδικοποιητή είναι οι **2^N εξοδοι δεδομένων** του αποπλέκτη*

Αποπλέκτης 1 σε 4

- Ο **αποπλέκτης 1 σε 4 συνδέει την είσοδο δεδομένων X** με μία από τις τέσσερεις εξόδους δεδομένων Y_0, Y_1, Y_2 και Y_3 με βάση την τιμή του διαύλου επιλογής $S_{1:0}$
  - *αν* $S_{1:0} = 00$, *τότε* $Y_0 = X$
  - *αν* $S_{1:0} = 01$, *τότε* $Y_1 = X$
  - *αν* $S_{1:0} = 10$, *τότε* $Y_2 = X$
  - *αν* $S_{1:0} = 11$, *τότε* $Y_3 = X$
- Υλοποιείται με έναν **δυαδικό αποκωδικοποιητή 2 σε 4 με έγκριση (enable)**

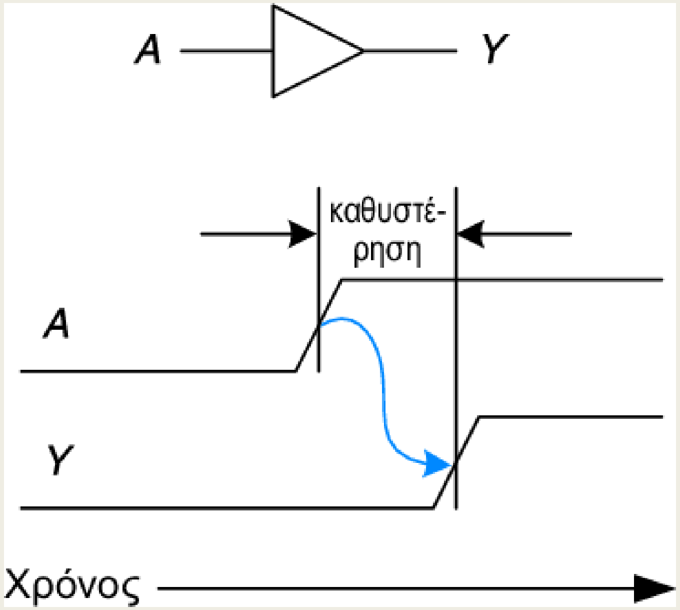


Χρονισμός

- Στη σχεδίαση ενός ψηφιακού κυκλώματος είναι σημαντικά:
  - *η **μείωση του κόστους υλοποίησης** με τη χρήση όσο το δυνατόν λιγότερων δομικών στοιχείων.*
  - *η **αύξηση της ταχύτητας λειτουργίας**, δηλαδή ο **χρονισμός (timing)***

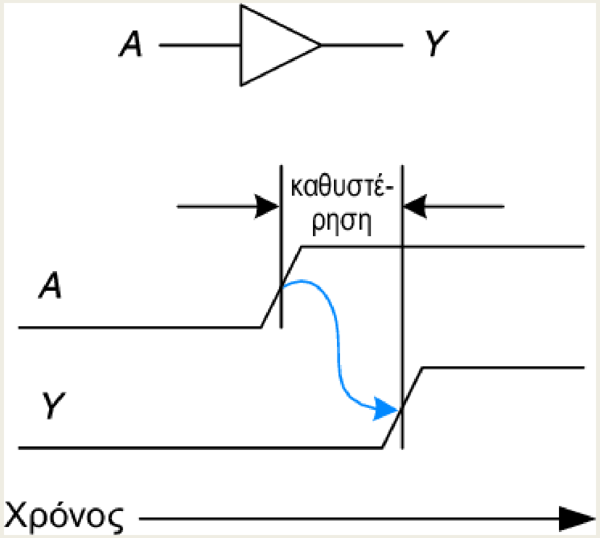
Χρονισμός: καθυστέρηση

- Μια έξοδος χρειάζεται χρόνο για να μεταβληθεί αντιδρώντας στη μεταβολή μιας εισόδου
- Στην εικόνα φαίνεται η καθυστέρηση (delay) (ο χρόνος) που μεσολαβεί από τη μεταβολή μιας εισόδου έως την επακόλουθη μεταβολή της εξόδου για έναν απομονωτή



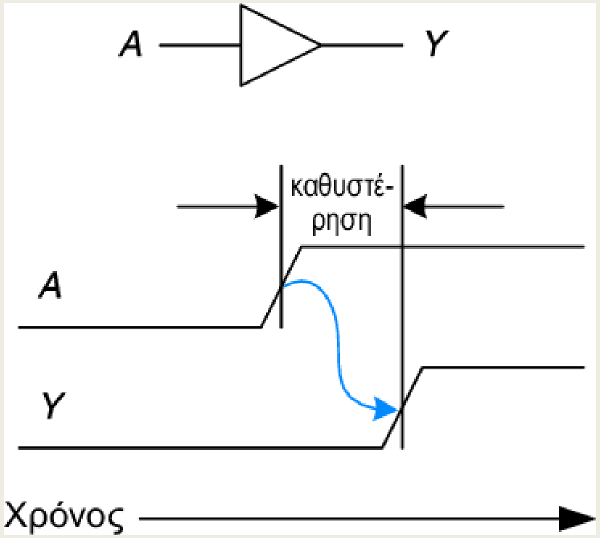
Χρονισμός: διάγραμμα χρονισμού

- Το παρακάτω σχήμα ονομάζεται **διάγραμμα χρονισμού**
  - *απεικονίζει τη **μεταβατική απόκριση** (transient response) του κυκλώματος του απομονωτή όταν μεταβάλλεται μια είσοδος*
  - *Η μετάβαση από την τιμή LOW στην τιμή HIGH ονομάζεται **ανερχόμενη ακμή** (rising edge)*
  - *Η μετάβαση από την τιμή HIGH στην τιμή LOW ονομάζεται **κατερχόμενη ακμή** (falling edge) (δεν φαίνεται στο σχήμα)*



Χρονισμός: διάγραμμα χρονισμού

- Το μπλε βέλος υποδεικνύει ότι η ανερχόμενη ακμή της εξόδου *Y* προκαλείται από την ανερχόμενη ακμή της εισόδου *A*
  - *Μετράμε την καθυστέρηση από το σημείο 50% του σήματος εισόδου A έως το σημείο 50% του σήματος εξόδου Y*
  - *Το σημείο 50% είναι εκείνο στο οποίο το σήμα βρίσκεται ακριβώς στη μέση (50%) της μετάβασης από την τιμή LOW στην τιμή HIGH και αντίστροφα*



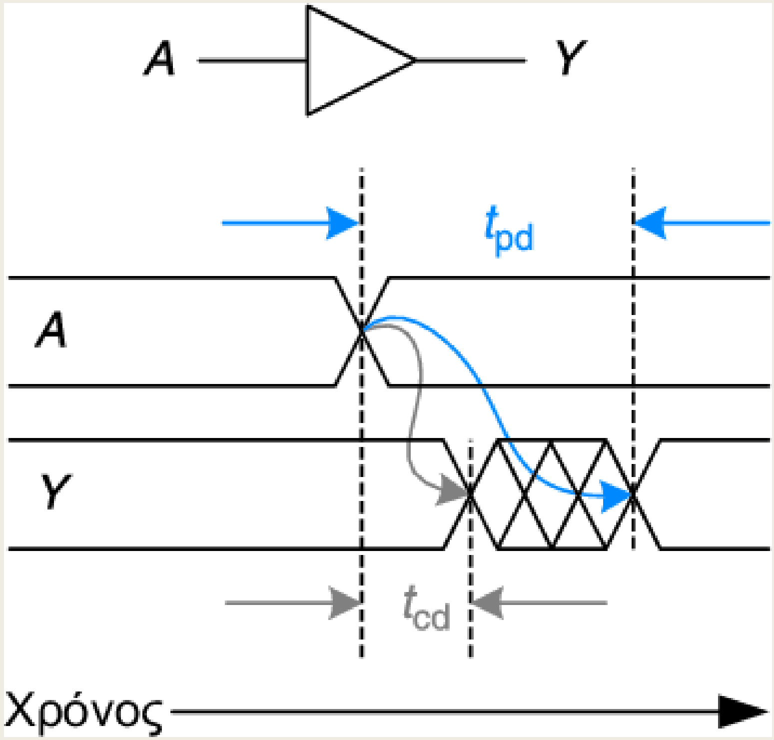
Καθυστέρηση διάδοσης / μόλυνσης

- Η συνδυαστική λογική χαρακτηρίζεται από
  - *την* ***καθυστέρηση διάδοσης*** t_{pd} *(propagation delay) και*
  - *την* ***καθυστέρηση μόλυνσης*** t_{cd} *(contamination delay)*

Η είσοδος *A* έχει αρχική τιμή είτε HIGH είτε LOW, και μεταβαίνει στην άλλη τιμή σε μια συγκεκριμένη στιγμή

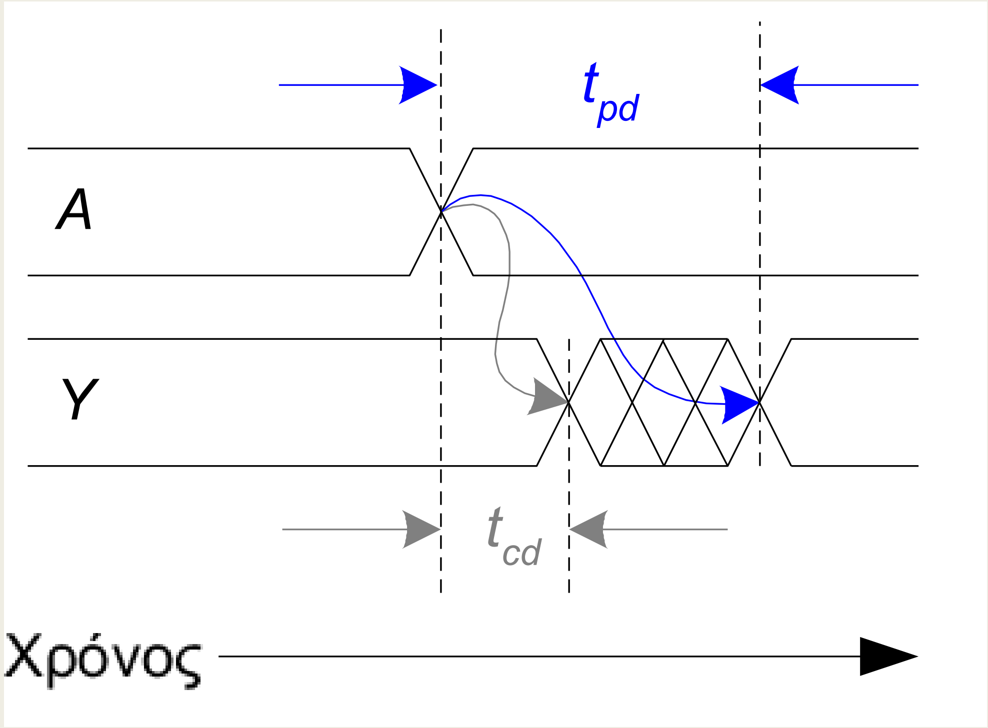
— δεν μας ενδιαφέρει η ακριβή τιμή

Αντιδρώντας στη μεταβολή, η έξοδος *Y* αλλάζει μετά από κάποιο χρονικό διάστημα.



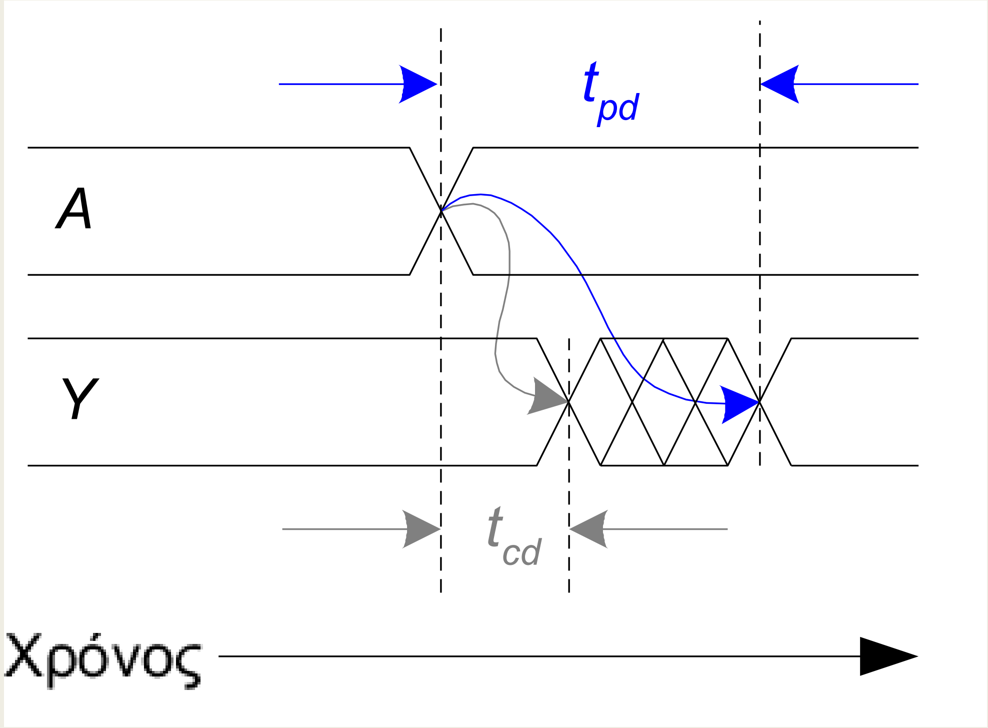
Καθυστέρηση διάδοσης

- Η **καθυστέρηση διάδοσης t_{pd}** είναι **ο μέγιστος χρόνος** που μεσολαβεί από τη στιγμή που μεταβάλλεται μια είσοδος έως τη στιγμή που η έξοδος ή οι έξοδοι αποκτούν την τελική τιμή τους
  – *είναι η **χειρότερη περίπτωση καθυστέρησης** (αυτό που ονομάζουμε απλά καθυστέρηση)*



Καθυστέρηση μόλυνσης

- Η **καθυστέρηση μόλυνσης** t_{cd} είναι ο **ελάχιστος χρόνος** που μεσολαβεί από τη στιγμή που μεταβάλλεται μια είσοδος έως τη στιγμή που οποιαδήποτε έξοδος αρχίζει να μεταβάλλει την τιμή της
  - *είναι η **καλλίτερη περίπτωση καθυστέρησης***
    (τη χρησιμοποιούμε σε συγκεκριμένες περιπτώσεις)

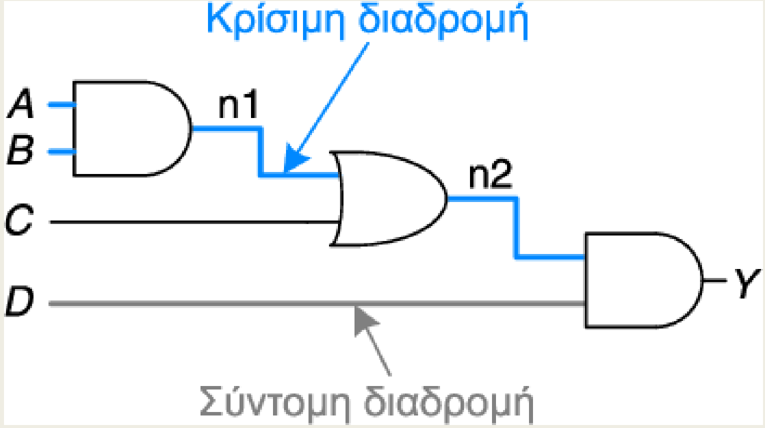


Καθυστέρηση διάδοσης / μόλυνσης

- **Αίτια καθυστέρησης σε επίπεδο πύλης**
  - *ο χρόνος που απαιτείται για τη φόρτιση της χωρητικότητας*
  - *περισσότερα τρανζίστορ στη σειρά μεγαλύτερη καθυστέρηση*
    - περισσότερες είσοδοι στην πύλη μεγαλύτερη καθυστέρηση
- **Οι τιμές των t_{pd} και t_{cd} ενδέχεται να είναι διαφορετικές**
  - *διαφορετικές καθυστερήσεις ανερχόμενης και κατερχόμενης ακμής*
    - Διαφορετική καθυστέρηση στα n-MOS από τα p-MOS τρανζίστορ
    - Διαφορετικό πλήθος τρανζίστορ στη σειρά στα n-MOS από τα p-MOS δίκτυα
  - *παρουσία περισσότερων από μίας εισόδων και εξόδων, κάποιες από τις οποίες εμφανίζουν πιο αργές ή πιο κρίσιμες διαδρομές*
  - *αύξηση και μείωση της καθυστέρησης όταν τα κυκλώματα θερμαίνονται και ψύχονται, αντίστοιχα*
- **Αίτια καθυστέρησης σε επίπεδο κυκλώματος**
  - *η **διαδρομή** που ακολουθεί το σήμα μέσα από τις πύλες, όπως διαδίδεται από τις εισόδους στις εξόδους του κυκλώματος*
  - *τα σήματα διαδίδονται κατά μήκος των συρμάτων με την **ταχύτητα του φωτός**, με τη μορφή ηλεκτρομαγνητικών κυμάτων (~15 cm/ns)*

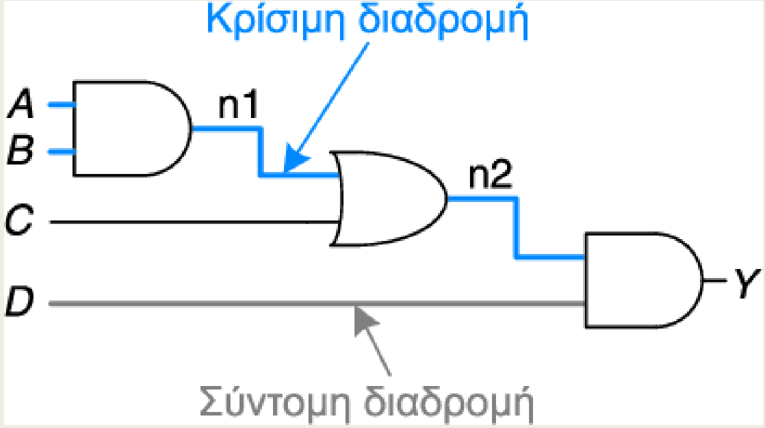
Κρίσιμη (αργή) διαδρομή

- Η **κρίσιμη (αργή) διαδρομή** είναι η μεγαλύτερη σε μήκος και συνεπώς η πιο αργή σε ταχύτητα
  - *αυτή που διέρχεται από τις περισσότερες πύλες, που έχουν και τις περισσότερες εισόδους*
  - *ονομάζεται κρίσιμη επειδή **περιορίζει τη συχνότητα λειτουργίας του κυκλώματος***
- Στο σχήμα ξεκινάει από την είσοδο A ή B και καταλήγει στην έξοδο Y περνώντας μέσα από τρεις πύλες



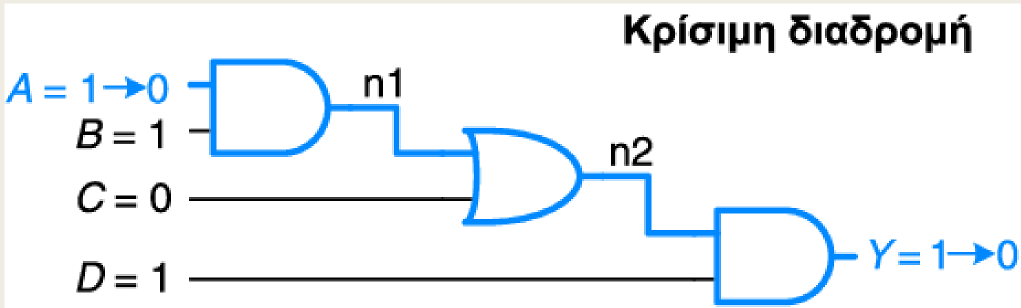
Σύντομη διαδρομή

- Η **σύντομη διαδρομή** είναι η μικρότερη σε μήκος και συνεπώς η πιο γρήγορη σε ταχύτητα
  - *αυτή που διέρχεται από τις λιγότερες πύλες, που έχουν και τις λιγότερες εισόδους*
- Στο σχήμα ξεκινάει από την είσοδο D και καταλήγει στην έξοδο Y περνώντας μέσα από μία πύλη

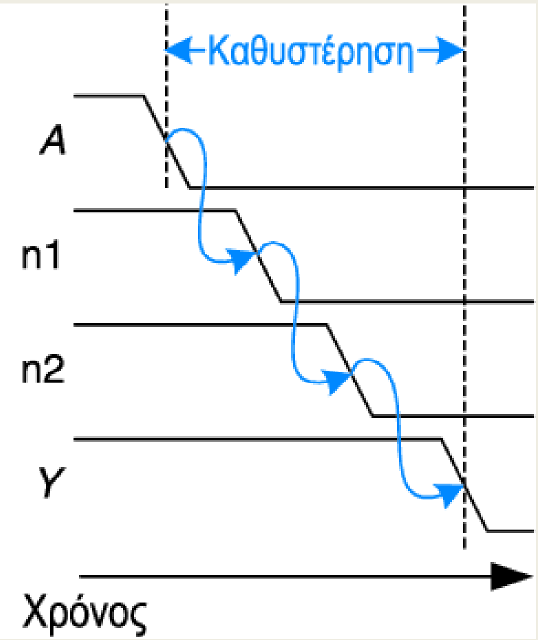


Υπολογισμός της καθυστέρησης t_{pd}

- Η **καθυστέρηση διάδοσης t_{pd}** ενός συνδυαστικού κυκλώματος είναι το άθροισμα των καθυστερήσεων διάδοσης για τη διέλευση από κάθε στοιχείο της **κρίσιμης (αργής) διαδρομής**
  - *Στο σχήμα:* $t_{pd} = 2t_{pd\_AND} + t_{pd\_OR}$
  - *Εμφανίζεται όταν οι είσοδοι του κυκλώματος μεταβάλλονται από ABCD = 1101 σε ABCD = 0101*

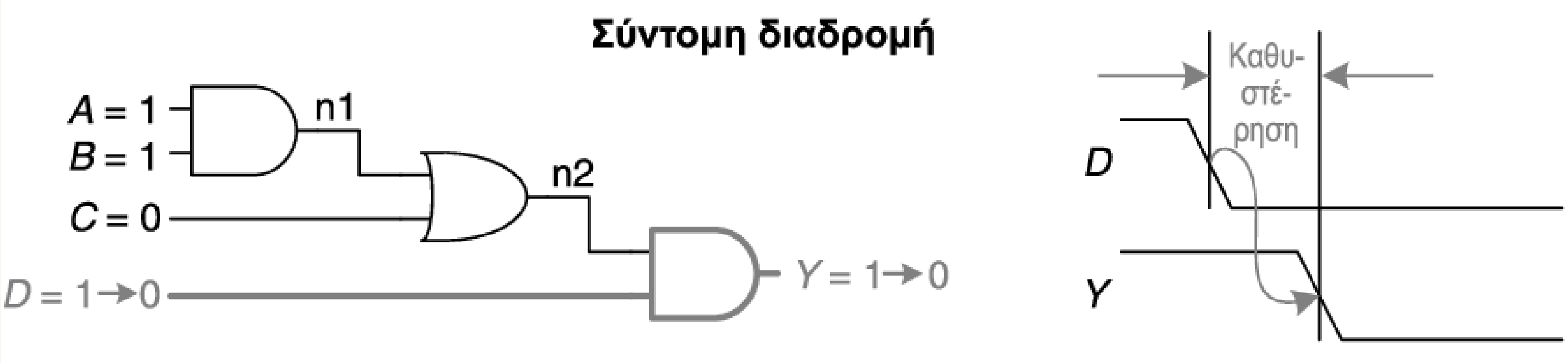


Προσοχή: Η κρίσιμη διαδρομή από την είσοδο A στην έξοδο Y ενεργοποιείται με συγκεκριμένες τιμές των υπολοίπων εισόδων που παραμένουν σταθερές, δηλαδή *B*=1, *C* = 0, *D* = 1



Υπολογισμός της καθυστέρησης t_{cd}

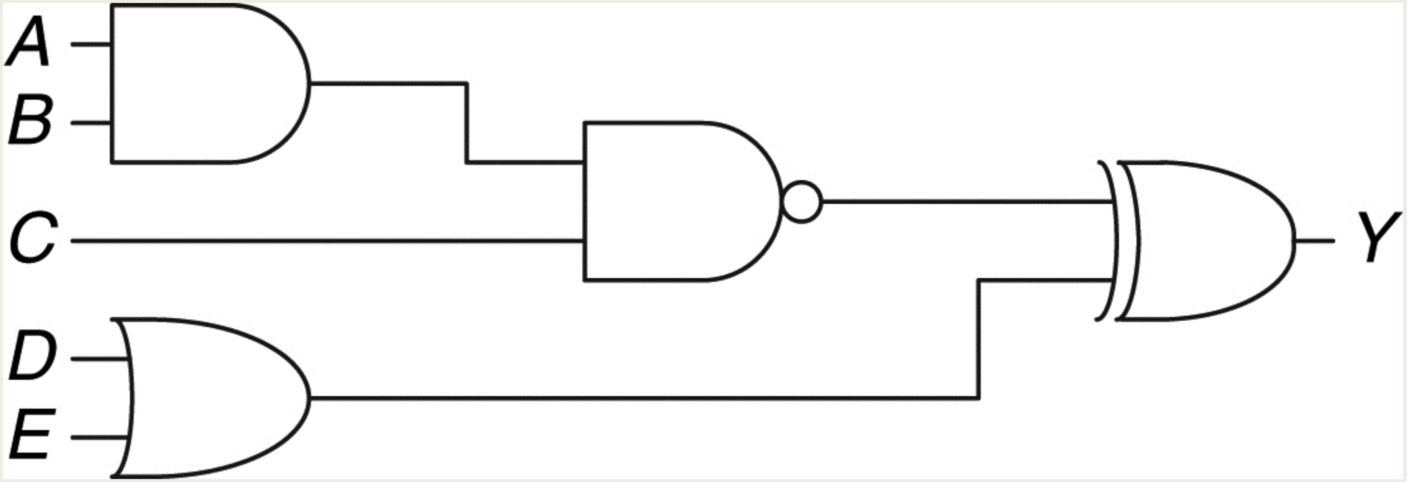
- Η **καθυστέρηση μόλυνσης t_{cd}** ενός συνδυαστικού κυκλώματος είναι το άθροισμα των καθυστερήσεων μόλυνσης για τη διέλευση από κάθε στοιχείο **της σύντομης διαδρομής**
  - *Στο σχήμα:* $t_{cd} = t_{cd\_AND}$
  - *Εμφανίζεται όταν οι είσοδοι του κυκλώματος μεταβάλλονται από ABCD = 1101 σε ABCD = 1100*



Προσοχή: Η σύντομη διαδρομή από την είσοδο D στην έξοδο Y ενεργοποιείται με συγκεκριμένες τιμές των υπολοίπων εισόδων που παραμένουν σταθερές, δηλαδή $A = 1, B = 1, C = 0 \Rightarrow n2 = 1$

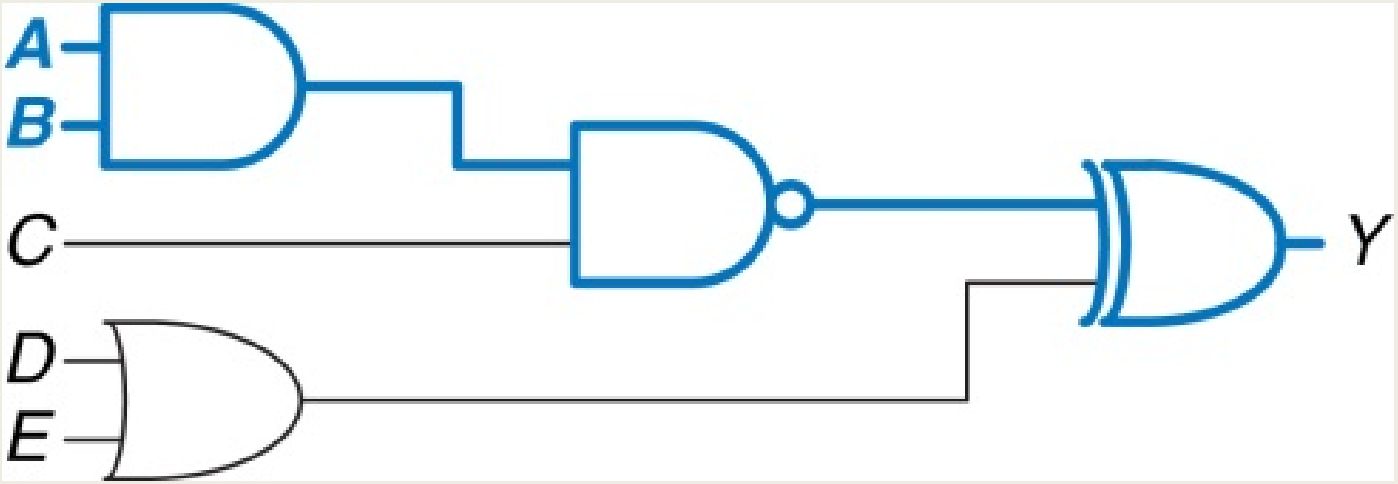
Παράδειγμα 2.15

- **Υπολογισμός καθυστερήσεων διάδοσης και μόλυνσης**
- Υπολογίστε την **καθυστέρηση διάδοσης** και την **καθυστέρηση μόλυνσης** του κυκλώματος
- Κάθε πύλη έχει καθυστέρηση διάδοσης ίση με **100 ps** και καθυστέρηση μόλυνσης ίση με **60 ps** (ps = picoseconds)



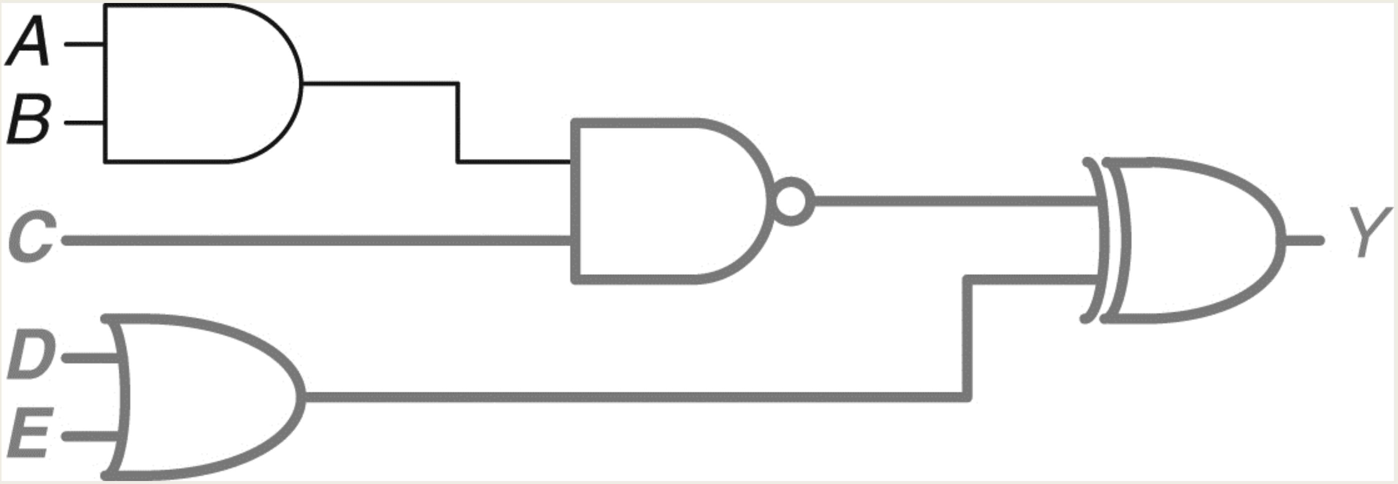
Παράδειγμα 2.15

- **Υπολογισμός καθυστερήσεων διάδοσης και μόλυνσης**
- Υπολογισμός της **καθυστέρησης διάδοσης** του κυκλώματος
  - *Κάθε πύλη έχει καθυστέρηση διάδοσης ίση με* ***100 ps***
  - *Η κρίσιμη διαδρομή αρχίζει από την είσοδο Α ή Β, διέρχεται από τρεις πύλες και καταλήγει στην έξοδο Υ*
  - *Άρα, η καθυστέρηση διάδοσης είναι:* $t_{pd} = 3 \times 100 = 300$ ***ps***



Παράδειγμα 2.15

- Υπολογισμός καθυστερήσεων διάδοσης και μόλυνσης
- Υπολογισμός της **καθυστέρησης μόλυνσης** του κυκλώματος
  - *Κάθε πύλη έχει καθυστέρηση μόλυνσης ίση με* ***60 ps***
  - *Η σύντομη διαδρομή αρχίζει από την είσοδο C ή D ή E, διέρχεται από δύο πύλες και καταλήγει στην έξοδο Y*
  - *Άρα, η καθυστέρηση μόλυνσης είναι:* $t_{cd} = 2 \times 60 = 120$ *ps*



Παράδειγμα 2.16

- **Χρονισμός πολυπλεκτών: κρισιμότητα από πλευράς ελέγχου και κρισιμότητα από πλευράς δεδομένων**

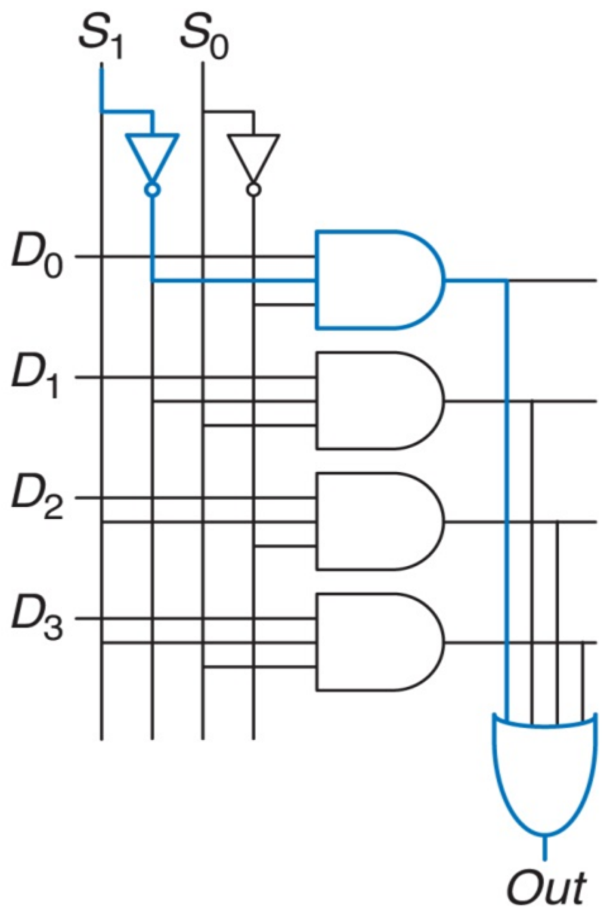
- Συγκρίνετε τον χρονισμό της χειρότερης περίπτωσης (εύρεση κρίσιμης διαδρομής και υπολογισμός καθυστέρησης διάδοσης) για τις τρεις διαφορετικές υλοποιήσεις των πολυπλεκτών 4 σε 1
  - *(α) λογική αθροίσματος γινομένων,*
  - *(β) απομονωτές τριών καταστάσεων,*
  - *(γ) δένδρο πολυπλεκτών 2 σε 1*

- Δίδονται οι τιμές της καθυστέρησης διάδοσης για τα επιμέρους στοιχεία

Πύλη	t_{pd} (ps)
NOT	30
AND με 2 εισόδους	60
AND με 3 εισόδους	80
OR με 4 εισόδους	90
απομονωτής τριών καταστάσεων (από *A* έως *Y*)	50
απομονωτής τριών καταστάσεων (από enable έως *Y*)	35

Παράδειγμα 2.16

- **Χρονισμός πολυπλεκτών: κρισιμότητα από πλευράς ελέγχου και κρισιμότητα από πλευράς δεδομένων**
- **Λογική αθροίσματος γινομένων**
- Η κρίσιμη διαδρομή αρχίζει από την είσοδο επιλογής S_1 ή S_0, διέρχεται από 3 πύλες (NOT, AND-3, OR-4) και καταλήγει στην έξοδο Y
- Η υλοποίηση είναι **κρίσιμη από πλευράς ελέγχου**
  - *Έλεγχος-έξοδος 200 ps*
  - *Δεδομένα-έξοδος 170 ps*

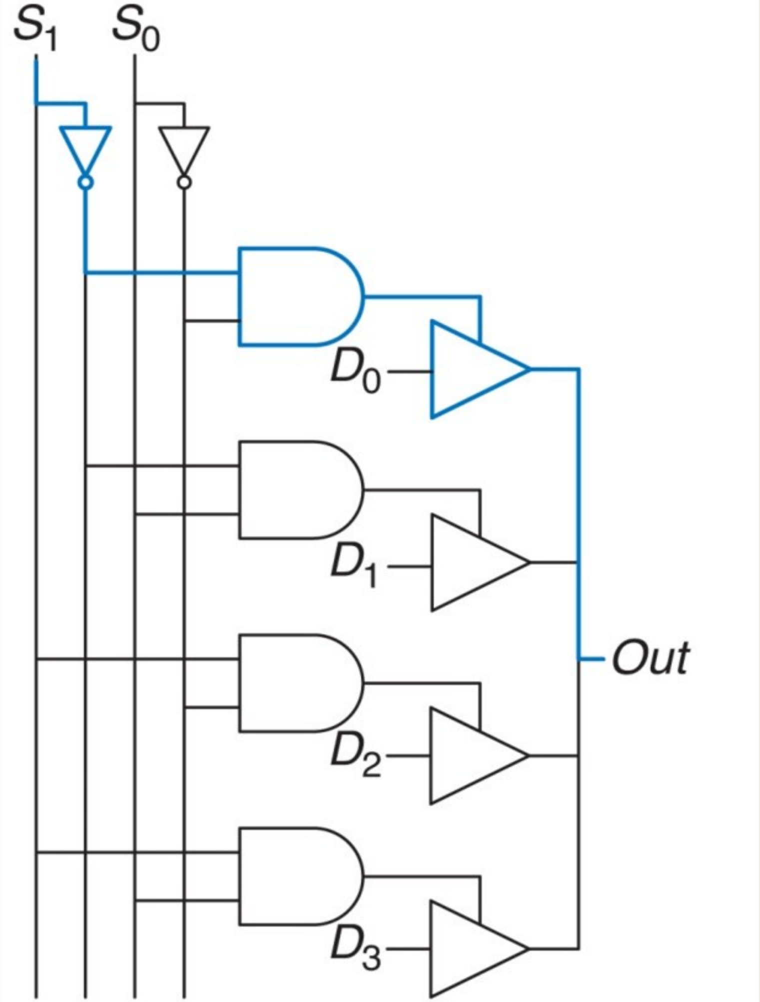


$$\begin{aligned} t_{pd\_sy} &= t_{pd\_\mathrm{INV}} + t_{pd\_\mathrm{AND3}} + t_{pd\_\mathrm{OR4}} \\ &= 30\ \mathrm{ps} + 80\ \mathrm{ps} + 90\ \mathrm{ps} \\ &= \mathbf{200\ ps} \end{aligned}$$

$$\begin{aligned} t_{pd\_dy} &= t_{pd\_\mathrm{AND3}} + t_{pd\_\mathrm{OR4}} \\ &= \mathbf{170\ ps} \end{aligned}$$

Παράδειγμα 2.16

- Χρονισμός πολυπλεκτών: κρισιμότητα από πλευράς ελέγχου και κρισιμότητα από πλευράς δεδομένων
- **Απομονωτές τριών καταστάσεων**
- Η κρίσιμη διαδρομή αρχίζει από την είσοδο επιλογής S_1 ή S_0, διέρχεται από 2 πύλες (NOT, AND-2) και τον απομονωτή τριών καταστάσεων (enable to Y), και καταλήγει στην έξοδο Y
- Η υλοποίηση είναι **κρίσιμη από πλευράς ελέγχου**
  - *Έλεγχος-έξοδος 125 ps*
  - *Δεδομένα-έξοδος 50 ps*

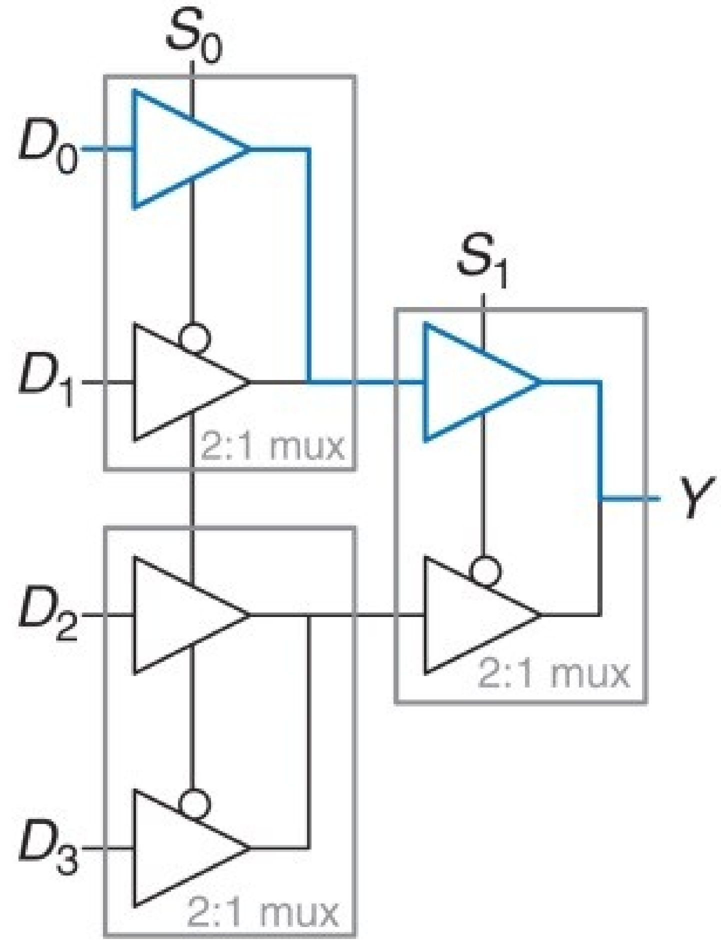


$$\begin{aligned} t_{pd\_sy} &= t_{pd\_INV} + t_{pd\_AND2} + t_{pd\_TRI\_sy} \\ &= 30\text{ ps} + 60\text{ ps} + 35\text{ ps} \\ &= \mathbf{125\ ps} \end{aligned}$$

$$\begin{aligned} t_{pd\_dy} &= t_{pd\_TRI\_ay} \\ &= \mathbf{50\ ps} \end{aligned}$$

Παράδειγμα 2.16

- Χρονισμός πολυπλεκτών: κρισιμότητα από πλευράς ελέγχου και κρισιμότητα από πλευράς δεδομένων
- **Δένδρο πολυπλεκτών 2 σε 1**
- Οι πολυπλέκτες 2 σε 1 υλοποιούνται με απομονωτές τριών καταστάσεων
- Η κρίσιμη διαδρομή αρχίζει από οποιαδήποτε είσοδο D, διέρχεται από 2 απομονωτές τριών καταστάσεων (A to Y), και καταλήγει στην έξοδο Y
- Η υλοποίηση είναι **κρίσιμη από πλευράς δεδομένων**
  - *Έλεγχος-έξοδος 85 ps*
  - *Δεδομένα-έξοδος 100 ps*



$$t_{pd\_s0y} = t_{pd\_TRI\_sy} + t_{pd\_TRI\_ay} = \mathbf{85\ ps}$$

$$t_{pd\_dy} = 2\ t_{pd\_TRI\_ay} = \mathbf{100\ ps}$$

Παράδειγμα 2.16

- Χρονισμός πολυπλεκτών: κρισιμότητα από πλευράς ελέγχου και κρισιμότητα από πλευράς δεδομένων
- Συμπεράσματα:
  - *Αν τα σήματα από τις εισόδους δεδομένων φτάνουν αρκετά πριν από εκείνα των εισόδων ελέγχου, επιλέγουμε την υλοποίηση με τη μικρότερη καθυστέρηση διάδοσης ελέγχου-προς-έξοδο (**85 ps**)*
    - Την υλοποίηση του **δένδρου πολυπλεκτών 2 σε 1**
  - *Αν τα σήματα από τις εισόδους ελέγχου φτάνουν αρκετά πριν από εκείνα των εισόδων δεδομένων, επιλέγουμε την υλοποίηση με τη μικρότερη καθυστέρηση δεδομένων-προς-έξοδο (**50 ps**)*
    - Την υλοποίηση των **απομονωτών τριών καταστάσεων**
- Η βέλτιστη επιλογή δεν εξαρτάται μόνο από την κρίσιμη διαδρομή μέσω του κυκλώματος και τους χρόνους άφιξης των σημάτων εισόδου, αλλά και από την κατανάλωση ισχύος, το κόστος και τη διαθεσιμότητα των στοιχείων

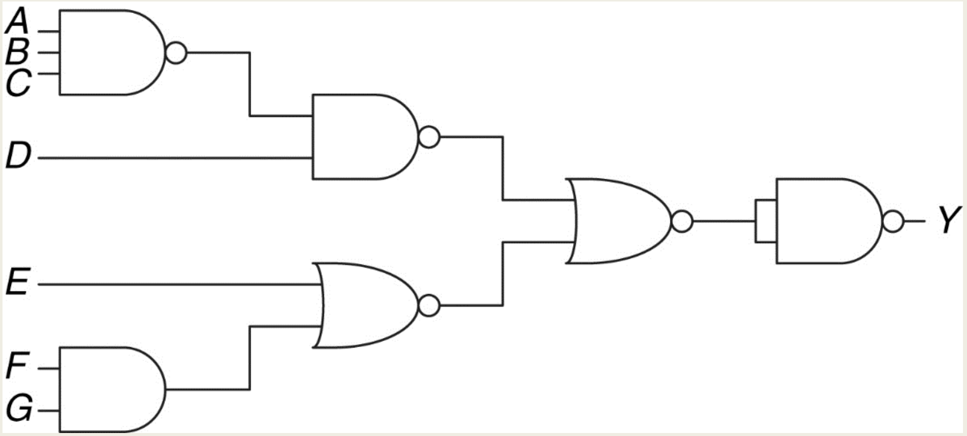
Επιλεγμένες ασκήσεις

- **Άσκηση 2.44**
  Υπολογίστε την καθυστέρηση διάδοσης και την καθυστέρηση μόλυνσης του κυκλώματος της εικόνας. Χρησιμοποιήστε τις καθυστερήσεις πυλών που δίνονται στον Πίνακα
  - *Κρίσιμη διαδρομή: F-Y*

$$t_{pd} = t_{pd\_AND2} + 2t_{pd\_NOR2} + t_{pd\_NAND2} = [30 + 2\,(30) + 20]\ \text{ps} = \mathbf{110\ ps}$$

  - *Σύντομη διαδρομή: D-Y*

$$t_{cd} = 2t_{cd\_NAND2} + t_{cd\_NOR2} = [2\,(15) + 25]\ \text{ps} = \mathbf{55\ ps}$$



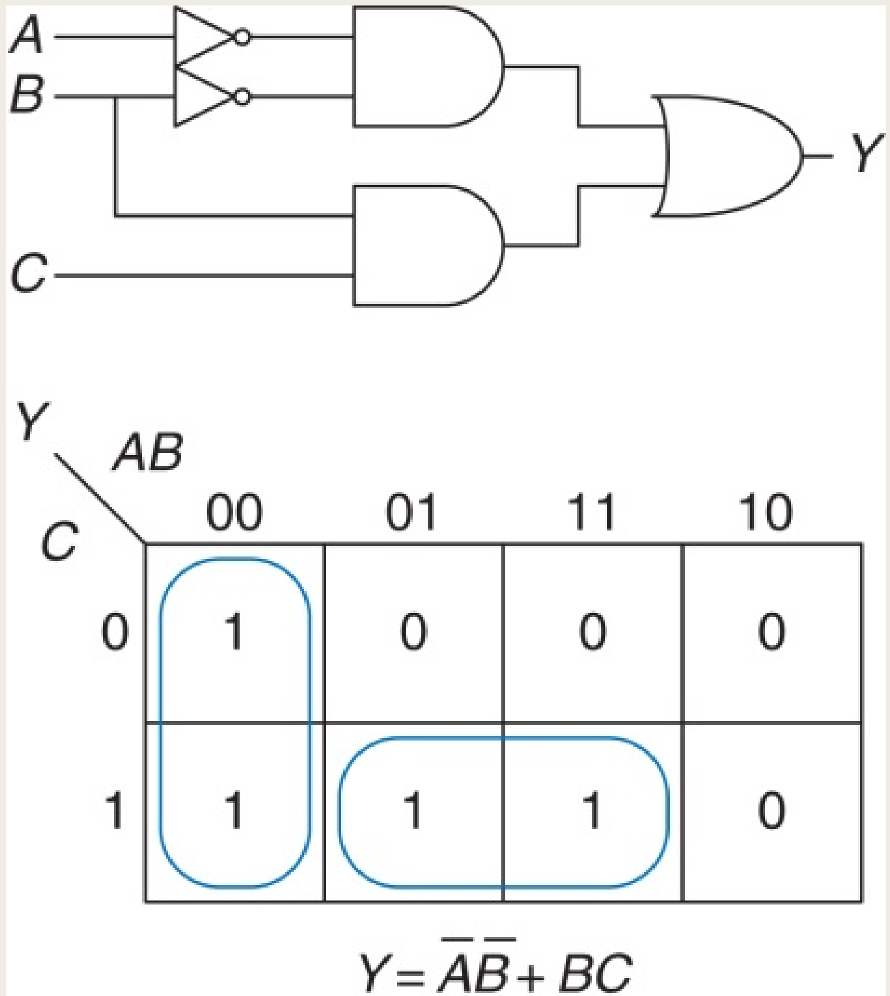
Πύλη	t_{pd} (ps)	t_{cd} (ps)
NOT	15	10
NAND με 2 εισόδους	20	15
NAND με 3 εισόδους	30	25
NOR με 2 εισόδους	30	25
NOR με 3 εισόδους	45	35
AND με 2 εισόδους	30	25
AND με 3 εισόδους	40	30
OR με 2 εισόδους	40	30
OR με 3 εισόδους	55	45
XOR με 2 εισόδους	60	40

Μεταβατικοί παλμοί (Glitches)

- Μέχρι τώρα έχουμε μελετήσει την περίπτωση μία μεταβολή εισόδου να προκαλεί μονάχα μία μεταβολή στην έξοδο
- Είναι όμως πιθανό μία μεταβολή εισόδου να προκαλέσει **πολλαπλές μεταβολές στην έξοδο**
  - *Αυτές ονομάζονται* ***μεταβατικοί παλμοί*** *(glitches) ή* ***κίνδυνοι*** *(hazards)*
- Με την προϋπόθεση ότι περιμένουμε να παρέλθει η **κρίσιμη καθυστέρηση διάδοσης** προτού βρεθούμε εξαρτημένοι από την έξοδο, οι μεταβατικοί παλμοί δεν αποτελούν πρόβλημα επειδή η έξοδος θα έχει προλάβει να σταθεροποιηθεί στη σωστή λογική τιμή
- Μας ενδιαφέρει να γνωρίζουμε ότι υπάρχουν στα περισσότερα ψηφιακά κυκλώματα και να τα διακρίνουμε όταν εξετάζουμε διαγράμματα χρονισμού σε προσομοιωτές ή παλμογράφους

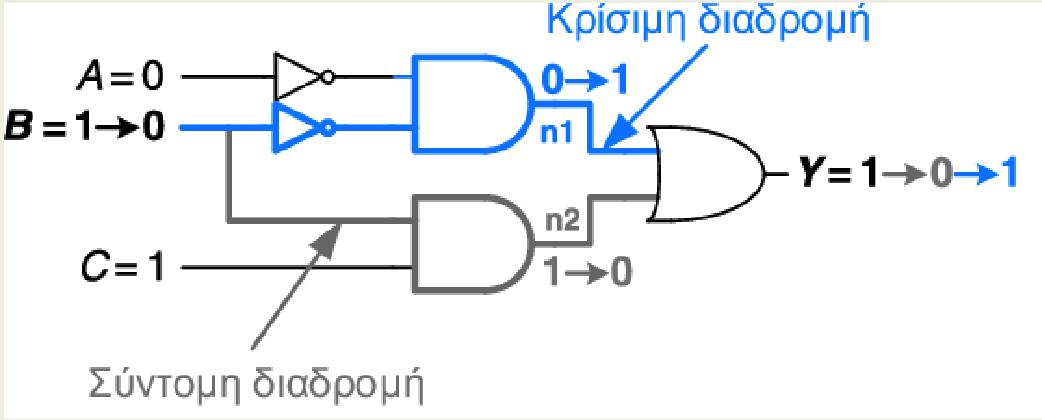
Μεταβατικοί παλμοί (Glitches)

- Τι θα συμβεί όταν A = 0, C = 1 και B μεταβαίνει από το 1 στο 0



Μεταβατικοί παλμοί (Glitches)

- Η είσοδος B συνδέεται με την έξοδο Y μέσω δύο διαδρομών:
  - *Την* ***κρίσιμη διαδρομή*** *που διέρχεται από τρείς πύλες (NOT, AND-2 και OR-2)*
  - *Τη* ***σύντομη διαδρομή*** *που διέρχεται από δύο πύλες (AND-2 και OR-2)*
  - *Η κρίσιμη διαδρομή διαφέρει από τη σύντομη διαδρομή στην καθυστέρηση διάδοσης* ***κατά μία πύλη NOT***
- Λόγω αυτής της διαφοράς εμφανίζεται ένας **μεταβατικός παλμός** στην έξοδο Y, όταν **A = 0, C = 1 και B μεταβαίνει από το 1 στο 0**



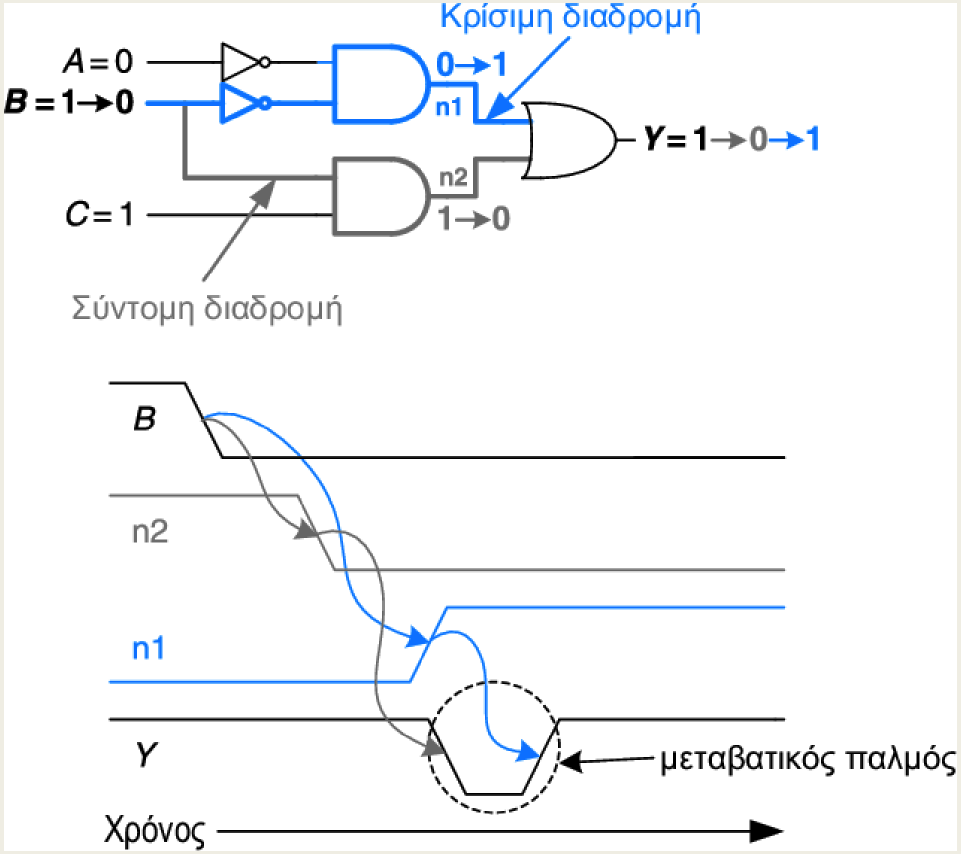
Μεταβατικοί παλμοί (Glitches)

Καθώς το *B* μεταβαίνει από το 1 στο 0, ο κόμβος n2 (στη σύντομη διαδρομή) υφίσταται κάθοδο προτού προλάβει να ολοκληρωθεί η άνοδος του κόμβου n1 (στην κρίσιμη διαδρομή)

Μέχρι να ολοκληρωθεί η άνοδος του n1, οι δύο είσοδοι της πύλης OR έχουν τιμή 0, και η έξοδος *Y* «πέφτει» στο 0

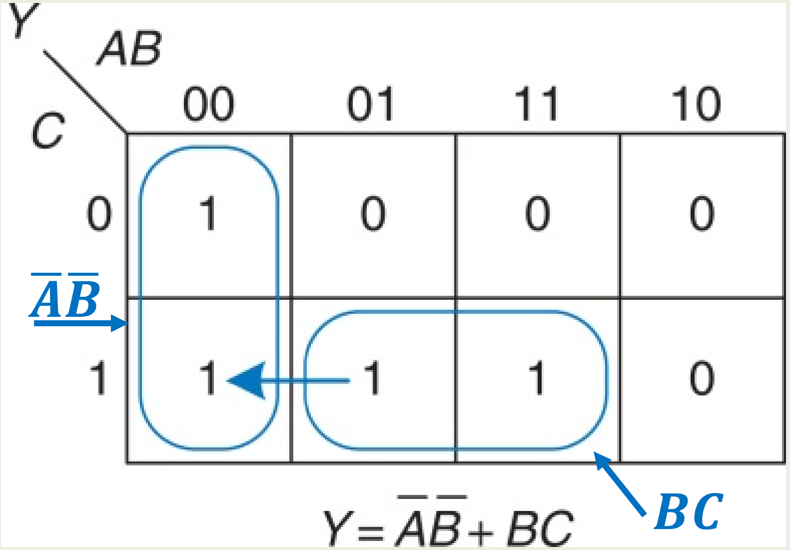
Μόλις ολοκληρωθεί η άνοδος του n1, το *Y* επιστρέφει στο 1

Η διάρκεια του μεταβατικού παλμού είναι η καθυστέρηση διάδοσης μίας πύλης NOT



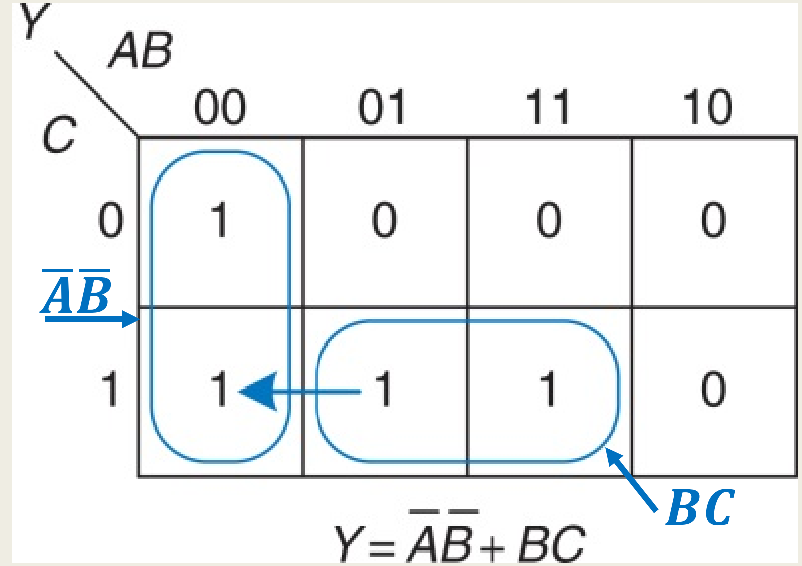
Αποφυγή μεταβατικού παλμού

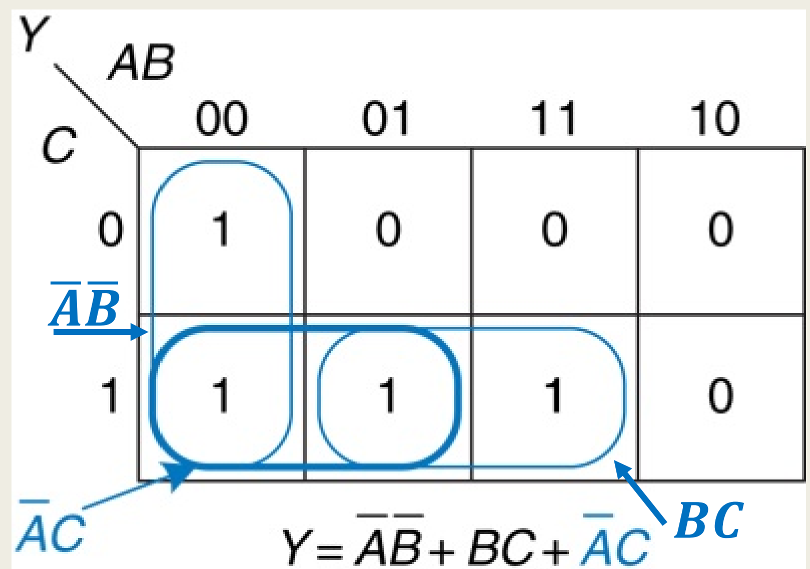
- Κατά τη μετάβαση των εισόδων *ABC* από το 011 στο 001, αν το κύκλωμα του πρώτου όρου ($\boldsymbol{BC}$) *απενεργοποιηθεί* προτού το κύκλωμα του άλλου πρώτου όρου ($\boldsymbol{\overline{A}\overline{B}}$) *ενεργοποιηθεί*, τότε παρατηρείται ένας μεταβατικός παλμός
- **Η μετάβαση που διασχίζει τα όρια δύο πρώτων όρων στον χάρτη Karnaugh αποτελεί ένδειξη πιθανού μεταβατικού παλμού**



Αποφυγή μεταβατικού παλμού

- Ο μεταβατικός παλμός αποφεύγεται εάν προσθέσουμε έναν επιπλέον κύκλο του οποίου ο όρος $\bar{A}C$ είναι **ανάμεσα** στους πρώτους όρους BC και $\bar{A}\bar{B}$
- Πρόκειται για το **θεώρημα της ομοφωνίας**, όπου ο επιπλέον προστιθέμενος όρος, το $\bar{A}C$, είναι ο **πλεονάζων όρος**





Αποφυγή μεταβατικού παλμού

- Ο μεταβατικός παλμός αποφεύγεται γιατί η πύλη AND που υλοποιεί τον **πλεονάζοντα όρο $\bar{A}C$** παράγει στην έξοδό της την τιμή 1 καθ' όλη τη διάρκεια της μεταβολής του B