ΤΕΧΝΟΛΟΓΙΚΟ ΕΚΠΑΙΔΕΥΤΙΚΟ ΙΔΡΥΜΑ
(Τ.Ε.Ι.) ΚΡΗΤΗΣ

Τμήμα Εφαρμοσμένης Πληροφορικής & Πολυμέσων

Ψηφιακή Σχεδίαση

Κεφάλαιο 2: Συνδυαστικά Λογικά Κυκλώματα

Γ. Κορνάρος

Περίγραμμα

- Μέρος 1 – Κυκλώματα Πυλών και Boolean Εξισώσεις
  - Δυαδική Λογική και Πύλες
  - Boolean Άλγεβρα
  - Standard Μορφές
- Μέρος 2 – Βελτιστοποίηση Κυκλωμάτων
  - Δι-επίπεδη Βελτιστοποίηση
  - Χειρισμός Χαρτών
  - Practical Optimization (Espresso)
  - Βελτιστοποίηση Κυκλωμάτων πολλών επιπέδων
- Μέρος 3 – Επιπλέον Πύλες και κυκλώματα
  - Άλλοι τύποι πυλών
  - Τελεστής Αποκλειστικού-Ή και πύλες
  - Έξοδοι υψηλής-εμπέδησης

Δυαδική Λογική και Πύλες

- Οι Δυαδικές μεταβλητές παίρνουν μία από τις δύο δυνατές τιμές.
- Οι λογικοί τελεστές λειτουργούν σε δυαδικές τιμές και δυαδικές μεταβλητές.
- Οι βασικοί λογικοί τελεστές είναι οι λογικές συναρτήσεις AND, OR και NOT.
- Οι Λογικές Πύλες υλοποιούν λογικές συναρτήσεις.
- Boolean Algebra: ένα χρήσιμο μαθηματικό σύστημα για περιγραφή και μετασχηματισμούς λογικών συναρτήσεων.
- Μελετούμε την Boolean algebra διότι είναι η βάση για σχεδιασμό και ανάλυση Ψηφιακών Συστημάτων!

Λογικές Λειτουργίες

- Οι τρεις βασικές λογικές πράξεις είναι:
  - AND
  - OR
  - NOT
- AND συμβολίζεται με (·)
- OR συμβολίζεται με πρόσθεση (+)
- NOT συμβολίζεται με μία παύλα (¯), μία απόστροφο (') μετά , ή ένα (~) πριν την μεταβλητή.

Πίνακας Αλήθειας

- *Truth table* :

  μία αναπαράσταση των τιμών μιας συνάρτησης σε πίνακα για όλους του δυνατούς συνδυασμούς των μεταβλητών (ορισμάτων) της συνάρτησης.

Boolean Άλγεβρα

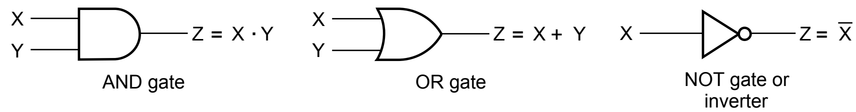
- Τρεις βασικές λειτουργίες μπορούν να γίνουν σε μεταβλητές Boole
  - και - and (juxtaposition) [•]: 0•0 = 0, 0•1 = 0, 1•0 = 0, 1•1 = 1
  - ή - or [+]: 0+0 = 0, 0+1 = 1, 1+0 = 1, 1+1 = 1
  - συμπλήρωμα - complement (not or negation) x' or x: 0' = 1, 1' = 0

- Οι πράξεις αυτές μπορούν να αναπαρασταθούν σε πίνακα αλήθειας (truth table):

A	B	A+B	A•B	A’	B’
0	0				
0	1				
1	0				
1	1				

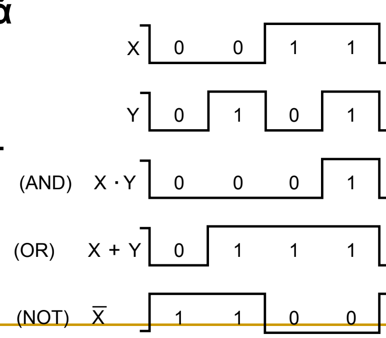
Σύμβολα Λογικών Πυλών - Συμπεριφορά

- Ειδικά σύμβολα για αναπαράσταση λογικών πυλών:



(a) Graphic symbols

- **Συμπεριφορά** αναπαράσταση στο χρόνο, κυματομορφές - waveform :



(b) Timing diagram

Λογικά Διαγράμματα και Εκφράσεις

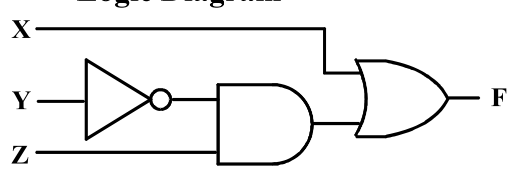
Πίνακας Αλήθειας

X Y Z	$F = X + \overline{Y} \cdot Z$
0 0 0	0
0 0 1	1
0 1 0	0
0 1 1	0
1 0 0	1
1 0 1	1
1 1 0	1
1 1 1	1

Εξίσωση

$$F = X + \overline{Y}\, Z$$

Logic Diagram



- Η ίδια συνάρτηση μπορεί να περιγράφεται με εξισώσεις Boolean, πίνακες αλήθειας και λογικά διαγράμματα!
  - Οι Πίνακες αλήθειας είναι μοναδικοί.
  - Οι εξισώσεις και τα λογικά διαγράμματα δεν είναι.

Άλγεβρα Boole : αξιωματικός ορισμός

- Η άλγεβρα (switching) Boole είναι ένα κλειστό αλγεβρικό σύστημα S,
  - περιέχει δύο στοιχεία {0,1},
  - δύο τελεστές + (OR) **.** (AND)
- Κλειστότητα
- Προσεταιριστικός νόμος
- Αντιμεταθετικός νόμος
- Ουδέτερο στοιχείο
- Αντίστροφο
- Επιμεριστικός νόμος

Αξιώματα και Θεωρήματα

• *Ταυτότητα*
– 1a: a+a = a
– 1b: a•a = a

• *Ουδέτερο στοιχείο για τους τελεστές + and •*
– 2a: a+1 = 1
– 2b: a•0 = 0

• *Διπλή άρνηση - Involution*
– 3: (a')' = a

• *Απορρόφηση*
– 4a: a+ab = a
– 4b: a(a+b) = a

– 5a: a+a'b = a+b
– 5b: a(a'+b) = ab
– 6a: ab+ab' = a
– 6b: (a+b)(a+b') = a
– 7a: ab+ab'c = ab+ac
– 7b: (a+b)(a+b'+c) = (a+b)(a+c)

Βασικά Θεωρήματα

- **DeMorgan's (Theorem 5)**
  - 8a: (a+b)' = a'•b'
  - 8b: (a•b)' = a'+b'

- Consensus (Theorem 9)
  - 9a: ab+a'c+bc = ab+a'c
  - 9b: (a+b)(a'+c)(b+c) = (a+b)(a'+c)
- Shannon's Expansion (Theorem 10)
  - 10a: $F(x_1, x_2, \ldots, x_n) =$
    $[x_1 \cdot F(1, x_2, \ldots, x_n)] + [x_1' \cdot F(0, x_2, \ldots, x_n)]$
  - 10b: $F(x_1, x_2, \ldots, x_n) =$
    $[x_1 + F(1, x_2, \ldots, x_n)] \cdot [x_1' + F(0, x_2, \ldots, x_n)]$

Θεώρημα : x + xy = x

Απόδειξη:

$$
\begin{aligned}
x + xy &= x \,.\, 1 + x\,y \\
&= x\,(1 + y) \\
&= x \,.\, 1 \\
&= x
\end{aligned}
$$

Λόγω δυϊσμού : $x \,.\, (x + y) = x$

Βασικά Θεωρήματα

- *n*-Variable Theorems (Generalized)
- Idempotency
  – T11a: x+x+…+x = x
  – T11b: x•x•…•x = x
- DeMorgan's
  –T12a: (x1 • x2 •…• xn)' = x1'+x2'+…+xn'
  – T12b: (x1+x2+…+xn)' = x1'• x2'•…• xn'

(A+B+C)' = (A+x)' ,όπου x= B+C
= A'x'
= A'(B+C)'
= A'(B'C')
= A'B'C'

Επανάληψη

1.	$X + 0 = X$	2.	$X \cdot 1 = X$	
3.	$X + 1 = 1$	4.	$X \cdot 0 = 0$	
5.	$X + X = X$	6.	$X \cdot X = X$	
7.	$X + \overline{X} = 1$	8.	$X \cdot \overline{X} = 0$	
9.	$\overline{\overline{X}} = X$			
10.	$X + Y = Y + X$	11.	$XY = YX$	Αντιμεταθετική
12.	$(X + Y) + Z = X + (Y + Z)$	13.	$(XY)Z = X(YZ)$	Προσεταιριστική
14.	$X(Y + Z) = XY + XZ$	15.	$X + YZ = (X + Y)(X + Z)$	Επιμεριστική
16.	$\overline{X + Y} = \overline{X} \cdot \overline{Y}$	17.	$\overline{X \cdot Y} = \overline{X} + \overline{Y}$	DeMorgan's

Υπολογισμός συνάρτησης Boolean

$F1 = xy\bar{z}$
$F2 = x + \bar{y}z$
$F3 = \bar{x}\bar{y}\bar{z} + \bar{x}\,y\,z + x\bar{y}$
$F4 = x\bar{y} + \bar{x}\,z$

x	y	z	F1	F2	F3	F4
0	0	0	0	0		
0	0	1	0	1		
0	1	0	0	0		
0	1	1	0	0		
1	0	0	0	1		
1	0	1	0	1		
1	1	0	1	1		
1	1	1	0	1		

Συμπλήρωμα συνάρτησης ?

F1' =

Απλοποίηση συναρτήσεων

- x(x´+y) = x´x + xy = 0 +xy = xy
- x+x´y = (x + x´)(x + y) = 1 (x + y) = x + y
- (x + y)(x + y´) = xx + xy + xy´ + yy´ =
  = x (1 + y + y´) = x
- xy + x´z + yz = xy + x´z + yz(x+x´)
  = xy + x´z + xyz + x´yz
  =xy(1 + z) + x´z (1 + y)
  = xy + x´z

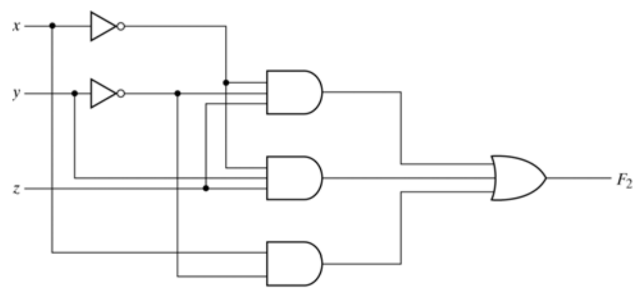
Απλοποίηση Έκφρασης

- Εφαρμογή της άλγεβρας Boole.
- Απλοποίηση ώστε να περιέχεται ο μικρότερος αριθμός <u>μεταβλητών</u> (είτε στην κανονική είτε ως συμπλήρωμα):

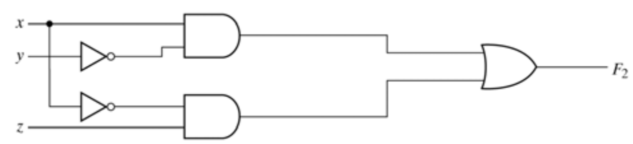
$$\mathbf{AB+\overline{A}CD+\overline{A}BD+\overline{A}C\overline{D}+ABCD}$$

- = AB + ABCD + A΄C D + A΄C D΄ + A΄ B D
- = AB + AB(CD) + A΄C (D + D΄) + A΄B D
- = AB + A΄C + A΄B D
- = B(A + A΄D) +A΄C
- = B(A + D) + A΄C

Έκφραση συνάρτησης Boole με πύλες



(a) $F_2 = x'y'z + x'yz + xy'$



(b) $F_2 = xy' + x'z$

Fig. 2-2 Implementation of Boolean function F_2 with gates

Κανονικές και Πρότυπες Μορφές

	Ελαχιστόροι		Μεγιστόροι	
x y z	Terms	Name	Terms	Name
0 0 0	x' y' z'	m_0	x+y+z	M_0
0 0 1	x' y' z	m_1	x+y+z'	M_1
0 1 0	x' y z'	m_2	x+y'+z	M_2
0 1 1	x' y z	m_3	x+y'+z'	M_3
1 0 0	x y' z'	m_4	x'+y+z	M_4
1 0 1	x y' z	m_5	x'+y+z'	M_5
1 1 0	x y z'	m_6	x'+y'+z	M_6
1 1 1	x y z	m_7	x'+y'+z'	M_7

Ελαχιστόροι

x	y	z	Corresponding minterm	Designation
0	0	0	x'y'z'	m_0
0	0	1	x'y'z	m_1
0	1	0	x'yz'	m_2
0	1	1	x'yz	m_3
1	0	0	xy'z'	m_4
1	0	1	xy'z	m_5
1	1	0	xyz'	m_6
1	1	1	xyz	m_7

- Μία μεθοδος αναπαράστασης συναρτήσεων Boole είναι η κανονική μορφή ελαχιστόρων (άθροισμα γινομένων ή SOP),
- F = x'y'z +xy'z + xyz' = m1 + m5 + m6 = Σ(1,5,6)

Ελαχιστόροι - παραδείγματα

x	y	z	F2 (Given)	Designation
0	0	0	1	m_0
0	0	1	1	m_1
0	1	0	1	m_2
0	1	1	1	m_3
1	0	0	0	
1	0	1	1	m_5
1	1	0	0	
1	1	1	0	

- F2 = Σ(0,1,2,3,5)
- = x'y'z' + x'y'z + x'yz' + x'yz + xy'z

Ελαχιστόροι – παραδείγματα – F'

x	y	z	F2 (Given)	Designation
0	0	0	1	m_0
0	0	1	1	m_1
0	1	0	1	m_2
0	1	1	1	m_3
1	0	0	0	
1	0	1	1	m_5
1	1	0	0	
1	1	1	0	

- (F2)' = Σ(all minterms not in F2) = Σ(4,6,7)
- = xy'z' + xyz' + xyz

Μεγιστόροι

x	y	z	Corresponding maxterm	Designation
0	0	0	x +y +z	M_0
0	0	1	x +y +z’	M_1
0	1	0	x +y’ +z	M_2
0	1	1	x +y’ +z’	M_3
1	0	0	x’ +y +z	M_4
1	0	1	x’ +y +z’	M_5
1	1	0	x’ +y’ +z	M_6
1	1	1	x’ +y’ +z’	M_7

- Ένας Μεγιστόρος είναι ένας όρος από OR
- Κάθε Μεγιστόρος έχει τιμή 0 για ένα συνδυασμό τιμών των N μεταβλητών

Μετατροπή μεταξύ Ελαχιστόρων - Μεγιστόρων

- m0 = x’y’z’ = (x+y+z)’ = (M0)’
- Γενικά, mi = (Mi)’

Παράδειγμα :
Μετατροπή σε γινόμενο αθροισμάτων

- F=xy+x'z
  - = (xy + x')(xy + z)
  - = (x + x')(y + x')(x + z)(y + z)
  - = (x' + y)(x + z)(y + z)

x'+y = x' + y + zz' = (x'+y+z)(x'+y+z')

.

.

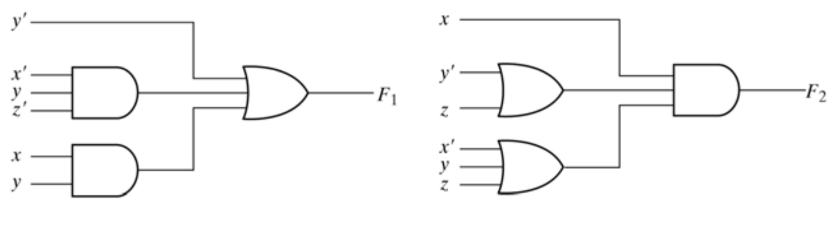
$F = M_0M_2M_4M_5$

Μεγιστόροι

x	y	z	F3 (Given)	Designation
0	0	0	0	M_0
0	0	1	1	
0	1	0	0	M_2
0	1	1	0	M_3
1	0	0	0	M_4
1	0	1	1	
1	1	0	1	
1	1	1	0	M_7

- F3 = (x+y+z)(x+y'+z)(x+y'+z')(x'+y+z)(x'+y'+z')
- = π(0,2,3,4,7)
- (F3)' = π(all maxterm not in F3)

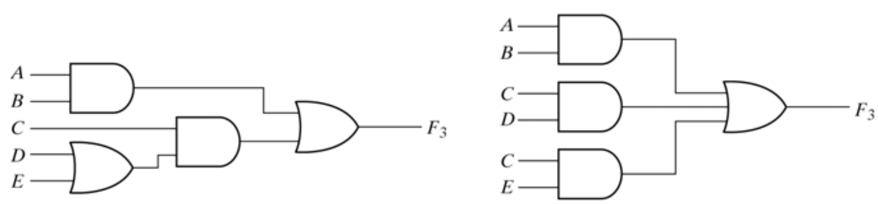
Υλοποίηση δύο επιπέδων



(a) Sum of Products (b) Product of Sums

Fig. 2-3 Two-level implementation

Υλοποίηση τριών επιπέδων και δύο επιπέδων



(a) $AB + C(D + E)$ (b) $AB + CD + CE$

Fig. 2-4 Three- and Two-Level implementation

Άλλες λογικές πράξεις

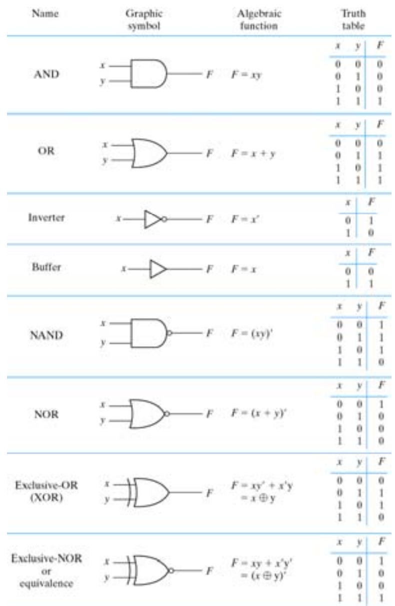


Fig. 2-5 Digital logic gates

Προσοχή! Προσεταιριστική ιδιότητα με NOR

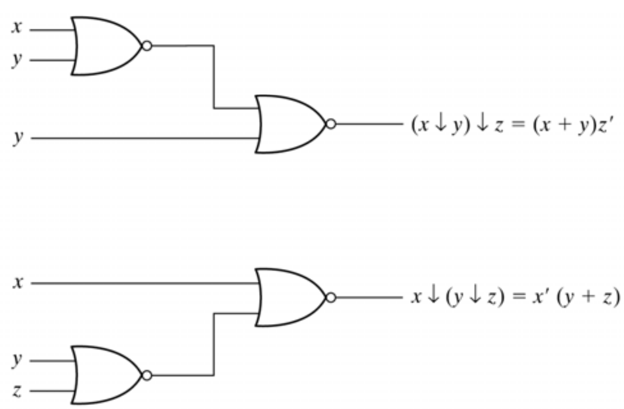
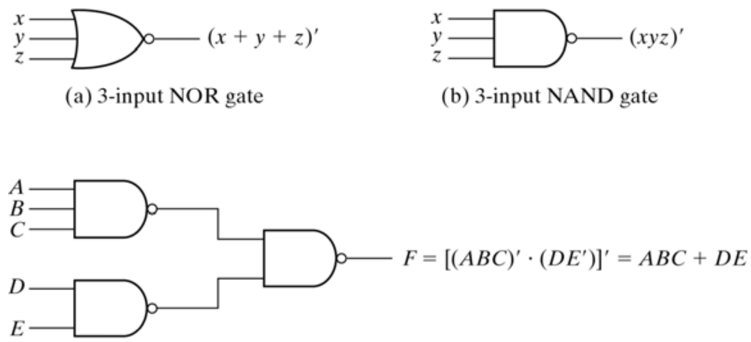


Fig. 2-6 Demonstrating the nonassociativity of the NOR operator; $(x \downarrow y) \downarrow z \neq x \downarrow (y \downarrow z)$

Πύλες πολλαπλών εισόδων



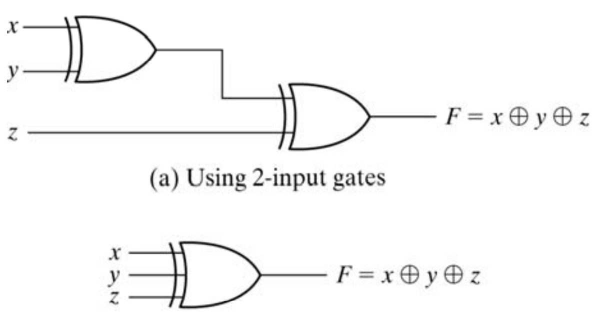
(a) 3-input NOR gate

(b) 3-input NAND gate

(c) Cascaded NAND gates

Fig. 2-7 Multiple-input and cascated NOR and NAND gates

Πύλη αποκλειστικού-Ή



(a) Using 2-input gates

(b) 3-input gate

x	y	z	F
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	1

(c) Truth table

Fig. 2-8 3-input exclusive-OR gate

Θετική και Αρνητική Λογική

- Θετική λογική: αντιστοίχιση του υψηλού δυναμικού στο λογικό “1”

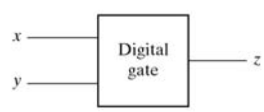
(a) Positive logic (b) Negative logic

Fig. 2-9 signal assignment and logic polarity

Θετική και Αρνητική Λογική

x	y	F
L	L	L
L	H	L
H	L	L
H	H	H

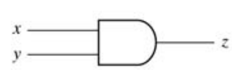
(a) Truth table with H and L



(b) Gate block diagram

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

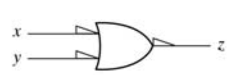
(c) Truth table for positive logic



(d) Positive logic AND gate

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

(e) Truth table for negative logic



(f) Negative logic OR gate

Fig. 2-10 Demonstration of positive and negative logic

Ολοκληρωμένα Κυκλώματα

- *Επίπεδα ολοκλήρωσης*
  - SSI
  - MSI
  - LSI
  - VLSI
- *Οικογένειες ψηφιακής λογικής*
  - TTL
  - ECL
  - CMOS
- *Χαρακτηριστικά*
  - ικανότητα οδήγησης (Fan-Out)
  - κατανάλωση ισχύος (power dissipation)
  - καθυστέρηση διάδοσης (propagation delay)
  - περιθώριο θορύβου (noise margin)

Παράρτημα - Απλοποίηση Λογικής

- Δεδομένης μιας έκφρασης Boole θέλουμε να ελαχιστοποιήσουμε :
  - αριθμό όρων
  - αριθμό μεταβλητών
  - αριθμό πυλών
- πχ η έκφραση :
  - f = B'C'D' + B'D + ABC + BC'D + ABD' + (A+B+D')'
- έχει 6 όρους, 4 μεταβλητές, 17 σύμβολα
- μπορεί να ελαχιστοποιηθεί ως :
  - f = B'C' + B'D + AB + C'D
- όπου έχουμε 4 όρους, 4 μεταβλητές και 8 σύμβολα

Παράδειγμα Ελαχιστοποίησης

$$\mathbf{f = B'C'D' + B'D + ABC + BC'D + ABD' + (A+B+D')'}$$

σημαντικό θεώρημα : $\boxed{x + xy = x}$

$$f = B'C'D' + \underline{B'D} + ABC + BC'D + ABD' + \underline{A'B'D}$$

$$= B'C'D' + ABC + BC'D + ABD' + B'D$$

$$= \underline{B'C'D'} + ABC + BC'D + ABD' + B'D + \underline{B'DC'}$$

$$= B'C'(D' + D) + ABC + BC'D + ABD' + B'D$$

$$= B'C' + \underline{B'C'D} + ABC + \underline{BC'D} + ABD' + B'D$$

$$= B'C' + C'D\ (B + B') + ABC + ABD' + B'D$$

$$= B'C' + B'D + C'D + ABC + ABD'$$

$$= B'C' + B'D + C'D + ABC'D + ABCD + \underline{ABCD'} + \underline{ABCD'} + ABC'D'$$

$$= B'C' + B'D + C'D + ABD(C+C') + ABD'(C+C')$$

$$= B'C' + B'D + C'D + ABD + ABD'$$

$$= B'C' + B'D + C'D + AB$$