$1^{η}$ Θεματική Ενότητα :
Αριθμητικά Κυκλώματα

Επιμέλεια διαφανειών:
Χρ. Καβουσιανός

Άθροιση

0	0	1	1	a
+0	+1	+0	+1	+b
0	1	1	10	2c+s

κρατούμενο

Μεταφέρεται στην επόμενη βαθμίδα σημαντικότητας

άθροισμα

Τελικό αποτέλεσμα (δεν περνά από επεξεργασία ξανά)

1 ← Κρατούμενο προηγούμενης βαθμίδας

$$\begin{array}{r} 1 \\ +1 \\ \hline 11 \end{array}$$

Κρατούμενο επόμενης βαθμίδας

Ημι-Αθροιστής (Half Adder)

1.Καθορισμός προβλήματος: κύκλωμα που να προσθέτει δύο δυαδικά ψηφία.

2.Πλήθος εισόδων/εξόδων: 2 είσοδοι – 2 έξοδοι.

3.Ονομασία εισόδων/εξόδων: έστω x, y οι δύο είσοδοι (προσθετέοι) και C (κρατούμενο), S (άθροισμα) οι δύο έξοδοι.

4.Πίνακας αλήθειας:

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

Ημι-Αθροιστής

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

(α)

$S = x'y + xy'$

$C = xy$

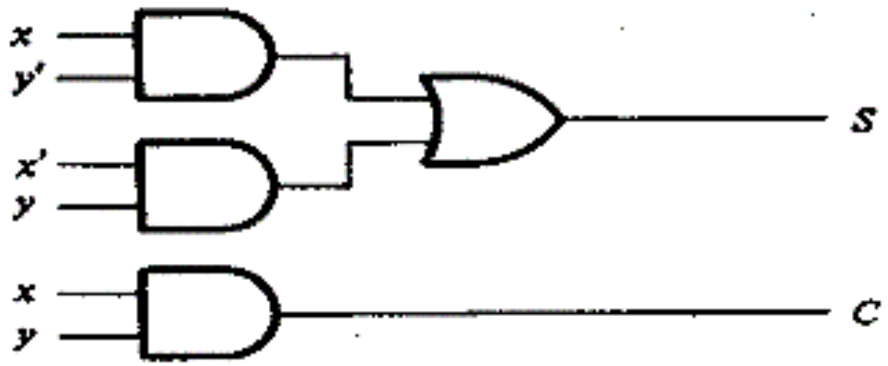
(β)

$S' = xy + x'y'$

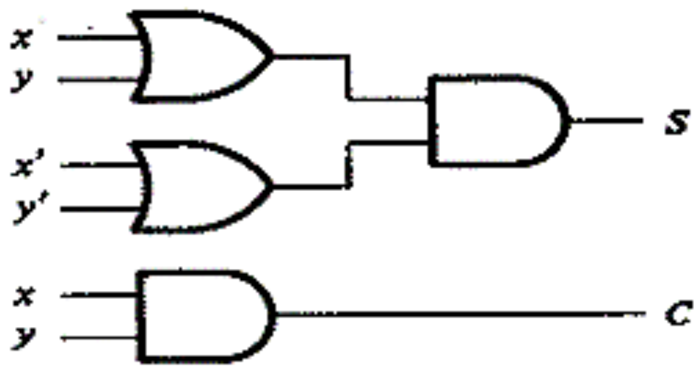
$S'' = S = (xy + x'y')' =$

$(x' + y')(x + y)$

$C = xy$



(α) $S = xy' + x'y$
$C = xy$



(β) $S = (x + y)(x' + y')$
$C = xy$

Ημι-Αθροιστής

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

(γ)

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

$S = (C + x'y')'$

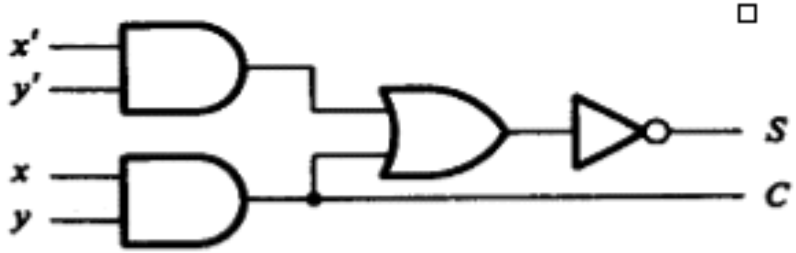
$C = xy$

(δ)

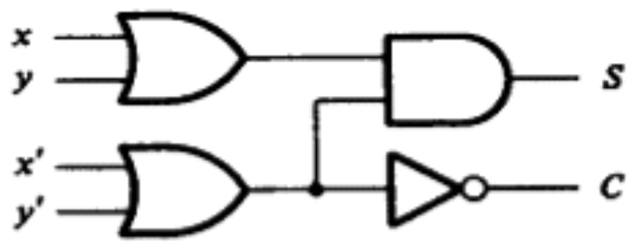
$C = (xy)'' = (x' + y')'$

$S = (C + x'y')'$

$S = C'(x + y) = (x'+y')(x+y)$



(γ) $S = (C + x'y')'$
$C = xy$



(δ) $S = (x + y)(x' + y')$
$C = (x' + y')'$

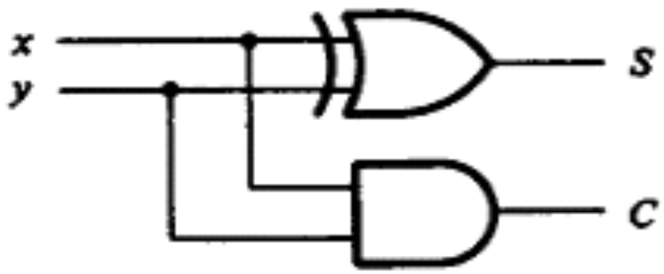
Ημι-Αθροιστής

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

(ε)

$S = x \oplus y$

$C = xy$



(ε) $S = x \oplus y$
$C = xy$

Πλήρης Αθροιστής (Full Adder)

1.Καθορισμός προβλήματος: κύκλωμα που να προσθέτει τρία δυαδικά ψηφία.

2.Πλήθος εισόδων/εξόδων: 3 είσοδοι – 2 έξοδοι.

3.Ονομασία εισόδων/εξόδων: έστω x, y οι δύο είσοδοι (προσθετέοι), z το κρατούμενο της προηγούμενης βαθμίδας και C (κρατούμενο), S (άθροισμα) οι δύο έξοδοι.

4.Πίνακας αληθείας:

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

Πλήρης-Αθροιστής

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

S

x \ yz	00	01	11	10
0		1		1
1	1		1	

C

x \ yz	00	01	11	10
0			1	
1		1	1	1

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

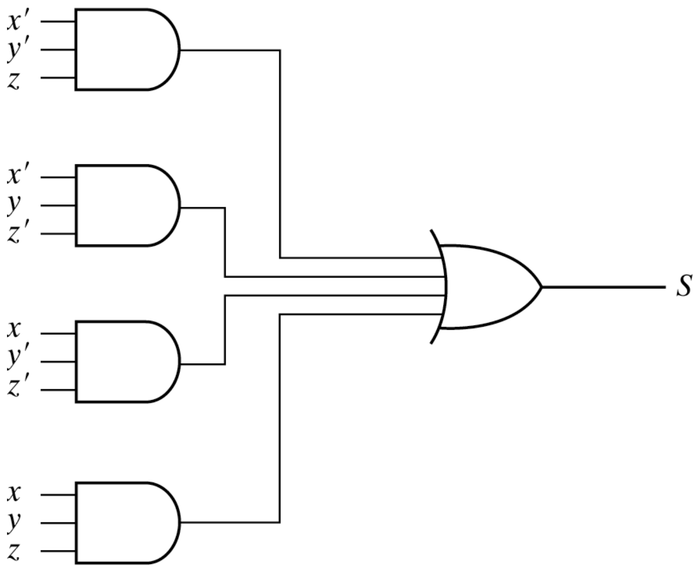
$S = (x'y' + xy)z + (x'y + xy')z' \Rightarrow$

$S = (x \oplus y)'z + (x \oplus y)z' \Rightarrow$

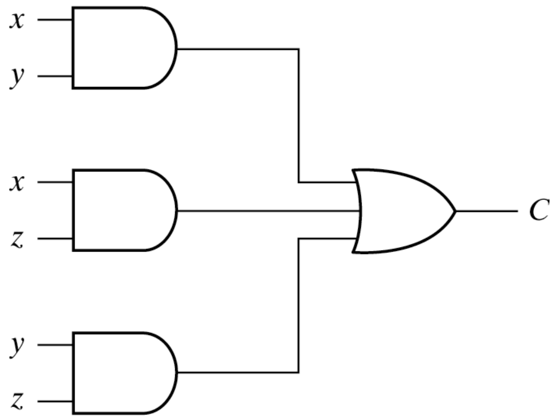
$S = x \oplus y \oplus z$

$C = xy + yz + xz$

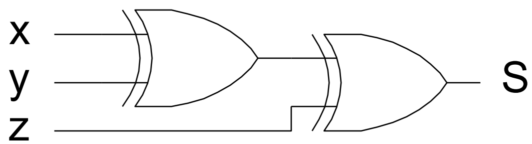
Πλήρης-Αθροιστής



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



$C = xy + yz + xz$



$S = x \oplus y \oplus z$

Πλήρης-Αθροιστής

Ένας πλήρης αθροιστής μπορεί να υλοποιηθεί με δύο ημιαθροιστές και μία πύλη Η.

$$S = x \oplus y \oplus z \qquad\qquad C = xy + yz + xz$$

Οι συναρτήσεις που υλοποιεί ο ημιαθροιστής HA_1 είναι εξ ορισμού:

$S_1 = x \oplus y$ $\qquad$ $C_1 = xy$ $\quad$ (HA_1 με εισόδους x,y, εξόδους S_1, C_1)

Το γενικό άθροισμα πρέπει να είναι $S = x \oplus y \oplus z = \underline{S_1 \oplus z}$

Το γενικό κρατούμενο πρέπει να είναι

$C = xy+yz+xz=xy+(x+x')yz+x(y+y')z=xy+xyz+x'yz+xyz+xy'z=$

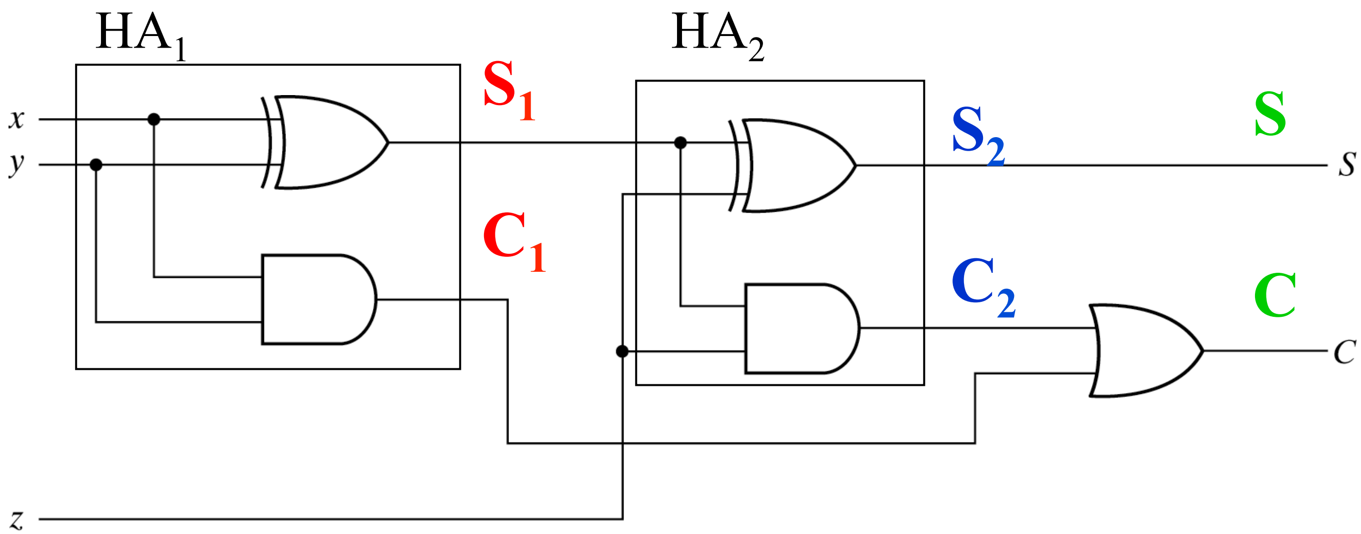
$=xy+xyz+(x'yz+xy'z)+xyz= xy+xyz+(x \oplus y)z=xy(1+z)+(x \oplus y)z=$

$= xy+(x \oplus y)z=C_1+\underline{S_1 z}$.

Πλήρης-Αθροιστής

$S_1 = x \oplus y$	$S_2 = S_1 \oplus z$	$S = S_2$
$C_1 = xy$	$C_2 = S_1 z$	$C = C_1 + C_2$

Αν ο 2ος ημιαθροιστής έχει εισόδους S_1, z, και εξόδους S_2, C_2 τότε το S_2 είναι το γενικό άθροισμα αφού $S_2 = S_1 \oplus z = S$, ενώ το κρατούμενο του είναι $C_2 = S_1 z$ οπότε το γενικό κρατούμενο γίνεται $C = C_1 + C_2$ (HA$_2$ με εισόδους S_1,z, εξόδους S_2, C_2)



Ημι-Αφαιρέτης (Half Subtractor)

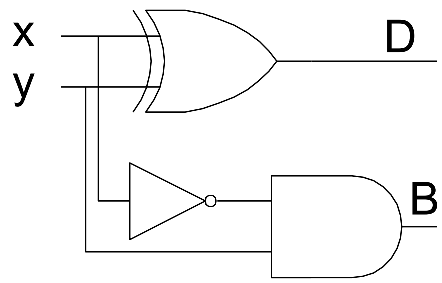
1.<u>Καθορισμός προβλήματος</u>: κύκλωμα που να αφαιρεί δύο δυαδικά ψηφία (x – y).

2.<u>Πλήθος εισόδων/εξόδων</u>: 2 είσοδοι – 2 έξοδοι.

3.<u>Ονομασία εισόδων/εξόδων</u>: έστω x, y οι δύο είσοδοι και B (δανεικό-κρατούμενο), D (διαφορά) οι δύο έξοδοι.

4.<u>Πίνακας αλήθειας</u>:

x	y	B	D
0	0	0	0
0	1	1	1
1	0	0	1
1	1	0	0



5. <u>Απλοποίηση συναρτήσεων</u>: D=x΄y+xy΄=x⊕y, B=x΄y

Πλήρης Αφαιρέτης (Full Subtractor)

1. Καθορισμός προβλήματος: κύκλωμα που να αφαιρεί δύο δυαδικά ψηφία και να λαμβάνει υπόψη πιθανό δανεικό από την προηγούμενη βαθμίδα.

2. Πλήθος εισόδων/εξόδων: 3 είσοδοι – 2 έξοδοι.

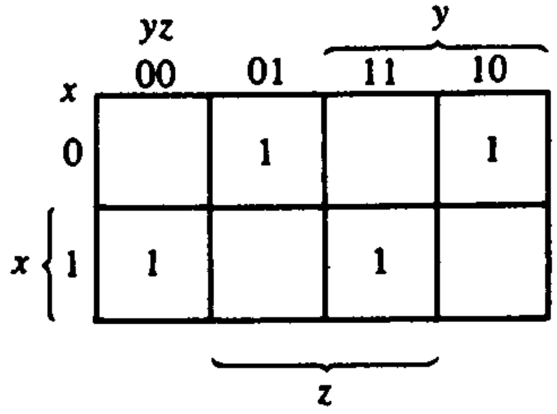
3. Ονομασία εισόδων/εξόδων: έστω x, y, z οι τρεις είσοδοι (μειωτέος, αφαιρετέος, δανεικό), και B (δανεικό-κρατούμενο), D (διαφορά) οι δύο έξοδοι.

4. Πίνακας αλήθειας: x-y-z

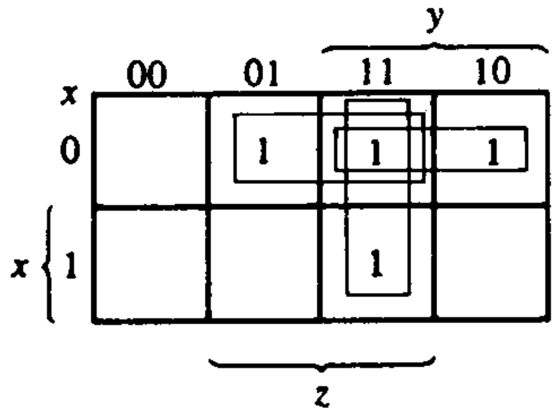
x	y	z	B	D
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	1	0
1	0	0	0	1
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

Πλήρης Αφαιρέτης

x	y	z	B	D
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	1	0
1	0	0	0	1
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1



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



$B = x'y + x'z + yz$

$D = x \oplus y \oplus z$

Η συνάρτηση D είναι ίδια με τη συνάρτηση S του πλήρη αθροιστή.

Πλήρης Αφαιρέτης

Η δυαδική αφαίρεση δεν γίνεται με τον κλασσικό τρόπο ⇔ Χρήση αριθμητικής συμπληρωμάτων ως προς 2

Παράδειγμα: 13-6

13 = 00001101

6 = 00000110 => -6 = 11111010

13 – 6 = 13 + (- 6)

00001101 – 00000110 =

00001101 + 11111010 =

(1)00000111 = +7

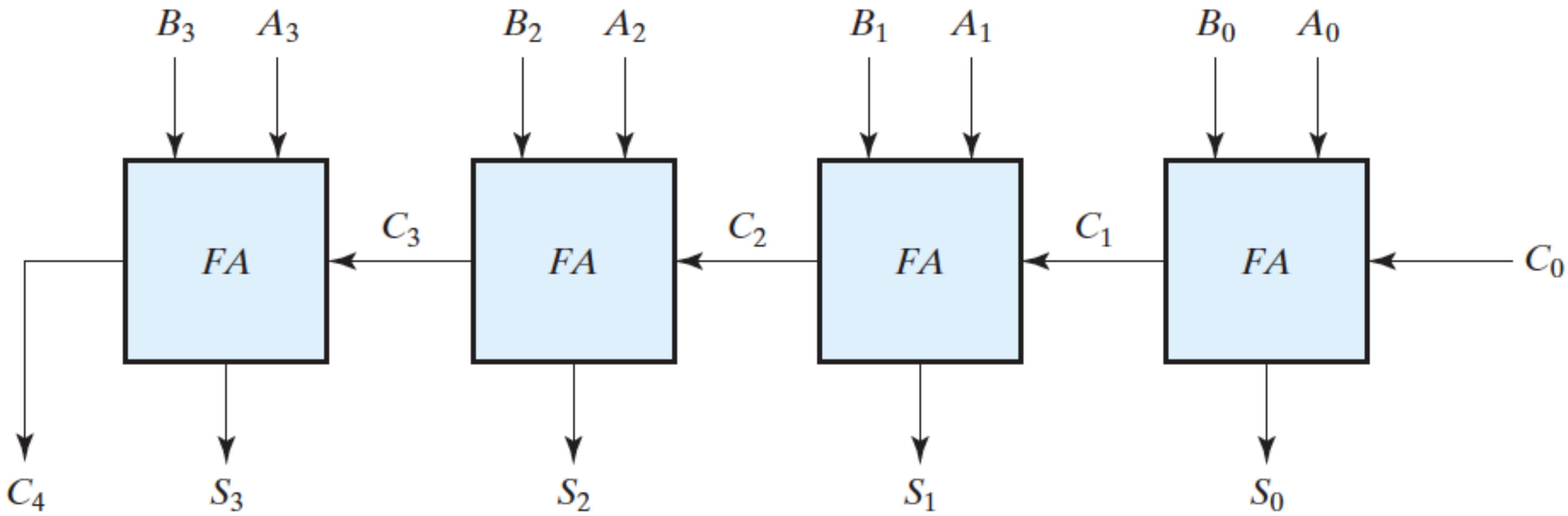
Η σημασία των αθροιστών είναι μεγάλη!!

Δυαδικός Αθροιστής

Δείκτης i	4	3	2	1	
Κρατούμενο εισόδου	0	1	1	0	C_i
Προσθετέος	1	0	1	1	A_i
Προσθετέος	0	0	1	1	B_i
Άθροισμα	1	1	1	0	S_i
Κρατούμενο εξόδου	0	0	1	1	C_{i+1}

Το άθροισμα δύο δυαδικών αριθμών των *n* bits μπορεί να παραχθεί είτε <u>σειριακά</u> είτε <u>παράλληλα</u>.

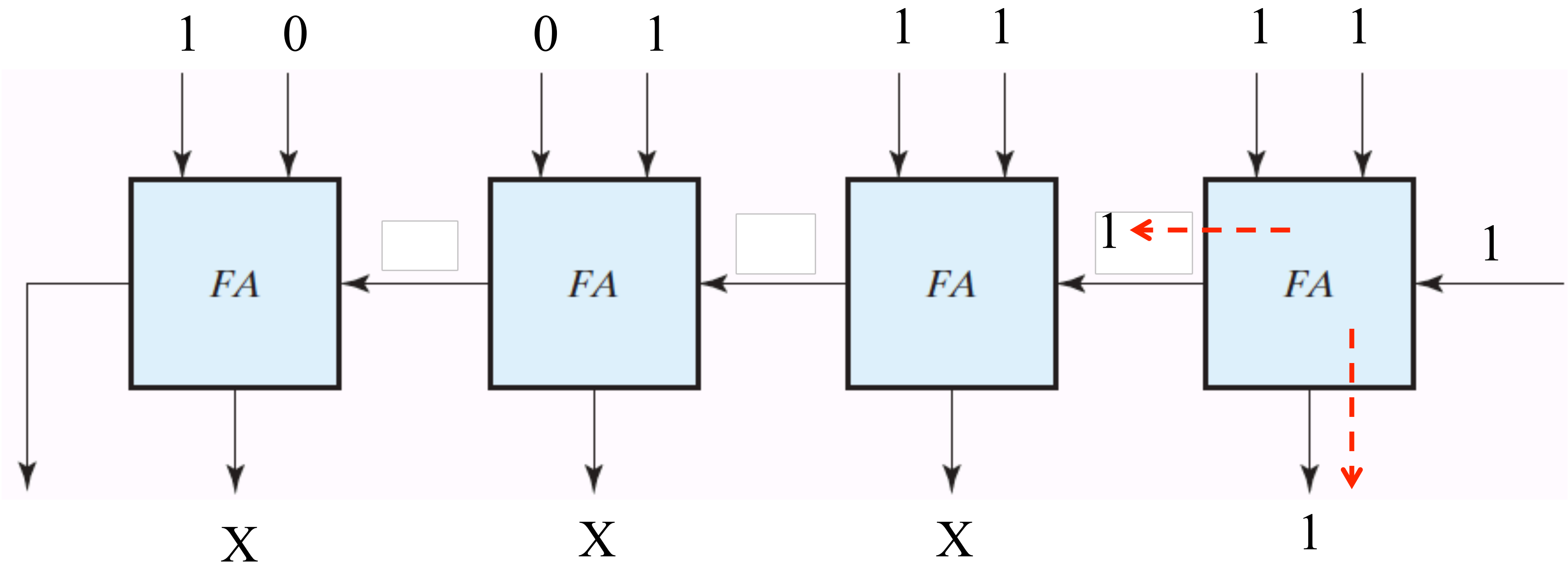
Παράλληλος Δυαδικός Αθροιστής



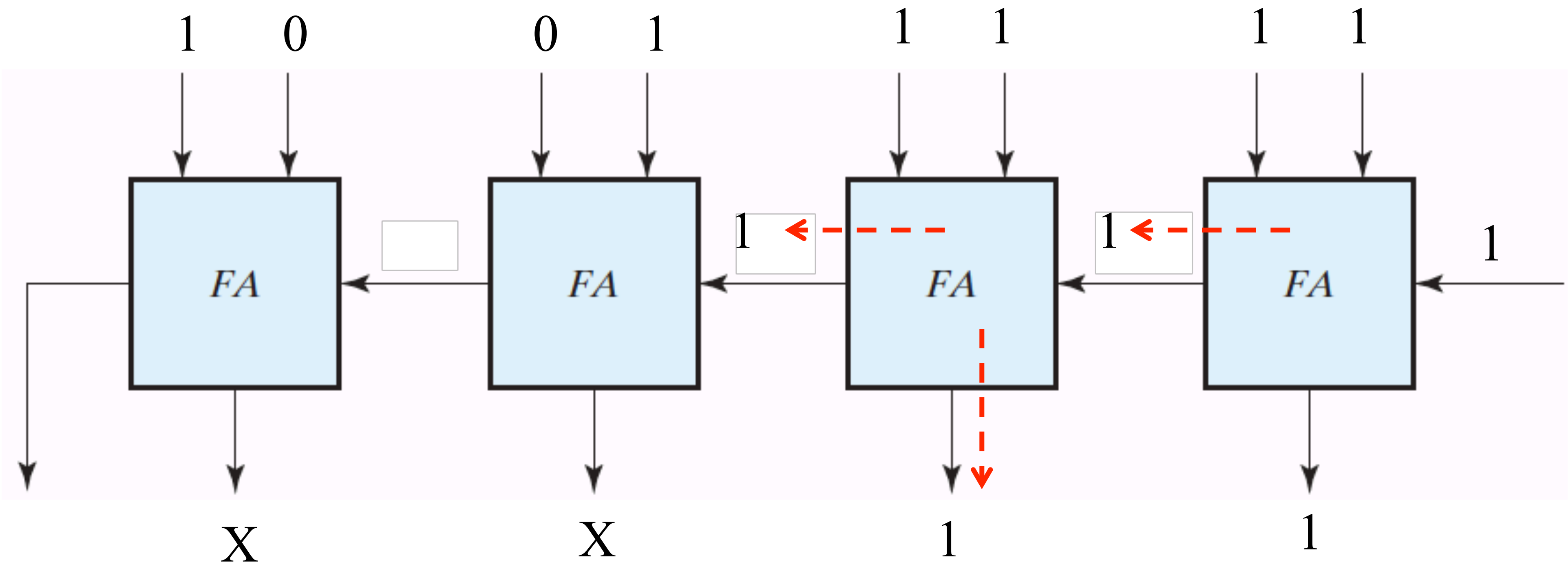
Ο <u>Παράλληλος Αθροιστής</u> αποτελείται από *n* πλήρεις αθροιστές και παράγει το αποτέλεσμα σε 1 κύκλο ρολογιού.

Υλοποίηση με συναρτήσεις: Πίνακας αλήθειας με 9 εισόδους και 2^9=512 καταστάσεις.

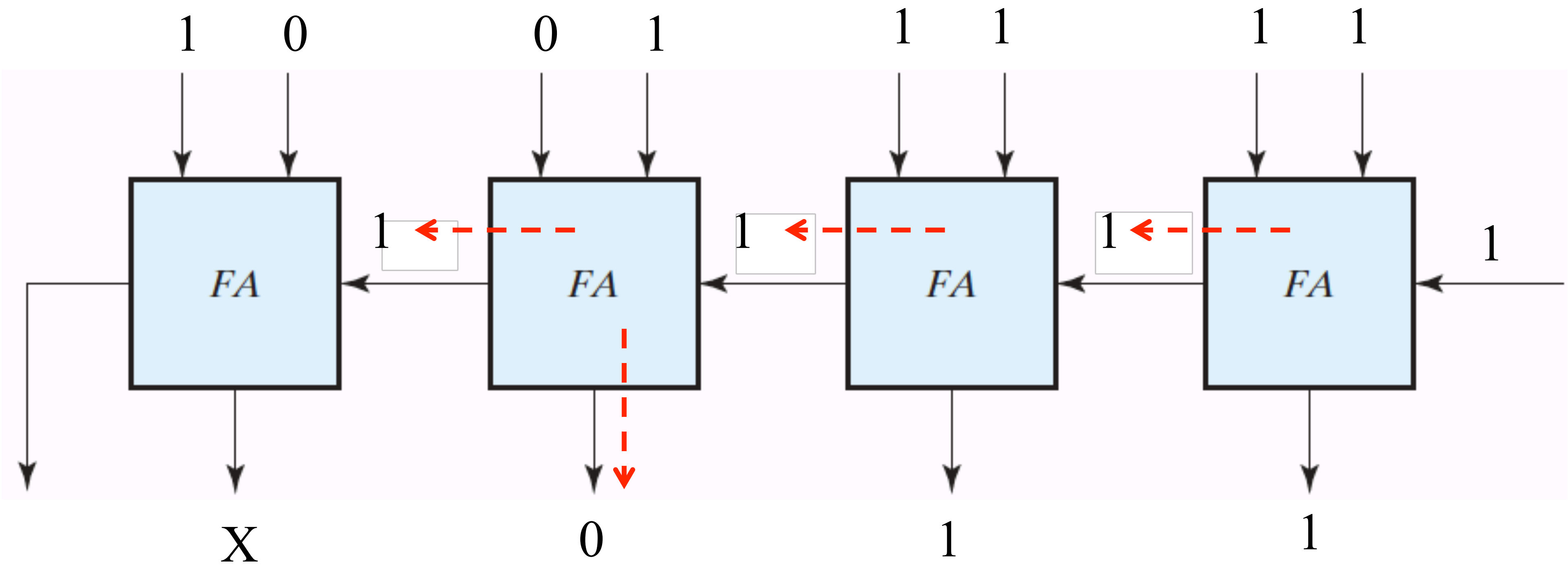
Παράδειγμα



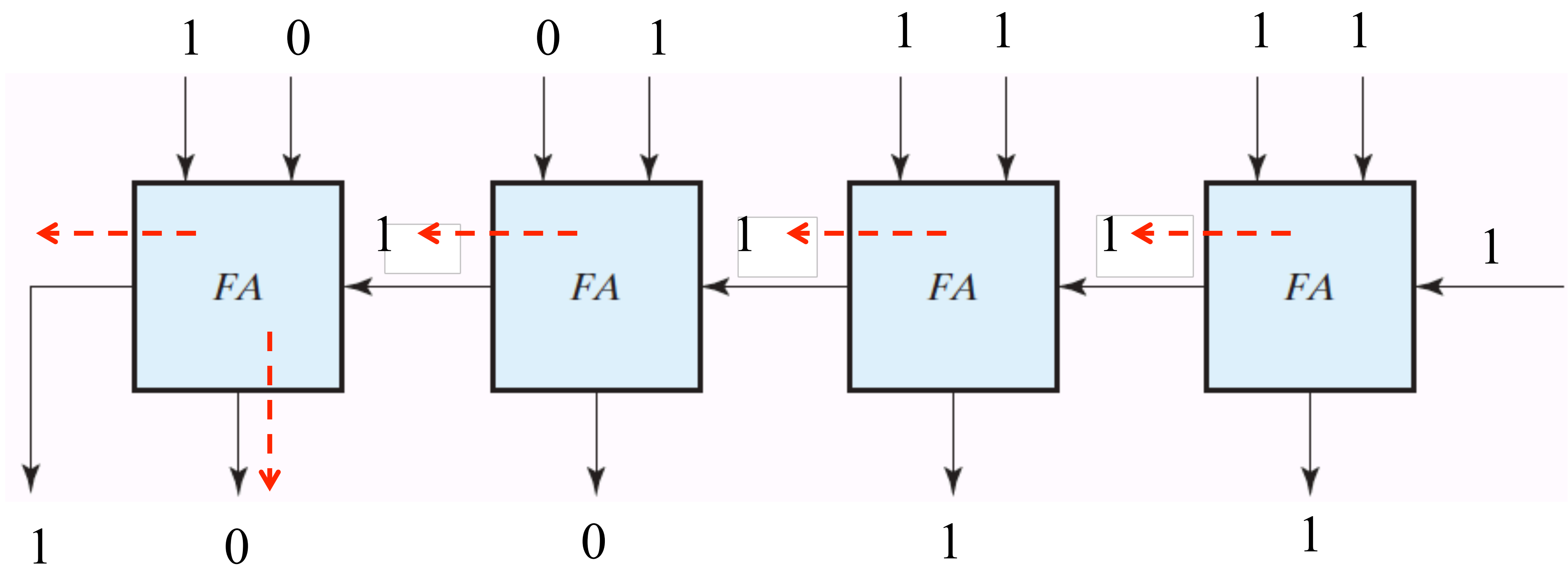
Παράδειγμα



Παράδειγμα

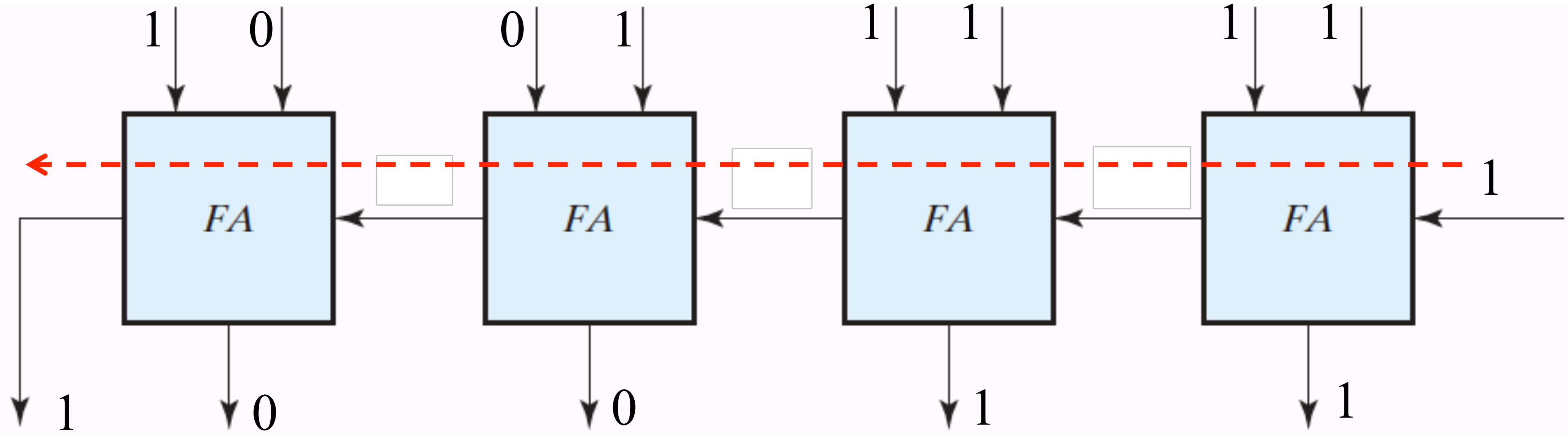


Παράδειγμα



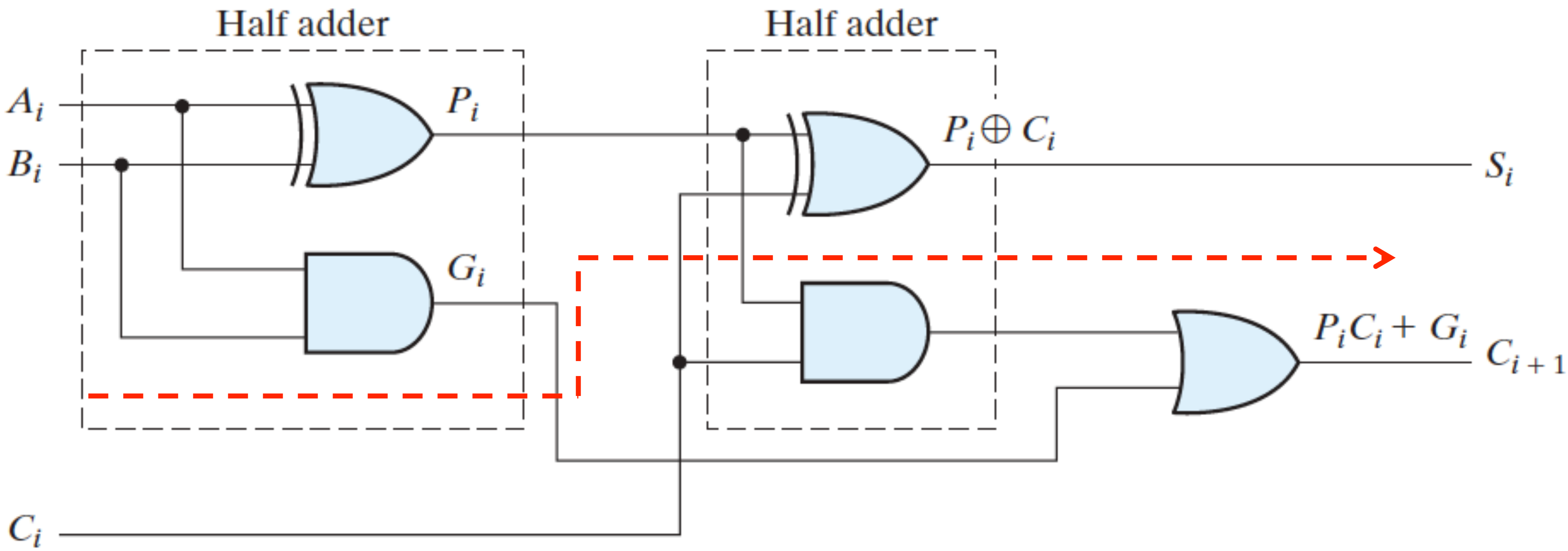
Διάδοση Κρατουμένου

Η διάδοση του κρατουμένου έχει καθυστέρηση η οποία αυξάνει με την αύξηση των βαθμίδων



Χρόνος Διάδοσης): Επίπεδα Πυλών × Καθυστέρηση Πύλης

Διάδοση Κρατουμένου



- 2 επίπεδα πυλών για κάθε διάδοση κρατουμένου.
- 2 × *n* επίπεδα για τον παράλληλο αθροιστή *n* bits.

Διάδοση Κρατουμένου

Ο χρόνος διάδοσης κρατουμένου είναι πολύ μεγάλος. ⟺ Η πρόσθεση είναι η συχνότερη πράξη.

Μείωση χρόνου διάδοσης κρατουμένου.

- Χρήση γρηγορότερων πυλών (άνω όριο).
- Αύξηση πολυπλοκότητας κυκλώματος (πρόβλεψη κρατουμένου).

Χρειαζόμαστε γρήγορους αθροιστές

Πρόβλεψη Κρατουμένου (Carry Look-Ahead)

a_i	b_i	c_i	c_{i+1}	s_i		
0	0	0	0	0	Διαγραφή	$d=a_i'b_i'$
0	0	1	0	1		
0	1	0	0	1	Διάδοση	$p=a_i \oplus b_i$
0	1	1	1	0		
1	0	0	0	1		
1	0	1	1	0		
1	1	0	1	0	Παραγωγή	$g=a_ib_i$
1	1	1	1	1		

$$c_{i+1}=a_ib_i+c_i(a_i \oplus b_i)$$

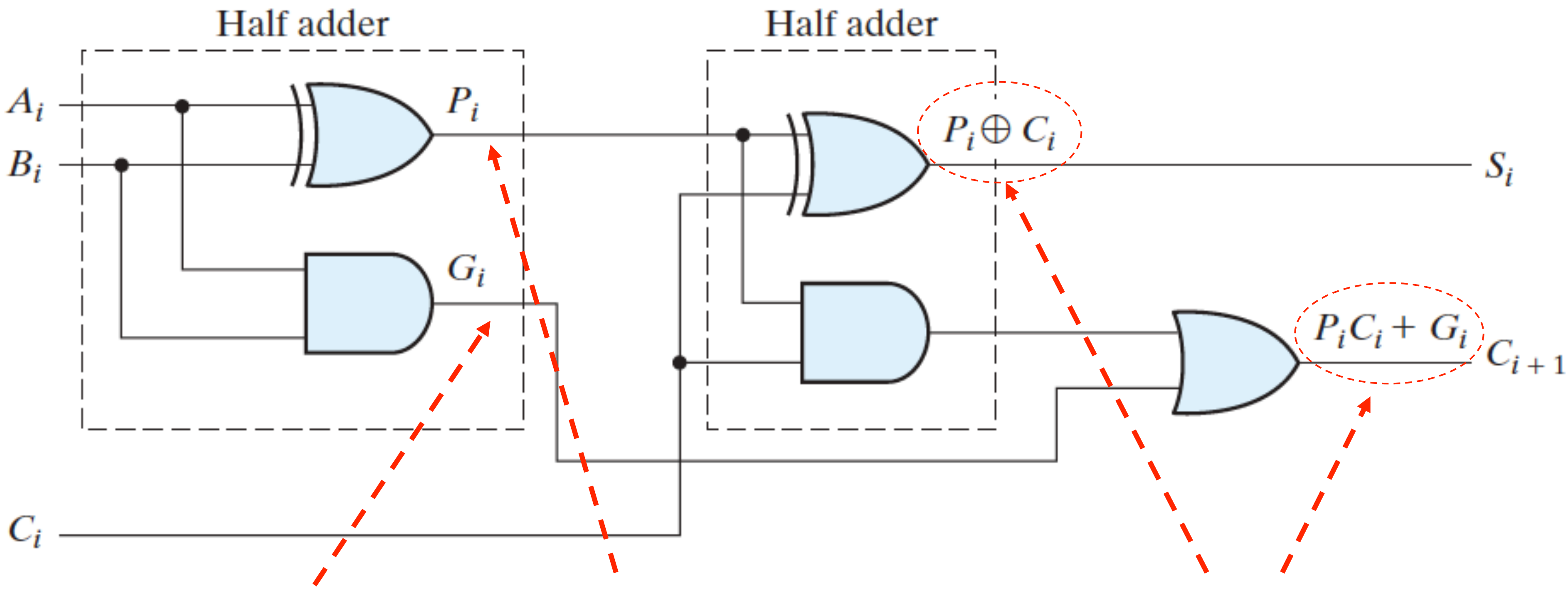
$$c_{i+1}=g_i+p_ic_i \qquad s_i=p_i \oplus c_i$$

Παρατήρηση:

Όταν $a_i=b_i$ τότε $p=0$ και το κρατούμενο είτε διαγράφεται είτε παράγεται

Δεν χρειάζεται αναμονή για το προηγούμενο κρατούμενο

Πρόβλεψη Κρατουμένου (Carry Look-Ahead)



$G_i = A_iB_i$

Γεννητής Κρατουμένου

$P_i = A_i \oplus B_i$

Διαδοτής Κρατουμένου

Άθροισμα και κρατούμενο παράγονται από τα P, G και το προηγούμενο κρατούμενο

Πρόβλεψη Κρατουμένου (Carry Look-Ahead)

$C_2 = G_1 + P_1C_1$

$C_3 = G_2 + P_2C_2 = G_2 + P_2(G_1 + P_1C_1) = G_2 + P_2G_1 + P_2P_1C_1$

$C_4 = G_3 + P_3C_3 = \ldots = G_3 + P_3G_2 + P_3P_2G_1 + P_3P_2P_1C_1$

Η αναδρομικότητα εισάγει την καθυστέρηση και την αναμονή των κρατουμένων

Χωρίς αναδρομικότητα έχουμε περισσότερο υλικό αλλά δεν υπάρχει αναμονή κρατουμένου

Πρόβλεψη Κρατουμένου (Carry Look-Ahead)

$$c_3=g_2+p_2g_1+p_2p_1g_0+p_2p_1p_0c_0$$

Παραγωγή κρατουμένου στην βαθμ. 2

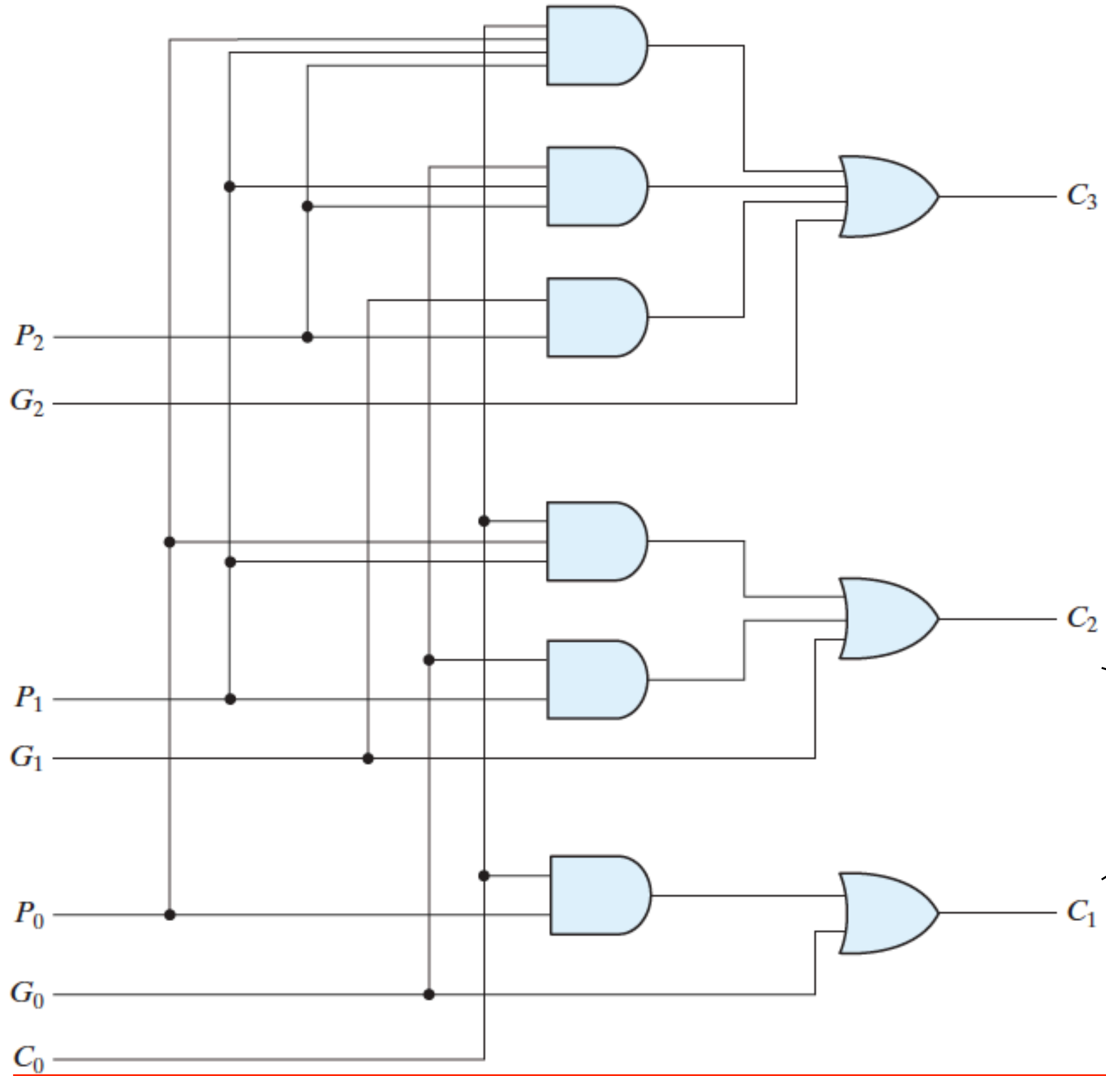
Παραγωγή κρατουμένου στην βαθμ. 1 και διάδοση από την βαθμ. 2

Παραγωγή κρατουμένου στην βαθμ. 0 και διάδοση από τις βαθμ. 1, 2

Διάδοση του κρατουμένου εισόδου από τις βαθμ. 0, 1, 2

Με τον ίδιο τρόπο μπορούμε να εκφράσουμε οποιοδήποτε κρατούμενο.

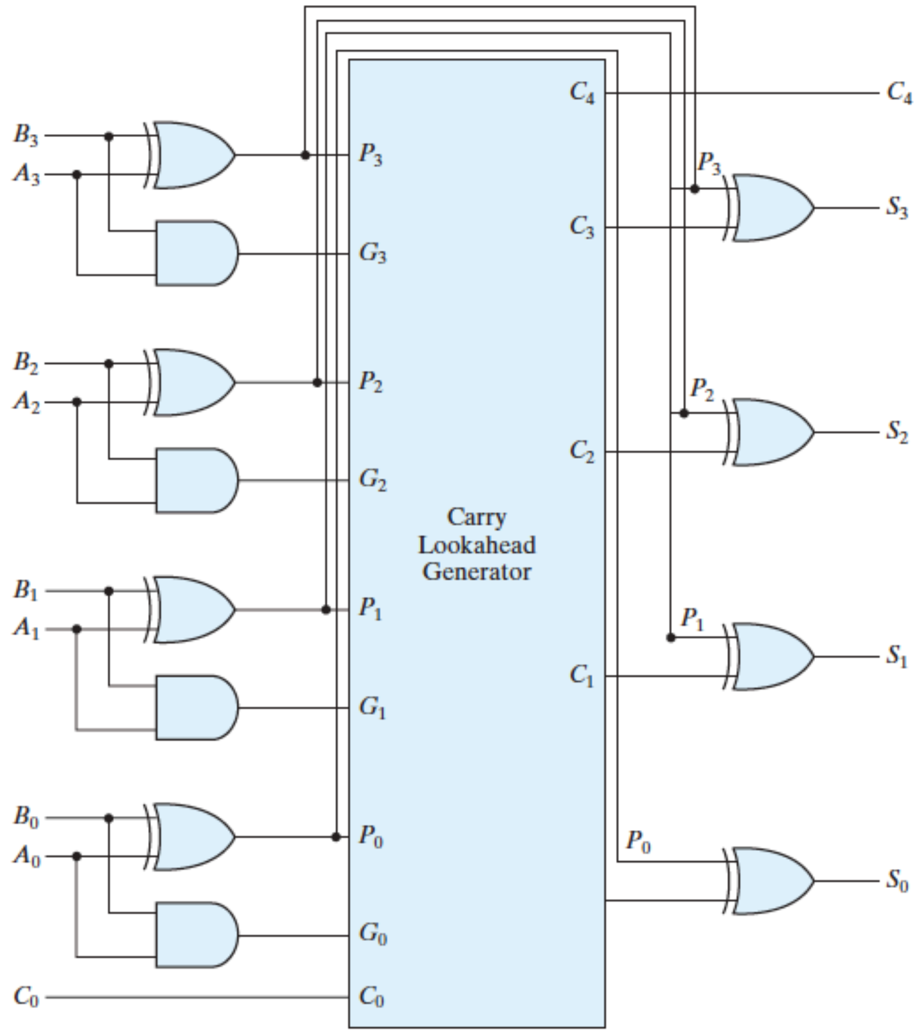
Γεννήτρια Πρόβλεψης Κρατουμένου



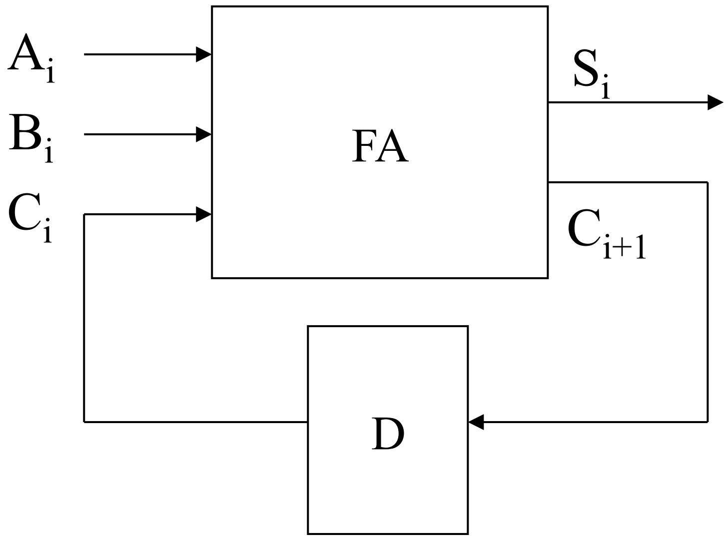
Τα ενδιάμεσα κρατούμενα χρειάζονται για την παραγωγή των αθροισμάτων

$S_i = P_i \oplus C_i$

Αθροιστής Πρόβλεψης Κρατουμένου

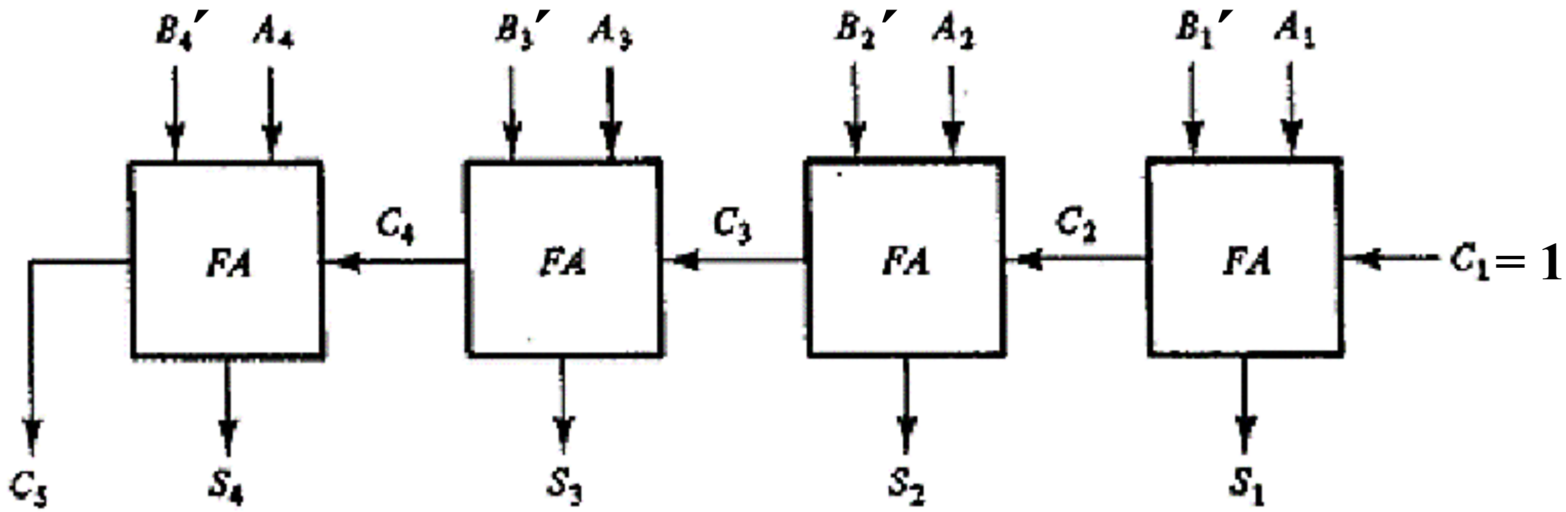


Σειριακός Δυαδικός Αθροιστής



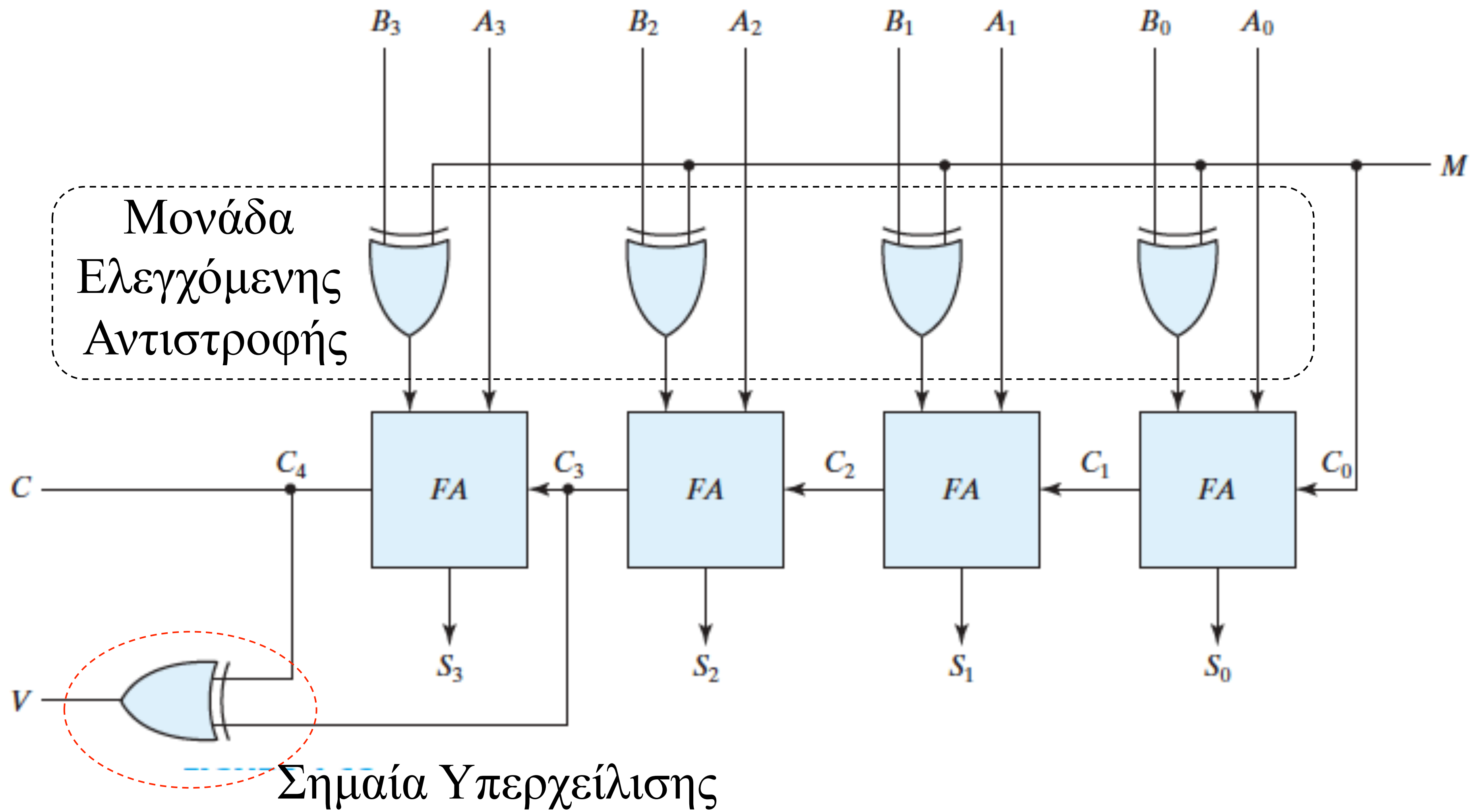
Ο <u>Σειριακός Αθροιστής</u> αποτελείται από 1 πλήρη αθροιστή και 1 στοιχείο μνήμης και παράγει το αποτέλεσμα σε *n* κύκλους ρολογιού.

Παράλληλος Δυαδικός Αφαιρέτης



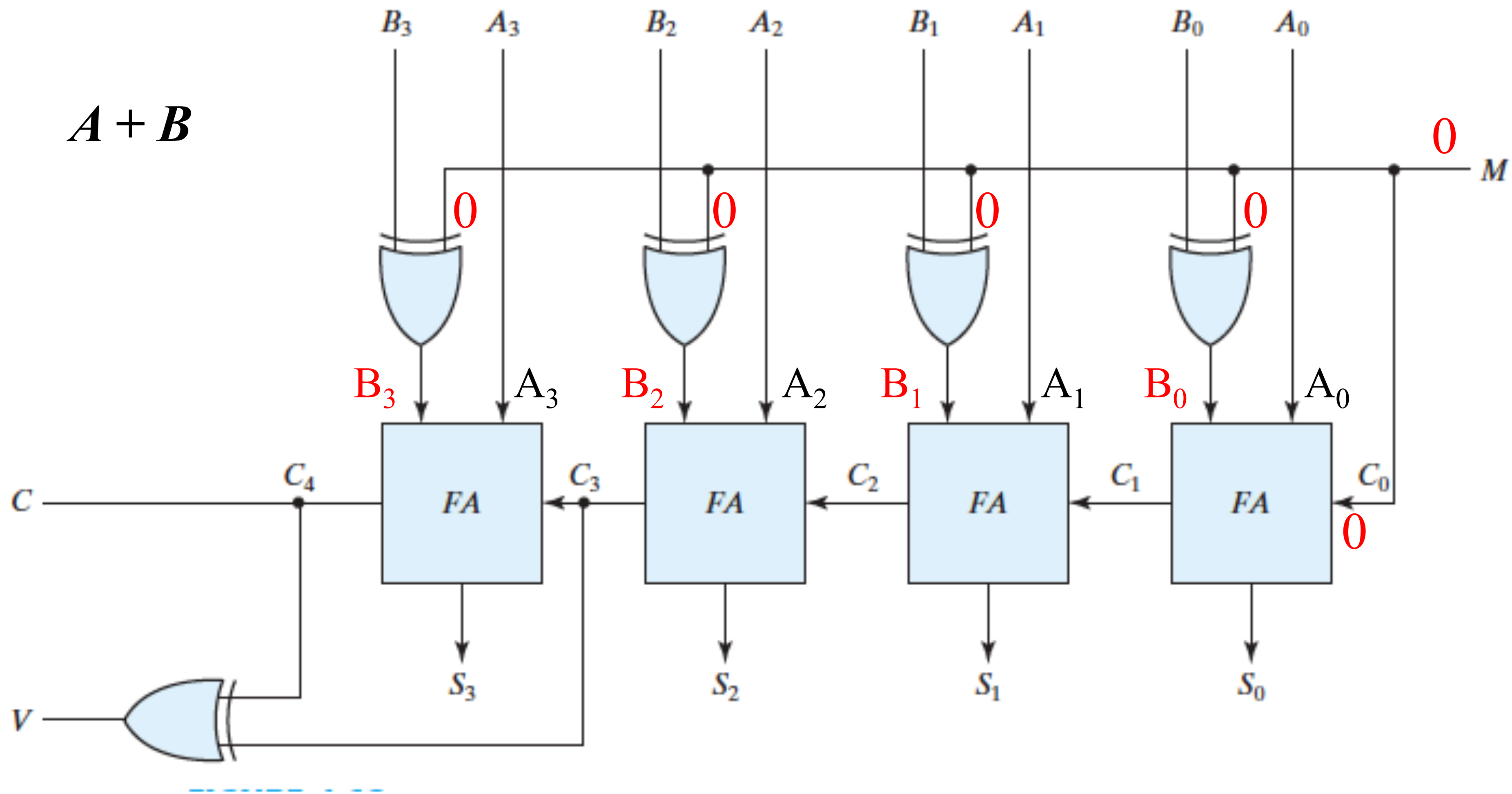
A – B = A + συμπλήρωμα-2(B) = A + B΄ + 1

Παράλληλος Δυαδικός Αθροιστής/Αφαιρέτης



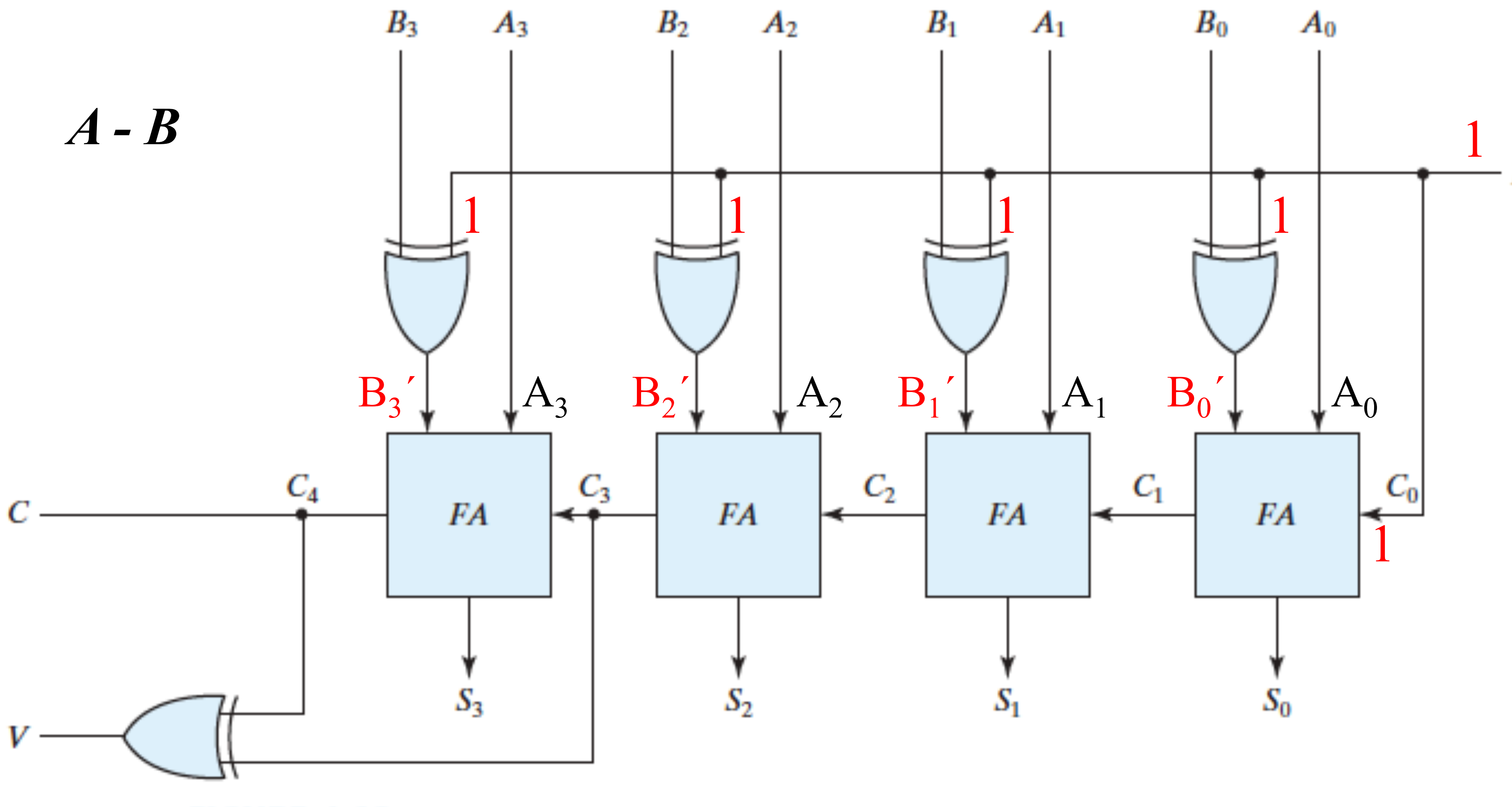
Παράδειγμα Πρόσθεσης

$A + B$



Παράδειγμα Αφαίρεσης

A - B



Υπερχείλιση

Το αποτέλεσμα της πρόσθεσης δύο δυαδικών αριθμών των ***n*** ψηφίων που απαιτεί ***n*+1** ψηφία έχει υποστεί υπερχείλιση.

↓

Σε ένα επεξεργαστικό σύστημα το αποτέλεσμα θα πρέπει να αποθηκευτεί σε ***n*** δυαδικά ψηφία (χάνεται το πιο σημαντικό ψηφίο)

↓

Το επεξεργαστικό σύστημα πρέπει να ανιχνεύσει την υπερχείλιση

↓

Ύψωση σημαίας υπερχείλισης (για προγραμματιστική χρήση)

Υπερχείλιση

Μη Προσημασμένος + Μη Προσημασμένος ⟺ Προσημασμένος + Προσημασμένος

(**κρατούμενο**) (**Xor 2 τελικών κρατουμέ**νων)

Θετικός + Αρνητικός

Θετικός – Θετικός

Αρνητικός - Αρνητικός

Δεν είναι δυνατόν να δώσουν υπερχείλιση

Παράδειγμα

Εστω προσημασμένοι αριθμοί των 8 δυαδικών ψηφίων σε αριθμητική συμπληρωμάτων ως προς 2

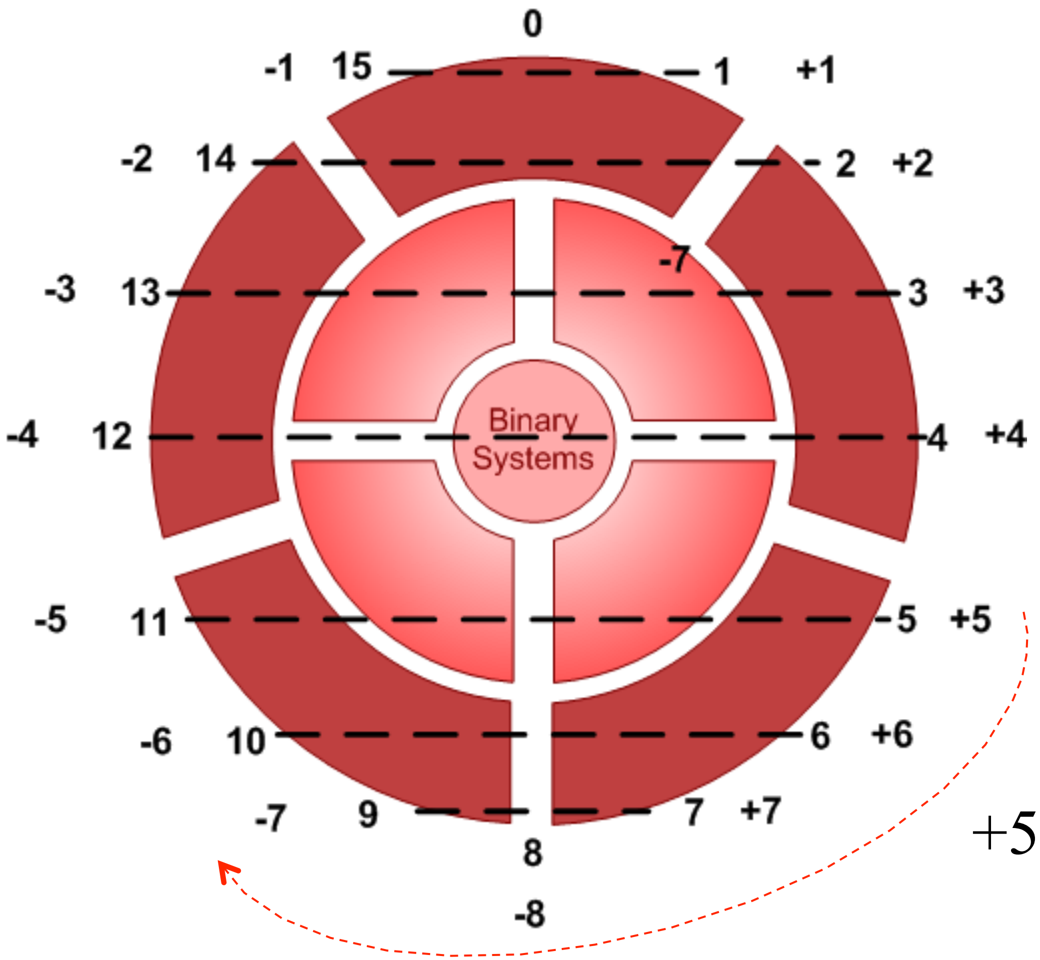
Εύρος: +127 ... -128

$$
\begin{array}{lr}
\text{carries:} & 0\ \ 1 \\
+70 & 0\ \ 1000110 \\
+80 & 0\ \ 1010000 \\
\hline
+150 & 1\ \ 0010110
\end{array}
\qquad
\begin{array}{lr}
\text{carries:} & 1\ \ 0 \\
-70 & 1\ \ 0111010 \\
-80 & 1\ \ 0110000 \\
\hline
-150 & 0\ \ 1101010
\end{array}
$$

$$1\ 0010110 \Downarrow -106 \qquad\qquad 0\ 1101010 \Downarrow +106$$

Παράδειγμα

Αρνητικοί αριθμοί

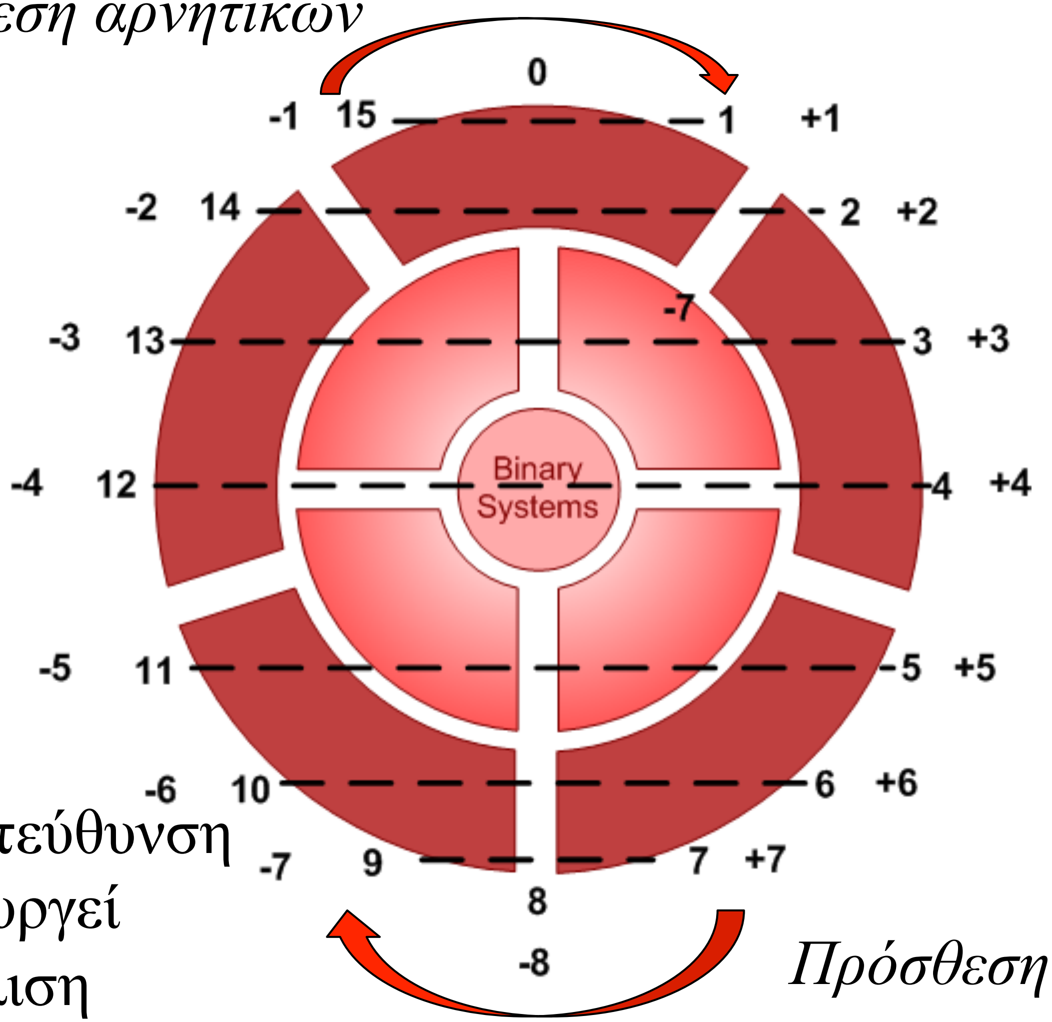


Θετικοί αριθμοί

+5+6 = -5 !!!

Παράδειγμα

Πρόσθεση αρνητικών



Η ανάποδη κατεύθυνση δεν δημιουργεί υπερχείλιση

Πρόσθεση Θετικών

Δεκαδικός Αθροιστής

Ένα ψηφιακό σύστημα που εκτελεί αριθμητικές πράξεις στο δεκαδικό σύστημα αναπαριστά τα ψηφία με κάποιο δυαδικό κώδικα (πχ Calculator).

⬇

Ένας αθροιστής για ένα τέτοιο σύστημα χρησιμοποιεί αριθμητικά κυκλώματα που δέχονται κωδικοποιημένους δεκαδικούς αριθμούς και παρουσιάζει τα αποτελέσματα στον κατάλληλο κώδικα.

⬇

Ένας δεκαδικός αθροιστής δύο ψηφίων απαιτεί 4+4+1=9 εισόδους και 4+1=5 εξόδους αφού κάθε ψηφίο κωδικοποιείται με 4 bit και χρειαζόμαστε επίσης κρατούμενα εισόδου/εξόδου.

Δεκαδικός Αθροιστής

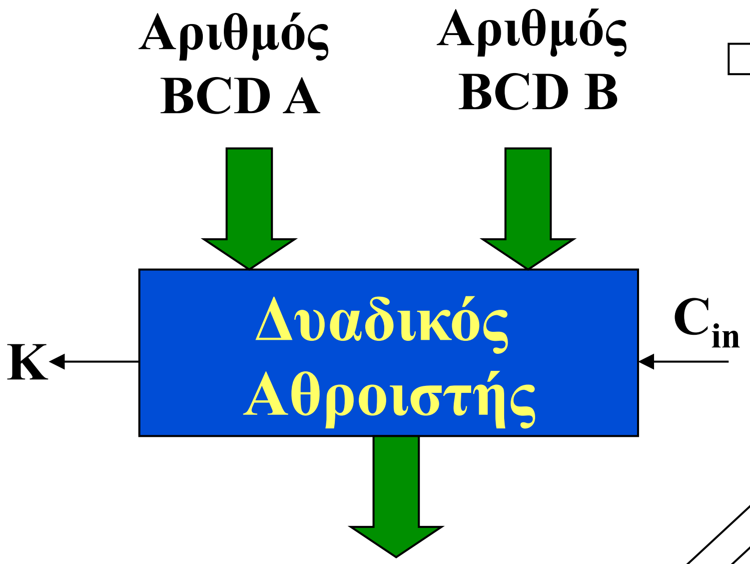
Τρόποι υλοποίησης:

Α) Σχεδιασμός συνδυαστικού κυκλώματος με 9 εισόδους και 5 εξόδους (512 δυνατοί συνδυασμοί στις εισόδους).

Β) Υλοποίηση με κυκλώματα πλήρη αθροιστή λαμβάνοντας υπόψη το γεγονός ότι 6 συνδυασμοί σε κάθε είσοδο 4 bits δε χρησιμοποιούνται.

Δεκαδικός Αθροιστής

Αρχικά προσθέτουμε τους BCD αριθμούς A, B και παίρνουμε το άθροισμα τους το οποίο είναι ένας δυαδικός αριθμός (όχι BCD).



Αποτέλεσμα $Z_8Z_4Z_2Z_1$

(Δυαδικός Αριθμός 4 bits)

A, B : 0000 … 1001

C_{in} : 0, 1

$KZ_8Z_4Z_2Z_1$: 00000 … 10011

0 19

Μετατροπή σε BCD

K	$Z_8Z_4Z_2Z_1$	⇔	C	$S_8S_4S_2S_1$
1	**0 0 1 1**		**1**	**1 0 0 1**

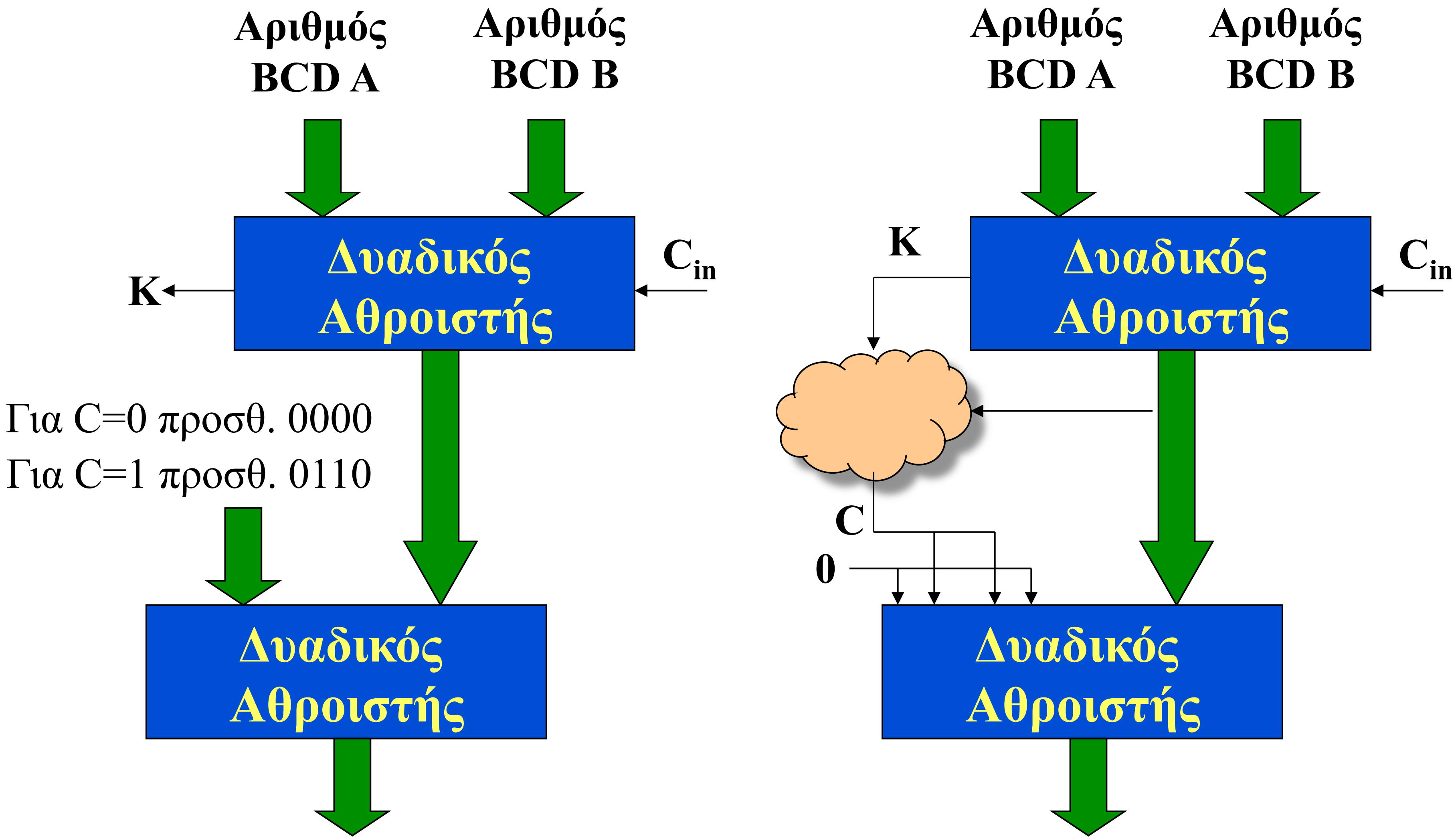
Αθροιστής BCD

Binary Sum					BCD Sum					Decimal
K	Z_8	Z_4	Z_2	Z_1	C	S_8	S_4	S_2	S_1	
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0	1	1
0	0	0	1	0	0	0	0	1	0	2
0	0	0	1	1	0	0	0	1	1	3
0	0	1	0	0	0	0	1	0	0	4
0	0	1	0	1	0	0	1	0	1	5
0	0	1	1	0	0	0	1	1	0	6
0	0	1	1	1	0	0	1	1	1	7
0	1	0	0	0	0	1	0	0	0	8
0	1	0	0	1	0	1	0	0	1	9
0	1	0	1	0	1	0	0	0	0	10
0	1	0	1	1	1	0	0	0	1	11
0	1	1	0	0	1	0	0	1	0	12
0	1	1	0	1	1	0	0	1	1	13
0	1	1	1	0	1	0	1	0	0	14
0	1	1	1	1	1	0	1	0	1	15
1	0	0	0	0	1	0	1	1	0	16
1	0	0	0	1	1	0	1	1	1	17
1	0	0	1	0	1	1	0	0	0	18
1	0	0	1	1	1	1	0	0	1	19

Δεν απαιτείται μετατροπή σε BCD

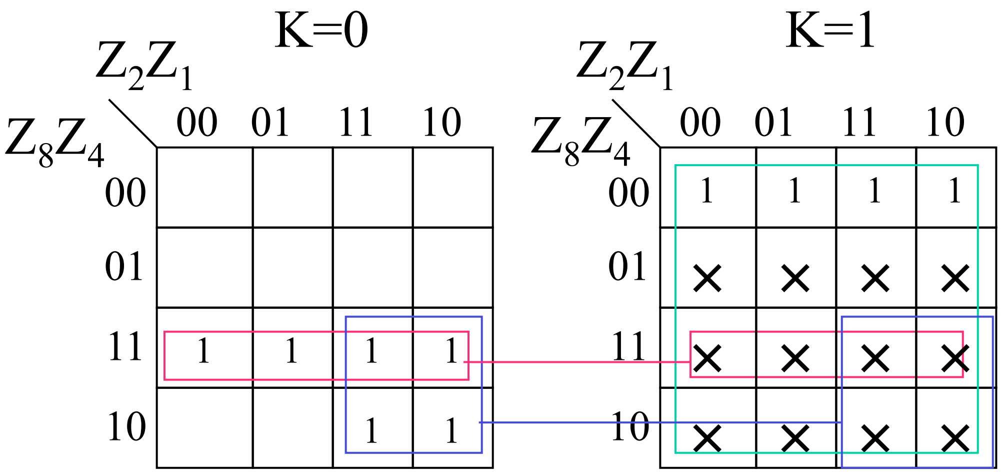
Απαιτείται πρόσθεση του +6

Αθροιστής BCD



Αθροιστής BCD

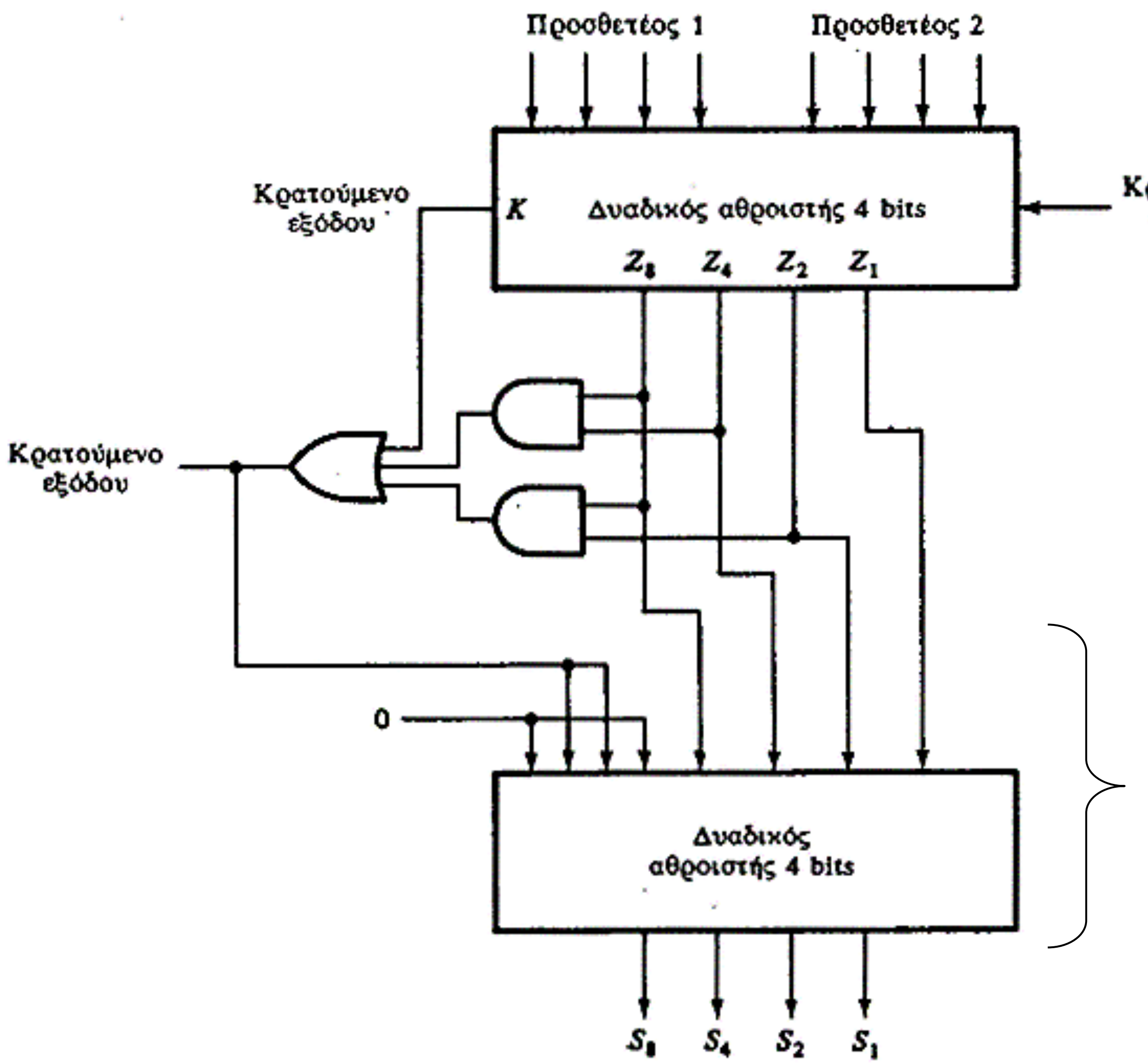
Binary Sum					
K	Z_8	Z_4	Z_2	Z_1	C
0	0	0	0	0	0
0	0	0	0	1	0
0	0	0	1	0	0
0	0	0	1	1	0
0	0	1	0	0	0
0	0	1	0	1	0
0	0	1	1	0	0
0	0	1	1	1	0
0	1	0	0	0	0
0	1	0	0	1	0
0	1	0	1	0	1
0	1	0	1	1	1
0	1	1	0	0	1
0	1	1	0	1	1
0	1	1	1	0	1
0	1	1	1	1	1
1	0	0	0	0	1
1	0	0	0	1	1
1	0	0	1	0	1
1	0	0	1	1	1



$$\boxed{C = K + Z_8Z_4 + Z_8Z_2}$$

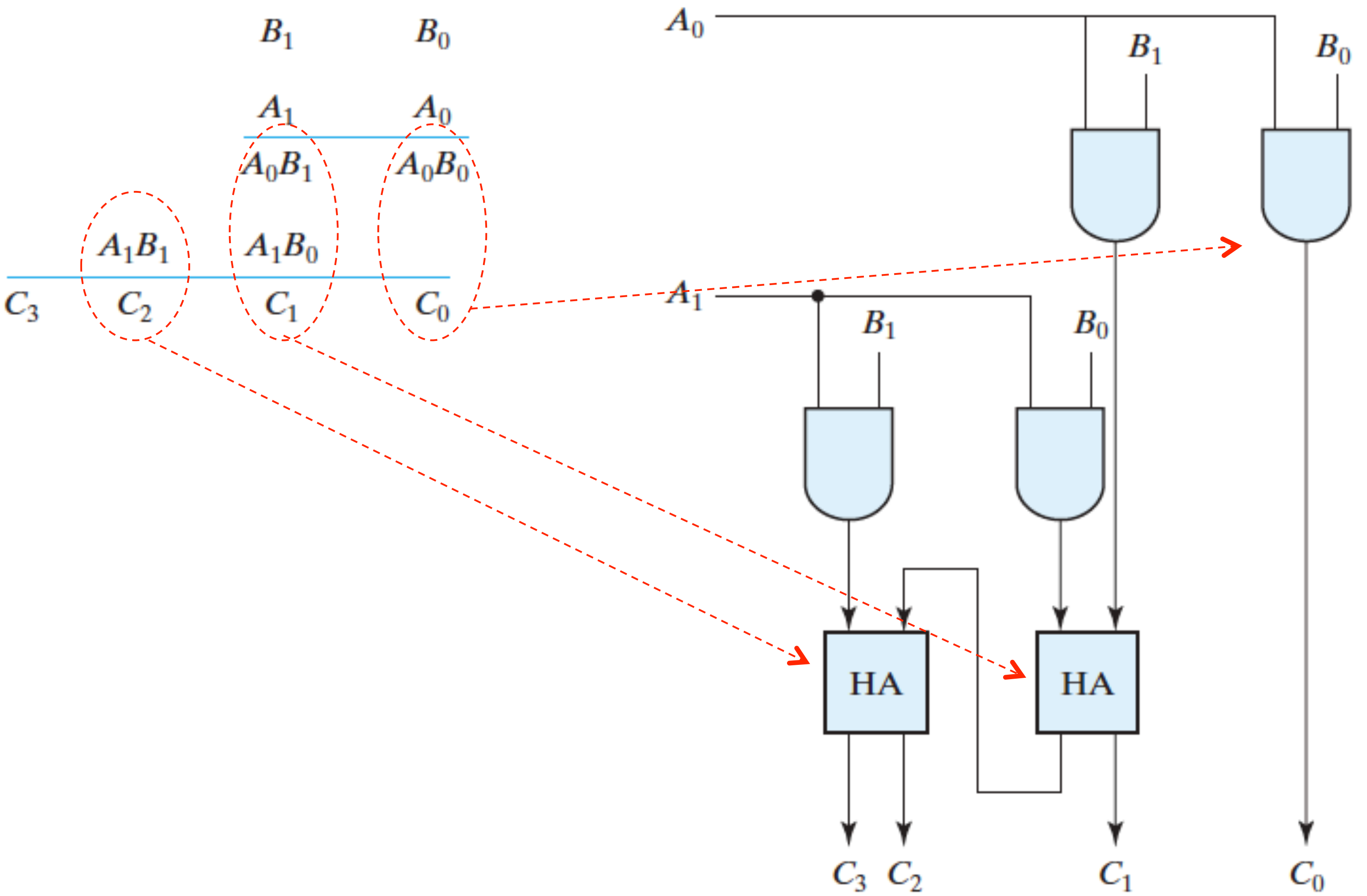
Αθροιστής BCD

$C = K + Z_8Z_4 + Z_8Z_2$



Από την τιμή 9 και πάνω προσθέτουμε την ποσότητα 0110 ώστε να έχουμε την επιθυμητή μετατροπή.

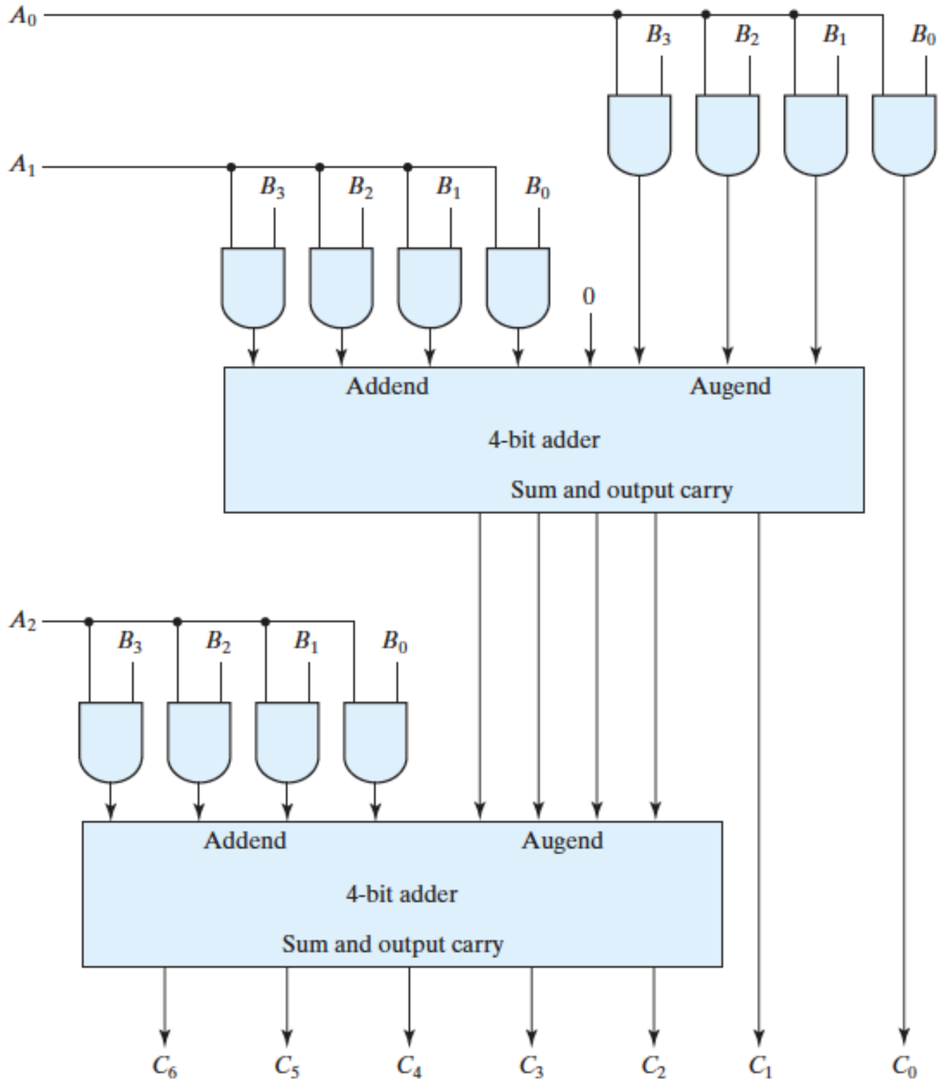
Δυαδικός Πολλαπλασιαστής



Δυαδικός Πολλαπλασιαστής

$$\mathbf{C_{6\ldots0} = B_{3\ldots0} \times A_{2\ldots0}}$$

$$
\begin{array}{ccccccc}
 & & & B_3 & B_2 & B_1 & B_0 \\
 & & \times & & A_2 & A_1 & A_0 \\
\hline
 & & & A_0B_3 & A_0B_2 & A_0B_1 & A_0B_0 \\
 & & A_1B_3 & A_1B_2 & A_1B_1 & A_1B_0 & \\
 & A_2B_3 & A_2B_2 & A_2B_1 & A_2B_0 & & \\
\hline
C_6 & C_5 & C_4 & C_3 & C_2 & C_1 & C_0
\end{array}
$$



Συγκριτής Μεγέθους (Comparator)

Συγκριτής Μεγέθους: συγκρίνει δύο αριθμούς και βρίσκει τη σχέση τους (<,>,=).

Για δύο αριθμούς των *n* bits έχουμε $2^n x 2^n = 2^{2n}$ συνδυασμούς.

Η μέθοδος χάρτη είναι ασύμφορη.

Το κύκλωμα του συγκριτή έχει αρκετή κανονικότητα.

- Έστω $A = A_3A_2A_1A_0$ και $B = B_3B_2B_1B_0$ οι δύο αριθμοί.

- Ισχύει A = B όταν όλα τα ζευγάρια (A_i, B_i) είναι ίσα, δηλαδή $A_3=B_3$ και $A_2=B_2$ και $A_1=B_1$ και $A_0=B_0$.

$$\boxed{(A=B) = x_3x_2x_1x_0 \qquad x_i = A_iB_i + A_i'B_i'}$$

Συγκριτής Μεγέθους

$$(A{=}B) = x_3x_2x_1x_0 \qquad x_i = A_iB_i + A_i'B_i'$$

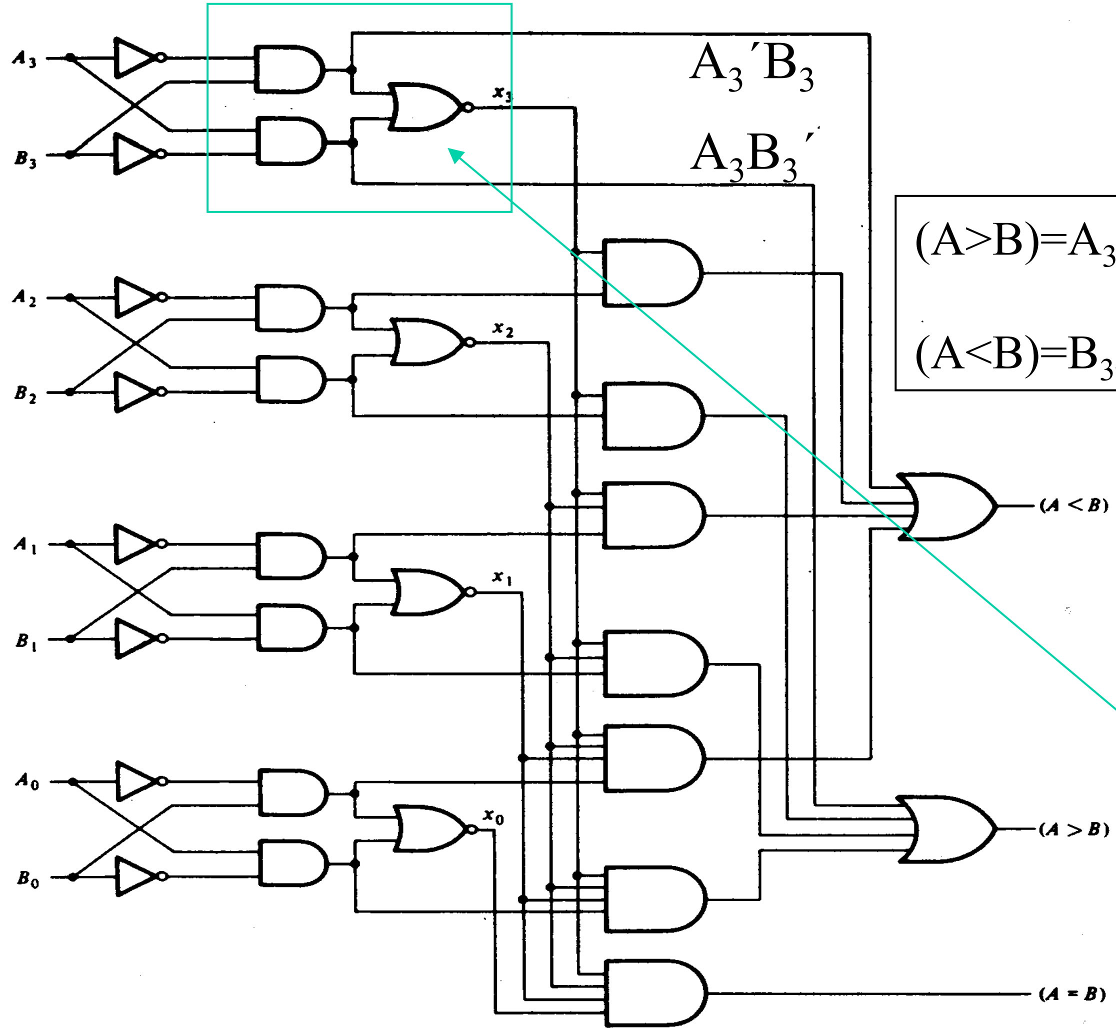
➢ Για να βρούμε εάν A<B ή A>B εξετάζουμε τα σχετικά μεγέθη των ζευγαριών ψηφίων ξεκινώντας από την πιο σημαντική θέση. Εάν τα δύο ψηφία είναι ίσα τότε συγκρίνουμε το επόμενο λιγότερο σημαντικό ζευγάρι ψηφίων.

➢ Εάν $A_i = 1$ και $B_i = 0$ τότε $A > B$, εάν $A_i = 0$ και $B_i = 1$ τότε $A < B$.

$$(A{>}B) = A_3B_3' + x_3A_2B_2' + x_3x_2A_1B_1' + x_3x_2x_1A_0B_0'$$

$$(A{<}B) = B_3A_3' + x_3B_2A_2' + x_3x_2B_1A_1' + x_3x_2x_1B_0A_0'$$

Συγκριτής Μεγέθους



$A_3{}'B_3$

$A_3B_3{}'$

$(A=B) = x_3x_2x_1x_0$

$x_i = A_iB_i + A_i{}'B_i{}'$

$(A>B)=A_3B_3{}'+x_3A_2B_2{}'+x_3x_2A_1B_1{}'+x_3x_2x_1A_0B_0{}'$

$(A<B)=B_3A_3{}'+x_3B_2A_2{}'+x_3x_2B_1A_1{}'+x_3x_2x_1B_0A_0{}'$

Μας συμφέρει στην υλοποίηση αντί για xnor να υλοποιήσουμε με AND-NOR τις σχέσεις

$x_i = (A_iB_i{}' + A_i{}'B_i)'$

καθώς τα $A_iB_i{}'$, $A_i{}'B_i$ διαμοιράζονται