ΕΛΛΗΝΙΚΟ ΑΝΟΙΚΤΟ ΠΑΝΕΠΙΣΤΗΜΙΟ
ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ ΚΑΙ ΤΕΧΝΟΛΟΓΙΑΣ
ΠΡΟΓΡΑΜΜΑ ΣΠΟΥΔΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ
ΘΕΜΑΤΙΚΗ ΕΝΟΤΗΤΑ: ΠΛΗ-21

ΨΗΦΙΑΚΗ ΣΧΕΔΙΑΣΗ

ΑΣΚΗΣΕΙΣ
ΓΡΑΠΤΩΝ ΕΡΓΑΣΙΩΝ
&
ΘΕΜΑΤΩΝ ΕΞΕΤΑΣΕΩΝ

ΣΥΝΤΕΛΕΣΤΕΣ (ΚΑΤ' ΑΛΦΑΒΗΤΙΚΗ ΣΕΙΡΑ):
Γ. ΑΛΕΞΙΟΥ
Χ. ΒΕΡΓΟΣ
Κ. ΕΥΣΤΑΘΙΟΥ
Γ. ΘΕΟΔΩΡΙΔΗΣ
Χ. ΚΑΒΟΥΣΙΑΝΟΣ
Ο. ΚΟΥΦΟΠΑΥΛΟΥ
Κ. ΛΑΜΠΡΙΝΟΥΔΑΚΗΣ
Φ. ΛΙΟΤΟΠΟΥΛΟΣ
Α. ΜΟΣΧΟΒΟΣ
Δ. ΜΠΑΚΑΛΗΣ
Σ. ΝΙΚΟΛΑΪΔΗΣ
Δ. ΝΙΚΟΛΟΣ
Β. ΠΑΛΙΟΥΡΑΣ
Ι. ΠΑΠΑΕΥΣΤΑΘΙΟΥ
Δ. ΠΑΠΑΚΩΣΤΑΣ
Α. ΣΤΟΥΡΑΙΤΗΣ
Α. ΣΚΟΔΡΑΣ
Β. ΦΩΤΟΠΟΥΛΟΣ
Α. ΧΑΤΖΟΠΟΥΛΟΣ

ΕΠΙΜΕΛΕΙΑ ΕΚΔΟΣΗΣ: Φ. ΛΙΟΤΟΠΟΥΛΟΣ – Δ. ΜΠΑΚΑΛΗΣ – Χ. ΚΑΒΟΥΣΙΑΝΟΣ

ΠΑΤΡΑ 2008

Το παρόν υλικό αποτελεί το κύριο τμήμα των ασκήσεων που δόθηκαν προς επίλυση στους φοιτητές του Ελληνικού Ανοικτού Πανεπιστημίου στα πλαίσια της Θεματικής Ενότητας **ΠΛΗ-21: Ψηφιακά Συστήματα** του Προγράμματος Σπουδών της Πληροφορικής κατά τα ακαδημαϊκά έτη 2001 - 2008.

Περιεχόμενα

I. Αριθμητικά Συστήματα 4
II. Κώδικες Αναπαράστασης Δεδομένων 27
III. Άλγεβρα Boole 33
IV. Σχεδίαση συνδυαστικών κυκλωμάτων με λογικές πύλες 54
V. Αποκωδικοποιητές – Πολυπλέκτες - Αθροιστές 89
VI. Ακολουθιακά κυκλώματα 128
VII. Σχεδίαση Καταχωρητών 165

I. Αριθμητικά Συστήματα

ΑΣΚΗΣΗ 1

Δίνονται οι αριθμοί: $\chi_{10}=244.75_{10}$ και $\psi_{10}=35.267_{10}$ σε δεκαδική αναπαράσταση.

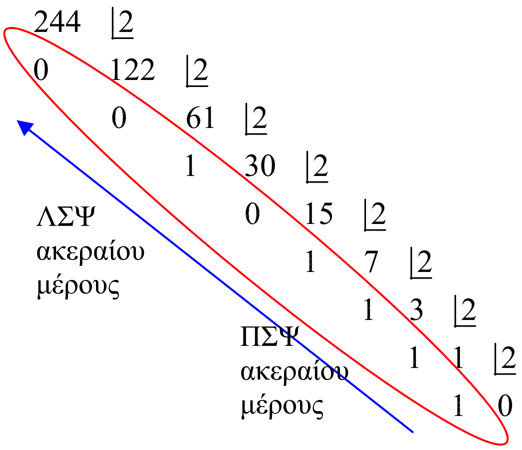
Α. Να μετατραπούν σε δυαδική αναπαράσταση (χ_2 , ψ_2) με 8 ψηφία για το ακέραιο μέρος και 4 ψηφία για το κλασματικό μέρος καθώς και σε δεκαεξαδική αναπαράσταση (χ_{16} , ψ_{16}) με 2 ψηφία για το ακέραιο μέρος και 1 ψηφίο για το κλασματικό μέρος. Η μετατροπή να εξηγηθεί αναλυτικά μόνο για τον αριθμό χ.

Β. Να μετατραπούν πάλι σε δεκαδική αναπαράσταση οι αριθμοί (χ_2 , ψ_2 και χ_{16} , ψ_{16}) που προέκυψαν από το ερώτημα Α. Σχολιάστε τις διαφορές που παρατηρείτε ανάμεσα στις δεκαδικές αναπαραστάσεις που υπολογίσατε στο ερώτημα Β και στις αρχικές.

(Υπόδειξη: Συμβουλευτείτε το Παράρτημα Α του Β' τόμου 'Αρχιτεκτονική Υπολογιστών Ι').

Λύση:

Α. Για να μετατρέψουμε τον αριθμό $\chi_{10}=244.75_{10}$ σε δυαδική μορφή αντιμετωπίζουμε ξεχωριστά το ακέραιο (244) από το κλασματικό (0.75) μέρος. Για τη μετατροπή του ακεραίου μέρους από δεκαδική σε δυαδική μορφή, εκτελούμε τη διαδικασία των διαδοχικών διαιρέσεων, ως εξής:



όπου διαιρούμε το πηλίκο κάθε διαίρεσης με τη βάση στην οποία θέλουμε να μετατρέψουμε την αρχική αναπαράσταση, δηλ. με το δύο, μέχρις ότου το πηλίκο να μηδενιστεί. Τότε, το τελικό αποτέλεσμα σχηματίζεται από τα υπόλοιπα των διαιρέσεων, με το περισσότερο σημαντικό ψηφίο να είναι το υπόλοιπο το οποίο προέκυψε τελευταίο. Άρα η δυαδική έκφραση του ακεραίου μέρους είναι **11110100_2**, το οποίο προκύπτει εκφρασμένο με 8 δυαδικά ψηφία. Για το κλασματικό μέρος, δηλ. το 0.75, εφαρμόζουμε τη μέθοδο των διαδοχικών πολλαπλασιασμών, ως εξής:

$$0.75\times 2 = 1.5 = 1 + 0.5$$
ΠΣΨ κλασματικού μέρους
$$0.5\times 2 = 1.0 = 1 + 0.0$$
$$0.0\times 2 = 0.0 = 0 + 0$$
$$0.0\times 2 = 0.0 = 0 + 0$$
ΛΣΨ κλασματικού μέρους

όπου πολλαπλασιάζουμε διαδοχικά με τη βάση, δηλ. το 2, το κλασματικό μέρος του αποτελέσματος του προηγούμενου πολλαπλασιασμού. Η διαδικασία τερματίζεται όταν το κλασματικό μέρος απομείνει μηδέν ή όταν εξαντληθούν τα διαθέσιμα για την αναπαράσταση δυαδικά ψηφία. Στην άσκηση αυτή ζητείται η έκφραση του κλασματικού μέρους με τέσσερα ψηφία. Το αποτέλεσμα προκύπτει από τις τιμές των ακεραίων μερών των γινομένων, με λιγότερο σημαντικό το ψηφίο που υπολογίστηκε τελευταίο. Άρα το κλασματικό μέρος εκφράζεται σε δυαδική αναπαράσταση ως

0.1100_2. Συνεπώς η τελική έκφραση του χ_{10} η οποία περιλαμβάνει ακέραιο και κλασματικό μέρος είναι **χ_2=11110100.1100_2**

Για τη μετατροπή στο δεκαεξαδικό, αντιστοιχίζουμε ένα δεκαεξαδικό ψηφίο σε κάθε τέσσερα δυαδικά. Έτσι:

$$\underbrace{1111}_{F}\underbrace{0100}_{4}.\underbrace{1100}_{C}$$

άρα η δεκαεξαδική αναπαράσταση είναι **χ_{16}= $F4.C_{16}$**

Ομοίως για το 35.267_{10}, προκύπτει ότι η δυαδική έκφραση είναι **ψ_2=00100011.0100_2**. Επειδή ζητείται η έκφραση με *οκτώ* δυαδικά ψηφία ακεραίων ενώ αρκούν έξι, έχουν προστεθεί δύο μηδενικά στις δύο περισσότερο σημαντικές θέσεις. Η μετατροπή στο δεκαεξαδικό γίνεται ως εξής:

$$\underbrace{0010}_{2}\underbrace{0011}_{3}.\underbrace{0100}_{4},$$

δηλ. η δεκαεξαδική έκφραση είναι **ψ_{16}=23.4_{16}**

Β. Η έκφραση χ_2 μετατρέπεται σε δεκαδική πολλαπλασιάζοντας κάθε δυαδικό ψηφίο με το αντίστοιχο βάρος εκφρασμένο σε δεκαδική μορφή:

$$1\times2^7+1\times2^6+1\times2^5+1\times2^4+0\times2^3+1\times2^2+0\times2^1+0\times2^0+1\times2^{-1}+1\times2^{-2}+0\times2^{-3}+0\times2^{-4}=244.75_{10}.$$

Αντίστοιχα, η δεκαεξαδική έκφραση χ_{16} μετατρέπεται σε δεκαδική ως ακολούθως:

$$F_{16}\times16^1+4_{16}\times16^0+C_{16}\times16^{-1}=15\times16^1+4\times16^0+12\times16^{-1}=244.75_{10}.$$

Παρατηρούμε ότι και στις δύο περιπτώσεις προκύπτει η αρχική έκφραση χ_{10}, ακριβώς. Αυτό συμβαίνει γιατί τα τέσσερα δυαδικά ψηφία είναι αρκετά για την ακριβή αναπαράσταση του συγκεκριμένου κλασματικού μέρους, μιας και το υπόλοιπο της διαδικασίας των διαδοχικών πολλαπλασιασμών για τη μετατροπή σε δυαδικό, είναι μηδέν. Επίσης τα οκτώ δυαδικά ψηφία αρκούν για την αναπαράσταση του ακεραίου μέρους, μιας και το 244 είναι μικρότερο από το 255(=2^8-1), ποσότητα η οποία είναι η μέγιστη αναπαραστάσιμη με 8 δυαδικά, όταν δεν έχουμε πρόσημο.

Επαναλαμβάνοντας τη διαδικασία για τα ψ_2 και ψ_{16}, προκύπτει ότι

$$0\times2^7+0\times2^6+1\times2^5+0\times2^4+0\times2^3+0\times2^2+1\times2^1+1\times2^0+0\times2^{-1}+1\times2^{-2}+0\times2^{-3}+0\times2^{-4}=35.25_{10}$$

και

$$2_{16}\times16^1+3_{16}\times16^0+4_{16}\times16^{-1}=2\times16^1+3\times16^0+4\times16^{-1}=35.25_{10}.$$

Παρατηρούμε ότι η ποσότητα που προκύπτει είναι κατά $35.267-35.25=0.017$ μικρότερη από την ψ_{10}. Αυτό οφείλεται στο ότι κατά τη μετατροπή της αναπαράστασης του κλασματικού μέρους σε δυαδική μορφή, με τους διαδοχικούς πολλαπλασιασμούς, έμεινε ένα μη μηδενικό υπόλοιπο.

ΑΣΚΗΣΗ 2

Απαντήστε στις ακόλουθες ερωτήσεις. Δικαιολογήστε τις απαντήσεις σας και αναλύστε τον τρόπο επίλυσης, όπου χρειάζεται.

Δίνονται οι αριθμοί α=375_{10}, β=568_{10}, γ=350_{10}, δ=106_{10}, ε=45.5_{10} και ζ=48.75_{10}.

α. Να γραφούν σε δυαδική και δεκαεξαδική αναπαράσταση. Να δηλωθεί σε κάθε περίπτωση, το περισσότερο και το λιγότερο σημαντικό ψηφίο.

β. Να γραφούν οι αντίθετοί τους στο δυαδικό και στο δεκαεξαδικό σε μορφή συμπληρώματος του δύο και του δεκαέξι αντίστοιχα.

γ. Να γίνουν αναλυτικά οι πράξεις α + β, β + γ, α – γ , α + δ στο δυαδικό και στο δεκαεξαδικό. Οι διαφορές να υπολογιστούν χρησιμοποιώντας συμπληρώματα ως προς τις αντίστοιχες βάσεις. Ομοίως, να υπολογιστεί στο δυαδικό το άθροισμα ε+ζ και η διαφορά ε – ζ.

Λύση:

Ερώτημα (α)

Το πλήθος των απαιτούμενων δυαδικών ψηφίων είναι

$$n = \lfloor \log_2 X \rfloor + 1,$$

άρα απαιτούνται $n = \lfloor \log_2 375 \rfloor + 1 = 9$ δυαδικά ψηφία. Άρα διαιρούμε με τις δυνάμεις του 2: 2^8, 2^7,...,2

$$\begin{array}{rlrlrlrlrlrlrlrl}
375 & \underline{|256} & 119 & \underline{|128} & 119 & \underline{|64} & 55 & \underline{|32} & 23 & \underline{|16} & 7 & \underline{|8} & 7 & \underline{|4} & 3 & \underline{|2} \\
-256 & 1 & 0 & 0 & -64 & 1 & -32 & 1 & -16 & 1 & 0 & 0 & -4 & 1 & -2 & 1 \\
119 & & 119 & & 55 & & 23 & & 7 & & 7 & & 3 & & 1 &
\end{array}$$

Συνεπώς, από τα πηλίκα των ανωτέρω διαδοχικών διαιρέσεων και το υπόλοιπο της τελευταίας, προκύπτει ότι

$$375_{10} = 101110111_2 .$$

Υπάρχουν δύο τρόποι για τη μετατροπή σε δεκαεξαδικό.

Τρόπος 1ος

Η διαδικασία επαναλαμβάνεται για βάση 16:

$$\begin{aligned} n &= \lfloor \log_{16} X \rfloor + 1 \\ &= \lfloor \log_{16} 375 \rfloor + 1 = 3. \end{aligned}$$

Συνεπώς, απαιτούνται τρία δεκαεξαδικά ψηφία, άρα διαιρούμε με τις δυνάμεις του 16, $16^2 = 256$ και 16:

$$\begin{array}{rlrl}
375 & \underline{|256} & 119 & \underline{|16} \\
-256 & 1 & -112 & 7 \\
119 & & 7 &
\end{array}$$

Άρα, το 375 γράφεται $375_2 = 177_{16}$.

Τρόπος 2ος

Επίσης από τη δυαδική αναπαράσταση του 375, προκύπτει ότι

$$375_{10} = 101110111_2 = \underbrace{0001}_{1}\underbrace{0111}_{7}\underbrace{0111}_{7} = 177_{16} .$$

Δυαδική αναπαράσταση: $(\underset{MSB}{1}\ 0111011\ \underset{LSB}{1})_2$.

Δεκαεξαδική: $375_{10} = (\underset{MSB}{1}\ 7\ \underset{LSB}{7})_{16}$

Ομοίως για τους άλλους αριθμούς. Τα αποτελέσματα δίνονται στον Πίνακα Ι.

Αριθμός	Δεκαδική	Δυαδική	Δεκαεξαδική
α	375	101110111	177
β	568	1000111000	238
γ	350	101011110	15E
δ	106	1101010	6A
ε	45.5	101101.1	2D.8
ζ	48.75	110000.11	30.C

Πίνακας Ι: Δυαδική και Δεκαεξαδική Αναπαράσταση.

Ερώτημα (β)

Το συμπλήρωμα ως προς δύο του 375 προκύπτει αντιστρέφοντας τα δυαδικά ψηφία και προσθέτοντας μονάδα στο λιγότερο σημαντικό ψηφίο. Έτσι για το 375 ισχύει ότι:

$$
\begin{array}{ll}
0101110111 \rightarrow 1010001000 & \text{αντιστροφη ψηφιων} \\
\qquad\qquad\qquad\qquad +1 & \\
\qquad\quad \rightarrow 1010001001 & \text{συμπληρωμα του 2}
\end{array}
$$

Ομοίως για την δεκαεξαδική αναπαράσταση προκύπτει ότι:

$$
\begin{array}{l}
0177 \rightarrow (15-0)(15-1)(15-7)(15-7) = (15)(14)88 = FE88 \\
\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad +1 \\
\qquad\qquad\qquad\qquad\qquad\qquad \rightarrow FE89 \quad \text{συμπληρωμα του 16}
\end{array}
$$

Αριθμός	Δεκαδική	Δυαδική Συμπληρώματος	Δεκαεξαδικό Συμπλήρωμα
α	375	1010001001	FE89
β	568	10111001000	FDC8
γ	350	1010100010	FEA2
δ	106	10010110	F96
ε	45.5	1010010.1	FD2.8
ζ	48.75	1001111.01	FCF.4

Πίνακας ΙΙ: Δυαδική και Δεκαεξαδική Αναπαράσταση των Αντιθέτων.

Άλλος τρόπος υπολογισμού του συμπληρώματος ως προς 2 του αριθμού A, είναι με τη χρήση του τύπου $2^n - A$, ως εξής:
Όπως προκύπτει από το ερώτημα (α), το 375 χρειάζεται 9 δυαδικά ψηφία, και η αναπαράσταση του είναι 101110111. Παρατηρούμε ότι επειδή το περισσότερο σημαντικό ψηφίο είναι 1, πρέπει να προστεθεί ένα αρχικό μηδέν για να δηλωθεί ότι ο αριθμός είναι θετικός, δηλαδή να εκφραστεί ως 0101110111. Άρα η δεκαδική αναπαράσταση του συμπληρώματος του 2 του 375 είναι $2^{10} - 375 = 649$, γιατί απαιτούνται 9 + 1 = 10 ψηφία, εξαιτίας του αρχικού 0. Μετατρέποντας το 649 σε δυαδική μορφή, προκύπτει ότι το συμπλήρωμα ως προς 2 του 375 είναι το 1010001001_2 το οποίο συμφωνεί με το αποτέλεσμα του άλλου τρόπου. Ομοίως εφαρμόζεται η διαδικασία για τους άλλους αριθμούς.

Ερώτημα (γ)

Το άθροισμα α + β υπολογίζεται ως εξής:

$$
\begin{array}{rcccccccccc l}
 & & & 1 & 1 & 1 & & & & & & \text{(Κρατούμενα)} \\
 & 0 & 1 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & \\
+ & \underline{1} & \underline{0} & \underline{0} & \underline{0} & \underline{1} & \underline{1} & \underline{1} & \underline{0} & \underline{0} & \underline{0} & \\
 & 1 & 1 & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 1 & \text{(Άθροισμα)}
\end{array}
$$

Στο α προστίθεται ένα αρχικό μηδέν για να έχουν ίσο μήκος λέξης οι δύο αριθμοί.

Το άθροισμα β +γ υπολογίζεται ως εξής

$$
\begin{array}{rcccccccccc}
 & & & & 1 & 1 & 1 & & & & \\
 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\
+ & \underline{0} & \underline{1} & \underline{0} & \underline{1} & \underline{0} & \underline{1} & \underline{1} & \underline{1} & \underline{1} & \underline{0} \\
 & 1 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 1 & 0
\end{array}
$$

Η διαφορά α – γ υπολογίζεται αθροίζοντας το α με το συμπλήρωμα του γ ως προς 2.

$$
\begin{array}{rcccccccccc l}
1 & 1 & 1 & 1 & 1 & & & 1 & 1 & & & \text{(κρατούμενα)} \\
 & 0 & 1 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & \\
+ & \underline{1} & \underline{0} & \underline{1} & \underline{0} & \underline{1} & \underline{0} & \underline{0} & \underline{0} & \underline{1} & \underline{0} & \\
1) & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & \text{(άθροισμα)}
\end{array}
$$

Στο α προστίθεται ένα αρχικό μηδέν για να έχουν ίσο μήκος λέξης οι δύο αριθμοί.
Στην περίπτωση αυτή το αποτέλεσμα είναι 25. Το κρατούμενο από την περισσότερο σημαντική θέση στην περίπτωση αυτή αγνοείται.

Το άθροισμα α + δ υπολογίζεται ως εξής.

```
    1 1 1 1 1 1       (κρατούμενα)
  1 0 1 1 1 0 1 1 1
+ 0 0 1 1 0 1 0 1 0
  1 1 1 1 0 0 0 0 1    (άθροισμα).
```

Στο **δεκαεξαδικό σύστημα**, τα αθροίσματα υπολογίζονται ως εξής:

 1 7 7 + 2 3 8 3 A F	1 2 3 8 + 1 5 E 3 9 6	1 1 0 1 7 7 + F E A 2 1) 0 0 1 9	 1 7 7 + 0 6 A 1 E 1

Με τον ίδιο τρόπο υπολογίζονται και τα αθροίσματα ε + ζ και η διαφορά ε – ζ.
Αρχικά σε δυαδικό:

```
  1       1 1
  1 0 1 1 0 1 . 1 0
+ 1 1 0 0 0 0 . 1 1
1 0 1 1 1 1 0 . 0 1
```

Η διαφορά υπολογίζεται με πρόσθεση σε συμπλήρωμα του 2.

```
        1  1  1  1
  0  1  0  1  1  0  1 . 1  0
+ 1  0  0  1  1  1  1 . 0  1
0)1  1  1  1  1  0  0 . 1  1
```

Σε δεκαεξαδικό, οι πράξεις γίνονται ως εξής:

```
    1
  2 D . 8
+ 3 0 . C
  5 E . 4
```

```
     1
  0  2 D . 8
+ F  C F . 4
 0)F F C . C
```

ΑΣΚΗΣΗ 3

Δίνονται οι αριθμοί α=247_{10}, β=932_{10} και γ=163_{10}.
Α. Να γραφούν οι αριθμοί α,β,γ σε δυαδική αναπαράσταση με 12 ψηφία και σε δεκαεξαδική αναπαράσταση με 3 ψηφία. Η μετατροπή να εξηγηθεί αναλυτικά μόνο για τον αριθμό α.
Β. Να γραφούν οι αντίθετοί τους στο δυαδικό και στο δεκαεξαδικό σε μορφή συμπληρώματος του δύο και του δεκαέξι, διατηρώντας το μήκος λέξης των αριθμών σε κάθε περίπτωση, δηλ. 12 δυαδικά ψηφία και 3 δεκαεξαδικά ψηφία.
Γ. Να γίνουν αναλυτικά οι πράξεις α + β και α – γ στο δυαδικό και στο δεκαεξαδικό σύστημα χρησιμοποιώντας την αναπαράσταση του πρώτου ερωτήματος. Οι διαφορές (α – γ) να υπολογιστούν χρησιμοποιώντας τα συμπληρώματα ως προς τις αντίστοιχες βάσεις από το ερώτημα Β.

Λύση:

Α. Το πλήθος των απαιτούμενων δυαδικών ψηφίων είναι

$$n = \lfloor \log_2 X \rfloor + 1,$$

άρα απαιτούνται $n = \lfloor \log_2 247 \rfloor + 1 = 8$ δυαδικά ψηφία. Άρα διαιρούμε με τις δυνάμεις του 2: $2^7, 2^6, \ldots, 2$.

$$\begin{array}{rl rl rl rl rl rl rl}
247 & \underline{|128} & 119 & \underline{|64} & 55 & \underline{|32} & 23 & \underline{|16} & 7 & \underline{|8} & 7 & \underline{|4} & 3 & \underline{|2} \\
-128 & 1 & -64 & 1 & -32 & 1 & -16 & 1 & 0 & 0 & -4 & 1 & -2 & 1 \\
119 & & 55 & & 23 & & 7 & & 7 & & 3 & & 1 &
\end{array}$$

Συνεπώς, από τα πηλίκα των ανωτέρω διαδοχικών διαιρέσεων και το υπόλοιπο της τελευταίας, προκύπτει ότι

$$247_{10} = 000011110111_2,$$

όπου απλώς προσθέτουμε τέσσερα μηδενικά στο περισσότερο σημαντικό μέρος, για να εκφραστεί ο αριθμός στη ζητούμενη μορφή των 12 δυαδικών ψηφίων.

Υπάρχουν δύο τρόποι για τη μετατροπή στο δεκαεξαδικό σύστημα αναπαράστασης.

Τρόπος 1ος

Η διαδικασία μετατροπής σε βάση 2 επαναλαμβάνεται για βάση 16:

$$\begin{aligned} n &= \lfloor \log_{16} X \rfloor + 1 \\ &= \lfloor \log_{16} 247 \rfloor + 1 = 2. \end{aligned}$$

Συνεπώς, απαιτούνται δύο δεκαεξαδικά ψηφία, άρα διαιρούμε με 16:

$$\begin{array}{rl}
247 & \underline{|16} \\
-240 & 15 \\
7 &
\end{array}$$

Άρα, το 247 γράφεται $247_{10} = F7_{16} = 0F7_{16}$, γιατί $15_{10} = F_{16}$.

Τρόπος 2ος

Επίσης η μετατροπή στο δεκαεξαδικό μπορεί να γίνει μετατρέποντας τα ψηφία της δυαδικής αναπαράστασης ανά τέσσερα σε ένα δεκαεξαδικό ψηφίο. Ξεκινώντας από τα λιγότερο σημαντικά ψηφία προκύπτει ότι:

$$247_{10} = 000011110111_2 = \underbrace{0000}_{0}\underbrace{1111}_{F}\underbrace{0111}_{7} = 0F7_{16}.$$

Ομοίως για τους άλλους αριθμούς. Τα αποτελέσματα δίνονται στον Πίνακα I.

Αριθμός	Αναπαράσταση		
	Δεκαδική	Δυαδική	Δεκαεξαδική
α	247	0000 1111 0111	0F7
β	932	0011 1010 0100	3A4
γ	163	0000 1010 0011	0A3

Πίνακας I: Δυαδική και Δεκαεξαδική Αναπαράσταση.

Β. Το συμπλήρωμα ως προς δύο του 247 προκύπτει αντιστρέφοντας τα δώδεκα δυαδικά ψηφία και προσθέτοντας μονάδα στο λιγότερο σημαντικό ψηφίο. Έτσι για το 247 ισχύει ότι:

$$\begin{array}{rl}
000011110111 \rightarrow 111100001000 & \text{αντιστροφη ψηφιων} \\
+1 & \\
\rightarrow 111100001001 & \text{συμπληρωμα του 2}
\end{array}$$

Με αντίστοιχο τρόπο, για την δεκαεξαδική αναπαράσταση προκύπτει ότι:

$$0F7 \rightarrow (15-0)(15-15)(15-7) = (15)(0)(8) = F08$$
$$+1$$
$$\rightarrow F09 \quad \text{συμπληρωμα του 16}$$

Αριθμός	Δεκαδική	Δυαδική Συμπληρώματος	Δεκαεξαδικό Συμπλήρωμα
α	247	1111 0000 1001	F09
β	932	1100 0101 1100	C5C
γ	163	1111 0101 1101	F5D
Πίνακας ΙΙ: Δυαδική και Δεκαεξαδική Αναπαράσταση των Αντιθέτων.			

Άλλος τρόπος υπολογισμού του συμπληρώματος ως προς 2 του αριθμού A, είναι με τη χρήση του τύπου $2^n - A$, ως εξής:
Όπως προκύπτει από το ερώτημα (α), το 247 αναπαριστάται με δώδεκα δυαδικά ψηφία, και η αναπαράσταση του είναι 000011110111. Άρα η δεκαδική αναπαράσταση του συμπληρώματος του 2 του 247 είναι $2^{12} - 247 = 3849$. Μετατρέποντας το 3849 σε δυαδική μορφή, προκύπτει ότι το συμπλήρωμα ως προς 2 του 247 είναι το 111100001001_2 το οποίο συμφωνεί με το αποτέλεσμα του πρώτου τρόπου υπολογισμού του συμπληρώματος, ανά ψηφίο. Ομοίως εφαρμόζεται η διαδικασία για τους άλλους αριθμούς.

Γ. Το άθροισμα α + β υπολογίζεται ως εξής:

```
   1 1 1 1 1 1   1        (Κρατούμενα)
  0 0 0 0 1 1 1 1 0 1 1 1
+ 0 0 1 1 1 0 1 0 0 1 0 0
  0 1 0 0 1 0 0 1 1 0 1 1  (Άθροισμα)
```

Η διαφορά α – γ υπολογίζεται αθροίζοντας το α με το συμπλήρωμα του γ ως προς 2.

```
1 1 1 1 1 1 1 1 1 1 1    (Κρατούμενα)
  0 0 0 0 1 1 1 1 0 1 1 1
 +1 1 1 1 0 1 0 1 1 1 0 1
1| 0 0 0 0 0 1 0 1 0 1 0 0   (Άθροισμα)
```

Το αποτέλεσμα στο δεκαδικό σύστημα είναι ίσο με 84. Το κρατούμενο από την περισσότερο σημαντική θέση στην περίπτωση αυτή αγνοείται.
Στο **δεκαεξαδικό σύστημα**, οι πράξεις υπολογίζονται ως εξής:

```
  1              1 1 1
  0 F 7            0 F 7
+ 3 A 4          + F 5 D
  4 9 B          1| 0 5 4
```

ΑΣΚΗΣΗ 4

Δίνονται οι αριθμοί α=247_{10}, β=352_{10}, γ=460_{10} και δ=101_{10}.

α. Να γραφούν οι αριθμοί α έως δ σε δυαδική αναπαράσταση με 12 ψηφία και σε δεκαεξαδική αναπαράσταση με 3 ψηφία. Η μετατροπή να εξηγηθεί αναλυτικά για τον αριθμό α και συνοπτικά για τους λοιπούς.

β. Να γραφούν οι αντίθετοί τους στο δυαδικό και στο δεκαεξαδικό σε μορφή συμπληρώματος του δύο και του δεκαέξι, διατηρώντας το μήκος λέξης των αριθμών σε κάθε περίπτωση, δηλ. 12 δυαδικά ψηφία και 3 δεκαεξαδικά ψηφία.

γ. Να γίνουν αναλυτικά οι πράξεις α + β, β + γ, α – γ και α + δ στο δυαδικό και στο δεκαεξαδικό σύστημα χρησιμοποιώντας την αναπαράσταση του πρώτου ερωτήματος. Οι διαφορές να υπολογιστούν χρησιμοποιώντας συμπληρώματα ως προς τις αντίστοιχες βάσεις.

Λύση:

Ερώτημα (α)
Το πλήθος των απαιτούμενων δυαδικών ψηφίων είναι

$$n = \lfloor \log_2 X \rfloor + 1,$$

άρα απαιτούνται $n = \lfloor \log_2 247 \rfloor + 1 = 8$ δυαδικά ψηφία. Άρα διαιρούμε με τις δυνάμεις του 2: $2^7, 2^6, \ldots, 2$

247	128	119	64	55	32	23	16	7	8	7	4	3	2
−128	1	−64	1	−32	1	−16	1	0	0	−4	1	−2	1
119		55		23		7		7		3		1	

Συνεπώς, από τα πηλίκα των ανωτέρω διαδοχικών διαιρέσεων και το υπόλοιπο της τελευταίας, προκύπτει ότι

$$247_{10} = 000011110111_2,$$

όπου απλώς προσθέτουμε τέσσερα μηδενικά στο περισσότερο σημαντικό μέρος, για να εκφραστεί ο αριθμός στη ζητούμενη μορφή των 12 δυαδικών ψηφίων.

Άλλος τρόπος για μετατροπή σε δυαδικό

Διαιρούμε διαδοχικά με τη βάση, δηλ. το 2, μέχρις ότου το πηλίκο να μηδενιστεί. Το τελικό αποτέλεσμα σχηματίζεται από τα υπόλοιπα των διαδοχικών διαιρέσεων, με περισσότερο σημαντικό ψηφίο το τελευταίο υπόλοιπο, δηλαδή όπως δείχνει το βέλος στο παρακάτω σχήμα.

247	2							
1	123	2						
	1	61	2					
		1	30	2				
			0	15	2			
				1	7	2		
					1	3	2	
						1	1	2
							1	0

Άρα, η δυαδική αναπαράσταση του 247 είναι 000011110111_2. (Τοποθετήθηκαν τέσσερα μηδενικά στην αρχή της δυαδικής λέξης, γιατί αυτή ζητείται με δώδεκα ψηφία.)
Υπάρχουν δύο τρόποι για τη μετατροπή στο δεκαεξαδικό σύστημα αναπαράστασης.

Τρόπος 1ος
Η διαδικασία μετατροπής σε βάση 2 επαναλαμβάνεται για βάση 16:

$$n = \lfloor \log_{16} X \rfloor + 1$$
$$= \lfloor \log_{16} 247 \rfloor + 1 = 2.$$

Συνεπώς, απαιτούνται δύο δεκαεξαδικά ψηφία, άρα διαιρούμε με 16:

247	16
−240	15
7	

Άρα, το 247 γράφεται $247_{10} = F7_{16} = 0F7_{16}$, γιατί $15_{10} = F_{16}$.

Τρόπος 2ος
Επίσης από τη δυαδική αναπαράσταση, προκύπτει ότι

$$247_{10} = 000011110111_2 = \underbrace{0000}_{0}\underbrace{1111}_{F}\underbrace{0111}_{7} = 0F7_{16}\ .$$

Ομοίως για τους άλλους αριθμούς. Τα αποτελέσματα δίνονται στον Πίνακα Ι.

Αριθμός	Αναπαράσταση		
	Δεκαδική	Δυαδική	Δεκαεξαδική
α	247	0000 1111 0111	0F7
β	352	0001 0110 0000	160
γ	460	0001 1100 1100	1CC
δ	101	0000 0110 0101	065

Πίνακας Ι: Δυαδική και Δεκαεξαδική Αναπαράσταση.

Ερώτημα (β)
Το συμπλήρωμα ως προς δύο του 247 προκύπτει αντιστρέφοντας τα δώδεκα δυαδικά ψηφία και προσθέτοντας μονάδα στο λιγότερο σημαντικό ψηφίο. Έτσι για το 247 ισχύει ότι:

$$\begin{aligned} 000011110111 \rightarrow\ & 111100001000 \quad \text{αντιστροφη ψηφιων} \\ & \qquad\qquad\ \ +1 \\ \rightarrow\ & 111100001001 \quad \text{συμπληρωμα του 2} \end{aligned}$$

Με αντίστοιχο τρόπο, για την δεκαεξαδική αναπαράσταση προκύπτει ότι:

$$\begin{aligned} 0F7 \rightarrow (15-0)(15-15)(15-7) = (15)(0)(8) = & F08 \\ & \ +1 \\ \rightarrow & F09 \quad \text{συμπληρωμα του 16} \end{aligned}$$

Αριθμός	Δεκαδική	Δυαδική Συμπληρώματος	Δεκαεξαδικό Συμπλήρωμα
α	247	1111 0000 1001	F09
β	352	1110 1001 0001	EA0
γ	460	1110 0011 0100	E34
δ	101	1111 1001 1011	F9B

Πίνακας ΙΙ: Δυαδική και Δεκαεξαδική Αναπαράσταση των Αντιθέτων.

Άλλος τρόπος υπολογισμού του συμπληρώματος ως προς 2 του αριθμού *A*, είναι με τη χρήση του τύπου $2^n - A$, ως εξής:
Όπως προκύπει από το ερώτημα (α), το 247 αναπαριστάται με δώδεκα δυαδικά ψηφία, και η αναπαράσταση του είναι 000011110111. Άρα η δεκαδική αναπαράσταση του συμπληρώματος του 2 του 247 είναι $2^{12} - 247 = 3849$. Μετατρέποντας το 3849 σε δυαδική μορφή, προκύπτει ότι το συμπλήρωμα ως προς 2 του 247 είναι το 111100001001_2 το οποίο συμφωνεί με το αποτέλεσμα του πρώτου τρόπου υπολογισμού του συμπληρώματος, ανά ψηφίο. Ομοίως εφαρμόζεται η διαδικασία για τους άλλους αριθμούς.

Ερώτημα (γ)
Το άθροισμα α + β υπολογίζεται ως εξής:

```
     1 1 1 1                 (Κρατούμενα)
   0 0 0 0 1 1 1 1 0 1 1 1
 + 0 0 0 1 0 1 1 0 0 0 0 0
   0 0 1 0 0 1 0 1 0 1 1 1   (Άθροισμα)
```

Το άθροισμα β + γ υπολογίζεται ως εξής

```
     1 1                     (Κρατούμενα)
  0 0 0 1 0 1 1 0 0 0 0 0
 +0 0 0 1 1 1 0 0 1 1 0 0
  0 0 1 1 0 0 1 0 1 1 0 0    (Άθροισμα)
```

Η διαφορά α – γ υπολογίζεται αθροίζοντας το α με το συμπλήρωμα του γ ως προς 2.

```
     1 1 1 1   1             (Κρατούμενα)
  0 0 0 0 1 1 1 1 0 1 1 1
 +1 1 1 0 0 0 1 1 0 1 0 0
  1 1 1 1 0 0 1 0 1 0 1 1    (Άθροισμα)
```

Το άθροισμα α + δ υπολογίζεται ως εξής.

```
     1 1 1   1 1 1   (κρατούμενα)
  0 0 0 0 1 1 1 1 0 1 1 1
+ 0 0 0 0 0 1 1 0 0 1 0 1
 0 0 0 1 0 1 0 1 1 1 0 0  (άθροισμα)
```

Στο δεκαεξαδικό σύστημα, τα αθροίσματα υπολογίζονται ως εξής:

```
 1        1        1        1
 0 F 7    1 6 0    0 F 7    0 F 7
+1 6 0   +1C C    + E 3 4  +0 6 5
 2 5 7    3 2 C    F 2 B    1 5 C
```

ΑΣΚΗΣΗ 5

Δίνονται οι δεκαεξαδικοί αριθμοί $85FA_{16}$, $453A_{16}$, $FFFF_{16}$, 10_{16}, 100_{16}. Να γραφούν σε δυαδική, οκταδική και δεκαδική μορφή. Να περιγραφεί αναλυτικά ο τρόπος μετατροπής.

Λύση:

α) Μετατροπή σε δυαδική μορφή.

$$\begin{aligned}85FA_{16} &= 2^3 \times \left(2^4\right)^3 + \left(2^2 + 2^0\right)\times\left(2^4\right)^2 + \left(2^3+2^2+2^1+2^0\right)\times 2^4 + 2^3 + 2 \\ &= 2^3 2^{12} + 2^2 2^8 + 2^8 + 2^3 2^4 + 2^2 2^4 + 2^1 2^4 + 2^0 2^4 + 2^3 + 2 \\ &= 2^{15} + 2^{10} + 2^8 + 2^7 + 2^6 + 2^5 + 2^4 + 2^3 + 2 \\ &= 1000010111111010_2.\end{aligned}$$

$$\begin{aligned}453A_{16} &= 2^2 \times \left(2^4\right)^3 + \left(2^2 + 2^0\right)\times\left(2^4\right)^2 + \left(2^1+2^0\right)\times\left(2^4\right) + 2^8 + 2^1 \\ &= 2^2 2^{12} + \left(2^2+2^0\right)2^8 + \left(2^1+2^0\right)2^4 + 2^3 + 2^1 \\ &= 2^{14} + 2^{10} + 2^8 + 2^5 + 2^4 + 2^3 + 2^1 \\ &= 100010100111010_2.\end{aligned}$$

$$\begin{aligned}FFFF_{16} &= \left(2^3+2^2+2^1+2^0\right)\times\left(2^4\right)^3 + \left(2^3+2^2+2^1+2^0\right)\times\left(2^4\right)^2 + \left(2^3+2^2+2^1+2^0\right)\times 2^4 + \left(2^3+2^2+2^1+2^0\right) \\ &= 2^3 2^{12} + 2^2 2^{12} + 2^1 2^{12} + 2^{12} + 2^3 2^8 + 2^2 2^8 + 2^1 2^8 + 2^0 2^8 + 2^3 2^4 + 2^2 2^4 + 2^1 2^4 + 2^0 2^4 + 2^3 + 2^2 + 2^1 + 2^0 \\ &= 2^{15} + 2^{14} + 2^{13} + 2^{12} + 2^{11} + 2^{10} + 2^9 + 2^8 + 2^7 + 2^6 + 2^5 + 2^4 + 2^3 + 2^2 + 2^1 + 2^0 \\ &= 1111111111111111_2.\end{aligned}$$

$$10_{16} = 1\times 2^4 + 0 = 10000_2.$$

$$100_{16} = 1\times\left(2^4\right)^2 = 1\times 2^8 = 100000000_2.$$

β) Μετατροπή σε οκταδικό. Χρησιμοποιούμε τις δυαδικές αναπαραστάσεις:

$$\begin{aligned}85FA_{16} &= 2^{15} + 2^{10} + 2^8 + 2^7 + 2^6 + 2^5 + 2^4 + 2^3 + 2 \\ &= 2^{3\times5} + 2^{1+3\times3} + 2^{2+3\times2} + 2^{1+3\times2} + 2^{0+3\times2} + 2^{2+3\times1} + 2^{1+3\times1} + 2^{0+3\times1} + 2^1 \\ &= \left(2^3\right)^5 + 2^1\times\left(2^3\right)^3 + \left(2^2+2^1+2^0\right)\times\left(2^3\right)^2 + \left(2^2+2^1+2^0\right)\left(2^3\right)^1 + 2^1 \\ &= 8^5 + 2\times 8^3 + 7\times 8^2 + 7\times 8 + 2 \\ &= 102772_8.\end{aligned}$$

$$\begin{aligned}453A_{16} &= 2^{14} + 2^{10} + 2^8 + 2^5 + 2^4 + 2^3 + 2^1 \\ &= 2^{2+3\times4} + 2^{1+3\times3} + 2^{2+2\times3} + 2^{2+3} + 2^{1+3} + 2^{0+3} + 2^1 \\ &= 2^2\left(2^3\right)^4 + 2\left(2^3\right)^3 + 2^2\left(2^3\right)^2 + 2^2 2^3 + 2^1 2^3 + 2^0 2^3 + 2^1 \\ &= 4\times 8^4 + 2\times 8^3 + 4\times 8^2 + 7\times 8 + 2 \\ &= 42472_8.\end{aligned}$$

$$\begin{aligned}FFFF_{16} &= 2^{15} + 2^{14} + 2^{13} + 2^{12} + 2^{11} + 2^{10} + 2^9 + 2^8 + 2^7 + 2^6 + 2^5 + 2^4 + 2^3 + 2^2 + 2^1 + 2^0 \\ &= 2^{3\times5} + 2^{2+3\times4} + 2^{1+3\times4} + 2^{0+3\times4} + 2^{2+3\times3} + 2^{1+3\times3} + 2^{3\times3} + \ldots + 2^2 + 2^1 + 2^0 \\ &= \left(2^3\right)^5 + \left(2^2+2^1+2^0\right)\left(2^3\right)^4 + \left(2^2+2^1+2^0\right)\left(2^3\right)^3 + \left(2^2+2^1+2^0\right)\left(2^3\right)^2 + \left(2^2+2^1+2^0\right)\left(2^3\right)^1 + 2^2 + 2^1 + 2^0 \\ &= 8^5 + 7\times 8^4 + 7\times 8^3 + 7\times 8^2 + 7\times 8 + 7 \\ &= 177777_8\end{aligned}$$

$$10_{16} = 1\times 2^4 + 0 = 2\times 2^3 = 2\times 8 = 20_8.$$

$$100_{16} = 1\times\left(2^4\right)^2 = 1\times 2^8 = 2^{2+2\times3} = 2^2\left(2^3\right)^2 = 4\times 8^2 = 400_8.$$

γ) Μετατροπή σε δεκαδική μορφή.

$$\begin{aligned}85FA_{16} &= 8\times16^3+5\times16^2+F\times16+A\\ &= 32768+1280+240+10\\ &= 34298.\end{aligned}$$

$$\begin{aligned}453A_{16} &= 4\times16^3+5\times16^2+3\times16+A\\ &= 16384+1280+48+10=17722.\end{aligned}$$

$$\begin{aligned}FFFF_{16} &= F\times16^3+F\times16^2+F\times16+F\\ &= 61440+3840+240+15=65535.\end{aligned}$$

$$10_{16}=1\times16+0=16.$$

$$100_{16}=1\times16^2+0\times16+0=256.$$

Άλλος τρόπος

Για μετατροπή σε δυαδικό, αντιστοιχίζουμε δυαδικά ψηφία σε κάθε δεκαεξαδικό.

$$\underbrace{8}_{1000}\quad\underbrace{5}_{0101}\quad\underbrace{F}_{1111}\quad\underbrace{A}_{1010}$$

Άρα το αποτέλεσμα είναι 1000010111111010.

Για μετατροπή σε οκταδικό, ξεκινώντας από τη δυαδική αναπαράσταση, προχωρούμε σε ορισμό τριάδων δυαδικών ψηφίων και αντιστοιχίζουμε σε κάθε τριάδα ένα οκταδικό ψηφίο.

$\underbrace{001}_{1}\underbrace{000}_{0}\underbrace{010}_{2}\underbrace{111}_{7}\underbrace{111}_{7}\underbrace{010}_{2}$. Άρα σε οκταδική αναπαράσταση το αποτέλεσμα είναι 102772_8.

ΑΣΚΗΣΗ 6

Δίνονται οι δεκαεξαδικοί αριθμοί $8D4E_{16}$, $34A2_{16}$, $FFFF_{16}$, 20_{16}, 200_{16}. Να γραφούν σε δυαδική, οκταδική και δεκαδική μορφή. Να περιγραφεί αναλυτικά ο τρόπος μετατροπής.

Λύση:

α) Μετατροπή σε δυαδική μορφή.

$$\begin{aligned}8D4E_{16} &= 2^3\times\left(2^4\right)^3+\left(2^3+2^2+2^0\right)\times\left(2^4\right)^2+\left(2^2\right)\times2^4+2^3+2^2+2^1\\ &= 2^32^{12}+2^32^8+2^22^8+2^02^8+2^22^4+2^3+2^2+2^1\\ &= 2^{15}+2^{11}+2^{10}+2^8+2^6+2^3+2^2+2^1\\ &= 1000110101001110_2.\end{aligned}$$

$$\begin{aligned}34A2_{16} &= \left(2^1+2^0\right)\times\left(2^4\right)^3+\left(2^2\right)\times\left(2^4\right)^2+\left(2^3+2^1\right)\times\left(2^4\right)+2^1\\ &= 2^12^{12}+2^02^{12}+2^22^8+2^32^4+2^12^4+2^1\\ &= 2^{13}+2^{12}+2^{10}+2^7+2^5+2^1\\ &= 0011010010100010_2.\end{aligned}$$

$$\begin{aligned}FFFF_{16} &= \left(2^3+2^2+2^1+2^0\right)\times\left(2^4\right)^3+\left(2^3+2^2+2^1+2^0\right)\times\left(2^4\right)^2+\left(2^3+2^2+2^1+2^0\right)\times2^4+\left(2^3+2^2+2^1+2^0\right)\\ &= 2^32^{12}+2^22^{12}+2^12^{12}+2^{12}+2^32^8+2^22^8+2^12^8+2^02^8+2^32^4+2^22^4+2^12^4+2^02^4+2^3+2^2+2^1+2^0\\ &= 2^{15}+2^{14}+2^{13}+2^{12}+2^{11}+2^{10}+2^9+2^8+2^7+2^6+2^5+2^4+2^3+2^2+2^1+2^0\\ &= 1111111111111111_2.\end{aligned}$$

$$20_{16}=2\times2^4+0=100000_2.$$

$$200_{16}=2\times\left(2^4\right)^2=2\times2^8=2^9=1000000000_2.$$

β) Μετατροπή σε οκταδικό. Χρησιμοποιούμε τις δυαδικές αναπαραστάσεις:

$$
\begin{aligned}
8D4E_{16} &= 2^{15}+2^{11}+2^{10}+2^{8}+2^{6}+2^{3}+2^{2}+2^{1} \\
&= 2^{3\times5}+2^{2+3\times3}+2^{1+3\times3}+2^{2+3\times2}+2^{0+3\times2}+2^{0+3\times1}+2^{2}+2 \\
&= \left(2^3\right)^5+\left(2^2+2^1\right)\times\left(2^3\right)^3+\left(2^2+2^0\right)\times\left(2^3\right)^2+\left(2^0\right)\left(2^3\right)^1+2^2+2^1 \\
&= 8^5+6\times8^3+5\times8^2+1\times8+6 \\
&= 106516_8.
\end{aligned}
$$

$$
\begin{aligned}
34A2_{16} &= 2^{13}+2^{12}+2^{10}+2^{7}+2^{5}+2^{1} \\
&= 2^{1+3\times4}+2^{0+3\times4}+2^{1+3\times3}+2^{1+3\times2}+2^{2+3\times1}+2^{1} \\
&= \left(2^1+2^0\right)\left(2^3\right)^4+2^1\left(2^3\right)^3+2^1\left(2^3\right)^2+2^2 2^3+2^1 \\
&= 3\times8^4+2\times8^3+2\times8^2+4\times8+2 \\
&= 32242_8.
\end{aligned}
$$

$$
\begin{aligned}
FFFF_{16} &= 2^{15}+2^{14}+2^{13}+2^{12}+2^{11}+2^{10}+2^{9}+2^{8}+2^{7}+2^{6}+2^{5}+2^{4}+2^{3}+2^{2}+2^{1}+2^{0} \\
&= 2^{3\times5}+2^{2+3\times4}+2^{1+3\times4}+2^{0+3\times4}+2^{2+3\times3}+2^{1+3\times3}+2^{3\times3}+\ldots+2^{2}+2^{1}+2^{0} \\
&= \left(2^3\right)^5+\left(2^2+2^1+2^0\right)\left(2^3\right)^4+\left(2^2+2^1+2^0\right)\left(2^3\right)^3+\left(2^2+2^1+2^0\right)\left(2^3\right)^2+\left(2^2+2^1+2^0\right)\left(2^3\right)^1+2^2+2^1+2^0 \\
&= 8^5+7\times8^4+7\times8^3+7\times8^2+7\times8+7 \\
&= 177777_8
\end{aligned}
$$

$$20_{16} = 2^1\times2^4+0 = 2^2\times2^3 = 4\times8 = 40_8.$$

$$200_{16} = 2\times\left(2^4\right)^2 = 2\times2^8 = 2^{3\times3} = \left(2^3\right)^3 = 1\times8^3 = 1000_8.$$

γ) Μετατροπή σε δεκαδική μορφή.

$$
\begin{aligned}
8D4E_{16} &= 8\times16^3+D\times16^2+4\times16+E \\
&= 32768+3328+64+14 \\
&= 36174.
\end{aligned}
$$

$$
\begin{aligned}
34A2_{16} &= 3\times16^3+4\times16^2+A\times16+2 \\
&= 12288+1024+160+2 \\
&= 13474.
\end{aligned}
$$

$$
\begin{aligned}
FFFF_{16} &= F\times16^3+F\times16^2+F\times16+F \\
&= 61440+3840+240+15 = 65535.
\end{aligned}
$$

$$20_{16} = 2\times16+0 = 32.$$

$$200_{16} = 2\times16^2+0\times16+0 = 512.$$

Άλλος τρόπος

Για μετατροπή δεκαεξαδικού σε δυαδικό, αντιστοιχίζουμε τέσσερα δυαδικά ψηφία σε κάθε δεκαεξαδικό ψηφίο.

$$\underbrace{8}_{1000}\quad\underbrace{D}_{1101}\quad\underbrace{4}_{0100}\quad\underbrace{E}_{1110}$$

Άρα το αποτέλεσμα είναι 1000110101001110.

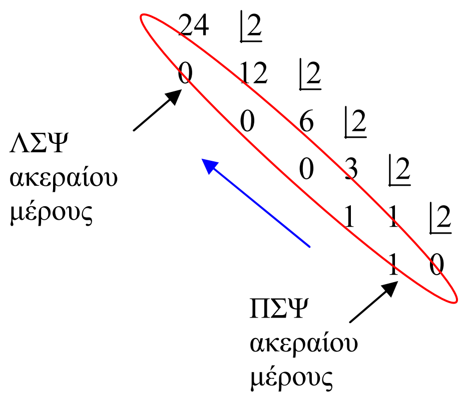
Για μετατροπή δεκαεξαδικού σε οκταδικό, ξεκινώντας από τη δυαδική αναπαράσταση, προχωρούμε σε ορισμό τριάδων δυαδικών ψηφίων και αντιστοιχίζουμε σε κάθε τριάδα ένα οκταδικό ψηφίο.

$\underbrace{001}_{1}\underbrace{000}_{0}\underbrace{110}_{6}\underbrace{101}_{5}\underbrace{001}_{1}\underbrace{110}_{6}$. Άρα σε οκταδική αναπαράσταση το αποτέλεσμα είναι 106516_8.

ΑΣΚΗΣΗ 7

Δίνονται οι αριθμοί: $\chi_{10}=24.625_{10}$ και $\psi_{10}=135.273_{10}$ σε δεκαδική αναπαράσταση.
Α. Να μετατραπούν σε δυαδική αναπαράσταση (χ_2 , ψ_2) με 8 ψηφία για το ακέραιο μέρος και 4 ψηφία για το κλασματικό μέρος καθώς και σε δεκαεξαδική αναπαράσταση (χ_{16} , ψ_{16}) με 2 ψηφία για το ακέραιο μέρος και 1 ψηφίο για το κλασματικό μέρος. Η μετατροπή να εξηγηθεί αναλυτικά μόνο για τον αριθμό χ.
Β. Να μετατραπούν πάλι σε δεκαδική αναπαράσταση οι αριθμοί (χ_2 , ψ_2 και χ_{16} , ψ_{16}) που προέκυψαν από το ερώτημα Α. Σχολιάστε τις διαφορές που παρατηρείτε ανάμεσα στις δεκαδικές αναπαραστάσεις που υπολογίσατε στο ερώτημα Β και στις αρχικές.
(Υπόδειξη: Συμβουλευτείτε το Παράρτημα Α του Β' τόμου 'Αρχιτεκτονική Υπολογιστών Ι').

Λύση
Α. Για να μετατρέψουμε τον αριθμό $\chi_{10}=24.625_{10}$ σε δυαδική μορφή αντιμετωπίζουμε ξεχωριστά το ακέραιο (24) από το κλασματικό (0.625) μέρος. Για τη μετατροπή του ακεραίου μέρους από δεκαδική σε δυαδική μορφή, εκτελούμε τη διαδικασία των διαδοχικών διαιρέσεων, ως εξής:



όπου διαιρούμε το πηλίκο κάθε διαίρεσης με τη βάση στην οποία θέλουμε να μετατρέψουμε την αρχική αναπαράσταση, δηλ. με το δύο, μέχρις ότου το πηλίκο να μηδενιστεί. Τότε, το τελικό αποτέλεσμα σχηματίζεται από τα υπόλοιπα των διαιρέσεων, με το περισσότερο σημαντικό ψηφίο να είναι το υπόλοιπο το οποίο προέκυψε τελευταίο. Άρα η δυαδική έκφραση του ακεραίου μέρους είναι $\mathbf{00011000_2}$. Επειδή ζητείται η έκφραση με *οκτώ* δυαδικά ψηφία ακεραίων ενώ αρκούν πέντε, έχουν προστεθεί τρία μηδενικά στις τρεις περισσότερο σημαντικές θέσεις. Για το κλασματικό μέρος, δηλ. το 0.625, εφαρμόζουμε τη μέθοδο των διαδοχικών πολλαπλασιασμών, ως εξής:

$$0.625\times 2 = 1.25 = 1 + 0.25$$
$$0.25\times 2 = 0.50 = 0 + 0.5$$
$$0.5\times 2 = 1.0 = 1 + 0$$
$$0.0\times 2 = 0.0 = 0 + 0$$

ΛΣΨ κλασματικού μέρους

όπου πολλαπλασιάζουμε διαδοχικά με τη βάση, δηλ. το 2, το κλασματικό μέρος του αποτελέσματος του προηγούμενου πολλαπλασιασμού. Η διαδικασία τερματίζεται όταν το κλασματικό μέρος απομείνει μηδέν ή όταν εξαντληθούν τα διαθέσιμα για την αναπαράσταση δυαδικά ψηφία. Στην άσκηση αυτή ζητείται η έκφραση του κλασματικού μέρους με τέσσερα ψηφία. Το αποτέλεσμα προκύπτει από τις τιμές των ακεραίων μερών των γινομένων, με λιγότερο σημαντικό το ψηφίο που υπολογίστηκε τελευταίο. Άρα το κλασματικό μέρος εκφράζεται σε δυαδική αναπαράσταση ως $\mathbf{0.1010_2}$. Συνεπώς η τελική έκφραση του χ_{10} η οποία περιλαμβάνει ακέραιο και κλασματικό μέρος είναι $\mathbf{\chi_2=00011000.1010_2}$

Για τη μετατροπή στο δεκαεξαδικό, αντιστοιχίζουμε ένα δεκαεξαδικό ψηφίο σε κάθε τέσσερα δυαδικά. Έτσι:

$$\underbrace{0001}_{1}\underbrace{1000}_{8}.\underbrace{1010}_{A}$$

άρα η δεκαεξαδική αναπαράσταση είναι $\mathbf{\chi_{16}= 18.A_{16}}$

Ομοίως για το 135.273_{10}, προκύπτει ότι η δυαδική έκφραση είναι $\boldsymbol{\psi_2}$**=10000111.0100**$_\mathbf{2}$. Η μετατροπή στο δεκαεξαδικό γίνεται ως εξής:

$\underbrace{1000}_{8}\underbrace{0111}_{7}.\underbrace{0100}_{4}$,

δηλ. η δεκαεξαδική έκφραση είναι $\boldsymbol{\psi_{16}}$**=87.4**$_\mathbf{16}$

Β. Η έκφραση χ_2 μετατρέπεται σε δεκαδική πολλαπλασιάζοντας κάθε δυαδικό ψηφίο με το αντίστοιχο βάρος εκφρασμένο σε δεκαδική μορφή:

$0\times2^7+0\times2^6+0\times2^5+1\times2^4+1\times2^3+0\times2^2+0\times2^1+0\times2^0+1\times2^{-1}+0\times2^{-2}+1\times2^{-3}+0\times2^{-4}=24.625_{10}$.

Αντίστοιχα, η δεκαεξαδική έκφραση χ_{16} μετατρέπεται σε δεκαδική ως ακολούθως:

$1_{16}\times16^1+8_{16}\times16^0+A_{16}\times16^{-1}=1\times16^1+8\times16^0+10\times16^{-1}=24.625_{10}$.

Παρατηρούμε ότι και στις δύο περιπτώσεις προκύπτει η αρχική έκφραση χ_{10}, ακριβώς. Αυτό συμβαίνει γιατί τα τέσσερα δυαδικά ψηφία είναι αρκετά για την ακριβή αναπαράσταση του συγκεκριμένου κλασματικού μέρους, μιας και το υπόλοιπο της διαδικασίας των διαδοχικών πολλαπλασιασμών για τη μετατροπή σε δυαδικό, είναι μηδέν.

Επαναλαμβάνοντας τη διαδικασία για τα ψ_2 και ψ_{16}, προκύπτει ότι

$1\times2^7+0\times2^6+0\times2^5+0\times2^4+0\times2^3+1\times2^2+1\times2^1+1\times2^0+0\times2^{-1}+1\times2^{-2}+0\times2^{-3}+0\times2^{-4}=135.25_{10}$

και

$8_{16}\times16^1+7_{16}\times16^0+4_{16}\times16^{-1}=8\times16^1+7\times16^0+4\times16^{-1}=135.25_{10}$.

Παρατηρούμε ότι η ποσότητα που προκύπτει είναι κατά 135.273 −135.25=0.023 μικρότερη από την ψ_{10}. Αυτό οφείλεται στο ότι κατά τη μετατροπή της αναπαράστασης του κλασματικού μέρους σε δυαδική μορφή, με τους διαδοχικούς πολλαπλασιασμούς, έμεινε μη μηδενικό υπόλοιπο. Επίσης τα οκτώ δυαδικά ψηφία αρκούν για την αναπαράσταση του ακεραίου μέρους, μιας και το 135 είναι μικρότερο από το 255, ποσότητα η οποία είναι η μέγιστη αναπαραστάσιμη με 8 δυαδικά, όταν δεν έχουμε πρόσημο.

ΑΣΚΗΣΗ 8

Δίνονται οι αριθμοί $\alpha=147_{10}$, $\beta=3276_{10}$ και $\gamma=307_{10}$.

Α. Να γραφούν οι αριθμοί α,β,γ σε δυαδική αναπαράσταση με 12 ψηφία και σε δεκαεξαδική αναπαράσταση με 3 ψηφία. Η μετατροπή να εξηγηθεί αναλυτικά μόνο για τον αριθμό α.

Β. Να γραφούν οι αντίθετοί τους στο δυαδικό και στο δεκαεξαδικό σε μορφή συμπληρώματος του δύο και του δεκαέξι, διατηρώντας το μήκος λέξης των αριθμών σε κάθε περίπτωση, δηλ. 12 δυαδικά ψηφία και 3 δεκαεξαδικά ψηφία.

Γ. Να γίνουν αναλυτικά οι πράξεις α + β και α − γ στο δυαδικό και στο δεκαεξαδικό σύστημα χρησιμοποιώντας την αναπαράσταση του πρώτου ερωτήματος. Οι διαφορές (α − γ) να υπολογιστούν χρησιμοποιώντας τα συμπληρώματα ως προς τις αντίστοιχες βάσεις από το ερώτημα Β.

Λύση

Α. Το πλήθος των απαιτούμενων δυαδικών ψηφίων είναι

$$n=\lfloor \log_2 X \rfloor+1,$$

άρα απαιτούνται $n=\lfloor \log_2 147 \rfloor+1=8$ δυαδικά ψηφία. Άρα διαιρούμε με $2^{n-1}=2^7=128$:

147	128	19	64	19	32	19	16	3	8	3	4	3	2
−128	1	0	0	0	0	−16	1	0	0	0	0	−2	1
19		19		19		3		3		3		1	

Συνεπώς, από τα πηλίκα των ανωτέρω διαδοχικών διαιρέσεων και το υπόλοιπο της τελευταίας, προκύπτει ότι

$$147_{10}=000010010011_2,$$

όπου απλώς προσθέτουμε τέσσερα μηδενικά στο περισσότερο σημαντικό μέρος, για να εκφραστεί ο αριθμός στη ζητούμενη μορφή των 12 δυαδικών ψηφίων.

Υπάρχουν δύο τρόποι για τη μετατροπή στο δεκαεξαδικό σύστημα αναπαράστασης.

Τρόπος 1ος
Η διαδικασία μετατροπής σε βάση 2 επαναλαμβάνεται για βάση 16:

$$n = \lfloor \log_{16} X \rfloor + 1$$
$$= \lfloor \log_{16} 147 \rfloor + 1 = 2.$$

Συνεπώς, απαιτούνται δύο δεκαεξαδικά ψηφία, άρα διαιρούμε με $16^{n-1} = 16^1 = 16$:

$$\begin{array}{r|l} 147 & 16 \\ \cline{2-2} -144 & 9 \\ 3 & \end{array}$$

Άρα, το 147 γράφεται $147_{10} = 93_{16} = 093_{16}$.

Τρόπος 2ος
Επίσης η μετατροπή στο δεκαεξαδικό μπορεί να γίνει μετατρέποντας τα ψηφία της δυαδικής αναπαράστασης ανά τέσσερα σε ένα δεκαεξαδικό ψηφίο. Ξεκινώντας από τα λιγότερο σημαντικά ψηφία προκύπτει ότι:

$$147_{10} = 000010010011_2 = \underbrace{0000}_{0}\underbrace{1001}_{9}\underbrace{0011}_{3} = 093_{16}.$$

Ομοίως για τους άλλους αριθμούς. Τα αποτελέσματα δίνονται στον Πίνακα Ι.

Αριθμός	Αναπαράσταση		
	Δεκαδική	Δυαδική	Δεκαεξαδική
Α	147	0000 1001 0011	093
Β	3276	1100 1100 1100	CCC
Γ	307	0001 0011 0011	133

Πίνακας Ι: Δυαδική και Δεκαεξαδική Αναπαράσταση.

Β. Το συμπλήρωμα ως προς δύο του 147 προκύπτει αντιστρέφοντας τα δώδεκα δυαδικά ψηφία και προσθέτοντας μονάδα στο λιγότερο σημαντικό ψηφίο. Έτσι για το 147 ισχύει ότι:

$$\begin{array}{ll} 000010010011 \rightarrow 111101101100 & \text{αντιστροφη ψηφιων} \\ \qquad\qquad\qquad\qquad +1 & \\ \qquad\quad \rightarrow 111101101101 & \text{συμπληρωμα του 2} \end{array}$$

Με αντίστοιχο τρόπο, για την δεκαεξαδική αναπαράσταση προκύπτει ότι:

$$\begin{array}{ll} 093 \rightarrow (15-0)(15-9)(15-3) = (15)(6)(12) = F6C & \\ \qquad\qquad\qquad\qquad\qquad\qquad\qquad +1 & \\ \qquad\qquad\qquad\qquad\qquad\quad \rightarrow F6D & \text{συμπληρωμα του 16} \end{array}$$

Αριθμός	Δεκαδική	Δυαδική Συμπληρώματος	Δεκαεξαδικό Συμπλήρωμα
α	147	1111 0110 1101	F6D
β	3276	-*	-*
γ	307	1110 1100 1101	ECD

Πίνακας ΙΙ: Δυαδική και Δεκαεξαδική Αναπαράσταση των Αντιθέτων.

* Ο αριθμός β=3276 και ο αντίθετός του δεν μπορούν να αναπαρασταθούν ως προσημασμένοι σε μορφή συμπληρώματος του δύο με 12 δυαδικά ψηφία καθώς η περιοχή των αριθμών που καλύπτεται με 12 δυαδικά ψηφία στο συμπλήρωμα ως προς 2 είναι από -2^{11}=-2048 ως 2^{11}-1=2047. Θα μπορούσε να αναπαρασταθεί με περισσότερα από 12 δυαδικά ψηφία (π.χ. με 16 δυαδικά ψηφία θα είχαμε β=3276=0000 1100 1100 1100=0CCC και ο αντίθετος του β θα ήταν 1111 0011 0011 0100=F334.

Άλλος τρόπος υπολογισμού του συμπληρώματος ως προς 2 του αριθμού A, είναι με τη χρήση του τύπου $2^n - A$, ως εξής:
Όπως προκύπτει από το ερώτημα (α), το 147 αναπαριστάται με δώδεκα δυαδικά ψηφία, και η αναπαράσταση του είναι 000010010011. Άρα η δεκαδική αναπαράσταση του συμπληρώματος του 2 του 147 είναι $2^{12} - 147 = 3949$. Μετατρέποντας το 3949 σε δυαδική μορφή, προκύπτει ότι το

συμπλήρωμα ως προς 2 του 147 είναι το 111101101101_2 το οποίο συμφωνεί με το αποτέλεσμα του πρώτου τρόπου υπολογισμού του συμπληρώματος, ανά ψηφίο.

Γ. Το άθροισμα α + β υπολογίζεται ως εξής:

```
      1                  (Κρατούμενα)
  0 0 0 0 1 0 0 1 0 0 1 1
+ 1 1 0 0 1 1 0 0 1 1 0 0
  1 1 0 1 0 1 0 1 1 1 1 1    (Άθροισμα)
```

Η διαφορά α – γ υπολογίζεται αθροίζοντας το α με το συμπλήρωμα του γ ως προς 2.

```
    1     1 1 1 1 1     (Κρατούμενα)
  0 0 0 0 1 0 0 1 0 0 1 1
+ 1 1 1 0 1 1 0 0 1 1 0 1
  1 1 1 1 0 1 1 0 0 0 0 0   (Άθροισμα)
```

Το αποτέλεσμα στο δεκαδικό σύστημα είναι ίσο με -160. Εύκολα μπορεί να αποδειχθεί ότι το -160 στο συμπλήρωμα ως προς 2 είναι το 111101100000.
Στο **δεκαεξαδικό σύστημα**, οι πράξεις υπολογίζονται ως εξής:

1 0 9 3 + C C C D 5 F	1 1 0 9 3 + E C D F 6 0

ΑΣΚΗΣΗ 9

Α. Δίνονται οι αριθμοί α=010010110100_2, σε δυαδική αναπαράσταση με 12 ψηφία και β=$0CD_{16}$ σε δεκαεξαδική αναπαράσταση με 3 ψηφία. Να γίνουν οι απαραίτητες μετατροπές ώστε να συμπληρωθεί ο παρακάτω πίνακας.

Αριθμός	Αναπαράσταση		
	Δεκαδική	Δυαδική (12 ψηφία)	Δεκαεξαδική (3 ψηφία)
α		0100 1011 0100	
β			0CD

Πίνακας: Δεκαδική, Δυαδική και Δεκαεξαδική Αναπαράσταση.

Β. Να γίνουν αναλυτικά οι πράξεις α + β και α – β στο δυαδικό και στο δεκαεξαδικό σύστημα. Η διαφορά (α – β) να υπολογιστεί χρησιμοποιώντας τα συμπληρώματα του αριθμού β ως προς 2 και 16 για το δυαδικό και το δεκαεξαδικό σύστημα αντίστοιχα. Επαληθεύστε τις πράξεις στο δεκαδικό.
Γ. Δίνεται ο αριθμός: $χ_{10}=124.65_{10}$ σε δεκαδική αναπαράσταση. Να μετατραπεί σε δυαδική αναπαράσταση ($χ_2$) με 8 ψηφία για το ακέραιο μέρος και 4 ψηφία για το κλασματικό μέρος καθώς και σε δεκαεξαδική αναπαράσταση ($χ_{16}$) με 2 ψηφία για το ακέραιο μέρος και 1 ψηφίο για το κλασματικό μέρος. Να μετατραπούν πάλι σε δεκαδική αναπαράσταση οι αριθμοί ($χ_2$ και $χ_{16}$) που προέκυψαν. Παρατηρείτε διαφορές ανάμεσα στις δεκαδικές αναπαραστάσεις που υπολογίσατε και στην αρχική; Σχολιάστε τις περιπτώσεις που θα εμφανιζόταν διαφορές.
(Υπόδειξη: Συμβουλευτείτε το Παράρτημα Α του Β' τόμου 'Αρχιτεκτονική Υπολογιστών Ι').

Λύση
Α. Ο αριθμός α=010010110100_2 μετατρέπεται σε δεκαδική μορφή πολλαπλασιάζοντας κάθε δυαδικό ψηφίο με το αντίστοιχο βάρος εκφρασμένο σε δεκαδική μορφή:

$$0\times2^{11}+1\times2^{10}+0\times2^{9}+0\times2^{8}+1\times2^{7}+0\times2^{6}+1\times2^{5}+1\times2^{4}+0\times2^{3}+1\times2^{2}+0\times2^{1}+0\times2^{0}=1204_{10}.$$

Για τη μετατροπή στο δεκαεξαδικό, αντιστοιχίζουμε ένα δεκαεξαδικό ψηφίο σε κάθε τέσσερα δυαδικά. Έτσι:

$\underbrace{0100}_{4}\underbrace{1011}_{B}\underbrace{0100}_{4}$

άρα η δεκαεξαδική αναπαράσταση είναι $α_{16}= 4B4_{16}$.

Η αντίστροφη μετατροπή δεκαεξαδικού σε δυαδικό γίνεται και πάλι με αντιστοίχιση τεσσάρων δυαδικών ψηφίων σε κάθε δεκαεξαδικό:

$\underbrace{0000}_{0}\underbrace{1100}_{C}\underbrace{1101}_{D}$

Η μετατροπή του δεκαεξαδικού σε δεκαδικό γίνεται πολλαπλασιάζοντας κάθε δεκαεξαδικό ψηφίο με το αντίστοιχο βάρος εκφρασμένο σε δεκαδική μορφή:

$0_{16}\times 16^2 + C_{16}\times 16^1 + D_{16}\times 16^0 = 12\times 16^1 + 13\times 16^0 = 205_{10}$

Με βάση τα παραπάνω συμπληρώνεται ο πίνακας.

Αριθμός	Αναπαράσταση		
	Δεκαδική	Δυαδική (12 ψηφία)	Δεκαεξαδική (3 ψηφία)
α	1204	0100 1011 0100	4B4
β	205	0000 1100 1101	0CD

Πίνακας: Δεκαδική, Δυαδική και Δεκαεξαδική Αναπαράσταση.

Β. Το άθροισμα (α+β) υπολογίζεται ως εξής:

```
      1 1 1 1 1 1           (Κρατούμενα)
  0 1 0 0 1 0 1 1 0 1 0 0
+ 0 0 0 0 1 1 0 0 1 1 0 1
  ───────────────────────
  0 1 0 1 1 0 0 0 0 0 0 1₂  (Άθροισμα)
```

010110000001_2 (Άθροισμα)

```
    1 1      (Κρατούμενα)
    4 B 4
  + 0 C D
  ───────
    5 8 1₁₆  (Άθροισμα)
```

581_{16} (Άθροισμα)

Στο δεκαδικό το άθροισμα γράφεται:

$0\times 2^{11} + 1\times 2^{10} + 0\times 2^9 + 1\times 2^8 + 1\times 2^7 + 0\times 2^6 + 0\times 2^5 + 0\times 2^4 + 0\times 2^3 + 0\times 2^2 + 0\times 2^1 + 1\times 2^0 = 1409_{10}$

ή

$5_{16}\times 16^2 + 8_{16}\times 16^1 + 1_{16}\times 16^0 = 1409_{10}$

οπότε επαληθεύεται η πράξη και στο δεκαδικό:

$1204 + 205 = 1409_{10}$

Το συμπλήρωμα ως προς δύο του β προκύπτει αντιστρέφοντας τα δώδεκα δυαδικά ψηφία και προσθέτοντας μονάδα στο λιγότερο σημαντικό ψηφίο. Έτσι για το 205 ισχύει ότι:

$0000\ 1100\ 1101 \rightarrow 111100110010$ αντιστροφη ψηφιων

$+1$

$\rightarrow 111100110011$ συμπληρωμα του 2

Με αντίστοιχο τρόπο, για την δεκαεξαδική αναπαράσταση προκύπτει ότι:

$0CD \rightarrow (15-0)(15-12)(15-13) = (15)(3)(2) = F32$

$+1$

$\rightarrow F33$ συμπληρωμα του 16

Η διαφορά α – β υπολογίζεται αθροίζοντας το α με το συμπλήρωμα του β ως προς την αντίστοιχη βάση (2 ή 16).

```
  1       1 1           (Κρατούμενα)
```

$$
\begin{array}{r}
0\,1\,0\,0\,1\,0\,1\,1\,0\,1\,0\,0 \\
+\,\underline{1\,1\,1\,1\,0\,0\,1\,1\,0\,0\,1\,1} \\
1)\,0\,0\,1\,1\,1\,1\,1\,0\,0\,1\,1\,1
\end{array}
\quad (\text{Διαφορά α-β})
$$

$$
\begin{array}{r}
4\,B\,4 \\
+\,\underline{F\,3\,3} \\
1)\,3\,E\,7
\end{array}
\quad (\text{Διαφορά α-β})
$$

Στο δεκαδικό η διαφορά γράφεται:

$$0\times 2^{11}+0\times 2^{10}+1\times 2^{9}+1\times 2^{8}+1\times 2^{7}+1\times 2^{6}+1\times 2^{5}+0\times 2^{4}+0\times 2^{3}+1\times 2^{2}+1\times 2^{1}+1\times 2^{0}=999_{10}$$

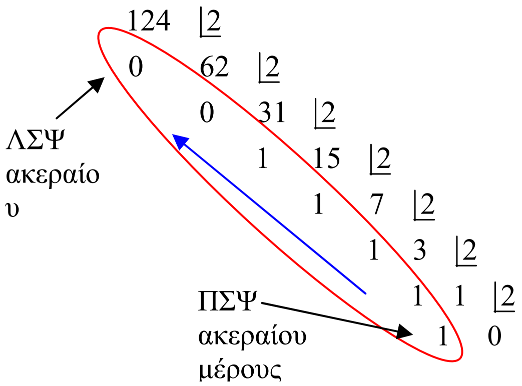
ή

$$3_{16}\times 16^{2}+E_{16}\times 16^{1}+7_{16}\times 16^{0}=3\times 16^{2}+14\times 16^{1}+7\times 16^{0}=999_{10}$$

οπότε επαληθεύεται η πράξη και στο δεκαδικό:

$1204 - 205 = 999_{10}$

Γ. Για να μετατρέψουμε τον αριθμό χ_{10}=124.65$_{10}$ σε δυαδική μορφή αντιμετωπίζουμε ξεχωριστά το ακέραιο (124) από το κλασματικό (0.65) μέρος. Για τη μετατροπή του ακεραίου μέρους από δεκαδική σε δυαδική μορφή, εκτελούμε τη διαδικασία των διαδοχικών διαιρέσεων, ως εξής:



όπου διαιρούμε το πηλίκο κάθε διαίρεσης με τη βάση στην οποία θέλουμε να μετατρέψουμε την αρχική αναπαράσταση, δηλ. με το δύο, μέχρις ότου το πηλίκο να μηδενιστεί. Τότε, το τελικό αποτέλεσμα σχηματίζεται από τα υπόλοιπα των διαιρέσεων, με το περισσότερο σημαντικό ψηφίο να είναι το υπόλοιπο το οποίο προέκυψε τελευταίο. Άρα η δυαδική έκφραση του ακεραίου μέρους είναι **0111 1100$_2$**. Επειδή ζητείται η έκφραση με ***οκτώ*** δυαδικά ψηφία ακεραίων ενώ αρκούν επτά, έχει προστεθεί ένα μηδενικό στην περισσότερο σημαντική θέση. Για το κλασματικό μέρος, δηλ. το 0.65, εφαρμόζουμε τη μέθοδο των διαδοχικών πολλαπλασιασμών, ως εξής:

$0.65\times 2=1.30=1+0.30$ ΠΣΨ κλασματικού μέρους

$0.30\times 2=0.60=0+0.60$

$0.60\times 2=1.20=1+0.20$

$0.20\times 2=0.40=0+0.40$ ΛΣΨ κλασματικού μέρους

όπου πολλαπλασιάζουμε διαδοχικά με τη βάση, δηλ. το 2, το κλασματικό μέρος του αποτελέσματος του προηγούμενου πολλαπλασιασμού. Η διαδικασία τερματίζεται όταν το κλασματικό μέρος απομείνει μηδέν ή όταν εξαντληθούν τα διαθέσιμα για την αναπαράσταση δυαδικά ψηφία. Στην άσκηση αυτή ζητείται η έκφραση του κλασματικού μέρους με τέσσερα ψηφία. Το αποτέλεσμα προκύπτει από τις τιμές των ακεραίων μερών των γινομένων, με λιγότερο σημαντικό το ψηφίο που υπολογίστηκε τελευταίο. Άρα το κλασματικό μέρος εκφράζεται σε δυαδική αναπαράσταση ως

0.1010_2. Συνεπώς η τελική έκφραση του χ_{10} η οποία περιλαμβάνει ακέραιο και κλασματικό μέρος είναι **χ_2=0111 1100.1010$_2$**
Για τη μετατροπή στο δεκαεξαδικό, αντιστοιχίζουμε ένα δεκαεξαδικό ψηφίο σε κάθε τέσσερα δυαδικά. Έτσι:

$$\underbrace{0111}_{7}\underbrace{1100}_{C}.\underbrace{1010}_{A}$$

άρα η δεκαεξαδική αναπαράσταση είναι **χ_{16}= 7C.A$_{16}$**
Στο ίδιο αποτέλεσμα καταλήγουμε αν εφαρμόσουμε για το ακέραιο μέρος τη μέθοδο των διαδοχικών διαιρέσεων με τη βάση (16) μέχρις ότου το πηλίκο να μηδενιστεί και για το κλασματικό μέρος τη μέθοδο των διαδοχικών πολλαπλασιασμών με τη βάση (16).
Για την μετατροπή των χ_2 και χ_{16} σε δεκαδική μορφή πολλαπλασιάζουμε κάθε ψηφίο με το αντίστοιχο βάρος στην αντίστοιχη βάση εκφρασμένο σε δεκαδική μορφή:

$$0\times 2^7+1\times 2^6+1\times 2^5+1\times 2^4+1\times 2^3+1\times 2^2+0\times 2^1+0\times 2^0+1\times 2^{-1}+0\times 2^{-2}+1\times 2^{-3}+0\times 2^{-4}=124.625_{10}$$

ή

$$7_{16}\times 16^1+C_{16}\times 16^0+A_{16}\times 16^{-1}=7\times 16^1+12\times 16^0+10\times 16^{-1}=124.625_{10}$$

Παρατηρούμε ότι η ποσότητα που προκύπτει είναι κατά 124.65 −124.625=0.025 μικρότερη από την χ_{10}. Αυτό οφείλεται στο ότι κατά τη μετατροπή της αναπαράστασης του κλασματικού μέρους σε δυαδική μορφή, με τους διαδοχικούς πολλαπλασιασμούς, έμεινε μη μηδενικό κλασματικό μέρος. Παρατηρούμε επίσης ότι τα οκτώ δυαδικά ψηφία αρκούν για την αναπαράσταση του ακεραίου μέρους, μιας και το 124 είναι μικρότερο από το 255, ποσότητα η οποία είναι η μέγιστη αναπαραστάσιμη με 8 δυαδικά, όταν δεν έχουμε πρόσημο.

ΑΣΚΗΣΗ 10

Α. Δίνεται ο αριθμός α= 1011 0100$_2$, σε δυαδική αναπαράσταση. Μετατρέψτε τον αριθμό στο δεκαδικό σύστημα. Μετακινήστε όλα τα ψηφία του δυαδικού κατά μία θέση **δεξιά** και συμπληρώσετε το περισσότερο σημαντικό ψηφίο με 0 (σημειώστε ότι το λιγότερο σημαντικό ψηφίο που χάνεται είναι 0). Μετατρέψτε πάλι τον αριθμό στο δεκαδικό σύστημα. Τι παρατηρείτε; Σχολιάστε τα αποτελέσματα.
Β. Δίνεται ο αριθμός β= 2F$_{16}$ σε δεκαεξαδική αναπαράσταση. Μετατρέψτε τον αριθμό στο δεκαδικό σύστημα και στο δυαδικό σύστημα. Μετακινήστε όλα τα ψηφία του δυαδικού κατά μία θέση ***αριστερά*** και συμπληρώστε το λιγότερο σημαντικό ψηφίο με 0. Μετατρέψτε τον νέο αριθμό από το δυαδικό στο δεκαδικό σύστημα. Τι παρατηρείτε; Σχολιάστε τα αποτελέσματα.
(Υπόδειξη: Μπορείτε να επαναλάβετε τις μετακινήσεις περισσότερες από μία φορές για να κατανοήσετε το αποτέλεσμα που προκαλούν).
Γ. Συμπληρώστε στον παρακάτω πίνακα τον μέγιστο και τον ελάχιστο ακέραιο αριθμό (στο δεκαδικό και στο αντίστοιχο σύστημα) καθώς και το πλήθος των αριθμών που μπορεί να παρασταθεί με συνολικά 6 δυαδικά ψηφία στο δυαδικό σύστημα (Bin8421), στο δυαδικό σύστημα με προσημασμένους αριθμούς και στο συμπλήρωμα ως προς 2. Εξηγείστε σύντομα την απάντησή σας.

Αριθμός	Αναπαράσταση		
	Δυαδική (Bin8421)	Δυαδική (προσημασμένοι αριθμοί)	Δυαδική (συμπλήρωμα ως προς 2)
Μέγιστος			
Ελάχιστος			
Πλήθος			

Λύση

Α. Ο αριθμός α= 1011 0100$_2$ μετατρέπεται σε δεκαδική μορφή πολλαπλασιάζοντας κάθε δυαδικό ψηφίο με το αντίστοιχο βάρος, εκφρασμένο σε δεκαδική μορφή:

$$1\times 2^7+0\times 2^6+1\times 2^5+1\times 2^4+0\times 2^3+1\times 2^2+0\times 2^1+0\times 2^0=180_{10}$$

Μετακινώντας τα ψηφία του δυαδικού κατά μία θέση δεξιά προκύπτει ο αριθμός: $0101\ 1010_2$ ο οποίος στο δεκαδικό γίνεται:

$$0\times 2^7 + 1\times 2^6 + 0\times 2^5 + 1\times 2^4 + 1\times 2^3 + 0\times 2^2 + 1\times 2^1 + 0\times 2^0 = 90_{10}$$

Παρατηρούμε ότι η μετακίνηση των δυαδικών ψηφίων κατά μία θέση δεξιά ισοδυναμεί με διαίρεση του αριθμού με το 2. Στη διαδικασία αυτή βασίζεται η πράξη της διαίρεσης στην κεντρική μονάδα του υπολογιστή. Να σημειωθεί ότι, αν το λιγότερο σημαντικό ψηφίο που χάνεται είναι 1, τότε παραμένοντας σε ακέραια αναπαράσταση υπάρχει απώλεια πληροφορίας.

Β. Η μετατροπή του δεκαεξαδικού σε δεκαδικό γίνεται πολλαπλασιάζοντας κάθε δεκαεξαδικό ψηφίο με το αντίστοιχο βάρος εκφρασμένο σε δεκαδική μορφή:

$$2_{16}\times 16^1 + F_{16}\times 16^0 = 2\times 16^1 + 15\times 16^0 = 47_{10}$$

Η μετατροπή δεκαεξαδικού σε δυαδικό γίνεται με αντιστοίχιση τεσσάρων δυαδικών ψηφίων σε κάθε δεκαεξαδικό ψηφίο:

$$\overbrace{0010}^{2}\overbrace{1111}^{F}$$

οπότε ο αριθμός είναι β= $0010\ 1111_2$ και, βέβαια, ισχύει:

$$0\times 2^7 + 0\times 2^6 + 1\times 2^5 + 0\times 2^4 + 1\times 2^3 + 1\times 2^2 + 1\times 2^1 + 1\times 2^0 = 47_{10}$$

Μετακινώντας τα ψηφία του δυαδικού κατά μία θέση αριστερά προκύπτει ο αριθμός: $0101\ 1110_2$ ο οποίος στο δεκαδικό γίνεται:

$$0\times 2^7 + 1\times 2^6 + 0\times 2^5 + 1\times 2^4 + 1\times 2^3 + 1\times 2^2 + 1\times 2^1 + 0\times 2^0 = 94_{10}$$

Παρατηρούμε ότι η μετακίνηση των δυαδικών ψηφίων κατά μία θέση αριστερά ισοδυναμεί με πολλαπλασιασμό του αριθμού με το 2. Στη διαδικασία αυτή βασίζεται η πράξη του πολλαπλασιασμού στην κεντρική μονάδα του υπολογιστή.

Γ. Η μεγαλύτερη τιμή που μπορεί να παρασταθεί από έναν δυαδικό αριθμό μήκους 6 ψηφίων είναι: 2^6-1=63 και η μικρότερη 0. Στους προσημασμένους δυαδικούς αριθμούς το πρώτο ψηφίο εκφράζει το πρόσημο και έτσι με τα υπόλοιπα 5 ψηφία η μεγαλύτερη τιμή που μπορεί να παρασταθεί είναι: 2^5-1=31 και προφανώς η μικρότερη είναι το -31. Τέλος στο συμπλήρωμα ως προς 2 η περιοχή αριθμών που καλύπτεται με 6 ψηφία είναι: από -2^{6-1} = -32 ως -2^{6-1}-1=31. Στους προσημασμένους δυαδικούς αριθμούς το μηδέν παριστάνεται είτε ως 000000 ή ως 100000, οπότε το πλήθος των αριθμών που μπορούν να παρασταθούν είναι κατά 1 μικρότερο.

Έτσι προκύπτει ο ακόλουθος πίνακας:

Αριθμός	Αναπαράσταση		
	Δυαδική (Bin8421)	Δυαδική (προσημασμένοι αριθμοί)	Δυαδική (συμπλήρωμα ως προς 2)
Μέγιστος	63 (111111)	+31 (011111)	+31 (011111)
Ελάχιστος	0 (000000)	-31 (111111)	-32 (100000)
Πλήθος	64	63	64

ΑΣΚΗΣΗ 11

Α. Δίνονται οι αριθμοί α= $0000\ 1001\ 0101_2$, σε δυαδική αναπαράσταση με 12 ψηφία και β=$0D4_{16}$ σε δεκαεξαδική αναπαράσταση με 3 ψηφία. Να γίνουν οι απαραίτητες μετατροπές ώστε να συμπληρωθεί ο παρακάτω πίνακας.

Πίνακας: Δεκαδική, Δυαδική και Δεκαεξαδική Αναπαράσταση.			
Αριθμός	Αναπαράσταση		
	Δεκαδική	Δυαδική (12 ψηφία)	Δεκαεξαδική (3 ψηφία)
α		0000 1001 0101	
β			0D4

Β. Να γίνουν αναλυτικά οι πράξεις α + β και α – β στο δυαδικό και στο δεκαεξαδικό σύστημα. Οι διαφορά (α – β) να υπολογιστεί χρησιμοποιώντας τα συμπληρώματα του αριθμού β ως προς 2 και 16 για το δυαδικό και το δεκαεξαδικό σύστημα αντίστοιχα. Επαληθεύστε τις πράξεις στο δεκαδικό.
Γ. Μετακινήστε όλα τα ψηφία του δυαδικού αριθμού α κατά τέσσερις θέσεις **αριστερά** και συμπληρώστε τα λιγότερο σημαντικά ψηφία με 0. Μετατρέψτε τον νέο αριθμό από το δυαδικό στο δεκαδικό σύστημα και το δεκαεξαδικό σύστημα. Τι παρατηρείτε; Σχολιάστε τα αποτελέσματα. (Θεωρούμε αναπαράσταση μη προσημασμένου αριθμού)

Λύση

Α. Ο αριθμός α=0000 1001 0101_2 μετατρέπεται σε δεκαδική μορφή πολλαπλασιάζοντας κάθε δυαδικό ψηφίο με το αντίστοιχο βάρος εκφρασμένο σε δεκαδική μορφή:

$$0\times 2^{11}+0\times 2^{10}+0\times 2^{9}+0\times 2^{8}+1\times 2^{7}+0\times 2^{6}+0\times 2^{5}+1\times 2^{4}+0\times 2^{3}+1\times 2^{2}+0\times 2^{1}+1\times 2^{0}=149_{10}.$$

Για τη μετατροπή στο δεκαεξαδικό, αντιστοιχίζουμε ένα δεκαεξαδικό ψηφίο σε κάθε τέσσερα δυαδικά. Έτσι:

$$\underbrace{0000}_{0}\underbrace{1001}_{9}\underbrace{0101}_{5}$$

άρα η δεκαεξαδική αναπαράσταση είναι $α_{16}$= 095_{16}.
Η αντίστροφη μετατροπή δεκαεξαδικού σε δυαδικό γίνεται και πάλι με αντιστοίχιση τεσσάρων δυαδικών ψηφίων σε κάθε δεκαεξαδικό:

$$\underbrace{0000}_{0}\underbrace{1101}_{D}\underbrace{0100}_{4}$$

Η μετατροπή του δεκαεξαδικού σε δεκαδικό γίνεται πολλαπλασιάζοντας κάθε δεκαεξαδικό ψηφίο με το αντίστοιχο βάρος εκφρασμένο σε δεκαδική μορφή:

$$0_{16}\times 16^{2}+D_{16}\times 16^{1}+4_{16}\times 16^{0}=13\times 16^{1}+4\times 16^{0}=212_{10}$$

Με βάση τα παραπάνω συμπληρώνεται ο πίνακας.

Πίνακας: Δεκαδική, Δυαδική και Δεκαεξαδική Αναπαράσταση.			
Αριθμός	Αναπαράσταση		
	Δεκαδική	Δυαδική (12 ψηφία)	Δεκαεξαδική (3 ψηφία)
α	149	0000 1001 0101	095
β	212	0000 1101 0100	0D4

Β. Το άθροισμα (α+β) υπολογίζεται ως εξής:

```
      1     1   1          (Κρατούμενα)
  0 0 0 0 1 0 0 1 0 1 0 1
+ 0 0 0 0 1 1 0 1 0 1 0 0
  0 0 0 1 0 1 1 0 1 0 0 1₂  (Άθροισμα)
```

```
    1          (Κρατούμενα)
    0 9 5
+   0 D 4
    1 6 9₁₆    (Άθροισμα)
```

Στο δεκαδικό το άθροισμα γράφεται:

$$0\times 2^{11}+0\times 2^{10}+0\times 2^{9}+1\times 2^{8}+0\times 2^{7}+1\times 2^{6}+1\times 2^{5}+0\times 2^{4}+1\times 2^{3}+0\times 2^{2}+0\times 2^{1}+1\times 2^{0}=361_{10}$$

ή

$$1_{16}\times 16^{2}+6_{16}\times 16^{1}+9_{16}\times 16^{0}=361_{10}$$

οπότε επαληθεύεται η πράξη και στο δεκαδικό:

$149 + 212 = 361_{10}$

Η αναπαράσταση του –β σε κώδικα συμπληρώματος ως προς δύο προκύπτει αντιστρέφοντας τα δώδεκα δυαδικά ψηφία και προσθέτοντας μονάδα στο λιγότερο σημαντικό ψηφίο. Έτσι για το –β ισχύει ότι:

$$
\begin{array}{ll}
0000\ 1101\ 0100 \rightarrow 111100101011 & \text{αντιστροφη ψηφιων} \\
\qquad\qquad\qquad\qquad\qquad +1 & \\
\qquad\qquad\quad \rightarrow 111100101100 & \text{συμπληρωμα του 2}
\end{array}
$$

Με αντίστοιχο τρόπο, για την αναπαράσταση σε συμπλήρωμα ως προς 16 προκύπτει ότι:

$$
\begin{array}{ll}
0D4 \rightarrow (15-0)(15-13)(15-4) = (15)(2)(11) = F2B & \\
\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad +1 & \\
\qquad\qquad\qquad\qquad\qquad\qquad\quad \rightarrow F2C & \text{συμπληρωμα του 16}
\end{array}
$$

Η διαφορά α – β υπολογίζεται αθροίζοντας το α με το –β σε κώδικα συμπληρώματος ως προς 2 ή 16, αντίστοιχα.

$$
\begin{array}{rl}
1\,1\,1\,1\qquad\qquad & \text{(Κρατούμενα)} \\
0\,0\,0\,0\,1\,0\,0\,1\,0\,1\,0\,1 & \\
+\ \underline{1\,1\,1\,1\,0\,0\,1\,0\,1\,1\,0\,0} & \\
1\,1\,1\,1\,1\,1\,0\,0\,0\,0\,0\,1 & \text{(Διαφορά α-β)}
\end{array}
$$

$$
\begin{array}{rl}
0\ 9\ 5 & \\
+\ \underline{F\ 2\ C} & \\
F\ C\ 1 & \text{(Διαφορά α-β)}
\end{array}
$$

Εφόσον το πρώτο ψηφίο στο δυαδικό είναι 1 (και στο δεκαεξαδικό F), το αποτέλεσμα είναι αρνητικός αριθμός . Αντιστρέφοντας όλα τα ψηφία της δυαδικής αναπαράστασης (αφαιρώντας όλα τα ψηφία από το FFF στη δεκαεξαδική αναπαράσταση) και προσθέτοντας 1 βρίσκουμε ότι η αναπαράσταση του μέτρου του αριθμού είναι 0000 0011 1111 (03F) = 63_{10}. Άρα το αποτέλεσμα είναι $–63_{10}$, οπότε επαληθεύεται η πράξη και στο δεκαδικό: $149 – 212 = –63_{10}$.

Γ. Μετακινώντας τα ψηφία του δυαδικού κατά 4 θέσεις αριστερά προκύπτει ο αριθμός: 1001 0101 0000_2 ο οποίος στο δεκαδικό (αναπαράσταση μη προσημασμένου αριθμού) γίνεται:

$$1\times 2^{11} + 0\times 2^{10} + 0\times 2^{9} + 1\times 2^{8} + 0\times 2^{7} + 1\times 2^{6} + 0\times 2^{5} + 1\times 2^{4} + 0\times 2^{3} + 0\times 2^{2} + 0\times 2^{1} + 0\times 2^{0} = 2384_{10} (= 149\times 16)$$

Παρατηρούμε ότι η μετακίνηση των δυαδικών ψηφίων κατά 4 θέσεις αριστερά ισοδυναμεί με πολλαπλασιασμό του αριθμού με το 2^4=16.

Για τη μετατροπή στο δεκαεξαδικό, αντιστοιχίζουμε ένα δεκαεξαδικό ψηφίο σε κάθε τέσσερα δυαδικά. Έτσι:

$$\underbrace{1001}_{9}\underbrace{0101}_{5}\underbrace{0000}_{0}$$

άρα η δεκαεξαδική αναπαράσταση είναι $α_{16}= 950_{16}$. Παρατηρούμε ότι η μετακίνηση των δυαδικών ψηφίων κατά 4 θέσεις αριστερά ισοδυναμεί με μετακίνηση των δεκαεξαδικών ψηφίων κατά μία θέση αριστερά.

Στη διαδικασία αυτή βασίζεται γενικά η πράξη του πολλαπλασιασμού στην κεντρική μονάδα του υπολογιστή. Να σημειωθεί ότι, αν κάποιο από τα περισσότερο σημαντικά ψηφία που χάνονται είναι 1, τότε παραμένοντας σε αναπαράσταση 12 ψηφίων, υπάρχει απώλεια πληροφορίας και δεν θα ισχύει ο πολλαπλασιασμός του αριθμού με το 16 (λάθος αποτέλεσμα).

ΙΙ. Κώδικες Αναπαράστασης Δεδομένων

ΑΣΚΗΣΗ 12

A. Να γραφούν σε κώδικα BCD οι δεκαδικοί αριθμοί 61, 25, 308 και 479 και να υπολογιστεί η τιμή του ψηφίου άρτιας ισοτιμίας για κάθε έναν αριθμό.
B. Αν οι κώδικες BCD των δεκαδικών αριθμών 61 και 25 θεωρηθούν δύο πηγαίες λέξεις να κατασκευαστεί για την καθεμία ο κώδικας Hamming τεσσάρων ψηφίων περιττής ισοτιμίας.

Λύση:

A. Ο κώδικας BCD κωδικοποιεί κάθε ψηφίο ενός αριθμού εκφρασμένου σε δεκαδική αναπαράσταση, με μία τετράδα δυαδικών ψηφίων, που είναι η αναπαράσταση του δεκαδικού ψηφίου σε δυαδική μορφή. Έτσι, εφαρμόζοντας τον ορισμό, προκύπτει ότι:

6	1	2	5	3	0	8	4	7	9
↓	↓	↓	↓	↓	↓	↓	↓	↓	↓
0110	0001	0010	0101	0011	0000	1000	0100	0111	1001

Άρα στον κώδικα BCD ο αριθμός 61_{10} αντιστοιχίζεται στη λέξη 01100001. Η λέξη 01100001 έχει τρία ψηφία με τιμή 1, άρα ο αριθμός μονάδων είναι περιττός. Συνεπώς, το ψηφίο άρτιας ισοτιμίας είναι 1. Ομοίως υπολογίζονται τα ψηφία άρτιας ισοτιμίας για τους υπόλοιπους αριθμούς. Τα αποτελέσματα φαίνονται στον παρακάτω πίνακα:

Δεκαδικός	BCD	Ψηφίο άρτιας ισοτιμίας
61	0110 0001	1
25	0010 0101	1
308	0011 0000 1000	1
479	0100 0111 1001	0

B. Έχουμε κωδικοποίηση τεσσάρων ψηφίων περιττής ισοτιμίας. Η πηγαία λέξη για τον αριθμό 61 είναι $m_0m_1m_2m_3m_4m_5m_6m_7 = 01100001_2$. Οι τιμές των δυαδικών ψηφίων $m_i, 0 \le i \le 7$, απεικονίζονται στον πίνακα

m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7
0	1	1	0	0	0	0	1

Τα ψηφία περιττής ισοτιμίας υπολογίζονται ως εξής:

$c_3 = 1: \quad p_3m_4m_5m_6m_7 = p_3 0001 \Rightarrow p_3 = 0$

$c_2 = 1: \quad p_2m_1m_2m_3m_7 = p_2 1101 \Rightarrow p_2 = 0$

$c_1 = 1: \quad p_1m_0m_2m_3m_5m_6 = p_1 01000 \Rightarrow p_1 = 0$

$c_0 = 1: \quad p_0m_0m_1m_3m_4m_6 = p_0 01000 \Rightarrow p_0 = 0.$

Άρα, αντικαθιστώντας τις τιμές των ψηφίων ισοτιμίας στη λέξη

$p_0p_1m_0p_2m_1m_2m_3p_3m_4m_5m_6m_7 =$

$p_0p_1 0 p_2 110 p_3 0001 = 000011000001.$

Ομοίως για την περίπτωση του 25, σχηματίζουμε τον πίνακα

m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7
0	0	1	0	0	1	0	1

Τα ψηφία περιττής ισοτιμίας υπολογίζονται ως εξής:

$c_3 = 1: \quad p_3 m_4 m_5 m_6 m_7 = p_3 0101 \Rightarrow p_3 = 1$

$c_2 = 1: \quad p_2 m_1 m_2 m_3 m_7 = p_2 0101 \Rightarrow p_2 = 1$

$c_1 = 1: \quad p_1 m_0 m_2 m_3 m_5 m_6 = p_1 01010 \Rightarrow p_1 = 1$

$c_0 = 1: \quad p_0 m_0 m_1 m_3 m_4 m_6 = p_0 00000 \Rightarrow p_0 = 1.$

Άρα η κωδικοποίηση κατά Hamming θα είναι:

$p_0 p_1 m_0 p_2 m_1 m_2 m_3 p_3 m_4 m_5 m_6 m_7 =$

$p_0 p_1 0 p_2 010 p_3 0101 = 110101010101.$

ΑΣΚΗΣΗ 13

α. Από τι είδους λάθη προστατεύει ο κώδικας Hamming, και με ποιό τρόπο;
β. Δίνεται η δυαδική λέξη 10010111_2. Να κωδικοποιηθεί σε κώδικα Hamming.
γ. Είναι δυνατόν η λέξη 101100110111_2 να είναι κωδικοποιημένη σε κώδικα Hamming των τεσσάρων ψηφίων αρτιας ισοτιμίας; Αν ναι, ποια είναι η λέξη πληροφορίας;

Λύση:

Ερώτημα (α)
Ο κώδικας Hamming μπορεί να χρησιμοποιηθεί για να διορθώσει λάθη μετατροπής ενός ψηφίου ή να χρησιμοποιηθεί για να ανιχνεύσει μέχρι και λάθη μετατροπής δύο ψηφίων.

Ερώτημα (β)
Έστω κωδικοποίηση τεσσάρων ψηφίων άρτιας ισοτιμίας. Ο πηγαίος κώδικας είναι $m_0 m_1 m_2 m_3 m_4 m_5 m_6 m_7 = 10010111_2$.

Τα ψηφία άρτιας ισοτιμίας υπολογίζονται ως εξής:

$c_3 = 1: \quad p_3 m_4 m_5 m_6 m_7 = p_3 0111 \Rightarrow p_3 = 1$

$c_2 = 1: \quad p_2 m_1 m_2 m_3 m_7 = p_2 0011 \Rightarrow p_2 = 0$

$c_1 = 1: \quad p_1 m_0 m_2 m_3 m_5 m_6 = p_1 10111 \Rightarrow p_1 = 0$

$c_0 = 1: \quad p_0 m_0 m_1 m_3 m_4 m_6 = p_0 10101 \Rightarrow p_0 = 1$

Άρα, αντικαθιστώντας τις τιμές των ψηφίων ισοτιμίας στη λέξη

$p_0 p_1 m_0 p_2 m_1 m_2 m_3 p_3 m_4 m_5 m_6 m_7 =$

$p_0 p_1 1 p_2 001 p_3 0111 =$

101000110111

Ερώτημα (γ)
Η λέξη έχει τέσσερα ψηφία ισοτιμίας, και συνολικό μήκος 12, άρα η πηγαία λέξη θα είναι μήκους 8 δυαδικών ψηφίων. Η οργάνωση της θα είναι όπως στο προηγούμενο ερώτημα. Έτσι, αντιστοιχίζοντας τις θέσεις της προηγούμενης αναπαράσταση με τη δοθείσα λέξη και θεωρώντας άρτια ισοτιμία, προκύπτουν οι τιμές των δυαδικών ψηφίων:

$p_0 p_1 m_0 p_2 m_1 m_2 m_3 p_3 m_4 m_5 m_6 m_7 = 101100110111$

$p_3: \quad p_3 m_4 m_5 m_6 m_7 = 10111 \Rightarrow c_3 = 0, \quad \sigma\omega\sigma\tau o$

$p_2: \quad p_2 m_1 m_2 m_3 m_7 = 10011 \Rightarrow c_2 = 1, \quad \lambda\alpha\theta o\varsigma$

$p_1: \quad p_1 m_0 m_2 m_3 m_5 m_6 = 010111 \Rightarrow c_1 = 0$

$p_0: \quad p_0 m_0 m_1 m_3 m_4 m_6 = 110101 \Rightarrow c_0 = 0$

Άρα υπάρχει ένα λάθος, το οποίο καταδεικνύεται από τον αριθμό θέσης 0100 δηλαδή 4. Συνεπώς, το σφάλμα είναι στο ψηφίο p_2 της αρχικής λέξης. Η λέξη πληροφορίας είναι 10010111.

ΑΣΚΗΣΗ 14

α. Δίνεται η δυαδική λέξη 11010111_2. Να κωδικοποιηθεί σε κώδικα Hamming τεσσάρων ψηφίων περιττής ισοτιμίας.
β. Είναι δυνατόν η λέξη 111010100110_2 να είναι κωδικοποιημένη σε κώδικα Hamming των τεσσάρων ψηφίων περιττής ισοτιμίας; Αν ναι, ποια είναι η λέξη πληροφορίας;

Λύση:

Ερώτημα (α)
Έστω κωδικοποίηση τεσσάρων ψηφίων περιττής ισοτιμίας. Ο πηγαίος κώδικας είναι $m_0m_1m_2m_3m_4m_5m_6m_7 = 11010111_2$.

Τα ψηφία περιττής ισοτιμίας υπολογίζονται ως εξής:

$c_3 = 1: \quad p_3m_4m_5m_6m_7 = p_3 0111 \Rightarrow p_3 = 0$

$c_2 = 1: \quad p_2m_1m_2m_3m_7 = p_2 1011 \Rightarrow p_2 = 0$

$c_1 = 1: \quad p_1m_0m_2m_3m_5m_6 = p_1 10111 \Rightarrow p_1 = 1$

$c_0 = 1: \quad p_0m_0m_1m_3m_4m_6 = p_0 11101 \Rightarrow p_0 = 1$

Άρα, αντικαθιστώντας τις τιμές των ψηφίων ισοτιμίας στη λέξη

$p_0p_1m_0p_2m_1m_2m_3p_3m_4m_5m_6m_7 =$

$p_0p_1 1 p_2 101 p_3 0111 = 111010100111$

Ερώτημα (β)
Η λέξη έχει τέσσερα ψηφία περιττής ισοτιμίας, και συνολικό μήκος 12, άρα η πηγαία λέξη θα είναι μήκους 8 δυαδικών ψηφίων. Η οργάνωση της θα είναι όπως στο προηγούμενο ερώτημα. Έτσι, αντιστοιχίζοντας τις θέσεις της προηγούμενης αναπαράστασης με τη δοθείσα λέξη προκύπτουν οι τιμές των δυαδικών ψηφίων:

$p_0p_1m_0p_2m_1m_2m_3p_3m_4m_5m_6m_7 = 111010100110$

$p_3: \quad p_3m_4m_5m_6m_7 = 00110 \Rightarrow c_3 = 1, \quad$ λαθος

$p_2: \quad p_2m_1m_2m_3m_7 = 01010 \Rightarrow c_2 = 1, \quad$ λαθος

$p_1: \quad p_1m_0m_2m_3m_5m_6 = 110111 \Rightarrow c_1 = 0, \quad$ σωστο

$p_0: \quad p_0m_0m_1m_3m_4m_6 = 111101 \Rightarrow c_0 = 0, \quad$ σωστο

Άρα υπάρχει ένα λάθος, το οποίο καταδεικνύεται από τον αριθμό θέσης $c_3c_2c_1c_0 = 1100_2 = 12_{10}$. Συνεπώς, το σφάλμα είναι στο ψηφίο m_7 της ληφθείσας λέξης, η ορθή τιμή του οποίου είναι '1' και όχι το '0'. Άρα η ορθή λέξη πληροφορίας προκύπτει από τις τιμές των $m_0m_1m_2m_3m_4m_5m_6m_7$ αντιστρέφοντας το m_7 και είναι 11010111.

ΑΣΚΗΣΗ 15

α. Να υπολογίσετε την τιμή ψηφίου άρτιας και περιττής ισοτιμίας για τις ακόλουθες περιπτώσεις: 101111_2, 1010_2, 1010101_2.
β. Σχεδιάστε ένα κύκλωμα που να υπολογίζει το ψηφίο άρτιας ισοτιμίας μίας λέξης δύο δυαδικών ψηφίων.
γ. Σχεδιάστε ένα κύκλωμα που να υπολογίζει το ψηφίο περιττής ισοτιμίας μίας λέξης δύο δυαδικών ψηφίων.

Λύση:

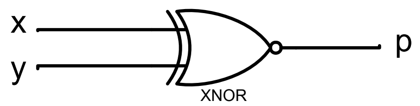
Ερώτημα (α)
Η λέξη 101111 έχει πέντε ψηφία με τιμή 1, άρα ο αριθμός μονάδων είναι περιττός. Συνεπώς, το ψηφίο άρτιας ισοτιμίας είναι 1 και το ψηφίο περιττής ισοτιμίας 0. Ομοίως, για την 1010, το ψηφίο άρτιας ισοτιμίας είναι 0 και το περιττής είναι 1, γιατί η λέξη έχει άρτιο (2) αριθμό ψηφίων με τιμή 1. Τέλος, η 1010101 έχει τέσσερα ψηφία με τιμή 1 και ψηφίο περιττής ισοτιμίας 1, ψηφίο άρτιας ισοτιμίας 0.

Ερώτημα (β)
Ο πίνακας αληθείας είναι ο εξής:

x	y	Πλήθος Ψηφίων τιμής 1	Άρτια ισοτιμία	Περιττή ισοτιμία
0	0	0	0	1
0	1	1	1	0
1	0	1	1	0
1	1	2	0	1

Παρατηρούμε ότι πρόκειται για τον πίνακα αλήθειας της XOR. Άρα το ψηφίο ισοτιμίας μπορεί να υπολογιστεί από μία πύλη XOR.

Ερώτημα (γ)
Αντίστοιχα, παρατηρούμε ότι για την περιττή ισοτιμία, έχουμε τον πίνακα αλήθειας της XNOR. Άρα το ψηφίο ισοτιμίας μπορεί να υπολογιστεί από μία πύλη XNOR, η αναπαράσταση της οποίας δίνεται στο ακόλουθο σχήμα.



ΑΣΚΗΣΗ 16

Α. Δίνεται η δυαδική λέξη 1101_2. Να κωδικοποιηθεί σε κώδικα Hamming τριών ψηφίων άρτιας ισοτιμίας.
Β. Αν στον προορισμό φθάσει η δυαδική λέξη 1000001_2 κωδικοποιημένη σε κώδικα Hamming των τριών ψηφίων άρτιας ισοτιμίας, βρείτε ποια είναι η σωστή δυαδική λέξη πληροφορίας που μεταδόθηκε από την πηγή.
Λύση:

Α. Έχουμε κωδικοποίηση τριών ψηφίων άρτιας ισοτιμίας. Η πηγαία λέξη αποτελείται από τέσσερα δυαδικά ψηφία και είναι $m_0 m_1 m_2 m_3 = 1101_2$. Οι τιμές των δυαδικών ψηφίων $m_i, 0 \le i \le 7$, απεικονίζονται στον πίνακα

$$\begin{array}{cccc} m_0 & m_1 & m_2 & m_3 \\ 1 & 1 & 0 & 1 \end{array}$$

Τα ψηφία άρτιας ισοτιμίας υπολογίζονται ως εξής:

$c_2 = 1: \quad p_2 m_1 m_2 m_3 = p_2 101 \Rightarrow p_2 = 0$

$c_1 = 1: \quad p_1 m_0 m_2 m_3 = p_1 101 \Rightarrow p_1 = 0$

$c_0 = 1: \quad p_0 m_0 m_1 m_3 = p_0 111 \Rightarrow p_0 = 1.$

Άρα, αντικαθιστώντας τις τιμές των ψηφίων ισοτιμίας στη λέξη

$p_0 p_1 m_0 p_2 m_1 m_2 m_3 =$

$p_0 p_1 1 p_2 101 = 1010101.$

Β. Αντιστοιχίζοντας ένα προς ένα τα αντίστοιχα δυαδικά ψηφία στην ισότητα $p_0 p_1 m_0 p_2 m_1 m_2 m_3 = 1000001$, προκύπτει ότι $p_0 = 1$, $p_1 = 0$ και $p_2 = 0$. Επίσης

m_0	m_1	m_2	m_3
0	0	0	1

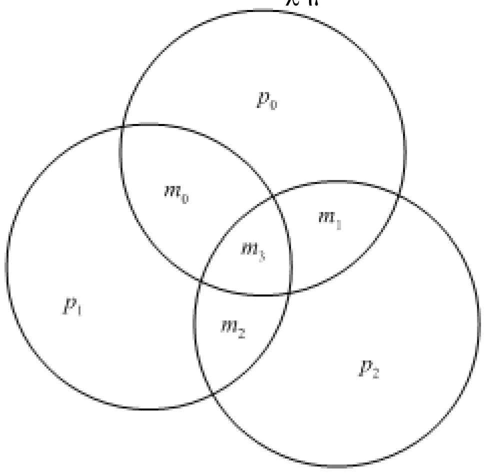
Στη συνέχεια ελέγχουμε την ισοτιμία κάθε ομάδας η οποία αποτελείται από ένα ψηφίο ισοτιμίας και τα αντίστοιχα ψηφία δεδομένων που προστατεύονται από αυτό, ως εξής:

$p_2 m_1 m_2 m_3 = 0001 \Rightarrow$ λαθος, $\quad c_2 = 1$

$p_1 m_0 m_2 m_3 = 0001 \Rightarrow$ λαθος, $\quad c_1 = 1$

$p_0 m_0 m_1 m_3 = 1001 \Rightarrow$ σωστο, $\quad c_0 = 0$

Όταν ο έλεγχος ισοτιμίας μιάς ομάδας δείξει λάθος, τότε το αντίστοιχο ψηφίο θέσης παίρνει την τιμή 1. Από τον ανωτέρω πίνακα, παρατηρούμε ότι έχει συμβεί λάθος στο ψηφίο που βρίσκεται στη θέση $c_2 c_1 c_0 = 110_2$, δηλ. στο έκτο ψηφίο της λέξης, το οποίο είναι το m_2. <u>Παρατηρήστε ότι το m_2 είναι το μόνο κοινό ψηφίο των δύο ομάδων στις οποίες έχει συμβεί λάθος.</u> Συνεπώς η ορθή τιμή του m_2 είναι η αντίθετη αυτής που λάβαμε, άρα $m_2 = 1$. Συνεπώς η λέξη που μεταδόθηκε είναι η 1000011. Η οργάνωση του κώδικα αυτού φαίνεται στο ακόλουθο σχήμα.



Παρατηρήστε ότι το κάθε ένα από τα ψηφία m_0, m_1 και m_2 προστατεύεται από δύο ψηφία ισοτιμίας.Το m_3 προστατεύεται και από τα τρία ψηφία ισοτιμίας. Αν συμβεί σφάλμα σε ψηφίο ισοτιμίας επηρρεάζεται μόνο το ίδιο. Συνεπώς, ένα σφάλμα σε ένα δυαδικό ψηφίο της κωδικοποιημένης λέξης εντοπίζεται γιατί προκαλεί μια αντίστοιχη και μοναδική παραβίαση των ελέγχων ισοτιμίας.

ΑΣΚΗΣΗ 17

Έστω λέξη Hamming άρτιας ισοτιμίας οργανωμένη ως $p_0 p_1 m_0 p_2 m_1 m_2 m_3$, όπου p είναι τα ψηφία ισοτιμίας και m τα ψηφία δεδομένων. Το p_0 είναι το ψηφίο άρτιας ισοτιμίας για τα m_0, m_1, m_3, το p_1 είναι το ψηφίο άρτιας ισοτιμίας για τα m_0, m_2, m_3 και το p_2 για τα m_1, m_2, m_3. Να σχεδιαστεί κύκλωμα το οποίο να δέχεται ως είσοδο τη λέξη αυτή και να παράγει δύο εξόδους:
α) μια έξοδο E, η οποία να δηλώνει αν υπάρχει λάθος στη λέξη ή όχι και
β) το ψηφίο m_0, διορθωμένο αν χρειάζεται.
Θεωρείστε ότι μπορεί να συμβεί λάθος σε ένα ψηφίο μόνο.

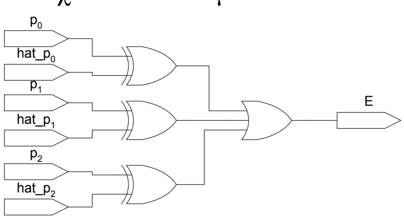
Λύση
Αρχικά υπολογίζουμε τις τιμές των ψηφίων ισοτιμίας με βάση τις τιμές των m_i.

m_0	m_1	m_3	$\hat{p}_0$
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

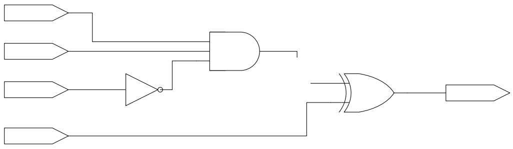
$m_0 \backslash m_1 m_3$	00	01	11	10
0		1		1
1	1		1	

$$\begin{aligned}\hat{p}_0 &= m_0 m_1' m_3' + m_0' m_1' m_3 + m_0 m_1 m_3 + m_0' m_1 m_3' \\ &= m_0 (m_1 \oplus m_3)' + m_0' (m_1 \oplus m_3)' \\ &= m_0 \oplus m_1 \oplus m_3\end{aligned}$$

Ομοίως υπολογίζουμε τις τιμές των $\hat{p}_1$ και $\hat{p}_2$. Στη συνέχεια ελέγχουμε τις τιμές των $\hat{p}$ σε σχέση με τις τιμές των p. Αν $\hat{p}_0 \neq p_0$ τότε έχει συμβεί λάθος. Η διαφορά ελέγχεται με το σήμα $e_0 = p_0 \oplus \hat{p}_0$, το οποίο υπολογίζεται με xor. Ομοίως για τα λοιπά ψηφία ισοτιμίας. Άρα η έξοδος E, η οποία δηλώνει αν υπάρχει λάθος ή όχι θα υπολογίζεται ως
$E = e_1 + e_2 + e_3$
Το σχετικό κύκλωμα θα είναι



Μιας και το m_0 καλύπτεται από το p_0 και το p_1 και ενδιαφέρει η περίπτωση απλού λάθους, το m_0 θα έχει λάθος τιμή αν το p_0 και το p_1 δείχνουν λάθος ενώ το p_2 όχι. Άρα το m_0 είναι λάθος όταν ισχύει ότι $M_0 = e_0 e_1 e_2' = 1$, και ένα κύκλωμα που διορθώνει το m_0 είναι το ακόλουθο:



γιατί αν $M_0 = 0$ τότε το m_0 δεν χρειάζεται διόρθωση. Πράγματι η έξοδος της xor όταν M_0=0 είναι m_0. Όταν M_0=1, τότε η έξοδος της xor είναι m_0', δηλαδή το διορθωμένο ψηφίο.

III. Άλγεβρα Boole

ΑΣΚΗΣΗ 18

Γιατί χρειάζεται η άλγεβρα Boole στην ανάλυση και σχεδίαση ψηφιακών συστημάτων; Ποια η σχέση της άλγεβρας Boole με τη Μαθηματική Λογική;

Λύση:

Η άλγεβρα Boole χρειάζεται στην ανάλυση και σχεδίαση ψηφιακών συστημάτων γιατί αποτελεί ένα μαθηματικό εργαλείο περιγραφής της λογικής συμπεριφοράς ενός ψηφιακού συστήματος μέσω των λογικών συναρτήσεων που παρέχει. Επίσης, με τη χρήση των διαφόρων θεωρημάτων που παρέχει, είναι δυνατόν να προκύψουν από μία αρχική περιγραφή άλλες απλούστερες, οι οποίες αντιστοιχούν σε καλύτερες υλοποιήσεις μιας δεδομένης συμπεριφοράς, όσον αφορά το πλήθος των πυλών ή την καθυστέρηση.

Η άλγεβρα Boole «μαθηματικοποιεί» τον λογισμό των προτάσεων.

ΑΣΚΗΣΗ 19

α. Υπολογίστε τις τιμές των **λογικών** παραστάσεων $ab'c$, $a+b'+c$, $a+b'c$, $ab'+c$, για *α*=1, *b*=1, και *c*=0.

β. Να λυθούν οι λογικές εξισώσεις $ab'c=1$ και $a+b'+c=0$, ως προς τις τιμές των δίτιμων μεταβλητών *a*, *b*, *c*.

Λύση:

Ερώτημα (α)
Αντικαθιστώ τις τιμές στις λογικές παραστάσεις:

$$ab'c=(1\cdot 1')\cdot 0=(1\cdot 0)\cdot 0=0\cdot 0=0$$
$$a+b'+c=1+1'+0=1+0+0=1$$
$$a+b'c=1+1'\cdot 0=1+0=1$$
$$ab'+c=1\cdot 1'+0=1\cdot 0+0=0$$

Ερώτημα (β)
Από τον πίνακα αληθείας της παράστασης $ab'c$ προκύπτει ότι η λύση είναι (*a*,*b’*,*c*)=(1,1,1) ή (*a*,*b*,*c*)=(1,0,1). Διαφορετικά, αν υπάρχει τουλάχιστον ένα μηδέν στην τριάδα (*a*,*b’*,*c*), τότε από τον ορισμό του λογικού AND προκύπτει ότι *ab’c*=0. Άρα δεν είναι δυνατόν κάποιο από τα (*a*,*b’*,*c*) να είναι 0. Άρα η τριάδα (1,0,1) είναι η μοναδική λύση.

Αντίστοιχα για την παράσταση $a+b'+c$, από τον πίνακα αληθείας η λύση είναι (*a*, *b’*,*c*) = (0,0,0) ή (*a, b, c*)=(0,1,0). Διαφορετικά, αν υπάρχει τουλάχιστον ένα 1 στις τιμές των (*a*, *b’*,*c*), τότε από τον ορισμό του λογικού OR, η παράσταση είναι 1.

ΑΣΚΗΣΗ 20

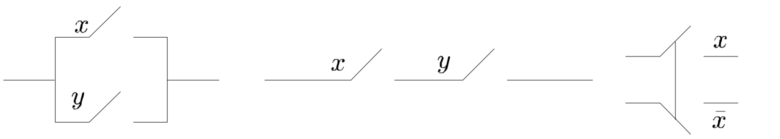
α. Πώς αντιστοιχίζονται οι πράξεις της δίτιμης άλγεβρας σε λογικά κυκλώματα;

β. Υπολογίστε τις τιμές των λογικών παραστάσεων abc, $a+b+c$, $a+bc$, $ab+c$, για a=1, b=1, c=0.
γ. Να λυθούν οι λογικές εξισώσεις $abc=1$ και $a+b+c=0$, ως προς τις τιμές των δίτιμων μεταβλητών a, b, c.
δ. Επίσης λύστε τις $\prod_i a_i = 1$ και $\sum_i a_i = 0$, για τις δίτιμες μεταβλητές a_i, και για $i < N$, έναν οποιονδήποτε ακέραιο.

Λύση:

Ερώτημα (α)
Οι πράξεις και οι μεταβλητές της δίτιμης άλγεβρας αντιστοιχίζονται σε λειτουργίες και καταστάσεις που χαρακτηρίζουν ένα κύκλωμα διακοπτών. Αντίστροφα ένα κύκλωμα διακοπτών μπορεί να περιγραφεί με τη δίτιμη άλγεβρα.

Ένας διακόπτης μπορεί να βρεθεί σε δύο καταστάσεις (ή ανοικτός ή κλειστός) κατά αντιστοιχία με τις μεταβλητές τις δίτιμης άλγεβρας. Επίσης αν κύκλωμα των διακοπτών βρίσκεται σε κατάσταση αγωγής θεωρούμε ότι συμβολίζει κατάσταση εξόδου '1', ενώ αν δεν βρίσκεται σε κατάσταση αγωγής η έξοδος είναι '0'. Με βάση τις θεωρήσεις αυτές, οι πράξεις της δίτιμης άλγεβρας αντιστοιχίζονται σε κυκλώματα διακοπτών ως εξής:



Παράλληλη OR   Σε σειρά AND   Αντιστροφή NOT

Η αντιστοιχία είναι έγκυρη γιατί:
Η πράξη OR αντιστοιχίζεται σε κύκλωμα δύο παράλληλων διακοπτών γιατί αποκαθίσταται αγώγιμος δρόμος (λογική κατάσταση '1') όταν τουλάχιστον ένας διακόπτης είναι κλειστός. Αντίστοιχα η πράξη $x+y$ δίνει '1' μόνο όταν υπάρχει τουλάχιστον ένα από τα x και y ίσο με '1'.

Η πράξη AND αντιστοιχίζεται σε κύκλωμα δύο διακοπτών σε σειρά γιατί αποκαθίσταται αγώγιμος δρόμος, όταν και οι δύο διακόπτες είναι κλειστοί. Αντίστοιχα ισχύει (από πίνακα αληθείας) ότι η πράξη xy δίνει '1' μόνο όταν και τα δύο x και y είναι ίσα με '1'.
Η αντιστροφή NOT αντιστοιχίζεται σε κύκλωμα δύο κλάδων και δύο συσχετισμένων διακοπτών τέτοιων ώστε όταν αποκαθίσταται αγώγιμος δρόμος στον ένα κλάδο, ο άλλος να διακόπτεται. Αντίστοιχα η πράξη $\bar{x}$ δίνει '1' μόνο όταν $x=0$.

Ερώτημα (β)
Αντικαθιστώ τις τιμές στις λογικές παραστάσεις:

$$abc = (1\cdot 1)\cdot 0 = 1\cdot 0 = 0$$

$$a+b+c = 1+1+0 = 1+0 = 1$$

$$a+bc = 1+1\cdot 0 = 1+0 = 1$$

$$ab+c = 1\cdot 1+0 = 1+0 = 1$$

Ερώτημα (γ)
Από τον πίνακα αληθείας της παράστασης abc προκύπτει ότι η λύση είναι $(a,b,c)=(1,1,1)$. Διαφορετικά, αν υπάρχει τουλάχιστον ένα μηδέν στην τριάδα, τότε από τον ορισμό του λογικού AND

προκύπτει ότι *abc*=0. Άρα δεν είναι δυνατόν να περιέχονται 0 στη λύση. Άρα η τριάδα (1,1,1) είναι η μοναδική λύση.
Αντίστοιχα για την παράσταση $a+b+c$, από τον πίνακα αληθείας η λύση είναι (0,0,0). Διαφορετικά, αν υπάρχει τουλάχιστον ένα 1, τότε από τον ορισμό του λογικού OR, η παράσταση είναι 1.

Ερώτημα (δ)
Από τον ορισμό της AND, αν υπάρχει τουλάχιστον ένα 0, στη N-αδα $(a_1, a_2, ..., a_N)$, τότε αυτή δεν είναι λύση της εξίσωσης. Επίσης από τον ορισμό, η $(a_1, a_2, ..., a_N) = (1,1,....,1)$ είναι λύση. Αντίστοιχα για την OR, μοναδική λύση είναι η $(a_1, a_2, ..., a_N) = (0,0,...,0)$. Αν υπήρχε τουλάχιστον ένα $a_i = 1$, από τον ορισμό της OR προκύπτει ότι το αποτέλεσμα θα είναι 1, άρα δεν είναι δυνατόν να περιλαμβάνεται 1 στη λύση.

ΑΣΚΗΣΗ 21

Πώς αντιστοιχίζονται οι λογικές συναρτήσεις σε πύλες; Περιγράψτε τη λογική λειτουργία των πυλών NAND, NOR, XOR. Με πόσους και ποιους τρόπους μπορεί να γίνει η περιγραφή αυτή;

Λύση:

Οι λογικές πύλες είναι υλικές συσκευές που υλοποιούν τις βασικές λογικές πράξεις. Αντίστοιχα με τις ανεξάρτητες μεταβλητές των συναρτήσεων οι πύλες έχουν εισόδους, στις οποίες καταλήγουν τα σήματα που θέλουμε να επεξεργαστούμε. Οι πύλες έχουν έξοδο, η οποία αντιστοιχεί στην τιμή της σχετικής λογικής συνάρτησης.
Η πύλη NAND υλοποιεί τη λογική συνάρτηση $(xy)'$.
Η πύλη NOR υλοποιεί τη λογική συνάρτηση $(x+y)'$.
Η πύλη XOR υλοποιεί τη λογική συνάρτηση $x \oplus y = xy' + x'y$.
Δύο τρόποι με τους οποίους μπορεί να περιγραφεί η λειτουργία μιας λογικής πύλης είναι ο πίνακας αληθείας και η αντίστοιχη λογική συνάρτηση.

ΑΣΚΗΣΗ 22

α. Να απλοποιήσετε τις συναρτήσεις: *xy'z+ x'yz+yz* και *xy'z+xy'z+xyz'*.
β. Να υπολογίσετε τις αντίστροφες συναρτήσεις.
γ. Να σχεδιάσετε κυκλώματα λογικών πυλών που να υλοποιούν τις αντίστροφες συναρτήσεις, επιδιώκοντας να χρησιμοποιήσετε όσο το δυνατόν λιγότερες λογικές πύλες.

Λύση:

Ερώτημα (α)

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

Εφαρμόζοντας το θεώρημα της απορρόφησης, προκύπτει ότι $xy' + y = y + x$,
άρα η αρχική συνάρτηση γίνεται $(y+x)z = yz + xz$.

Για τη δεύτερη συνάρτηση:

$$\begin{aligned} xy'z + xy'z + xyz' &= xy'z(1+1) + xyz' \\ &= xy'z + xyz' \\ &= x(y'z + yz') \\ &= x(y \oplus z) \end{aligned}$$

Ερώτημα (β)

Εφαρμόζεται το Θεώρημα του DeMorgan,

$$\begin{aligned} f' &= (xz + yz)' = (x' + z')(y' + z') \\ &= x'y' + x'z' + z'y' + z'z' \\ &= x'y' + x'z' + (y'+1)z' = x'y' + x'z' + z' \\ &= x'y' + (x'+1)z' \\ &= x'y' + z' \end{aligned}$$

Εναλλακτικά: f'=(x'+z')(y'+z')=z'+x'y' (επιμεριστική ιδιότητα)

Ομοίως για τη δεύτερη συνάρτηση:

$$f' = (x(y \oplus z))' = x' + (y \oplus z)'$$

Δεύτερος Τρόπος για το Ερώτημα (β)

Εφαρμόζουμε στην αρχική έκφραση το γενικευμένο DeMorgan στην πρώτη συνάρτηση και στη συνέχεια εκτελούμε τις πράξεις:

$$\begin{aligned} f' &= (xy'z + x'yz + yz)' \\ &= (x' + y + z')(x + y' + z')(y' + z') \\ &= (x'y' + x'z' + xy + yz' + z')(y' + z') \\ &= (x'y' + xy + z'[x' + y + 1])(y' + z') \\ &= (x'y' + xy + z')(y' + z') \\ &= x'y' + z'y' + z'x'y' + z'xy + z' \\ &= z'(y' + x'y' + 1) + x'y' \\ &= z' + x'y' \end{aligned}$$

Εφαρμόζουμε στην αρχική έκφραση το γενικευμένο θεώρημα DeMorgan και στη συνέχεια εκτελούμε τις πράξεις:

$$\begin{aligned} f' &= (xy'z + xy'z + xyz')' \\ &= (x' + y + z')(x' + y + z')(x' + y' + z) \\ &= (x' + x'y + x'z' + yx' + y + yz' + z'x' + yz' + z')(x' + y' + z) \\ &= (x'[1 + y + z'] + y + z'[1 + y])(x' + y' + z) \\ &= (x' + y + z')(x' + y' + z) \\ &= x' + x'y + x'z + yx' + yz + z'x' + z'y' \\ &= x'[1 + y + z + z'] + yz + z'y' \\ &= x' + yz + y'z' \\ &= x' + (y \oplus z)' = [x(y \oplus z)]' \end{aligned}$$

Ερώτημα (γ)

Η πρώτη συνάρτηση γράφεται $((x'y'+z')')'=((x'y')'z)'=((x+y)z)'$, άρα είναι μία πύλη OAI. Αν δεν είναι διαθέσιμη σύνθετη πύλη, απαιτείται μια OR και μια NAND. Η δεύτερη συνάρτηση υλοποιείται με μία NAND και μια XOR.

ΑΣΚΗΣΗ 23

α. Να απλοποιήσετε τις συναρτήσεις: $f = x'y'z + x'yz + y'z$ και $g = xy'z + xz + xyz'$.

β. Να υπολογίσετε τις αντίστροφες συναρτήσεις f' και g', χρησιμοποιώντας τις απλοποιημένες εκφράσεις που προέκυψαν στο βήμα α.

γ. Να γραφούν οι f και g ως άθροισμα ελαχιστόρων.

δ. Αποδείξτε ότι ισχύει $f'f = 0$ και $g'g = 0$, χρησιμοποιώντας τις απλοποιημένες εκφράσεις των συναρτήσεων f, g, f' και g' που προέκυψαν στα βήματα α και β και εκτελώντας πράξεις.

ε. Να σχεδιάσετε κυκλώματα λογικών πυλών που να υλοποιούν τις αντίστροφες συναρτήσεις, επιδιώκοντας να χρησιμοποιήσετε όσο το δυνατόν λιγότερες λογικές πύλες.

Λύση:

Ερώτημα α

Για τη συνάρτηση *f* ισχύει:

$$f = x'y'z + x'yz + y'z = (x'+1)y'z + x'yz$$
$$= y'z + x'yz = (y'+x'y)z = (y'+x')z$$

Εφαρμόζοντας το θεώρημα της απορρόφησης, προκύπτει ότι $y'+x'y = y'+x'$,

άρα η *f* γίνεται $(y'+x')z = y'z + x'z$.

Εναλλακτικά, με χρήση της επιμεριστικής ιδιότητας: *f = x'(y'+y)z+y'z = x'1z+y'z = x'z+y'z = (x'+y')z*

Για τη συνάρτηση *g*:

$$g = xy'z + zx + xyz' = (y'+1)zx + xyz'$$
$$= zx + xyz'$$
$$= x(z + yz')$$
$$= x(z + y) = xz + xy$$

Ερώτημα β

Εφαρμόζεται το Θεώρημα του DeMorgan στις απλοποιημένες μορφές των συναρτήσεων με σκοπό να μειωθούν οι πράξεις,

$$f' = (y'z + zx')' = (y+z')(z'+x)$$
$$= yz' + xy + z'z' + z'x$$
$$= yz' + xy + z' + z'x = z'(y+1+x) + xy$$
$$= z' + xy$$

Εναλλακτικά, με χρήση της επιμεριστικής ιδιότητας: *f'=(y'z+zx')'=(y+z')(z'+x)=z'+xy*

Ομοίως για τη συνάρτηση *g*:

g' = [x(z+y)]' = x'+(z+y)' = x'+z'y'

Ερώτημα γ

Σε κάθε όρο της απλοποιημένης μορφής εμφανίζουμε ως παράγοντα άθροισμα των αντίθετων μεταβλητών που δεν περιλαμβάνονται στο όρο, ως εξής

$$f = y'z + x'z = \left(x + x'\right)y'z + x'\left(y + y'\right)z$$
$$= xy'z + x'y'z + x'yz + x'y'z$$
$$= xy'z + x'y'z + x'yz$$

g = xz+xy = xz(y+y')+xy(z+z') =
= xyz+xy'z+xyz+xyz' =
= xy'z+xyz+xyz'=Σ(5,6,7)

Ερώτημα δ

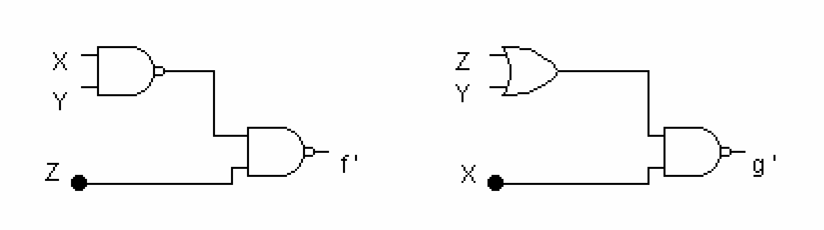
Εκτελούμε τις πράξεις ως εξής

$$f'f = \left(z' + xy\right)\left(y'z + x'z\right) = z'y'z + z'x'z + xyy'z + xyx'z$$
$$= 0 + 0 + 0 + 0 = 0$$

γιατί σε κάθε όρο του αθροίσματος γινομένων περιλαμβάνονται γινόμενα αντιθέτων μεταβλητών. Ομοίως για την *g*
g'g = (x'+z'y')(xz+xy) = x'xz+x'xy+z'y'xz+z'y'xy = 0+0+0+0 = 0

Ερώτημα ε

Από το ερώτημα (β) ισχύει *f' = z'+xy*, η οποία υλοποιείται με τρεις πύλες, μία AND, μία OR και μία NOT. Αν εφαρμόσουμε τον νόμο του De Morgan η *f'* γίνεται *f' = (z(xy)')'* η οποία υλοποιείται με δύο πύλες NAND όπως φαίνεται στο παρακάτω σχήμα αριστερά.
Για την *g'* από το ερώτημα (β) ισχύει *g'=x'+z'y'*. Αν εφαρμόσουμε τον νόμο του De Morgan γίνεται *g'=x'+(z+y)'=(x(z+y))'* η οποία υλοποιείται με δύο πύλες, μία NAND και μία OR όπως φαίνεται στο παρακάτω σχήμα δεξιά.



ΑΣΚΗΣΗ 24

Δίνεται η συνάρτηση *f(x,y,z)=xy+y'z'+xy'z'*. Να γραφεί ως άθροισμα ελαχιστόρων. Να κατασκευαστεί ο πίνακας αληθείας. Να υπολογιστεί η τιμή της λογικής παράστασης: *[f(0,0,0)+f(0,1,0)]•f(1,1,0)*.

Λύση:
(Σημείωση: Το σύμβολο της λογικής πράξης AND (•) ανάμεσα στις μεταβλητές παραλείπεται για λόγους απλότητας στην παρουσίαση. Έτσι π.χ. η παράσταση *x'y* σημαίνει *x'•y*.)

Για την μετατροπή της συνάρτησης σε άθροισμα ελαχιστόρων πρέπει σε κάθε όρο της απλοποιημένης μορφής να εμφανίσουμε ως παράγοντα, άθροισμα των αντίθετων μεταβλητών που δεν περιλαμβάνονται στο όρο, όπως φαίνεται παρακάτω:

f(x,y,z) = xy+y'z'+xy'z'

= xy(z+z’)+y’z’(x+x’)+xy’z’
= xyz+xyz’+xy’z’+x’ y’z’+xy’z’
= xyz+xyz’+xy’z’+x’y’z’
= x’y’z’+xy’z’+xyz’+xyz
=Σ(0, 4, 6, 7)

Ο πίνακας αληθείας είναι ο ακόλουθος:

x	y	z	x’y’z’	xy'z'	xyz'	xyz	f(x,y,z)=xyz+ xy'z'+ x'yz'+ x'y'z
0	0	0	1	0	0	0	1
0	0	1	0	0	0	0	0
0	1	0	0	0	0	0	0
0	1	1	0	0	0	0	0
1	0	0	0	1	0	0	1
1	0	1	0	0	0	0	0
1	1	0	0	0	1	0	1
1	1	1	0	0	0	1	1

Για τον υπολογισμό της τιμής της λογικής παράστασης υπάρχουν δύο τρόποι:

1ος τρόπος
Αντικαθιστώ τις τιμές των μεταβλητών και βρίσκω την τιμή καθενός όρου της λογικής παράστασης:
f(0,0,0) = 0• 0+0’• 0’+0• 0’• 0’ = 0• 0+1• 1+0• 1• 1’ = 0 + 1 + 0 = 1
Όμοια βρίσκω ότι: f(0,1,0)=0 και f(1,1,0)=1.
Τελικά η λογική παράσταση ισούται με:
[f(0,0,0)+f(0,1,0)]• f(1,1,0) = [1+0]• 1 = 1• 1 = 1

2ος τρόπος
Τόσο από την μετατροπή της συνάρτησης σε άθροισμα ελαχιστόρων όσο και από τον πίνακα αλήθειας, μπορώ εύκολα να υπολογίσω τις τιμές των όρων της λογικής παράστασης. Για τον όρο *f(0,0,0)* παρατηρώ ότι αντιστοιχεί στον ελαχιστόρο m_0 ή στην πρώτη γραμμή του πίνακα αλήθειας και ισούται με 1. Όμοια βρίσκω ότι ο όρος *f(0,1,0)* αντιστοιχεί στον ελαχιστόρο m_2 ή στην τρίτη γραμμή του πίνακα αλήθειας και ισούται με 0 καθώς και ότι ο όρος *f(1,1,0)* αντιστοιχεί στον ελαχιστόρο m_6 ή στην προτελευταία γραμμή του πίνακα αλήθειας και ισούται με 1.
Τελικά υπολογίζω την τιμή της λογικής παράστασης όπως προηγουμένως:
[f(0,0,0)+f(0,1,0)]• f(1,1,0) = [1+0]• 1 = 1• 1 = 1

ΑΣΚΗΣΗ 25

α. Δίνεται η λογική παράσταση $f(x,y) = xy + x'y'$. Να κατασκευαστεί ο αντίστοιχος πίνακας αληθείας. Θεωρώντας τα *x*, *y* δυαδικά σήματα, μπορείτε να σχεδιάσετε κύκλωμα που να υλοποιεί τη λογική παράσταση με μία πύλη;
β. Δίνεται η συνάρτηση $f(x,y,z) = xyz + xy'z' + x'yz' + x'y'z$. Να κατασκευαστεί ο πίνακας αληθείας. Μπορείτε να υλοποιήσετε τη συνάρτηση με κύκλωμα δύο πυλών;

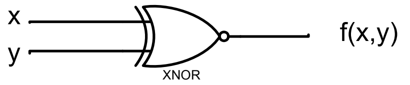
Λύση:

Ερώτημα (α)
Ο πίνακας αληθείας είναι ο ακόλουθος:

x	y	x'	y'	xy	x'y'	xy+ x'y'
0	0	1	1	0	1	1
0	1	1	0	0	0	0

1	0	0	1	0	0	0
1	1	0	0	1	0	1

Το κύκλωμα είναι λογική πύλη XNOR δύο εισόδων.



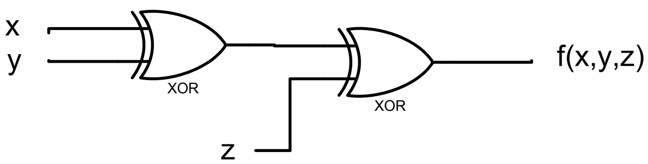
Ερώτημα (β)
Ο πίνακας αληθείας είναι ο ακόλουθος:

x	y	z	xyz	xy'z'	x'yz'	x'y'z	xyz+ xy'z'+ x'yz'+ x'y'z
0	0	0	0	0	0	0	0
0	0	1	0	0	0	1	1
0	1	0	0	0	1	0	1
0	1	1	0	0	0	0	0
1	0	0	0	1	0	0	1
1	0	1	0	0	0	0	0
1	1	0	0	0	0	0	0
1	1	1	1	0	0	0	1

$$
\begin{aligned}
xyz + xy'z' + x'yz' + x'y'z &= x\left(yz + y'z'\right) + x'\left(yz' + y'z\right) \\
&= x\left(y \oplus z\right)' + x'\left(y \oplus z\right) \\
&= x \oplus \left(y \oplus z\right).
\end{aligned}
$$

όπου χρησιμοποιήθηκε ότι $\left(y \oplus z\right)' = \left(yz + y'z'\right)$, το οποίο αποδεικνύεται με πράξεις.

Άρα το κύκλωμα είναι δύο XOR σε σειρά. Από τον πίνακα αλήθειας προκύπτει ότι είναι 1 <u>μόνο</u> σε περίπτωση που έχουμε περιττό αριθμό μονάδων.



ΑΣΚΗΣΗ 26

α. Δίνεται η λογική παράσταση $f\left(x, y\right) = xy' + x'y$. Να κατασκευαστεί ο αντίστοιχος πίνακας αληθείας. Θεωρώντας τα *x*, *y* δυαδικά σήματα, μπορείτε να σχεδιάσετε κύκλωμα που να υλοποιεί τη λογική παράσταση με μία πύλη;

β. Δίνεται η συνάρτηση $f\left(x, y, z\right) = xyz + xy'z + x'yz' + x'y'z$. Να κατασκευαστεί ο πίνακας αληθείας. Μπορείτε να υλοποιήσετε τη συνάρτηση με κύκλωμα δύο πυλών;

Λύση:

Ερώτημα (α)
Ο πίνακας αληθείας είναι ο ακόλουθος:

x	y	x'	y'	xy'	x'y	xy'+ x'y
0	0	1	1	0	0	0
0	1	1	0	0	1	1

1	0	0	1	1	0	1
1	1	0	0	0	0	0

Το κύκλωμα είναι λογική πύλη XOR δύο εισόδων.

Ερώτημα (β)
Ο πίνακας αληθείας είναι ο ακόλουθος:

x	y	z	xyz	xy'z'	x'yz'	x'y'z	xyz+ xy'z'+ x'yz'+ x'y'z
0	0	0	0	0	0	0	0
0	0	1	0	0	0	1	1
0	1	0	0	0	1	0	1
0	1	1	0	0	0	0	0
1	0	0	0	1	0	0	1
1	0	1	0	0	0	0	0
1	1	0	0	0	0	0	0
1	1	1	1	0	0	0	1

$$
\begin{aligned}
xyz + xy'z' + x'yz' + x'y'z &= x\left(yz + y'z'\right) + x'\left(yz' + y'z\right) \\
&= x\left(y \oplus z\right)' + x'\left(y \oplus z\right) \\
&= x \oplus \left(y \oplus z\right).
\end{aligned}
$$

όπου χρησιμοποιήθηκε ότι $\left(y \oplus z\right)' = \left(yz + y'z'\right)$, το οποίο αποδεικνύεται με πράξεις.

Άρα το κύκλωμα είναι δύο XOR σε σειρά. Από τον πίνακα αλήθειας προκύπτει ότι είναι 1 <u>μόνο</u> σε περίπτωση που έχουμε περιττό αριθμό μονάδων.

ΑΣΚΗΣΗ 27

Υλοποιήστε τις τρεις βασικές λογικές πύλες (NOT, AND, OR) χρησιμοποιώντας μόνο πύλες NAND. (Υπόδειξη: Ξεκινήστε από την λογική σχέση της βασικής πύλης και προσπαθήστε να την μετασχηματίσετε σε σχέση που περιέχει όρους της μορφής $\overline{x \cdot y}$ που αντιστοιχούν μόνο στην πύλη NAND).

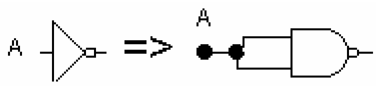
Λύση:

Ξεκινάμε από την λογική σχέση της πύλης NOT ($\overline{\mathrm{A}}$) και την μετασχηματίζουμε κατάλληλα εφαρμόζοντας αξιώματα και ιδιότητες της άλγεβρας Boole:

$\overline{\mathrm{A}}$ (ισχύει ότι: $\mathrm{X} = \mathrm{X} \cdot \mathrm{X}$, για X=A προσθέτοντας άρνηση και στα δύο μέλη έχουμε)

$\overline{\mathrm{A} \cdot \mathrm{A}}$

Υλοποιώντας μόνο με πύλες NAND:



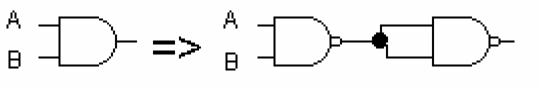
Ξεκινώντας από την λογική σχέση της πύλης AND έχουμε:

$\mathrm{A} \cdot \mathrm{B}$ (προσθέτοντας δύο αρνήσεις – που ισοδυναμούν με μία κατάφαση - έχουμε)

$\overline{\overline{\mathrm{A} \cdot \mathrm{B}}}$ (ισχύει ότι: $\mathrm{X} = \mathrm{X} \cdot \mathrm{X}$, εφαρμόζοντας για $\mathrm{X} = \overline{\mathrm{A} \cdot \mathrm{B}}$ έχουμε)

$\overline{\left(\overline{\mathrm{A} \cdot \mathrm{B}}\right) \cdot \left(\overline{\mathrm{A} \cdot \mathrm{B}}\right)}$

Υλοποιώντας μόνο με πύλες NAND:



Ξεκινώντας από την λογική σχέση της πύλης OR έχουμε:

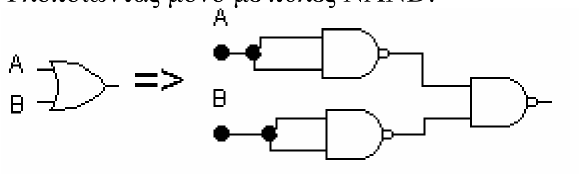
$A+B$ (προσθέτοντας δύο αρνήσεις – που ισοδυναμούν με μία κατάφαση - έχουμε)

$\overline{\overline{A+B}}$ (εφαρμόζουμε τον νόμο De Morgan για την σχέση $\overline{A+B}$)

$\overline{\overline{A}\cdot\overline{B}}$ (ισχύει ότι: $X = X \cdot X$, εφαρμόζοντας δύο φορές για $X = A$ και $X = B$ έχουμε)

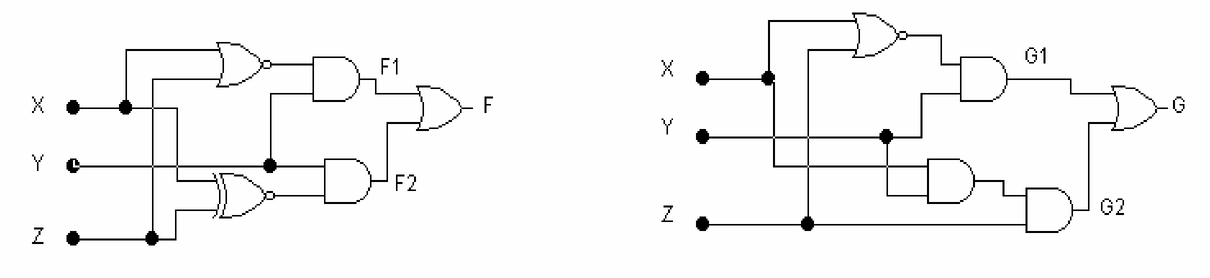
$\overline{\overline{A\cdot A}\cdot\overline{B\cdot B}}$

Υλοποιώντας μόνο με πύλες NAND:



ΑΣΚΗΣΗ 28

Βρείτε τις λογικές συναρτήσεις F(x,y,z) και G(x,y,z) που υλοποιούνται από τα κυκλώματα των παρακάτω σχημάτων. Είναι αυτές ισοδύναμες μεταξύ τους;



Λύση:

Έχοντας υπόψη τη λογική παράσταση εξόδου κάθε πύλης, βρίσκω αρχικά τις λογικές σχέσεις για τα F1, F2 και έπειτα για την F.

F1 = (x + z)’ • y

F2 = (x ⊕ z)’ • y = (x• z+x’• z’) • y

F = F1+F2 = [(x+z)’ • y] + [(x• z+x’• z’) • y]

Δουλεύω όμοια στο άλλο σχήμα για την εύρεση της G.

G1 = (x + z)’ • y

G2 = (x• y) • z

G = G1 + G2 = [(x+z)’ • y] + [(x• y) • z]

Οι λογικές συναρτήσεις F(x,y,z) και G(x,y,z) που υπολόγισα δεν είναι σε κανονική μορφή και έτσι δεν μπορώ να απαντήσω σχετικά με την ισοδυναμία τους. Εκφράζω τις συναρτήσεις σε κανονική μορφή:

F = [(x+z)’ • y] + [(x• z+x’• z’) • y]
  = [(x’• z’) • y] + [x• y• z+x’• y• z’]
  = x’• y• z’ + x• y• z + x’• y• z’
  = x’• y• z’ + x• y• z = Σ(2,7)

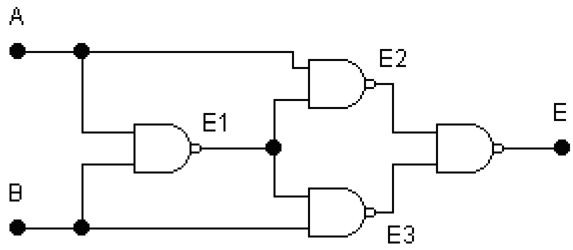
G = [(x+z)’ • y] + [(x• y) • z]
  = [(x’• z’) • y] + x• y• z

$= x' \bullet y \bullet z' + x \bullet y \bullet z = \Sigma(2,7)$

Τώρα είναι εύκολο να παρατηρήσω ότι οι δύο συναρτήσεις είναι ισοδύναμες.

ΑΣΚΗΣΗ 29

Βρείτε τη σχέση εξόδου του παρακάτω κυκλώματος και απλοποιήστε την ώστε να δείτε με ποια γνωστή λογική πύλη ισοδυναμεί. (Υπόδειξη: Προχωρήστε διαδοχικά βρίσκοντας πρώτα τις σχέσεις για τα E1, E2, E3 και τελικά για την έξοδο E).



Λύση:

Από την σχέση εξόδου καθεμίας πύλης NAND έχουμε:

$E1 = \overline{A \cdot B}$, $E2 = \overline{A \cdot E1}$, $E3 = \overline{B \cdot E1}$ και $E = \overline{E2 \cdot E3}$

Αντικαθιστώντας διαδοχικά βρίσκουμε την E σε συνάρτηση με τα A, B:

$E = \overline{\overline{A \cdot \overline{A \cdot B}} \cdot \overline{B \cdot \overline{A \cdot B}}}$ (εφαρμόζουμε το νόμο De Morgan με $X = A \cdot \overline{A \cdot B}$ και $Y = B \cdot \overline{A \cdot B}$)

$A \cdot \overline{A \cdot B} + B \cdot \overline{A \cdot B}$ (κοινός παράγοντας το $\overline{A \cdot B}$)

$(A + B) \cdot (\overline{A \cdot B})$ (εφαρμόζουμε τον νόμο De Morgan για την σχέση $\overline{A \cdot B}$)

$(A + B) \cdot (\overline{A} + \overline{B})$ (κάνουμε τις πράξεις για να φύγουν οι παρενθέσεις)

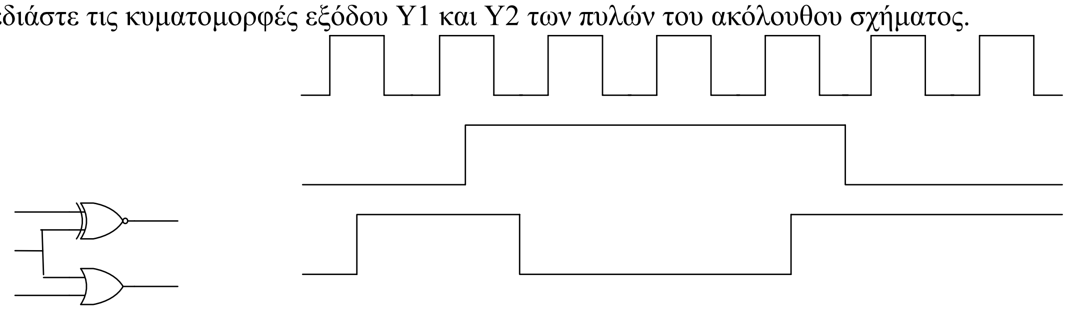
$A \cdot \overline{A} + A \cdot \overline{B} + B \cdot \overline{A} + B \cdot \overline{B}$ (ισχύει ότι: $X \cdot \overline{X} = 0$ και: $X + 0 = X$, οπότε προκύπτει)

$A \cdot \overline{B} + B \cdot \overline{A}$

Όπως είναι γνωστό η τελευταία σχέση είναι η σχέση εξόδου της πύλης XOR, οπότε το παραπάνω κύκλωμα με τις τέσσερις πύλες NAND ισοδυναμεί με μία πύλη XOR.

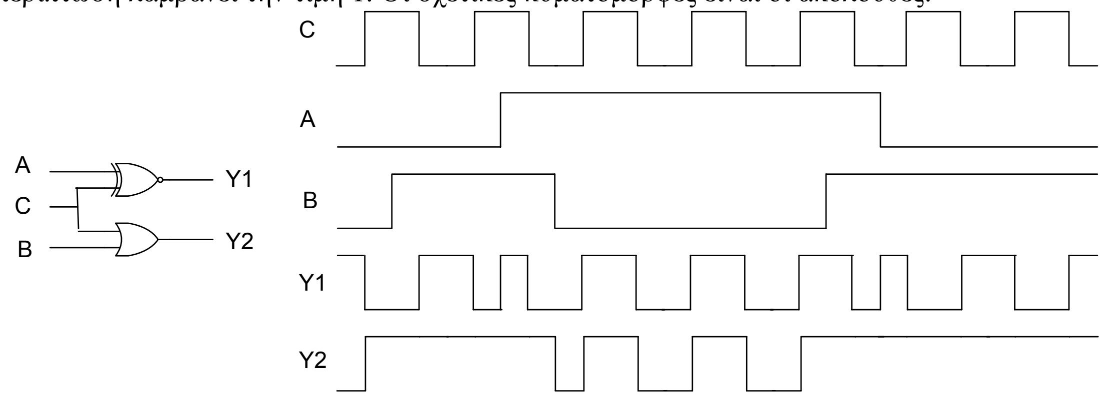
ΑΣΚΗΣΗ 30

Σχεδιάστε τις κυματομορφές εξόδου Y1 και Y2 των πυλών του ακόλουθου σχήματος.



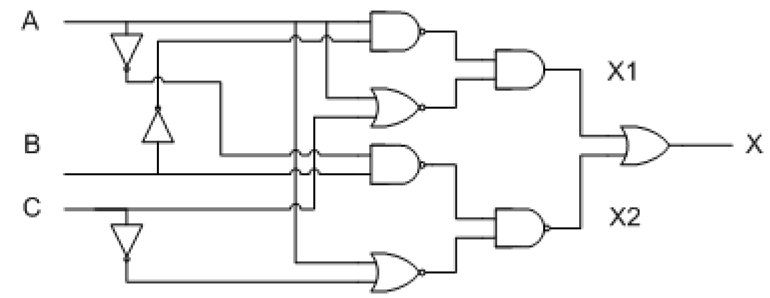
Λύση:

Η έξοδος Y1 προκύπτει ως Y1 = A □ C = (A ⊕ C)’=AC+A’C’. Παρατηρούμε ότι όταν A=0, τότε Y1=C’, ενώ όταν A=1, τότε Y1=C. Η παρατήρηση αυτή διευκολύνει την εύρεση της κυματομορφής. Η έξοδος Y2 = C + B, λαμβάνει την τιμή 0, όταν και τα δύο C και B είναι μηδέν, ενώ σε κάθε άλλη περίπτωση λαμβάνει την τιμή 1. Οι σχετικές κυματομορφές είναι οι ακόλουθες:



ΑΣΚΗΣΗ 31

Απλοποιείστε τη λογική συνάρτηση X του παρακάτω σχήματος, εκφράζοντας αυτή αρχικά ως άθροισμα των δύο όρων X1 και X2. Σχεδιάστε το λογικό κύκλωμα της απλοποιημένης συνάρτησης μόνο με πύλες NAND.



Λύση:

Η έξοδος γράφεται ως X = X1 + X2.

Η X1 είναι η έξοδος μιας πύλης AND, οι είσοδοι στην οποία προέρχονται από μία πύλη NAND και μία πύλη NOR. Άρα,

X1 = (AB’)’(A+C)’
= (AB’ + A + C)’ [De Morgan]
= (A(B’+1) + C)’ [A, κοινός παράγοντας, z + 1 = 1]
= (A ·1 + C)’
= (A + C)’ = A’C’ [De Morgan]

Αντίστοιχα για την X2:

X2 = ((A’B)’(A+C’)’)’
= ((A’B + A + C’)’)’ [DeMorgan]
= A’B + A + C’ [(z’)’ = z]
= A + B + C’ [απορόφηση]

Άρα για την συνολική συνάρτηση X, έχουμε

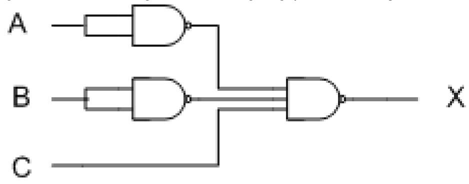
X = X1 + X2 = A’C’ + (A + B + C’) = A’C’ + A + B + C’
= (A’+1) C’ + A + B
= A + B + C’

η οποία είναι και η απλοποιημένη έκφραση. Αυτή γράφεται ως εξής

X = A + B + C’

= ((A + B + C’)’)’[(z’)’ = z]
= (A’B’C)’ [DeMorgan]

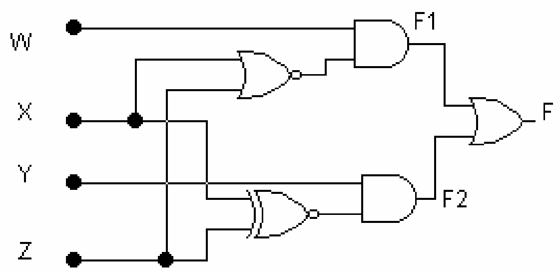
Η έκφραση αυτή οδηγεί και στην ακόλουθη υλοποίηση με πύλες NAND:

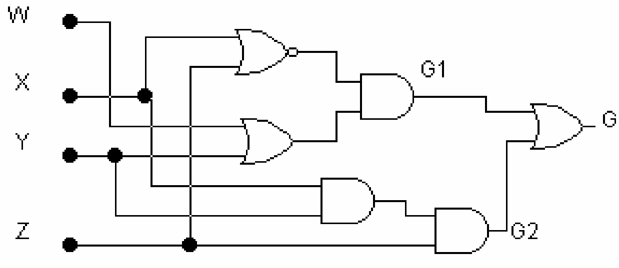


Παρατηρήστε ότι (A A)’ = A’.

ΑΣΚΗΣΗ 32

Βρείτε τις λογικές συναρτήσεις F(w,x,y,z) και G(w,x,y,z) που υλοποιούνται από τα κυκλώματα των παρακάτω σχημάτων. Είναι αυτές ισοδύναμες μεταξύ τους;





Λύση

Έχοντας υπόψη την λογική παράσταση εξόδου κάθε πύλης, βρίσκω αρχικά τις λογικές σχέσεις για τα F1, F2 και έπειτα για την F.

F1 = (x+z)’ •w
F2 = (x ⊕ z)’ •y = (x•z+x’•z’) • y
F = F1+F2 = [(x+z)’ • w] + [(x•z+x’•z’) •y]

Δουλεύω όμοια στο άλλο σχήμα για την εύρεση της G.

G1 = (x + z)’ • (w + y)
G2 = (x• y) • z
G = G1 + G2 = [(x+z)’ •(w + y)] + [(x• y) • z]

Οι λογικές συναρτήσεις F(w,x,y,z) και G(w,x,y,z) που υπολόγισα δεν είναι σε κανονική μορφή και έτσι δεν μπορώ να απαντήσω σχετικά με την ισοδυναμία τους.

Εκφράζω τις συναρτήσεις σε κανονική μορφή:

F= [(x+z)’ •w] + [(x•z+x’•z’) •y]
= [(x’• z’) • w] + [x•y•z+x’•y•z’]
= (w•x’• z’) + (x•y•z) +(x’•y•z’)
= [(w•x’• z’)•(y + y’)] + [(x•y•z) •(w + w’)] + [(x’•y•z’) •(w + w’)]
=(w•x’•y• z’)+(w•x’•y’• z’)+(w•x•y•z)+(w’•x•y•z)+(w•x’•y•z’)+(w’•x’•y•z’)
=Σ(10, 8, 15, 7, 10, 2)=Σ(2, 7, 8, 10 ,15)

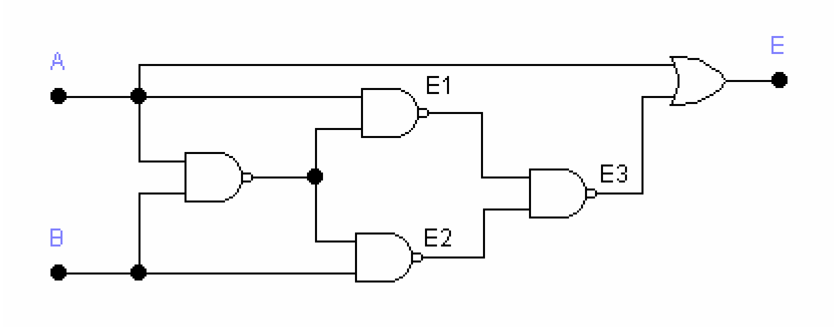
G= [(x+z)’ •(w + y)] + [(x• y) • z]
= [(x’• z’) •(w + y)] + (x•y•z)
= [(w•x’• z’) + (x’• y• z’)] + [x•y•z]
= [(w•x’• z’)•(y + y’) + (x’• y• z’) •(w + w’)] + [(x•y•z) •(w + w’)]
=(w•x’•y•z’)+(w•x’•y’•z’)+(w•x’•y• z’)+(w’•x’•y•z’)+(w•x•y•z)+(w’•x•y•z)
=Σ(10, 8, 10, 2,15,7)
=Σ(2,7,8,10 ,15)

Τώρα είναι εύκολο να παρατηρήσω ότι οι δύο συναρτήσεις είναι ισοδύναμες.

Στο ίδιο αποτέλεσμα μπορώ να καταλήξω συγκρίνοντας τους πίνακες αλήθειας ή τους χάρτες Karnaugh των δύο συναρτήσεων.

ΑΣΚΗΣΗ 33

Αποδείξτε ότι η έξοδος (E) του παρακάτω κυκλώματος αντιστοιχεί σε μία γνωστή λογική πύλη.



Λύση

Αρχικά βρίσκουμε τις λογικές συναρτήσεις των εξόδων E1, E2, E3 και έπειτα της E.

$E1 = ((AB)'A)'$

$E2 = ((AB)'B)'$

$E3 = (E1\ E2)' = (\ ((AB)'A)'\ ((AB)'B)'\ )'$

$E = E3 + A = (\ ((AB)'A)'\ ((AB)'B)'\ )' + A$

Στη συνέχεια απλοποιούμε την λογική συνάρτηση της εξόδου E ως εξής:

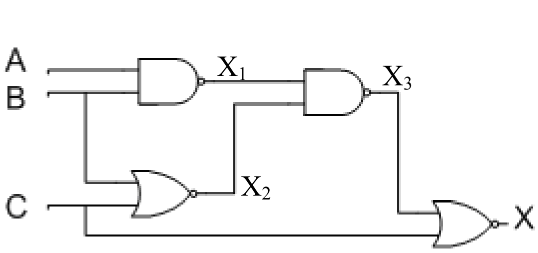
$E = (\ ((AB)'A)'\ ((AB)'B)'\ )' + A$

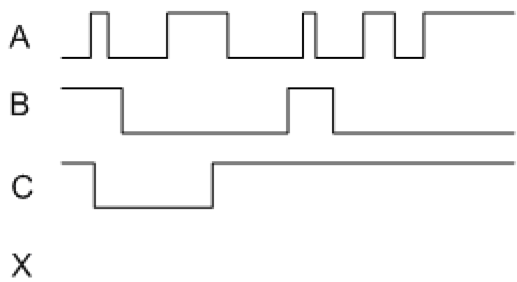
$= (\ (AB)'A + (AB)'B\ ) + A$	De Morgan
$= (\ (AB)'\ (A+B)\ ) + A$	
$= (\ (A'+B')\ (A+B)\ ) + A$	De Morgan
$= (\ A'A+A'B+B'A+B'B) + A$	
$= (\ A'B+B'A) + A$	$A'A=0\ \&\ X+0=X$
$= A'B+ (B'A + A)$	$A'A=0\ \&\ X+0=X$
$= A'B + A$	$A+AX=A$
$= B + A$	$A+A'B=A+B$

Έτσι αποδεικνύεται ότι η έξοδος (E) του κυκλώματος αντιστοιχεί στη πύλη OR.

ΑΣΚΗΣΗ 34

Απλοποιήστε το λογικό κύκλωμα του σχήματος και σχεδιάστε τη κυματομορφή εξόδου X. (Υπόδειξη: Σχεδιάστε τη κυματομορφή εξόδου X στο παρόν φύλλο αμέσως μετά από τις κυματομορφές των εισόδων A, B, C).





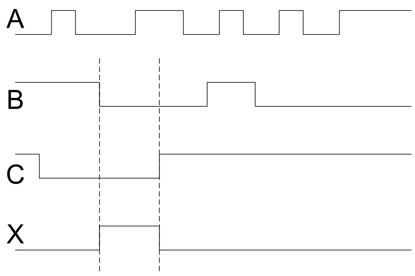
Λύση

$X_1 = (AB)'$

$X_2 = (B+C)'$

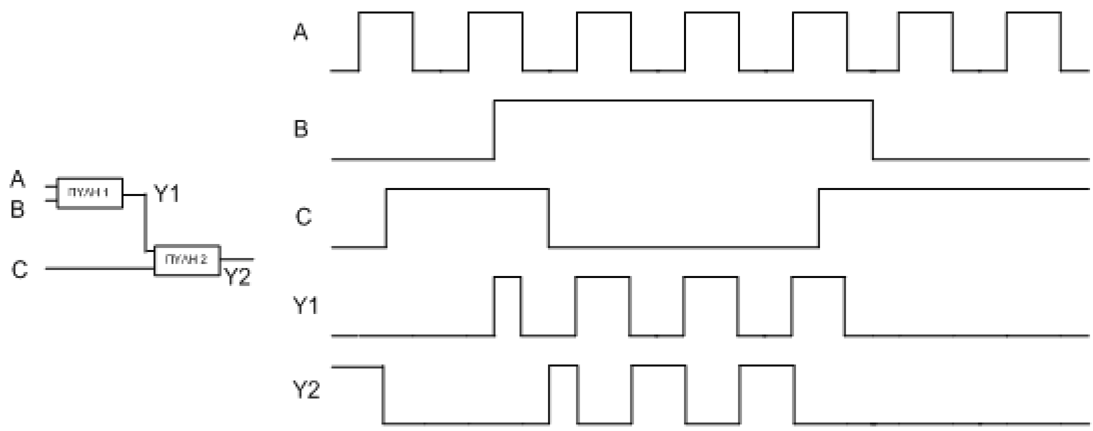
$X_3 = (X_1X_2)' = X_1' + X_2' = AB+B+C = B+C$

$X = (C+X_3)' = (C+B+C)' = (B+C)'$



ΑΣΚΗΣΗ 35

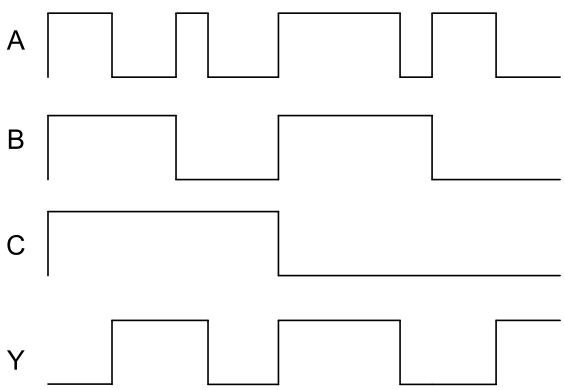
Δίνονται οι κυματομορφές A,B,C και Y1,Y2 του ακόλουθου σχήματος. Γνωρίζοντας ότι κάθε μία από τις δύο πύλες μπορεί να είναι AND, NAND, OR ή NOR, βρείτε σε ποια λογική πύλη αντιστοιχούν οι ΠΥΛΗ 1 και ΠΥΛΗ 2 του σχήματος.



Λύση:
Η έξοδος Y1 προκύπτει από τις εισόδους A, B της πύλης 1. Παρατηρώντας τις κυματομορφές A,B,Y1 θα προσπαθήσουμε να βρούμε σε ποια λογική πύλη αντιστοιχεί η ΠΥΛΗ 1. Παρατηρώντας τα χρονικά διαστήματα στα οποία η έξοδος Y1 είναι στο λογικό ‘1’ βλέπουμε ότι αυτό γίνεται μόνο όταν και οι δύο είσοδοι A,B είναι επίσης στο λογικό ‘1’, ενώ η έξοδος Y1 είναι στο λογικό ‘0’ σε όλες τις άλλες περιπτώσεις των εισόδων A,B. Έτσι, με βάση τη λειτουργία των λογικών πυλών AND, NAND, OR και NOR, η ΠΥΛΗ 1 αντιστοιχεί στην πύλη AND.
Η έξοδος Y2 προκύπτει από τις εισόδους Y1, C της πύλης 2. Παρατηρώντας τις κυματομορφές Y1,C,Y2 θα προσπαθήσουμε να βρούμε σε ποια λογική πύλη αντιστοιχεί η ΠΥΛΗ 2. Παρατηρώντας τα χρονικά διαστήματα στα οποία η έξοδος Y2 είναι στο λογικό ‘1’ βλέπουμε ότι αυτό γίνεται μόνο όταν και οι δύο είσοδοι Y1,C είναι στο λογικό ‘0’, ενώ η έξοδος Y2 είναι στο λογικό ‘0’ σε όλες τις άλλες περιπτώσεις των εισόδων Y1,C. Έτσι, με βάση τη λειτουργία των λογικών πυλών AND, NAND, OR και NOR, η ΠΥΛΗ 2 αντιστοιχεί στην πύλη NOR.

ΑΣΚΗΣΗ 36

Δίνονται οι ακόλουθες κυματομορφές τάσης - χρόνου, όπου A, B, C είναι είσοδοι και Y είναι η έξοδος. Η υψηλή τιμή αντιστοιχεί στο λογικό ‘1’ και η χαμηλή στο λογικό ‘0’.



Α. Βρείτε τη λογική συνάρτηση που υλοποιείται στην έξοδο Y.
Β. Κατασκευάστε έναν αντιστροφέα χρησιμοποιώντας μόνο μια πύλη XOR δύο εισόδων.
Γ. Μετασχηματίστε τη λογική συνάρτηση της εξόδου Y ώστε το κύκλωμα να υλοποιείται χρησιμοποιώντας μόνο πύλες XOR δύο εισόδων.
(Υπόδειξη: Στη θέση του αντιστροφέα χρησιμοποιήστε μία πύλη XOR δύο εισόδων όπως προκύπτει από το ερώτημα Β).

Λύση

Α. Από τις κυματομορφές καταστρώνουμε τον παρακάτω πίνακα αληθείας:

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

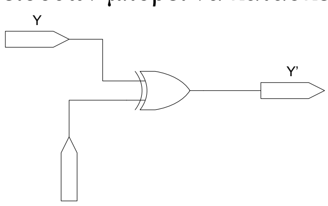
Γράφοντας την Y ως άθροισμα ελαχιστόρων δίνεται μια αλγεβρική έκφραση της ζητούμενης συνάρτησης:

$$Y = a'b'c' + a'bc + ab'c + abc'$$

Β. Ο πίνακας αληθείας της XOR είναι

X	y	$x \oplus y$
0	0	0
0	1	1
1	0	1
1	1	0

Παρατηρούμε ότι όταν το x = 0 τότε ότι η έξοδος ταυτίζεται με την είσοδο y. Όταν x=1, παρατηρούμε ότι η έξοδος είναι συμπληρωματική της εισόδου y. Άρα ένας αντιστροφέας με πύλη XOR δύο εισόδων μπορεί να κατασκευαστεί ως εξής:

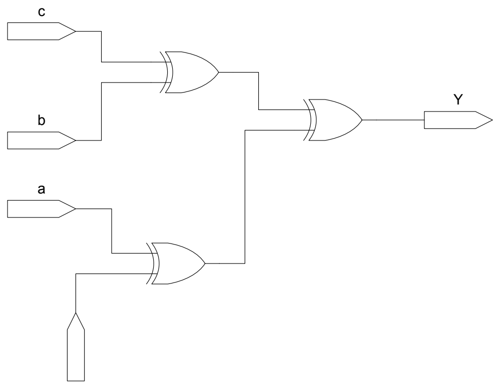


Γ. Μετασχηματίζοντας την έκφραση της Y την οποία βρήκαμε στο ερώτημα Α προκύπτει:

$$Y = a'\left(b'c' + bc\right) + a\left(b'c + bc'\right)$$

$$= a'\left(b \oplus c\right)' + a\left(b \oplus c\right)$$

για την εξαγωγή της οποίας χρησιμοποιήθηκαν οι εξής ιδιότητες: $x \oplus y = x'y + xy'$ και $\left(x \oplus y\right)' = x'y' + xy$. Ορίζοντας $d = a'$, η έκφραση της Y μπορεί να γραφεί ως $Y = d\left(b \oplus c\right)' + d'\left(b \oplus c\right)$, άρα $Y = d \oplus \left(b \oplus c\right)$. Επειδή σύμφωνα με την απάντηση στο ερώτημα Β, ο αντίστροφέας για τον υπολογισμό της d μπορεί να υλοποιηθεί με μια πύλη XOR, προκύπτει το ακόλουθο κύκλωμα:



ΑΣΚΗΣΗ 37

Γνωρίζοντας ότι AB=0 και A+B=1 αποδείξτε με αλγεβρικούς μετασχηματισμούς ότι ισχύει η σχέση: (A+C)(A’+B)(B+C) = BC

Λύση

1^{ος} τρόπος:

Ξεκινάμε από το αριστερό μέρος και προσπαθούμε να απλοποιήσουμε εκμεταλλευόμενοι τις δύο σχέσεις που γνωρίζουμε ότι ισχύουν:

(A+C) (A’+B) (B+C) =
= (AA’ + AB + A’C + BC) (B+C) = (AA’=0, AB=0)
= A’BC + A’CC + BBC + BCC = (BB=B, X+X=X)
= A’BC + A’C + BC =
= A’C (B + 1) + BC = (B+1=1, A+B=1)
= A’C (A + B) + BC =
= A’AC + A’BC + BC = (AA’=0)
= (A’ +1) BC = (X+1=1)
= BC

2^{ος} τρόπος:

Καταστρώνουμε τον πίνακα αλήθειας για τις μεταβλητές A, B και τις σχέσεις που μας δίνονται:

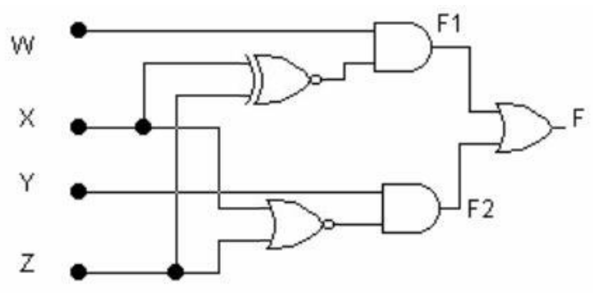
A	B	AB	A+B
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	1

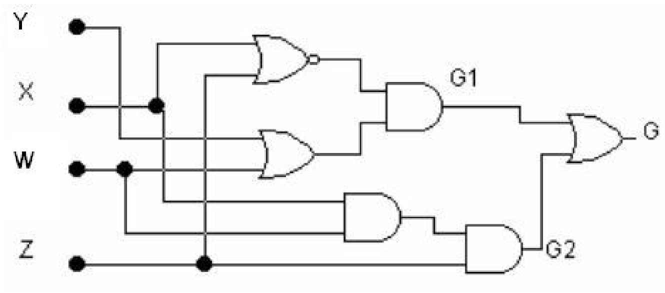
Παρατηρούμε ότι οι σχέσεις AB=0 και A+B=1 επαληθεύονται συγχρόνως στη 2^{η} και 3^{η} γραμμή του πίνακα αλήθειας όπου ισχύει ότι A=B’. Χρησιμοποιώντας αυτή τη σχέση (A=B’) το αριστερό μέρος της σχέσης που θέλουμε να αποδείξουμε γίνεται:

$(A+C)(A'+B)(B+C) =$
$= (B'+C)(B+B)(B+C) =$ (B+B=B)
$= B(B'B + B'C + CB + CC) =$ (BB'=0, CC=C, X+0=X)
$= B(B'C + BC + C) =$
$= B(C(B' + B) + C) =$ (B'+B=1)
$= B(C \cdot 1 + C) =$ (C·1=C, C+C=C)
$= BC$

ΑΣΚΗΣΗ 38

Βρείτε τις λογικές συναρτήσεις F(w,x,y,z) και G(w,x,y,z) που υλοποιούνται από τα κυκλώματα των παρακάτω σχημάτων, χρησιμοποιώντας αλγεβρικούς μετασχηματισμούς. Είναι αυτές οι λογικές συναρτήσεις ισοδύναμες μεταξύ τους;





Λύση

Έχοντας υπόψη την λογική παράσταση εξόδου κάθε πύλης, βρίσκω αρχικά τις λογικές σχέσεις για τα F1, F2 και έπειτα για την F.

$F1 = (x \oplus z)' \bullet w = (x \bullet z + x' \bullet z') \bullet w$

$F2 = (x + z)' \bullet y$

$F = F1 + F2 = [(x \bullet z + x' \bullet z') \bullet w] + [(x+z)' \bullet y]$

Δουλεύω όμοια στο άλλο σχήμα για την εύρεση της G.

$G1 = (x + z)' \bullet (w + y)$

$G2 = (x \bullet w) \bullet z$

$G = G1 + G2 = [(x+z)' \bullet (w + y)] + [(x \bullet w) \bullet z]$

Οι λογικές συναρτήσεις F(w,x,y,z) και G(w,x,y,z) που υπολόγισα δεν είναι σε κανονική μορφή και έτσι δεν μπορώ να απαντήσω σχετικά με την ισοδυναμία τους.

Εκφράζω τις συναρτήσεις σε κανονική μορφή:

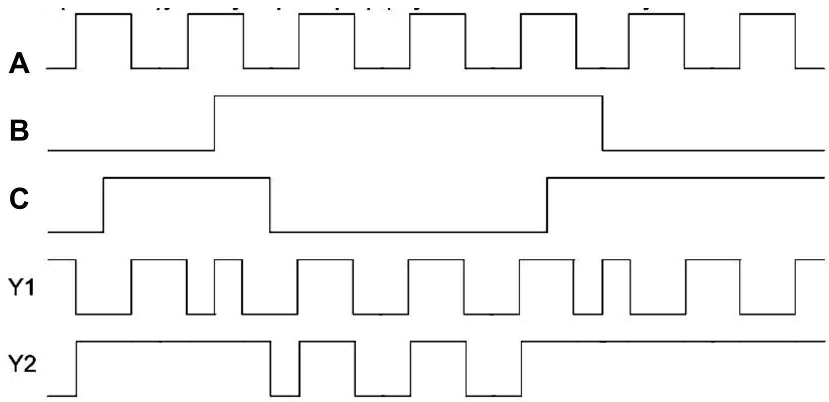
$F = [(x+z)' \bullet y] + [(x \bullet z + x' \bullet z') \bullet w]$
$= [(x' \bullet z') \bullet y] + [x \bullet w \bullet z + x' \bullet w \bullet z']$
$= (y \bullet x' \bullet z') + (x \bullet w \bullet z) + (x' \bullet w \bullet z')$
$= [(y \bullet x' \bullet z') \bullet (w + w')] + [(x \bullet w \bullet z) \bullet (y + y')] + [(x' \bullet w \bullet z') \bullet (y + y')]$
$= (w \bullet x' \bullet y \bullet z') + (w' \bullet x' \bullet y \bullet z') + (w \bullet x \bullet y \bullet z) + (w \bullet x \bullet y' \bullet z) + (w \bullet x' \bullet y \bullet z') + (w \bullet x' \bullet y' \bullet z')$
$= \Sigma(10, 2, 15, 13, 10, 8)$
$= \Sigma(2, 8, 10, 13, 15)$

$G = [(x+z)' \bullet (w + y)] + [(x \bullet w) \bullet z]$
$= [(x' \bullet z') \bullet (w + y)] + (x \bullet w \bullet z)$
$= [(w \bullet x' \bullet z') + (x' \bullet y \bullet z')] + [x \bullet w \bullet z]$
$= [(w \bullet x' \bullet z') \bullet (y + y') + (x' \bullet y \bullet z') \bullet (w + w')] + [(x \bullet w \bullet z) \bullet (y + y')]$
$= (w \bullet x' \bullet y \bullet z') + (w \bullet x' \bullet y' \bullet z') + (w \bullet x' \bullet y \bullet z') + (w' \bullet x' \bullet y \bullet z') + (w \bullet x \bullet y \bullet z) + (w \bullet x \bullet y' \bullet z)$
$= \Sigma(10, 8, 10, 2, 15, 13)$
$= \Sigma(2, 8, 10, 13, 15)$

Τώρα είναι εύκολο να παρατηρήσω ότι οι δύο συναρτήσεις είναι ισοδύναμες.
Στο ίδιο αποτέλεσμα μπορώ να καταλήξω συγκρίνοντας τους πίνακες αλήθειας ή τους χάρτες Karnaugh των δύο συναρτήσεων.
Επίσης, μετασχηματίζοντας αλγεβρικά τη συνάρτηση F προκύπτει:
$F=w(x'z'+xz)+y(z+x)'$
$=wx'z'+wxz+y(x'z')$
$= x'z'(w+y) + wxz$
$=(x+z)'(w+y) + wxz$
$=G$
Άρα οι δύο συναρτήσεις είναι ισοδύναμες μεταξύ τους.

ΑΣΚΗΣΗ 39

Δίνονται οι ακόλουθες κυματομορφές τάσης - χρόνου, όπου A, B, C είναι είσοδοι και Y1, Y2 είναι οι έξοδοι. Η υψηλή τιμή αντιστοιχεί στο λογικό ‘1’ και η χαμηλή στο λογικό ‘0’.



A. Να βρεθεί η λογική συνάρτηση που υλοποιείται στις εξόδους Y1 και Y2.
B. Να σχεδιαστεί το κύκλωμα που υλοποιεί τις Y1 και Y2 χρησιμοποιώντας μόνο πύλες δύο εισόδων.

Λύση

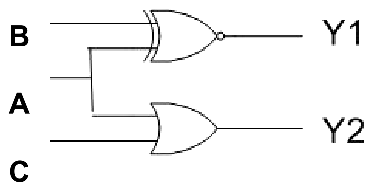
A. Από τις κυματομορφές καταστρώνουμε τον παρακάτω πίνακα αληθείας:

A	B	C	Y1	Y2
0	0	0	1	0
0	0	1	1	1
0	1	0	0	0
0	1	1	0	1
1	0	0	0	1
1	0	1	0	1
1	1	0	1	1
1	1	1	1	1

Γράφοντας την Y1 ως άθροισμα ελαχιστόρων δίνεται μια αλγεβρική έκφραση της ζητούμενης συνάρτησης:
$Y1 = A'B'C' + A'B'C + ABC' + ABC$

Για την Y2 παρατηρούμε ότι είναι απλούστερο να γράψουμε την Y2′, οπότε θα είναι:
$Y2' = A'B'C' + A'BC' = A'C' \Rightarrow Y2 = A + C$

B. Η Y1 γράφεται: $Y1 = A'B' + AB = (A \oplus B)'$, οπότε το κύκλωμα θα είναι:



ΑΣΚΗΣΗ 40

Είναι γνωστό ότι μια λογική συνάρτηση μπορεί να γραφεί είτε σαν άθροισμα ελαχιστόρων (SOP) είτε σαν γινόμενο μεγιστόρων (POS). Για να εξασκηθείτε λοιπόν στην άλγεβρα Boole, αποδείξτε ότι για μια συνάρτηση F(x,y,z), ισχύει Σ(2,4,5,6)=Π(0,1,3,7).

Λύση

Α' Τρόπος:

Έχουμε

Σ(2,4,5,6)= (Σ(0,1,3,7))′=
=(x′y′z′+x′y′z+x′yz+xyz)′
=((x+y+z)′+(x+y+z′)′+(x+y′+z′)′+(x′+y′+z′)′)′
=(x+y+z)″(x+y+z′)″(x+y′+z′)″(x′+y′+z′)″
=(x+y+z)(x+y+z′)(x+y′+z′)(x′+y′+z′)
=M0 M1 M3 M7
=Π(0,1,3,7)

Β' Τρόπος:

Μια άλλη λογική, θα ήταν να ξεκινήσουμε από το κάθε μέρος της ισότητας ξεχωριστά και να καταλήξουμε σε μια κοινή απλοποιημένη μορφή.

Έχουμε λοιπόν ότι:

Σ(2,4,5,6)=x'yz'+xy'z'+xy'z+xyz'=z'y(x+x')+xy'(z+z')=yz'+xy'

Επίσης,

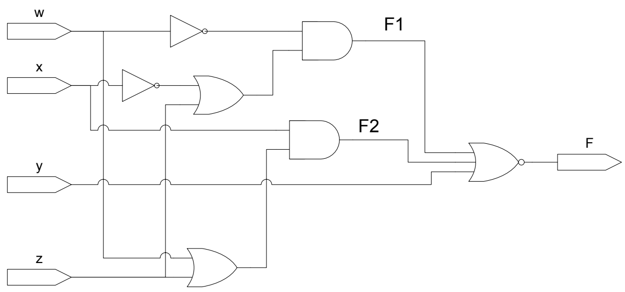
Π(0,1,3,7)=(x+y+z)(x+y+z')(x+y'+z')(x'+y'+z')=
=(xx+xy+xz'+xy+yy+yz'+xz+yz+zz')(xx'+xy'+xz'+y'x'+y'y'+y'z'+z'x'+z'y'+z'z')
=(x+xy+xz'+xy+xz+y+yz'+yz+zz')(xx'+xy'+y'x'+y'+y'z'+z'y'+xz'+z'x'+z')
=(x+y)(y'+z')
=xy'+xz'+yy'+yz'
=xy'+xz'+yz'
=xy'+xy'z'+xyz'+yz'
=xy'+yz'

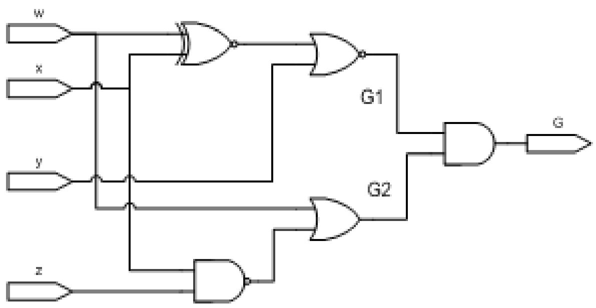
Σημείωση: με κόκκινο και μπλέ χρώμα φαίνονται οι εφαρμογές των θεωρημάτων απορρόφησης (A+AB=A) ως προς y και x αντίστοιχα, ενώ με πράσινο, η εφαρμογή της γνωστής ιδιότητας AA'=0.

ΑΣΚΗΣΗ 41

Βρείτε τις λογικές συναρτήσεις F(w,x,y,z) και G(w,x,y,z) που υλοποιούνται από τα κυκλώματα των παρακάτω σχημάτων. Είναι αυτές ισοδύναμες μεταξύ τους;

(Υπόδειξη: Προχωρήστε διαδοχικά βρίσκοντας πρώτα τις λογικές συναρτήσεις F1,F2 και G1,G2.)





Λύση

Η συνάρτηση F γράφεται ως

$F = (F1 + F2 + y)'$

γιατί προκύπτει ως έξοδος πύλης NOR τριων εισόδων. Οι είσοδοι της πύλης προκύπτουν ως

$F1 = w'(x' + z)$ [έξοδος πύλης AND με είσοδο την x' και την έξοδο της πύλης OR]

$F2 = x\,(w+z)$ [έξοδος πύλης AND με μια είσοδο την x και δεύτερη είσοδο την έξοδο της πύλης OR]

$F = (F1 + F2 + y)' = F1'\, F2'\, y'$ [de Morgan]

$= (w'(x' + z))'\,(x(w+z))'\,y'$ [de Morgan]

$= (w + (x' + z)')\,(x' + (w+z)')\,y'$ [de Morgan]

$= (w + xz')(x' + w'z')y'$ [de Morgan]

$= (wx' + w\,w'z' + xz'x' + xz'\,w'z')\,y'$ [επιμεριστική]

$= (wx' + xz'w')\,y'$ [ιδιότητες $a\,a' = 0$, $a\,a = a$]

$= wx'\,y' + w'x\,y'\,z'$

Η συνάρτηση F γράφεται ως άθροισμα ελαχιστόρων

$F = wx'\,y'\,(z+z') + w'x\,y'\,z' = w'x\,y'\,z' + wx'\,y'\,z' + wx'\,y'\,z + = \Sigma(4,8,9)$.

Η συνάρτηση G γράφεται ως

$G = G1\,G2$

Όπου

$G1 = (\text{xnor}(w,x) + y)' = \text{xor}(w,x)\,y'$ [de Morgan]

$= (w'x + wx')\,y'$ [από πίνακα αληθείας της xor]

$= w'x\,y' + wx'y'$

$G2 = w + (x\,z)' = w + x' + z'$

Συνολικά

$G = (w'x\,y' + wx'y')\,(w + x' + z')$

$= 0 + 0 + w'x\,y'\,z' + w\,wx'y' + wx'\,x'y' + wx'y'\,z'$

$= w'x\,y'\,z' + wx'y' + wx'y' + wx'y'\,z'$ [ιδιότητες $a\,a' = 0$, $a\,a = a$]

$= w'x\,y'\,z' + wx'y' + wx'y'\,z'$

$= w'x\,y'\,z' + wx'y'\,(1 + z')$ [κοινός παράγοντας το $wx'y'$]

$= w'x\,y'\,z' + wx'y'$ [ιδιότητα $1 + a = 1$]

Η συνάρτηση G γράφεται ως άθροισμα ελαχιστόρων:

$G = w'x\,y'\,z' + wx'y'(z+z') = w'x\,y'\,z' + wx'y'z' + wx'y'z = \Sigma\,(4, 9, 8)$.

Επειδή οι εκφράσεις των συναρτήσεων F και G ως αθροίσματα ελαχιστόρων ταυτίζονται, οι συναρτήσεις είναι ισοδύναμες.

Σημείωση: στη συγκεκριμένη περίπτωση, οι συναρτήσεις F και G καταλήγουν και στην ίδια αλγεβρική έκφραση, άρα είναι ισοδύναμες.

IV. Σχεδίαση συνδυαστικών κυκλωμάτων με λογικές πύλες

ΑΣΚΗΣΗ 42

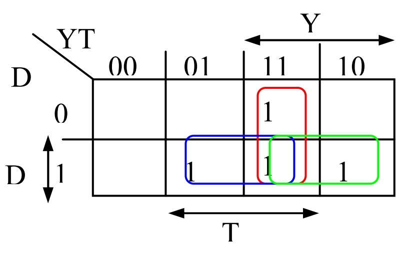
Το σύστημα ασφάλειας του χρηματοκιβωτίου μιας τράπεζας έχει δυο πόρτες, την εξωτερική (X) και την εσωτερική (S) που οδηγεί στο χώρο που φυλάγεται ο χρυσός της τράπεζας. Οι δυο αυτές πόρτες έχουν ηλεκτρονικές κλειδαριές με θέσεις για τρία κλειδιά τα οποία έχουν:
Ο Διευθυντής (D) - Ο Υποδιευθυντής (Y) - Ο Ταμίας (T).
Η εξωτερική πόρτα ανοίγει με τα κλειδιά οποιονδήποτε δύο από τους παραπάνω υπαλλήλους.
Η εσωτερική πόρτα ανοίγει μόνο με τα κλειδιά και των τριών υπαλλήλων.
Συμβολίστε με λογικό ‘1’ την παρουσία ενός υπαλλήλου με το κλειδί και με λογικό ‘0’ το αντίθετο. Συμβολίστε με λογικό ‘1’ την κατάσταση μία πόρτα να είναι ανοιχτή και με λογικό ‘0’ την κατάσταση να είναι κλειστή.
A. Αν το σύστημα ασφαλείας υλοποιείται με ένα συνδυαστικό κύκλωμα, καθορίστε τις εισόδους και τις εξόδους του και κατασκευάστε τον πίνακα αλήθειας.
B. Απλοποιείστε το κύκλωμα χρησιμοποιώντας χάρτες Karnaugh και σχεδιάστε το με λογικές πύλες.

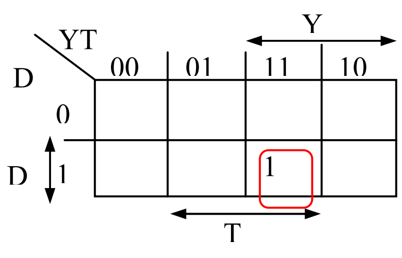
Λύση:

A. Είναι προφανές ότι στο συνδυαστικό κύκλωμα οι είσοδοι αντιστοιχούν στα τρία κλειδιά που έχουν οι υπάλληλοι (D,Y,T) ενώ οι έξοδοι του είναι οι δύο πόρτες (X,S). Λαμβάνοντας υπόψη τα δεδομένα της εκφώνησης ο πίνακας αλήθειας συμπληρώνεται όπως φαίνεται παρακάτω:

D	Y	T	X	S
0	0	0	0	0
0	0	1	0	0
0	1	0	0	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

B. Χρειαζόμαστε 2 χάρτες Karnaugh των τριών μεταβλητών, έναν για κάθε έξοδο.

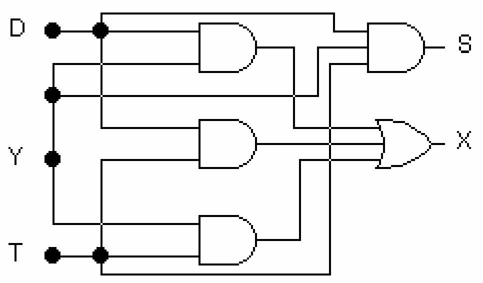




X = YT + DT + DY

S = DYT

Με βάση τις απλοποιημένες συναρτήσεις σχεδιάζουμε το παρακάτω κύκλωμα.

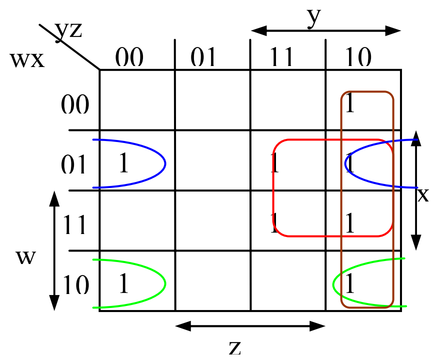


ΑΣΚΗΣΗ 43

Δίνεται η λογική συνάρτηση: F (w,x,y,z) = Σ(2,4,6,7,8,10,14,15)
A. Να απλοποιηθεί με χρήση του χάρτη Karnaugh.
B. Να σχεδιαστεί το λογικό της κύκλωμα με τον μικρότερο δυνατό αριθμό πυλών.

Λύση:

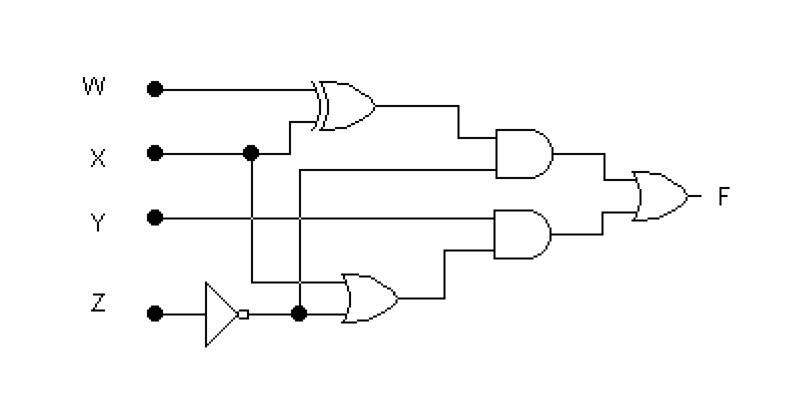
A. Σχηματίζουμε το χάρτη Karnaugh, προσέχοντας ιδιαίτερα την σωστή τοποθέτηση των ελαχιστόρων, και βρίσκουμε την απλοποιημένη συνάρτηση.



F(w,x,y,z) = x y + y z' + w'x z'+ w x' z'

B. Η απλοποίηση με χάρτη Karnaugh οδηγεί στην ελάχιστη υλοποίηση με πύλες AND-OR. Παρατηρώντας όμως τους όρους της συνάρτησης μπορούμε να την μετασχηματίσουμε έτσι ώστε να υλοποιηθεί με τον μικρότερο δυνατό αριθμό οποιονδήποτε πυλών, όπως φαίνεται παρακάτω:

$$\begin{aligned} F(w,x,y,z) &= xy + yz' + w'xz' + wx'z' \\ &= (x + z')\, y + (w'x + wx')\, z' \\ &= (x + z')\, y + (w \oplus x)\, z' \end{aligned}$$



ΑΣΚΗΣΗ 44

Ένα συνδυαστικό κύκλωμα δέχεται στην είσοδο έναν τετραψήφιο δυαδικό αριθμό ($X=X_3X_2X_1X_0$) και δίνει στην έξοδο έναν δυαδικό αριθμό (Y) ίσο με Y=X-9. Π.χ. για τον αριθμό $X=12_{10}=1100_2$, η έξοδος πρέπει να δίνει $Y=X-9=12-9=3_{10}=11_2$.
Όταν η έξοδος έχει αρνητική τιμή δεν μας ενδιαφέρει να εμφανίζεται η σωστή ένδειξη.
Α. Καθορίστε τις εισόδους και τις εξόδους του κυκλώματος και κατασκευάστε τον πίνακα αλήθειας.
Β. Απλοποιείστε τις συναρτήσεις εξόδου του κυκλώματος χρησιμοποιώντας χάρτες Karnaugh.
Γ. Υλοποιήστε το κύκλωμα χρησιμοποιώντας μόνο πύλες NAND.

Λύση
Α. Η μεγαλύτερη δεκαδική τιμή της εξόδου προκύπτει για την μεγαλύτερη δεκαδική τιμή της εισόδου $X=15_{10}$ και είναι ίση με: $Y=X-9=15-9=6_{10}=110_2$. Άρα η μεγαλύτερη δυαδική έξοδος απαιτεί τρία ψηφία. Έτσι το κύκλωμα αποτελείται από 4 εισόδους ($X=X_3X_2X_1X_0$), μία για κάθε δυαδικό ψηφίο εισόδου και 3 εξόδους ($Y=Y_2Y_1Y_0$) μία για κάθε δυαδικό ψηφίο εξόδου.
Επειδή, όταν η έξοδος έχει αρνητική τιμή δεν μας ενδιαφέρει να εμφανίζεται η σωστή ένδειξη, στον πίνακα αλήθειας θα εμφανίζονται αδιάφοροι όροι (X) στις αντίστοιχες θέσεις. Με βάση τα παραπάνω ο πίνακας αλήθειας είναι ο ακόλουθος:

Δυαδική Είσοδος				Δεκαδικοί Αριθμοί		Δυαδική Έξοδος		
X_3	X_2	X_1	X_0	X	Y	Y_2	Y_1	Y_0
0	0	0	0	0	-9	X	X	X
0	0	0	1	1	-8	X	X	X
0	0	1	0	2	-7	X	X	X
0	0	1	1	3	-6	X	X	X
0	1	0	0	4	-5	X	X	X
0	1	0	1	5	-4	X	X	X
0	1	1	0	6	-3	X	X	X
0	1	1	1	7	-2	X	X	X
1	0	0	0	8	-1	X	X	X
1	0	0	1	9	0	0	0	0
1	0	1	0	10	1	0	0	1
1	0	1	1	11	2	0	1	0
1	1	0	0	12	3	0	1	1
1	1	0	1	13	4	1	0	0
1	1	1	0	14	5	1	0	1
1	1	1	1	15	6	1	1	0

B. Σχηματίζουμε τους χάρτες Karnaugh, προσέχοντας τη σωστή χρήση των αδιάφορων όρων στην εύρεση των απλοποιημένων συναρτήσεων.
Για λόγους ομοιομορφίας με τους χάρτες προηγούμενων ασκήσεων, διατηρούμε την ονομασία των μεταβλητών (w,x,y,z) σε αντιστοιχία με τις μεταβλητές ($X_3X_2X_1X_0$).

wx \ yz	00	01	11	10
00	X	X	X	X
01	X	X	X	X
11		1	1	1
10	X			

Y_2(w,x,y,z) = x y + x z

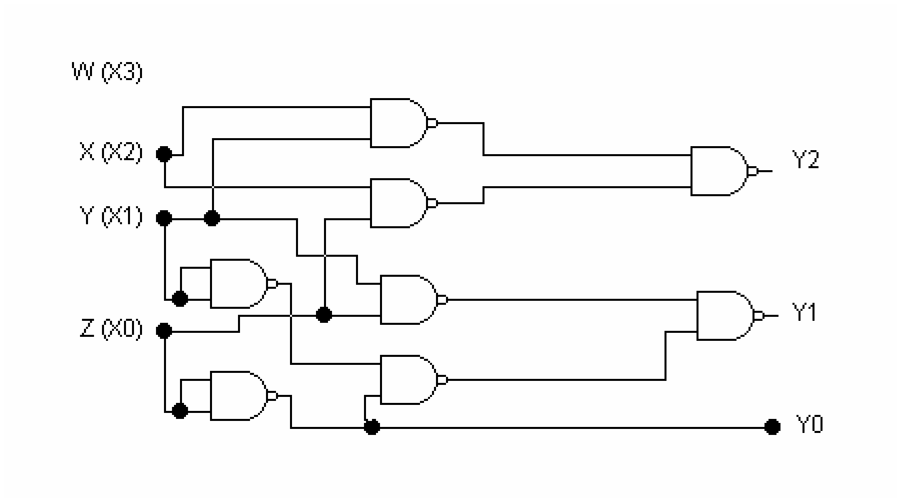
wx \ yz	00	01	11	10
00	X	X	X	X
01	X	X	X	X
11	1		1	
10	X		1	

Y_1(w,x,y,z) = y’z’ + y z

wx \ yz	00	01	11	10
00	X	X	X	X
01	X	X	X	X
11	1			1
10	X			1

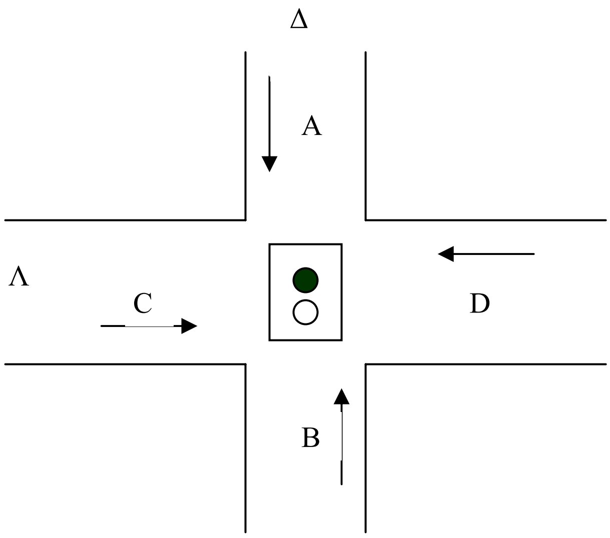
Y_0(w,x,y,z) = z’

Γ. Για την υλοποίηση του κυκλώματος μόνο με πύλες NAND οι συναρτήσεις των εξόδων πρέπει να τροποποιηθούν ως εξής:
Y_2 = xy+xz = [(xy)’ (xz)’]’
Y_1 = yz+y’z’ = [(yz)’ (y’z’)’]’
Y_0 = z’ = (zz)’
Η υλοποίηση φαίνεται στο παρακάτω σχήμα:



ΑΣΚΗΣΗ 45

Το παρακάτω σχήμα δείχνει τη διασταύρωση μιας λεωφόρου Λ με ένα δρόμο Δ. Στις γραμμές A, B, C, D έχουν τοποθετηθεί ανιχνευτές οχημάτων που δίνουν έξοδο ‘0’, όταν δεν είναι κατειλημμένες για 50 μέτρα πριν τη διασταύρωση και ‘1’, όταν συμβαίνει το αντίθετο.

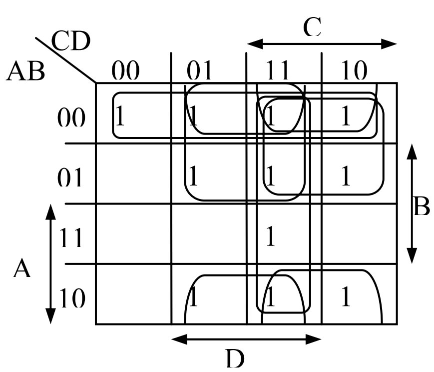
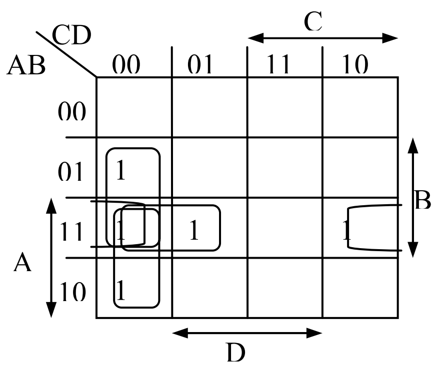
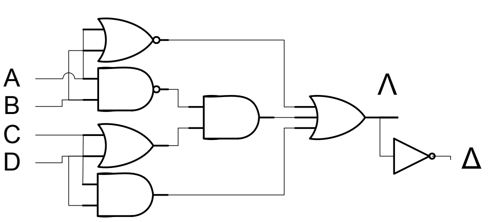
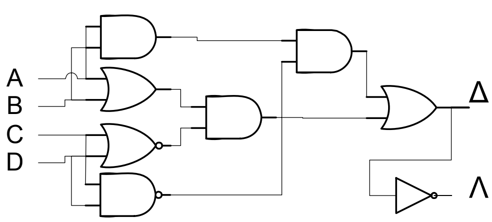


Το σύστημα ελέγχου της κυκλοφορίας λειτουργεί με τον ακόλουθο τρόπο:
Το φανάρι που επιτρέπει την κυκλοφορία στη λεωφόρο Λ γίνεται πράσινο και το φανάρι στο δρόμο Δ γίνεται κόκκινο, όταν συμβαίνει τουλάχιστον μία από τις παρακάτω περιπτώσεις:
Και οι δύο γραμμές C και D είναι κατειλημμένες.
Δεν υπάρχουν οχήματα στις γραμμές A,B,C και D.
Μία τουλάχιστον από τις γραμμές C,D είναι κατειλημμένη, αλλά οι γραμμές A,B δεν είναι και οι δύο κατειλημμένες.

Σε οποιαδήποτε άλλη περίπτωση το φανάρι που επιτρέπει την κυκλοφορία στη λεωφόρο Λ είναι κόκκινο και το φανάρι που επιτρέπει την κυκλοφορία στο δρόμο Δ είναι πράσινο.
Με βάση την παραπάνω λογική να καταστρωθεί ο πίνακας αλήθειας και ο αντίστοιχος χάρτης Karnaugh.
Να σχεδιαστεί λογικό κύκλωμα με την ελάχιστη υλοποίηση του προβλήματος.

Λύση:
Για τις μεταβλητές A, B, C, D ορίζουμε το λογικό 1 να αντιπροσωπεύει την κατάσταση «γραμμή κατειλημμένη». Επίσης, θεωρούμε ότι Λ=1 συμβολίζει το πράσινο φανάρι και Λ=0 το κόκκινο. Η συνάρτηση Δ είναι η συμπληρωματική της Λ, δηλ. Δ=Λ'.
Ο πίνακας αλήθειας της συνάρτησης Λ, ακολουθεί:

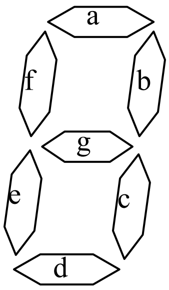
A	B	C	D	Λ	Δ
0	0	0	0	1	0
0	0	0	1	1	0
0	0	1	0	1	0
0	0	1	1	1	0
0	1	0	0	0	1
0	1	0	1	1	0
0	1	1	0	1	0
0	1	1	1	1	0
1	0	0	0	0	1
1	0	0	1	1	0
1	0	1	0	1	0
1	0	1	1	1	0
1	1	0	0	0	1
1	1	0	1	0	1
1	1	1	0	0	1
1	1	1	1	1	0

1ος ΤΡΟΠΟΣ	2ος ΤΡΟΠΟΣ
	
$\mathbf{\Lambda} = A'B'+CD+A'C+A'D+B'C+B'D =$ $= (A+B)' + CD + (A'+B')(C+D) =$ $= (A+B)' + CD + (AB)'(C+D)$	$\mathbf{\Delta} = BC'D'+AC'D'+ABC'+ABD' =$ $= (A+B)(C'D') + (AB)(C'+D') =$ $= (A+B)(C+D)' + (AB)(CD)'$
	

ΑΣΚΗΣΗ 46

Να σχεδιάσετε έναν αποκωδικοποιητή «BCD σε απεικόνιση-7-τμημάτων» (BCD to 7-segment display). Χρησιμοποιήστε απλοποίηση με χάρτες Karnaugh για την υλοποίηση των 7 συναρτήσεων: *a, b, c, d, e, f, g*.

Τα 10 BCD ψηφία απεικονίζονται στο display ως εξής:
[0] = {a,b,c,d,e,f}
[1] = {b,c}
[2] = {a,b,d,e,g}
[3] = {a,b,c,d,g}
[4] = {b,c,f,g}
[5] = {a,c,d,f,g}
[6] = {a,c,d,e,f,g}
[7] = {a,b,c}
[8] = {a,b,c,d,e,f,g}
[9] = {a,b,c,d,f,g}
(Οι τιμές 10 έως 15 δεν ενδιαφέρει πώς θα απεικονίζονται.)

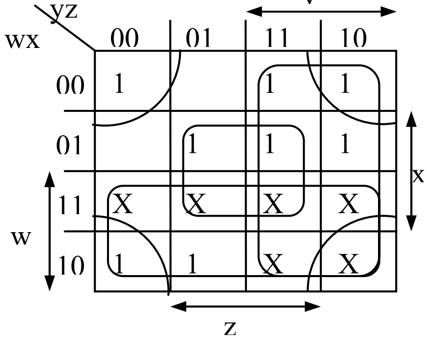


Λύση:

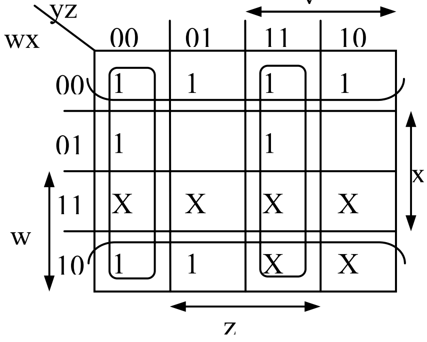
Θεωρώντας ότι η BCD τιμή σχηματίζεται από τις μεταβλητές w, x, y, z (όπου w είναι το ΠΣΨ), συμπληρώνουμε τον πίνακα αλήθειας των 7 συναρτήσεων a, b, c, d, e, f, g:

w	x	y	z	Αριθμός	a	b	c	d	e	f	g
0	0	0	0	0	1	1	1	1	1	1	0
0	0	0	1	1	0	1	1	0	0	0	0
0	0	1	0	2	1	1	0	1	1	0	1
0	0	1	1	3	1	1	1	1	0	0	1
0	1	0	0	4	0	1	1	0	0	1	1
0	1	0	1	5	1	0	1	1	0	1	1
0	1	1	0	6	1	0	1	1	1	1	1
0	1	1	1	7	1	1	1	0	0	0	0
1	0	0	0	8	1	1	1	1	1	1	1
1	0	0	1	9	1	1	1	1	0	1	1
1	X	1	X	A,B,E,F	X	X	X	X	X	X	X
1	1	X	X	C,D	X	X	X	X	X	X	X

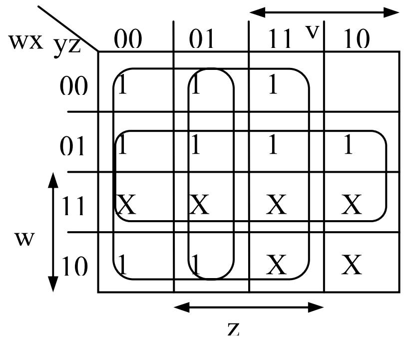
Στη συνέχεια, εφαρμόζοντας τη μέθοδο του χάρτη Karnaugh, βρίσκουμε τις απλοποιημένες εκφράσεις των 7 συναρτήσεων, σε μορφή αθροίσματος γινομένων και τις υλοποιούμε με δύο επίπεδα AND-OR (ή ισοδύναμα, αντικαθιστώντας τις πύλες AND και OR με πύλες NAND):



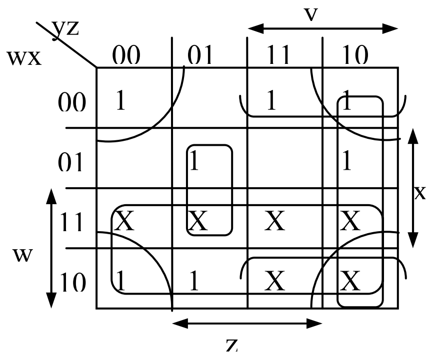
a = w + y + x·z + x'·z'



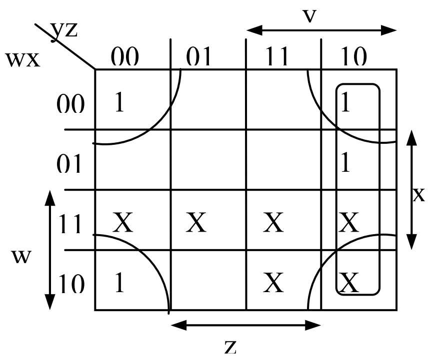
b = x' + y·z + y'·z'



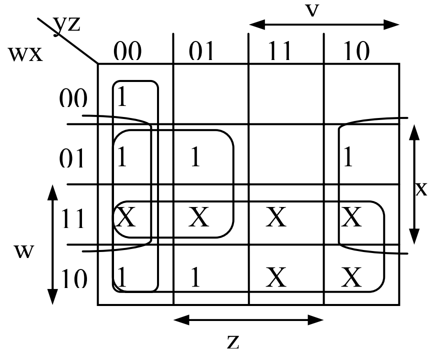
c = x + y’ + z



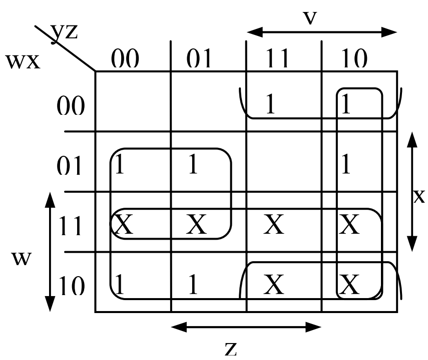
d = w + x·y’·z + x’·y + x’·z’ + y·z’



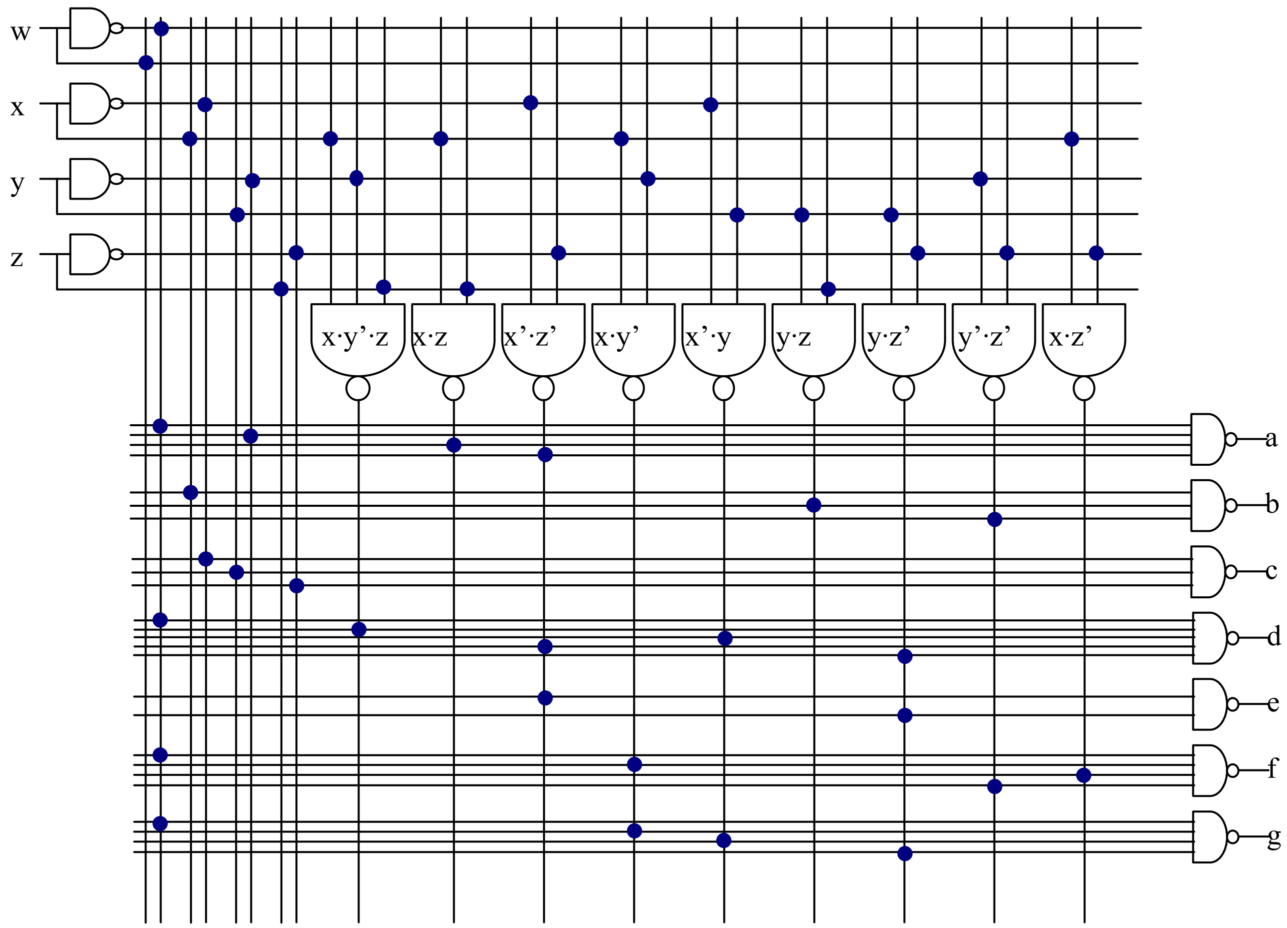
e = x’·z’+ y·z’



f = w + x·y’ + x·z’ + y’·z’



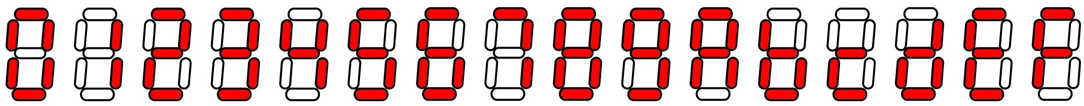
g = w + x·y’ + x’·y + y·z’

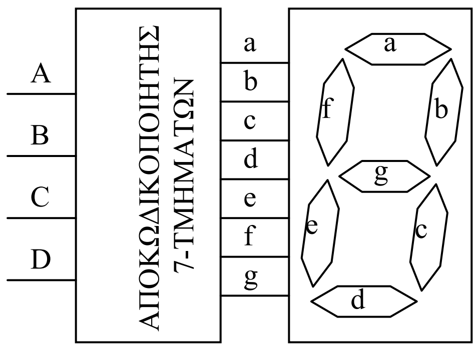


ΑΣΚΗΣΗ 47

Να σχεδιάσετε έναν αποκωδικοποιητή «HEX σε απεικόνιση 7-τμημάτων» (7-segment decoder). Χρησιμοποιήστε χάρτες Karnaugh για την απλοποίηση των 7 λογικών συναρτήσεων *a, b, c, d, e, f, g.*

Η απεικόνιση των αριθμών του δεκαεξαδικού συστήματος (0-F) στο display θα γίνεται ως εξής:





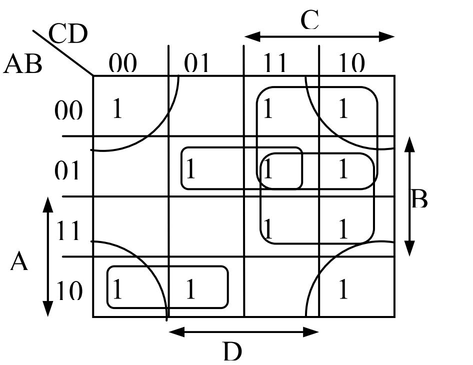
[0]={a,b,c,d,e,f}	[8]={a,b,c,d,e,f,g}
[1]={b,c}	[9]={a,b,c,d,f,g}
[2]={a,b,d,e,g}	[A]={a,b,c,e,f,g}
[3]={a,b,c,d,g}	[B]={c,d,e,f,g}
[4]={b,c,f,g}	[C]={d,e,g}
[5]={a,c,d,f,g}	[D]={b,c,d,e,g}
[6]={a,c,d,e,f,g}	[E]={a,d,e,f,g}
[7]={a,b,c}	[F]={a,e,f,g}

Να γράψετε όλες τις απλοποιημένες συναρτήσεις και να σχεδιάσετε τα λογικά κυκλώματα μόνο για τις συναρτήσεις *f* και *g*. Θεωρείστε ότι οι 4 είσοδοι του αποκωδικοποιητή είναι A, B, C, D (όπου A είναι το ΠΣΨ). Καθεμιά από τις διόδους led *a, b, c, d, e, f, g,* του display φωτοβολεί όταν η αντίστοιχη είσοδός της βρίσκεται σε λογικό «1».

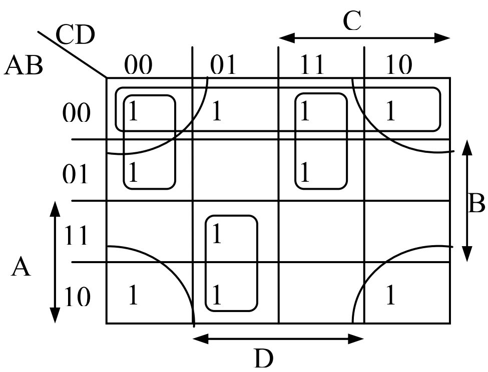
Λύση:
Θεωρώντας ότι η δεκαεξαδική τιμή σχηματίζεται από τις μεταβλητές A, B, C, D, (όπου A είναι το ΠΣΨ), συμπληρώνουμε τον πίνακα αλήθειας των 7 συναρτήσεων a, b, c, d, e, f, g:

A	B	C	D	Αριθμός	a	b	c	d	e	f	g
0	0	0	0	0	1	1	1	1	1	1	0
0	0	0	1	1	0	1	1	0	0	0	0
0	0	1	0	2	1	1	0	1	1	0	1
0	0	1	1	3	1	1	1	1	0	0	1
0	1	0	0	4	0	1	1	0	0	1	1
0	1	0	1	5	1	0	1	1	0	1	1
0	1	1	0	6	1	0	1	1	1	1	1
0	1	1	1	7	1	1	1	0	0	0	0
1	0	0	0	8	1	1	1	1	1	1	1
1	0	0	1	9	1	1	1	1	0	1	1
1	0	1	0	A	1	1	1	0	1	1	1
1	0	1	1	B	0	0	1	1	1	1	1
1	1	0	0	C	0	0	0	1	1	0	1
1	1	0	1	D	0	1	1	1	1	0	1
1	1	1	0	E	1	0	0	1	1	1	1
1	1	1	1	F	1	0	0	0	1	1	1

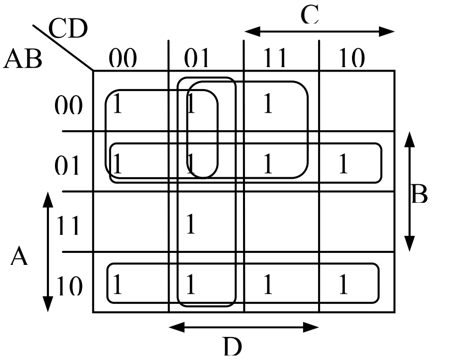
Στη συνέχεια, εφαρμόζοντας τη μέθοδο του χάρτη Karnaugh, βρίσκουμε τις απλοποιημένες εκφράσεις των 7 συναρτήσεων, σε μορφή αθροίσματος γινομένων:



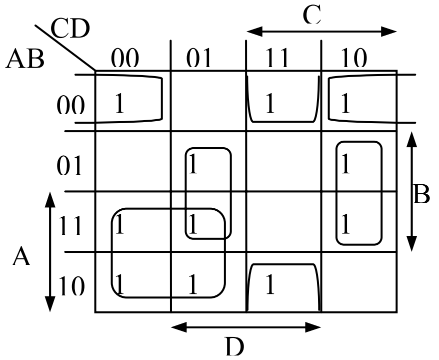
a = A’C+BC+B’D’+A’BD+AB’C’



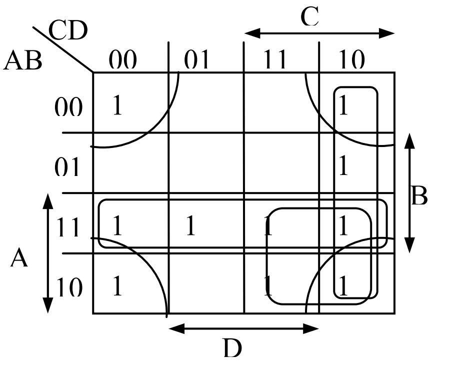
b = A’B’+B’D’+A’C’D’+A’CD+AC’D



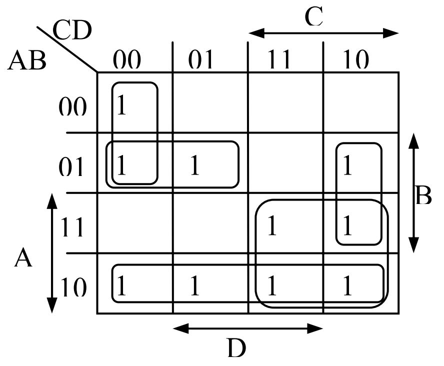
c = A’C’+A’D+A’B+AB’+C’D



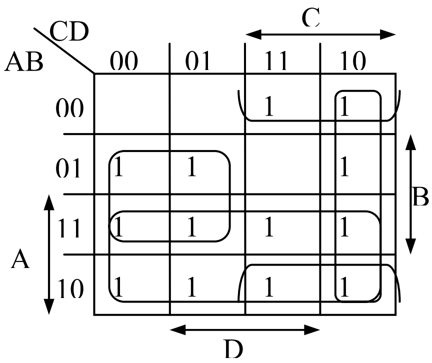
d = AC’+BC’D+BCD’+A’B’D’+B’CD



e = AB+AC+B’D’+CD’

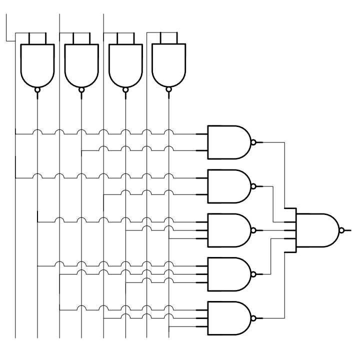
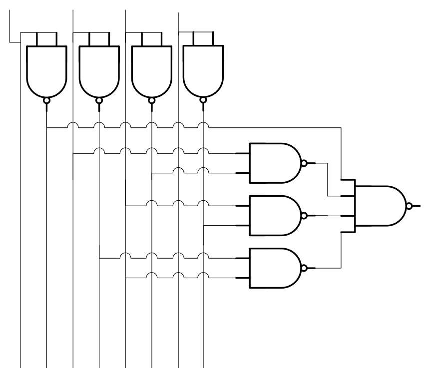


f = AB’+AC+A’C’D’+A’BC’+BCD’



g = A+BC’+CD’+B’C

Στη συνέχεια, υλοποιούμε τις συναρτήσεις f και g με τρία επίπεδα (NOT-AND-OR) (ή ισοδύναμα, αντικαθιστώντας τις πύλες NOT, AND και OR με πύλες NAND):

	
f = AB’+AC+A’C’D’+A’BC’+BCD’	**g** = A+BC’+CD’+B’C

ΑΣΚΗΣΗ 48

Α. Να συμπληρώσετε πίνακα, η αριστερή στήλη του οποίου να περιέχει όλους τους συνδυασμούς ($C_3C_2C_1C_0$, όπου C_3 το ΠΣΨ) σε μορφή «*συμπληρώματος προς 2 (2’s complement)*», ενώ η δεξιά στήλη τους αντίστοιχους αριθμούς του δεκαδικού συστήματος.
Β. Χρησιμοποιώντας τον ελάχιστο δυνατό αριθμό λογικών πυλών, να σχεδιάσετε κύκλωμα το οποίο να μετατρέπει προσημασμένους αριθμούς των 4 bits ($C_3C_2C_1C_0$, όπου C_3 το ΠΣΨ) από μορφή «*συμπληρώματος ως προς 2 (2’s complement)*» σε μορφή «*προσημασμένου μέτρου (signed magnitude)*».

Λύση:

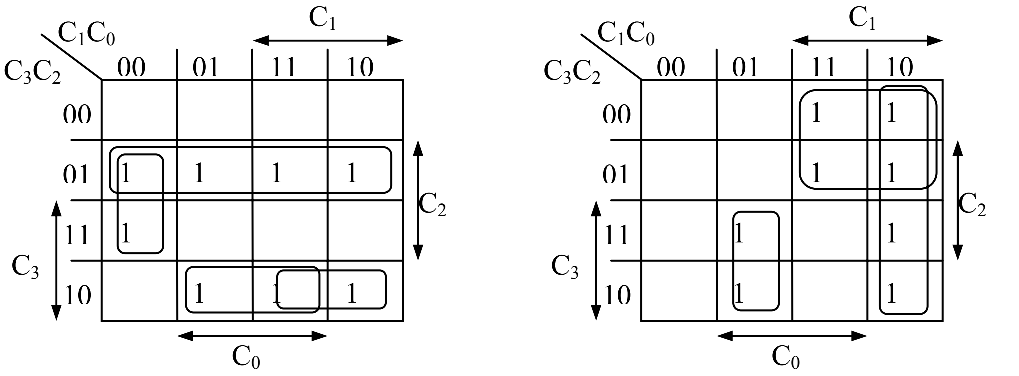
Α. Οι δυνατοί συνδυασμοί των 4 bits είναι 16 συνολικά, όπως φαίνεται στο αριστερό τμήμα του πίνακα που ακολουθεί. Αφού οι αριθμοί είναι σε αναπαράσταση συμπληρώματος ως προς 2, το bit C_3 θα παίζει το ρόλο του προσήμου (sign). Θυμηθείτε ότι η αξία (στο δεκαδικό) ενός αριθμού σε μορφή συμπληρώματος ως προς 2 υπολογίζεται από τη σχέση: $X= -C_3 2^3 + C_2 2^2 + C_1 2^1 + C_0 2^0$.

Συμπλήρωμα ως προς 2				Αριθμός Δεκαδικού Συστήματος
C_3	C_2	C_1	C_0	
0	0	0	0	0
0	0	0	1	+1
0	0	1	0	+2
0	0	1	1	+3
0	1	0	0	+4
0	1	0	1	+5
0	1	1	0	+6
0	1	1	1	+7
1	0	0	0	-8
1	0	0	1	-7
1	0	1	0	-6
1	0	1	1	-5
1	1	0	0	-4
1	1	0	1	-3
1	1	1	0	-2
1	1	1	1	-1

Συμπλήρωμα ως προς 2				Αριθμός Δεκαδικού Συστήματος	Προσημασμένο Μέτρο				
C_3	C_2	C_1	C_0		M_4	M_3	M_2	M_1	M_0
0	0	0	0	0	0	0	0	0	0
0	0	0	1	+1	0	0	0	0	1
0	0	1	0	+2	0	0	0	1	0
0	0	1	1	+3	0	0	0	1	1
0	1	0	0	+4	0	0	1	0	0
0	1	0	1	+5	0	0	1	0	1
0	1	1	0	+6	0	0	1	1	0
0	1	1	1	+7	0	0	1	1	1
1	0	0	0	-8	1	1	0	0	0
1	0	0	1	-7	1	0	1	1	1
1	0	1	0	-6	1	0	1	1	0
1	0	1	1	-5	1	0	1	0	1
1	1	0	0	-4	1	0	1	0	0
1	1	0	1	-3	1	0	0	1	1
1	1	1	0	-2	1	0	0	1	0
1	1	1	1	-1	1	0	0	0	1

Β. Αρχικά, παρατηρούμε ότι $M_4=C_3$, $M_3=C_3C_2'C_1'C_0'= C_3(C_0+C_1+C_2)'$ και $M_0=C_0$.

Στη συνέχεια, εφαρμόζοντας τη μέθοδο του χάρτη Karnaugh, βρίσκουμε τις απλοποιημένες εκφράσεις των M_2 και M_1, σε μορφή αθροίσματος γινομένων και τις υλοποιούμε με τρία επίπεδα NOT-AND-OR (ή ισοδύναμα, αντικαθιστώντας τις πύλες NOT, AND και OR με πύλες NAND):

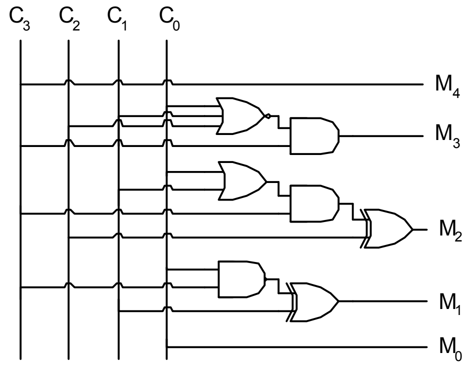


$\mathbf{M_2}$=$C_2C_3'+C_0'C_1'C_2+C_0C_2'C_3+C_1C_2'$	$\mathbf{M_1}$ = $C_0'C_1+C_1C_3'+C_0C_1'C_3$

Στη συνέχεια ξαναγράφουμε ισοδύναμα τις παραπάνω συναρτήσεις, ώστε να μπορούν να υλοποιηθούν με λιγότερες πύλες, μικρότερου αριθμού εισόδων:

$$M_2 = C_2C_3'+C_0'C_1'C_2+C_0C_2'C_3+C_1C_2'C_3 = C_2(C_3'+C_0'C_1') + C_2'C_3(C_0+C_1) =$$
$$= C_2[C_3(C_0+C_1)]' + C_2'[(C_0+C_1)C_3] = C_2 \oplus [C_3(C_0+C_1)].$$

$$M_1 = C_0'C_1+C_1C_3'+C_0C_1'C_3 = C_1(C_0'+C_3') + C_1'C_0C_3 = C_1(C_0C_3)' + C_1'(C_0C_3) = C_1 \oplus (C_0C_3).$$



ΑΣΚΗΣΗ 49

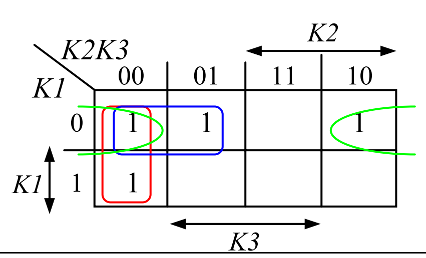
Στην άρση βαρών μια προσπάθεια ενός αθλητή θεωρείται έγκυρη όταν τη δέχονται τουλάχιστον δύο από τους τρεις κριτές (Κ1,Κ2,Κ3) και άκυρη διαφορετικά. Σχεδιάστε συνδυαστικό κύκλωμα, χρησιμοποιώντας μόνο πύλες NOR, το οποίο να δίνει έξοδο (X) ίση με λογικό '1' όταν μια προσπάθεια θεωρείται <u>άκυρη</u>.
Συμβολίστε με λογικό '1' την αποδοχή της προσπάθειας από έναν κριτή και με λογικό '0' το αντίθετο.

Λύση

Είναι προφανές ότι στο συνδυαστικό κύκλωμα οι είσοδοι αντιστοιχούν στους τρεις κριτές (Κ1,Κ2,Κ3) ενώ η μοναδική έξοδος (X) δίνει λογικό '1' όταν μια προσπάθεια θεωρείται <u>άκυρη</u>. Λαμβάνοντας υπόψη τα δεδομένα της εκφώνησης ο πίνακας αληθείας συμπληρώνεται όπως φαίνεται παρακάτω:

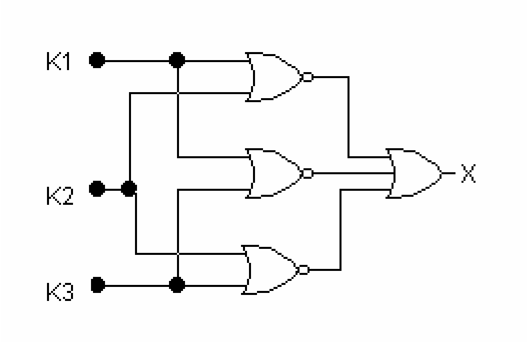
K1	K2	K3	X
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	0

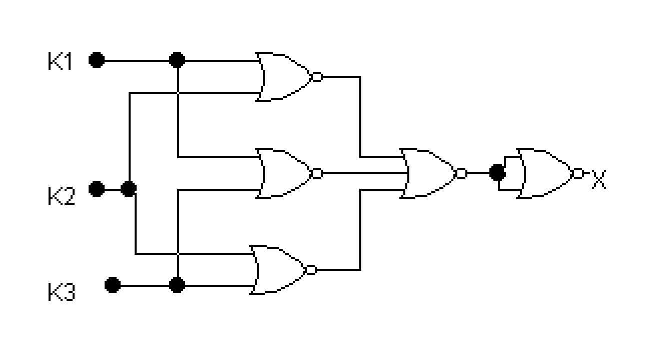
Χρειαζόμαστε 1 χάρτη Karnaugh τριών μεταβλητών για να απλοποιήσουμε την έξοδο X.



X = K2'K3' + K1'K2' + K1'K3'

Πρέπει να μετασχηματίσουμε την απλοποιημένη συνάρτηση ώστε να προκύψουν όροι που να οδηγούν στη σχεδίαση με πύλες NOR. Αρχικά, παρατηρούμε ότι εφαρμόζοντας τον νόμο DeMorgan, σε κάθε ένα από τα τρία γινόμενα, προκύπτουν όροι που οδηγούν στη σχεδίαση με πύλες NOR. Έτσι η συνάρτηση εξόδου γίνεται: X = (K2+K3)' + (K1+K2)' + (K1+K3)' και σχεδιάζουμε το κύκλωμα που φαίνεται αριστερά. Αντικαθιστώντας την πύλη OR 3-εισόδων με NOR, για να διατηρηθεί η ισοδυναμία του κυκλώματος, χρειάζεται να προσθέσουμε στην έξοδο μία επιπλέον πύλη NOR, συνδεμένη ως NOT, όπως φαίνεται στο κύκλωμα δεξιά το οποίο αποτελείται μόνο από πύλες NOR.





ΑΣΚΗΣΗ 50

Ένα συνδυαστικό κύκλωμα δέχεται στην είσοδο ένα τετραψήφιο δυαδικό αριθμό ($X=X_3X_2X_1X_0$) και δίνει στην έξοδο έναν δυαδικό αριθμό (Y) ίσο με Y=X-10. Π.χ. για τον αριθμό $X=12_{10}=1100_2$, η έξοδος πρέπει να δίνει $Y=X-10=12-10=2_{10}$.
Όταν η έξοδος έχει αρνητική τιμή δεν μας ενδιαφέρει να εμφανίζεται η σωστή ένδειξη.
Α. Καθορίστε τις εισόδους και τις εξόδους του κυκλώματος και κατασκευάστε τον πίνακα αλήθειας.
Β. Απλοποιείστε τις συναρτήσεις εξόδου του κυκλώματος χρησιμοποιώντας χάρτες Karnaugh.
Γ. Υλοποιήστε το κύκλωμα χρησιμοποιώντας μόνο πύλες NAND.
Δ. Βρείτε τις απλοποιημένες συναρτήσεις εξόδου που προκύπτουν χωρίς την εκμετάλλευση των αδιάφορων όρων.

Λύση
Α. Η μεγαλύτερη δεκαδική τιμή της εξόδου προκύπτει για την μεγαλύτερη δεκαδική τιμή της εισόδου $X=15_{10}$ και είναι ίση με: $Y=X-10=15-10=5_{10}=101_2$. Άρα η μεγαλύτερη δυαδική έξοδος απαιτεί τρία

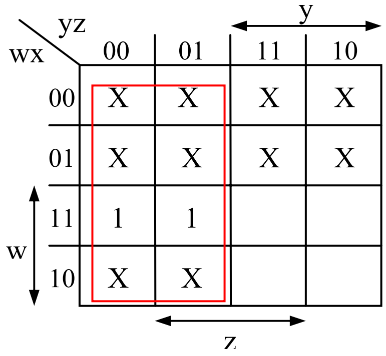
ψηφία. Έτσι το κύκλωμα αποτελείται από 4 εισόδους ($X=X_3X_2X_1X_0$), μία για κάθε δυαδικό ψηφίο εισόδου και 3 εξόδους ($Y=Y_2Y_1Y_0$) μία για κάθε δυαδικό ψηφίο εξόδου.
Επειδή, όταν η έξοδος έχει αρνητική τιμή δεν μας ενδιαφέρει να εμφανίζεται η σωστή ένδειξη, στον πίνακα αλήθειας θα εμφανίζονται αδιάφοροι όροι (X) στις αντίστοιχες θέσεις. Με βάση τα παραπάνω ο πίνακας αλήθειας είναι ο ακόλουθος:

Δυαδική Είσοδος				**Δεκαδικοί Αριθμοί**		**Δυαδική Έξοδος**		
X_3	**X_2**	**X_1**	**X_0**	**X**	**Y**	**Y_2**	**Y_1**	**Y_0**
0	0	0	0	0	-10	X	X	X
0	0	0	1	1	-9	X	X	X
0	0	1	0	2	-8	X	X	X
0	0	1	1	3	-7	X	X	X
0	1	0	0	4	-6	X	X	X
0	1	0	1	5	-5	X	X	X
0	1	1	0	6	-4	X	X	X
0	1	1	1	7	-3	X	X	X
1	0	0	0	8	-2	X	X	X
1	0	0	1	9	-1	X	X	X
1	0	1	0	10	0	0	0	0
1	0	1	1	11	1	0	0	1
1	1	0	0	12	2	0	1	0
1	1	0	1	13	3	0	1	1
1	1	1	0	14	4	1	0	0
1	1	1	1	15	5	1	0	1

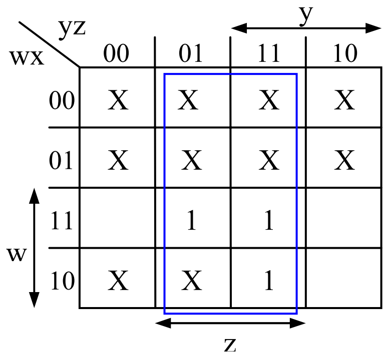
B. Σχηματίζουμε τους χάρτες Karnaugh, προσέχοντας την σωστή χρήση των αδιάφορων όρων στην εύρεση των απλοποιημένων συναρτήσεων.
Για λόγους ομοιομορφίας στους χάρτες Karnaugh 4 μεταβλητών, διατηρούμε την ονομασία των μεταβλητών (w,x,y,z) σε αντιστοιχία με τις μεταβλητές ($X_3X_2X_1X_0$).

wx \ yz	00	01	11	10
00	X	X	X	X
01	X	X	X	X
11			1	1
10	X	X		

$Y_2(w,x,y,z) = x\,y$

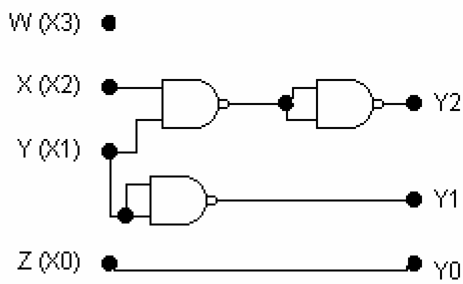


$Y_1(w,x,y,z) = y'$



$Y_0(w,x,y,z) = z$

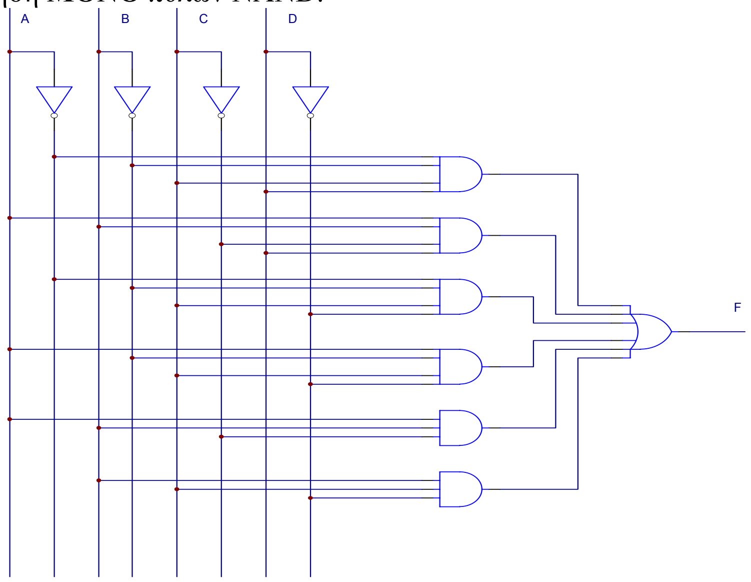
Γ. Για την υλοποίηση του κυκλώματος μόνο με πύλες NAND οι συναρτήσεις των εξόδων πρέπει να τροποποιηθούν ως εξής:
$Y_2 = xy = [(xy)']' = [(xy)'(xy)']'$
$Y_1 = y' = (yy)'$
$Y_0 = z$
Η υλοποίηση φαίνεται στο παρακάτω σχήμα:



Δ. Με βάση τους χάρτες Karnaugh του ερωτήματος Β και χωρίς την εκμετάλλευση των αδιάφορων όρων προκύπτουν οι παρακάτω απλοποιημένες συναρτήσεις των εξόδων:
$Y_2 = wxy$
$Y_1 = wxy'$
$Y_0 = wxz + wyz$

ΑΣΚΗΣΗ 51

Να βρεθεί η απλοποιημένη λογική συνάρτηση του παρακάτω κυκλώματος και να υλοποιηθεί με τη χρήση **ΜΟΝΟ** πυλών NAND.



Λύση:
Από το παραπάνω κύκλωμα κύκλωμα προκύπτει η λογική εξίσωση:

F = A'B'CD + ABC'D + A'B'CD' + AB'CD' + ABC' + BCD'

Η τοποθέτηση των όρων σε πίνακα Karnaugh είναι η παρακάτω:

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

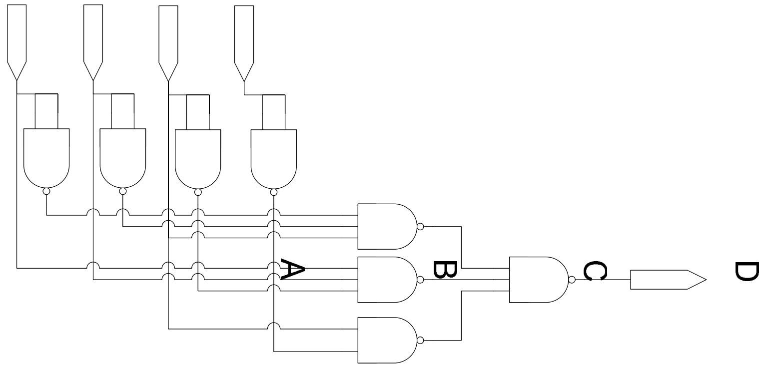
Από τη παραπάνω ομαδοποίηση προκύπτει η λογική εξίσωση:

F = A'B'C + ABC' + CD'

Με την εφαρμογή του θεωρήματος DeMorgan προκύπτει η παρακάτω μορφή που περιέχει μόνο όρους NAND:

F = (A'B'C + ABC' + CD')'' = ((A'B'C)' (ABC')' (CD')')'

και τελικά προκύπτει το παρακάτω κύκλωμα:



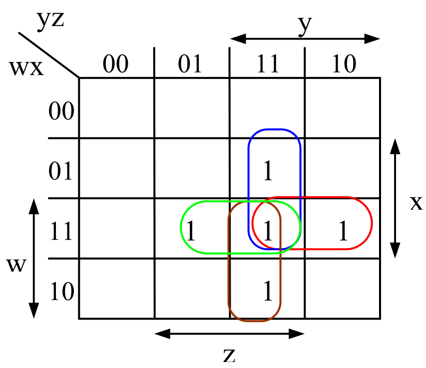
ΑΣΚΗΣΗ 52

Κύκλωμα ‘πλειοψηφίας’ τεσσάρων εισόδων λέγεται το κύκλωμα το οποίο δέχεται στην είσοδο τέσσερα δυαδικά ψηφία (w,x,y,z) και δίνει στην έξοδο E λογικό ‘1’ όταν στις εισόδους το πλήθος των λογικών ‘1’ είναι μεγαλύτερο από το πλήθος των λογικών ‘0’.
A. Κατασκευάστε τον πίνακα αλήθειας του κυκλώματος.
B. Απλοποιείστε τη συνάρτηση εξόδου του κυκλώματος χρησιμοποιώντας χάρτη Karnaugh.
Γ. Στην περίπτωση που δεν μας ενδιαφέρει το αποτέλεσμα της εξόδου όταν το πλήθος των λογικών ‘1’ είναι ίσο με το πλήθος των λογικών ‘0’, υλοποιήστε το κύκλωμα χρησιμοποιώντας μόνο πύλες NAND.

Λύση
A. Το κύκλωμα αποτελείται από τέσσερις εισόδους (w,x,y,z) και μία έξοδο E. Ο πίνακας αλήθειας έχει 16 γραμμές και είναι ο ακόλουθος:

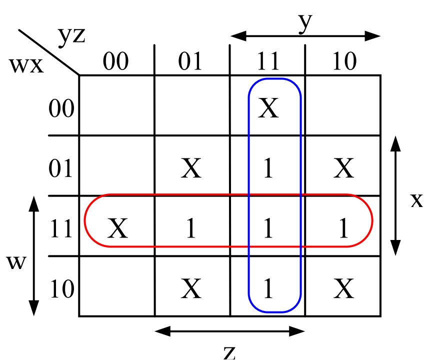
w	x	y	z	E
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	0
0	1	0	1	0
0	1	1	0	0
0	1	1	1	1
1	0	0	0	0
1	0	0	1	0
1	0	1	0	0
1	0	1	1	1
1	1	0	0	0
1	1	0	1	1
1	1	1	0	1
1	1	1	1	1

B. Ο χάρτης Karnaugh των τεσσάρων μεταβλητών φαίνεται παρακάτω. Οι γειτονιές ελαχιστόρων φαίνονται σε διάφορα χρώματα. Η συνάρτηση εξόδου και τα γινόμενα που δίνουν φαίνονται με αντίστοιχα χρώματα στο πλαίσιο.



E = $wxy + xyz + wxz + wyz$

Γ. Αφού δεν μας ενδιαφέρει το αποτέλεσμα της εξόδου όταν το πλήθος των λογικών ‘1’ είναι ίσο με το πλήθος των λογικών ‘0’ σε αυτές τις περιπτώσεις προκύπτουν αδιάφοροι όροι. Ο χάρτης Karnaugh των τεσσάρων μεταβλητών με τους αδιάφορους όρους φαίνεται παρακάτω. Οι γειτονιές ελαχιστόρων με την εκμετάλλευση των αδιάφορων όρων φαίνονται σε διάφορα χρώματα. Η συνάρτηση εξόδου και τα γινόμενα που δίνουν φαίνονται με αντίστοιχα χρώματα στο πλαίσιο.

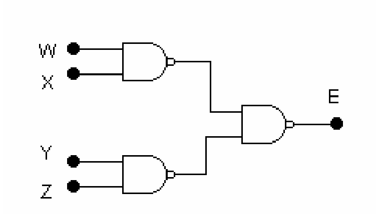


E = $wx + yz$

Για την υλοποίηση του κυκλώματος μόνο με πύλες NAND η συνάρτηση εξόδου πρέπει να τροποποιηθεί ως εξής:

$E = wx + yz = [(wx + yz)']' = [(wx)'(yz)']'$

Η υλοποίηση φαίνεται στο παρακάτω σχήμα:



ΑΣΚΗΣΗ 53

Κύκλωμα ‘μειοψηφίας’ πέντε εισόδων λέγεται το κύκλωμα το οποίο δέχεται στην είσοδο πέντε δυαδικά ψηφία (v,w,x,y,z) και δίνει στην έξοδο E λογικό ‘1’ όταν στις εισόδους το πλήθος των λογικών ‘1’ είναι μικρότερο από το πλήθος των λογικών ‘0’.

Α. Κατασκευάστε τον πίνακα αλήθειας του κυκλώματος.

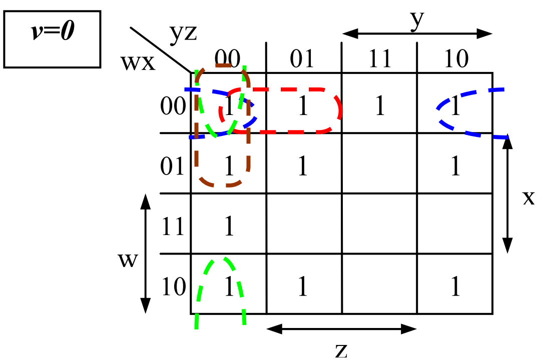
Β. Απλοποιείστε τη συνάρτηση εξόδου του κυκλώματος χρησιμοποιώντας χάρτες Karnaugh.

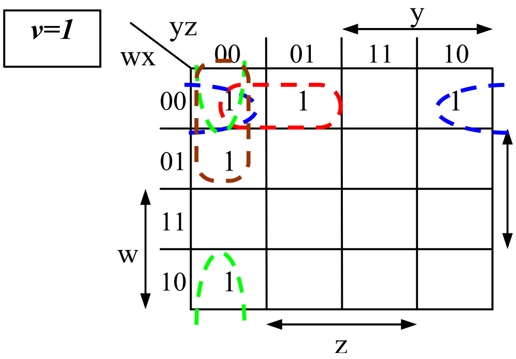
Λύση

Α. Το κύκλωμα αποτελείται από πέντε εισόδους (v,w,x,y,z) και μία έξοδο Ε. Ο πίνακας αλήθειας έχει 32 γραμμές και είναι ο ακόλουθος:

v	**w**	**x**	**y**	**z**	**E**
0	0	0	0	0	1
0	0	0	0	1	1
0	0	0	1	0	1
0	0	0	1	1	1
0	0	1	0	0	1
0	0	1	0	1	1
0	0	1	1	0	1
0	0	1	1	1	0
0	1	0	0	0	1
0	1	0	0	1	1
0	1	0	1	0	1
0	1	0	1	1	0
0	1	1	0	0	1
0	1	1	0	1	0
0	1	1	1	0	0
0	1	1	1	1	0
1	0	0	0	0	1
1	0	0	0	1	1
1	0	0	1	0	1
1	0	0	1	1	0
1	0	1	0	0	1
1	0	1	0	1	0
1	0	1	1	0	0
1	0	1	1	1	0
1	1	0	0	0	1
1	1	0	0	1	0
1	1	0	1	0	0
1	1	0	1	1	0
1	1	1	0	0	0
1	1	1	0	1	0
1	1	1	1	0	0
1	1	1	1	1	0

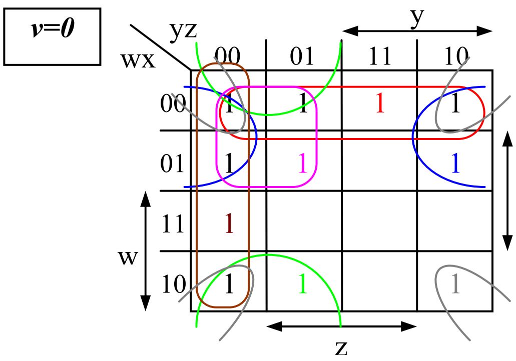
Β. Αφού έχουμε πέντε εισόδους πρέπει να σχηματίσουμε δύο χάρτες Karnaugh των τεσσάρων μεταβλητών, ένα χάρτη για v=0 και ένα χάρτη για v=1, προσέχοντας ιδιαίτερα την σωστή τοποθέτηση των ελαχιστόρων. Οι τέσσερις γειτονιές ελαχιστόρων που εμφανίζονται στις ίδιες θέσεις και στους δύο χάρτες (για v=0 και για v=1) φαίνονται παρακάτω με διακεκομμένες γραμμές σε διάφορα χρώματα. Τα γινόμενα που δίνουν φαίνονται με αντίστοιχα χρώματα στο πλαίσιο που ακολουθεί.





$w'x'y' + w'y'z' + w'x'z' + x'y'z'$

Όπως φαίνεται καλύπτονται όλοι οι ελαχιστόροι (λογικά ‘1’) του κάτω χάρτη (για v=1), ενώ στον επάνω χάρτη (για v=0) υπάρχουν έξι ελαχιστόροι που δεν έχουν καλυφθεί. Οι έξι αυτοί ελαχιστόροι δεν είναι γειτονικοί και άρα δεν μπορούν να συνδυαστούν με αντίστοιχους ελαχιστόρους του κάτω χάρτη. Άρα πρέπει να συνδυαστούν με γειτονικούς ελαχιστόρους που βρίσκονται μόνο στον επάνω χάρτη. Παρακάτω φαίνεται ο επάνω χάρτης, χωρίς να απεικονίζονται οι τέσσερις γειτονιές που έχουν ήδη καλυφθεί, ώστε να είναι κατά το δυνατόν ευδιάκριτες οι γειτονιές ελαχιστόρων που προσθέτονται. Οι γειτονιές ελαχιστόρων που προσθέτονται φαίνονται παρακάτω με συνεχόμενες γραμμές σε διάφορα χρώματα. Κάθε ελαχιστόρος που καλύπτεται μόνο από μία γειτονιά φαίνεται με το ίδιο χρώμα στο χάρτη. Τα γινόμενα που δίνει κάθε γειτονιά φαίνονται με αντίστοιχα χρώματα στο πλαίσιο που ακολουθεί.



$v'w'x' + v'y'z' + v'w'z' + v'x'y' + v'w'y' + v'x'z'$

Έτσι, τελικά η συνάρτηση της εξόδου E ισούται με:

$$E = w'x'y'+w'y'z'+w'x'z'+x'y'z'+v'w'x'+v'y'z'+v'w'z'+v'x'y'+v'w'y'+v'x'z'$$

ΑΣΚΗΣΗ 54

Έστω κύκλωμα το οποίο δέχεται στην είσοδο τέσσερα δυαδικά ψηφία (w,x,y,z) και δίνει στην έξοδο E λογικό ‘1’ όταν το πλήθος των λογικών ‘0’ των τεσσάρων δυαδικών ψηφίων είναι μεγαλύτερο ή ίσο από 2.

Α. Κατασκευάστε τον πίνακα αλήθειας του κυκλώματος.

Β. Βρείτε την απλοποιημένη συνάρτηση εξόδου χρησιμοποιώντας χάρτη Karnaugh.

Γ. Τροποποιείστε την απλοποιημένη συνάρτηση εξόδου ώστε το κύκλωμα να υλοποιείται χρησιμοποιώντας το πολύ **7 πύλες** των δύο εισόδων η καθεμία και όσες πύλες NOT χρειάζεστε.

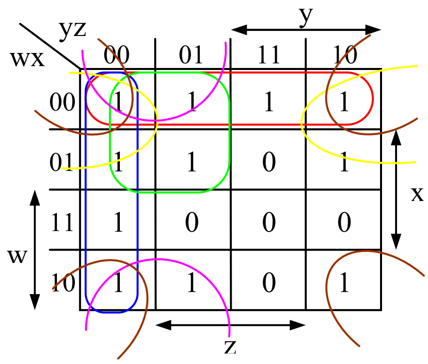
Δ. Στην περίπτωση που δεν μας ενδιαφέρει το αποτέλεσμα της εξόδου όταν το πλήθος των λογικών '1' είναι ίσο με το πλήθος των λογικών '0', υλοποιήστε το κύκλωμα χρησιμοποιώντας μόνο πύλες NAND.

Λύση

Α. Το κύκλωμα αποτελείται από τέσσερις εισόδους (w,x,y,z) και μία έξοδο Ε. Ο πίνακας αλήθειας έχει 16 γραμμές και είναι ο ακόλουθος:

w	**x**	**y**	**z**	**E**
0	0	0	0	1
0	0	0	1	1
0	0	1	0	1
0	0	1	1	1
0	1	0	0	1
0	1	0	1	1
0	1	1	0	1
0	1	1	1	0
1	0	0	0	1
1	0	0	1	1
1	0	1	0	1
1	0	1	1	0
1	1	0	0	1
1	1	0	1	0
1	1	1	0	0
1	1	1	1	0

Β. Ο χάρτης Karnaugh των τεσσάρων μεταβλητών φαίνεται παρακάτω. Οι γειτονιές ελαχιστόρων φαίνονται σε διάφορα χρώματα. Η συνάρτηση εξόδου και τα γινόμενα που δίνουν φαίνονται με αντίστοιχα χρώματα στο πλαίσιο.

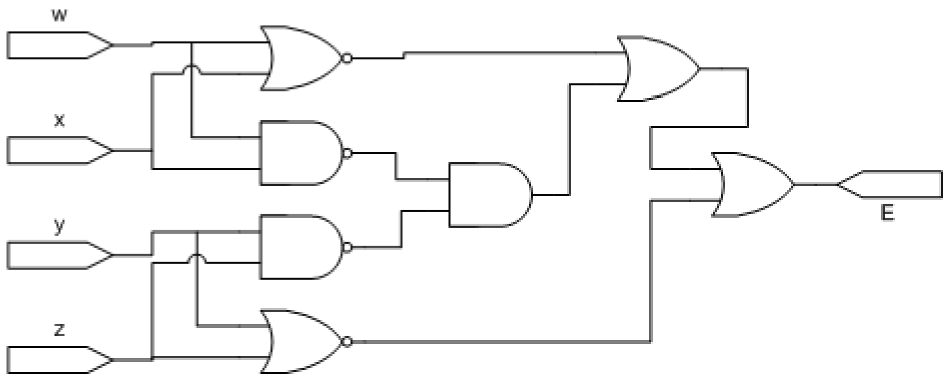


E = $w'x' + w'y' + w'z' + x'y' + x'z' + y'z'$

Γ. Για την υλοποίηση του κυκλώματος με τον μικρότερο δυνατό αριθμό πυλών δύο εισόδων η συνάρτηση εξόδου μπορεί να τροποποιηθεί ως εξής:

$$
\begin{aligned}
E &= w'x' + w'y' + w'z' + x'y' + x'z' + y'z' \\
&= w'x' + w'(y' + z') + x'(y' + z') + y'z' \\
&= w'x' + (w' + x')(y' + z') + y'z' \\
&= (w+x)' + (w\,x)'(y\,z)' + (y+z)'
\end{aligned}
$$

Η υλοποίηση φαίνεται στο παρακάτω σχήμα:



Δ. Αφού δεν μας ενδιαφέρει το αποτέλεσμα της εξόδου όταν το πλήθος των λογικών ‘1’ είναι ίσο με το πλήθος των λογικών ‘0’ σε αυτές τις περιπτώσεις προκύπτουν αδιάφοροι όροι. Ο χάρτης Karnaugh των τεσσάρων μεταβλητών με τους αδιάφορους όρους για την αντίστροφη συνάρτηση Ε’ φαίνεται παρακάτω. Οι γειτονιές ελαχιστόρων με την εκμετάλλευση των αδιάφορων όρων φαίνονται σε διάφορα χρώματα. Η συνάρτηση εξόδου Ε’ και τα γινόμενα που δίνουν φαίνονται με αντίστοιχα χρώματα στο πλαίσιο.

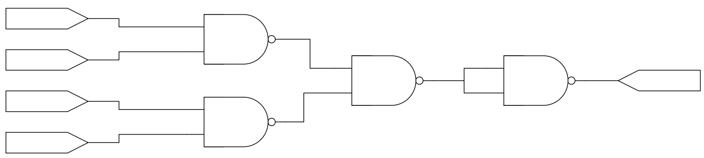
wx \ yz	00	01	11	10
00			X	
01		X	1	X
11	X	1	1	1
10		X	1	X

E’ = $wx + yz$

Για την υλοποίηση του κυκλώματος μόνο με πύλες NAND η συνάρτηση εξόδου πρέπει να τροποποιηθεί ως εξής:

$$E = (E')' = (wx + yz)' = (wx)'(yz)' = [(wx)'(yz)']'' = \big[\, [(wx)'(yz)']' \cdot [(wx)'(yz)']' \,\big]'$$

Η υλοποίηση με τέσσερις πύλες NAND φαίνεται στο παρακάτω σχήμα:



Παρατηρείστε ότι η προσπάθεια υλοποίησης του κυκλώματος μόνο με πύλες NAND ξεκινώντας από τη συνάρτηση Ε και όχι την Ε’ οδηγεί σε κύκλωμα με περισσότερες από τέσσερις πύλες NAND.

ΑΣΚΗΣΗ 55

Έχετε στην διάθεσή σας **5 πύλες** των δύο εισόδων η καθεμία και όσες πύλες NOT χρειάζεστε. Σχεδιάστε κύκλωμα το οποίο να δέχεται ως είσοδο δύο ακέραιους αριθμούς **Α** και **Β** σε παράσταση **συμπληρώματος του 2,** των **δύο δυαδικών** ψηφίων ο καθένας, και να έχει μία έξοδο L τέτοια ώστε: L=**1 αν Α<Β**, και L=**0 διαφορετικά**.

Λύση

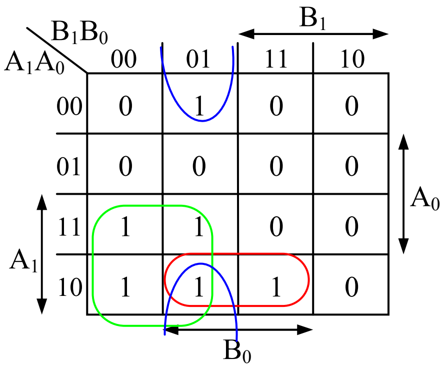
Έστω $A=A_1A_0$ και $B=B_1B_0$. Τότε, δεδομένου ότι σύμφωνα με την αναπαράσταση συμπληρώματος του δύο ισχύει ότι

10	-2
11	-1
00	0
01	1

κατασκευάζουμε τον ακόλουθο πίνακα αληθείας

A_1	A_0	B_1	B_0	L
0	0	0	0	0
0	0	0	1	1
0	0	1	0	0
0	0	1	1	0
0	1	0	0	0
0	1	0	1	0
0	1	1	0	0
0	1	1	1	0
1	0	0	0	1
1	0	0	1	1
1	0	1	0	0
1	0	1	1	1
1	1	0	0	1
1	1	0	1	1
1	1	1	0	0
1	1	1	1	0

Ο χάρτης Karnaugh των τεσσάρων μεταβλητών φαίνεται παρακάτω. Οι γειτονιές ελαχιστόρων φαίνονται σε διάφορα χρώματα. Η συνάρτηση εξόδου και τα γινόμενα που δίνουν φαίνονται με αντίστοιχα χρώματα στο πλαίσιο.



$$\mathbf{E} = A_1B_1' + A_1A_0'B_0 + A_0'B_1'B_0$$

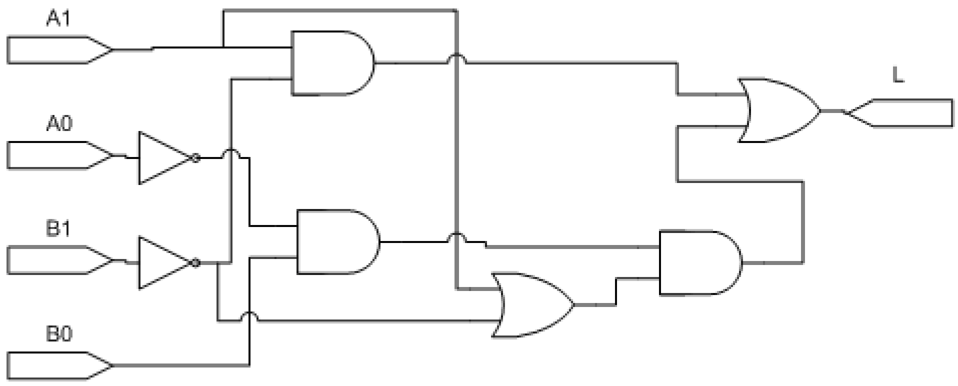
Για την υλοποίηση του κυκλώματος με την παραπάνω συνάρτηση εξόδου απαιτούνται 7 πύλες δύο εισόδων και 2 πύλες NOT. Αν η συνάρτηση εξόδου τροποποιηθεί ως εξής:

$\mathbf{E} = A_1B_1' + A_1A_0'B_0 + A_0'B_1'B_0 =$

$= A_1B_1' + A_0'B_0\,(A_1 + B_1')$

απαιτούνται 5 πύλες δύο εισόδων και 2 πύλες NOT.

Η υλοποίηση φαίνεται στο παρακάτω σχήμα:

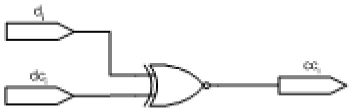


ΑΣΚΗΣΗ 56

Θεωρούμε τις λέξεις των τριών δυαδικών ψηφίων $d_2d_1d_0$ και τη κωδικοποίηση αυτών ως $d_2dc_2d_1dc_1d_0dc_0p$ όπου dc_i συμβολίζει το συμπληρωματικό δυαδικό ψηφίο του d_i, και p το ψηφίο άρτιας ισοτιμίας της $d_2d_1d_0$. Έτσι, για παράδειγμα, η λέξη δεδομένων 110 κωδικοποιείται σαν 1010010. Να σχεδιαστεί κύκλωμα το οποίο να **διορθώνει λάθος** σε λέξη η οποία ακολουθεί την κωδικοποίηση αυτή, θεωρώντας ότι μπορεί να συμβεί λάθος σε ένα μόνο (αλλά οποιοδήποτε) ψηφίο.

Λύση

Λάθος μπορεί να συμβεί στο ψηφίο ισοτιμίας p ή σε ψηφία των ζευγών (ψηφίο δεδομένων d_i, σύμπληρωμά του dc_i). Παρατηρούμε ότι τα ζεύγη ψηφίων θα πρέπει να έχουν πάντα δύο διαφορετικές τιμές. Επειδή μπορεί να συμβεί μόνο ένα λάθος, αρκεί να ελέγξουμε ότι όλα τα ζεύγη περιέχουν διαφορετικά ψηφία. Άρα λοιπόν για το ζεύγος i παράγουμε το σήμα $cc_i = (d_i \oplus dc_i)'$, το οποίο θα πρέπει να είναι μηδέν αν το ζεύγος δεν περιέχει λάθος και παράγεται ως εξής:

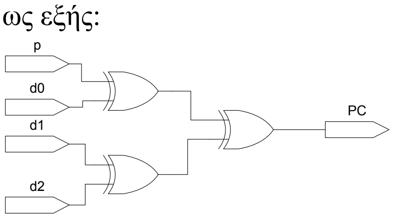


Μια λογική έκφραση του σήματος προκύπτει από τον ακόλουθο πίνακα αληθείας:

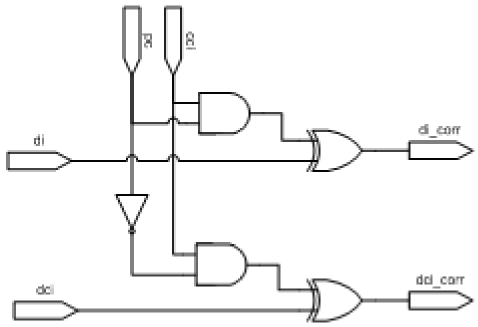
d_i	dc_i	cc_i
0	0	1
0	1	0
1	0	0
1	1	1

Από όπου $cc_i = d_i' dc_i' + d_i dc_i = (d_i \oplus dc_i)'$.

Σε περίπτωση που εντοπιστεί λάθος, για να διευκρινίσουμε ποιο από τα δύο ψηφία του ζεύγους απαιτεί διόρθωση, χρησιμοποιούμε έλεγχο ψηφίου ισοτιμίας. Αν έχει παραβιαστεί η ισοτιμία τότε το d_i απαιτεί διόρθωση, διαφορετικά το dc_i. Συνεπώς κατασκευάζουμε και ένα σήμα ελέγχου ισοτιμίας ως εξής:

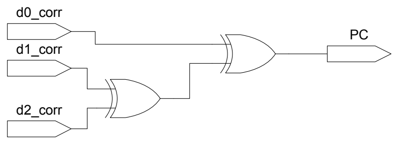


Τελικά το κύκλωμα που διορθώνει το ζεύγος i είναι το ακόλουθο:



όπου το σήμα cc_i καθορίζει αν θα αλλάξει τιμές κάποιο από τα ψηφία του ζεύγους (αν είναι μηδέν, τότε οι έξοδοι των πυλών AND είναι μηδέν, δηλ. η μία είσοδος κάθε XOR είναι μηδέν, άρα οι XOR δεν αλλάζουν τις τιμές των άλλων εισόδων τους). Όταν το PC έχει την τιμή ένα, αλλάζει η τιμή του d_i, διαφορετικά αλλάζει η τιμή του dc_i. Παρατηρείστε ότι δεν έχει σημασία αν το PC είναι σωστό ή όχι, δεδομένου ότι μπορεί να συμβεί μόνο ένα λάθος. Άρα αν το PC έχει λάθος τιμή λόγω λάθους στο ψηφίο ισοτιμίας, τότε η έξοδος δεν επηρεάζεται γιατί τα cci θα έχουν τιμές μηδέν.

Τέλος μια πάντα σωστή τιμή του ψηφίου ισοτιμίας προκύπτει υπολογίζοντάς το χρησιμοποιώντας τις διορθωμένες τιμές των ψηφίων δεδομένων:



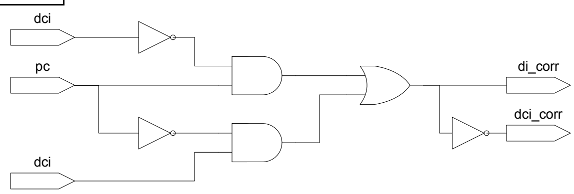
Άλλος τρόπος:

Είναι δυνατό να μην υπολογιστούν αναλυτικά τα σήματα cc_i. Θεωρώντας ότι έχουμε το σήμα pc όπως νωρίτερα, και παρατηρώντας ότι το σήμα dci_corr προκύπτει ως αντίθετο του di_corr, μπορούμε να φτιάξουμε πίνακα αληθείας, ο οποίος δίνει απευθείας το di_corr:

d_i	dc_i	pc	d_i_corr
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	0

d_i \ $dc_i pc$	00	01	11	10
0		1		
1	1	1		1

Από την απλοποίηση με το χάρτη Karnaugh προκύπτει ότι $d_i\_corr = dc_i' pc + d_i pc'$, το οποίο υλοποιείται ως ακολούθως:



ΑΣΚΗΣΗ 57

Θεωρείστε ότι σε έναν δυαδικό κώδικα είναι επιτρεπτές μόνο οι λέξεις οι οποίες δεν έχουν σε συνεχόμενες θέσεις ψηφία με τιμή ‘1’.

Α) Κατασκευάστε τον πίνακα αληθείας λογικής συνάρτησης G η οποία λαμβάνει την τιμή 1 αν μια λέξη μήκους τεσσάρων δυαδικών ψηφίων είναι επιτρεπτή σύμφωνα με τον κώδικα αυτό και την τιμή 0 αν δεν είναι.
Β) Απλοποιήστε τη συνάρτηση G και σχεδιάστε το αντίστοιχο λογικό κύκλωμα. Εξηγήστε.
Γ) Σχεδιάστε το λογικό κύκλωμα της συνάρτησης G χρησιμοποιώντας τον ελάχιστο δυνατό αριθμό πυλών. Εξηγήστε.

Λύση

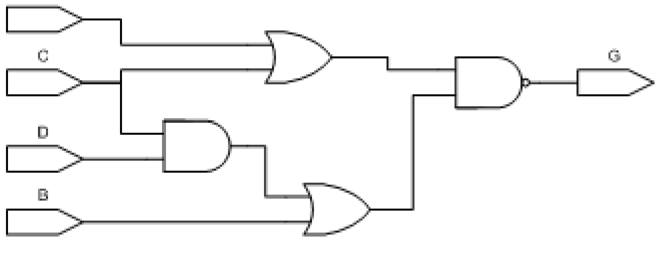
Α) Ο πίνακας αληθείας

	ABCD	G
0	0000	1
1	0001	1
2	0010	1
3	0011	0
4	0100	1
5	0101	1
6	0110	0
7	0111	0
8	1000	1
9	1001	1
10	1010	1
11	1011	0
12	1100	0
13	1101	0
14	1110	0
15	1111	0

Γ) Η σχέση $G = A'C' + B'D' + B'C'$ χρησιμοποιώντας DeMorgan γράφεται ως

$$
\begin{aligned}
G &= A'C' + B'(D' + C') \\
&= A'C' + B'(DC)' \\
&= (A+C)' + (B+DC)' \\
&= ((A+C)(B+DC))'
\end{aligned}
$$

Η υλοποίηση αυτής της έκφρασης με πύλες, έχει ως εξής:



ΑΣΚΗΣΗ 58

A) Απλοποιήστε τη λογική συνάρτηση F η οποία δίνεται ως το κανονικό άθροισμα ελαχιστόρων **F(A,B,C,D) =Σ(1,3,5,7,9,11,12,14)**.

B) Υλοποιήστε την F με **δύο μόνον πύλες** των **δύο εισόδων** η καθεμία.

<u>Απάντηση</u>

A) Ο πίνακας αληθείας οργανώνεται ως ακολούθως

	ABCD	F
0	0000	0
1	0001	1
2	0010	0
3	0011	1
4	0100	0
5	0101	1
6	0110	0
7	0111	1
8	1000	0
9	1001	1
10	1010	0
11	1011	1
12	1100	1
13	1101	0
14	1110	1
15	1111	0

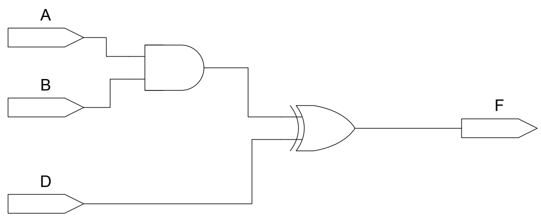
B) Ο αντίστοιχος χάρτης Karnaugh έχει ως εξής

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

$$
\begin{aligned}
F &= A'D + B'D + ABD' \\
&= (A'+B')D + ABD' \\
&= (AB)'D + (AB)D' \\
&= (AB) \oplus D
\end{aligned}
$$

(με χρήση De Morgan)

Η υλοποίηση της συνάρτησης *F* με δύο λογικές πύλες έχει ως εξής:



ΑΣΚΗΣΗ 59

Έστω λογικό κύκλωμα στο οποίο είναι δυνατόν να εμφανιστούν ως είσοδοι **μόνο οι ακέραιοι από 1 έως και 9**, κωδικοποιημένοι κατά 8421. Το κύκλωμα αυτό παράγει μια έξοδο F η οποία είναι 1 αν ο ακέραιος αριθμός της εισόδου διαιρείται ακριβώς με το 3, και μηδέν διαφορετικά. Να υλοποιηθεί το κύκλωμα με 4 πύλες των 2 εισόδων η καθεμία.

Λύση

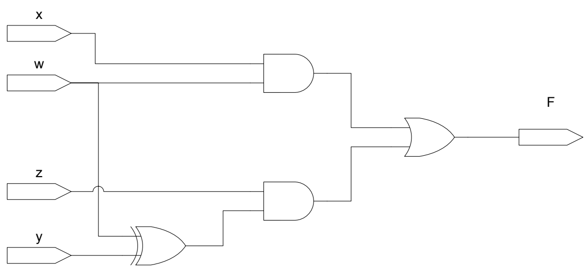
Δεδομένου ότι ως είσοδοι είναι οι αριθμοί 1 έως και 9, απαιτούνται τέσσερα δυαδικά ψηφία ως είσοδοι. Κατασκευάζουμε τον πίνακα αληθείας της συνάρτησης η οποία υλοποιείται από το κύκλωμα αυτό. Παρατηρούμε ότι επειδή οι τιμές εισόδου 0 και 10, 11, 12, 13, 14, 15 δεν είναι δυνατόν να συμβούν, μπορούν να χρησιμοποιηθούν ως αδιάφοροι όροι. Ο πίνακας αληθείας έχει ως εξής:

xyzw	**F**
0000	X
0001	0
0010	0
0011	1
0100	0
0101	0
0110	1
0111	0
1000	0
1001	1
1010	X
1011	X
1100	X
1101	X
1110	X
1111	X

Στη συνέχεια με βάση τον πίνακα αληθείας κατασκευάζουμε το χάρτη Karnaugh

xy \ zw	00	01	11	10
00	X		1	
01				1
11	X	X	X	X
10		1	X	X

από τον οποίο προκύπτει ότι η συνάρτηση F απλοποιείται ως F = xw + yzw' + y'zw = xw + z(yw'+y'w) = x w + z (y ⊕ w). Η σχετική υλοποίηση έχει ως εξής:



ΑΣΚΗΣΗ 60

Έστω ψηφιακό σύστημα τεσσάρων εισόδων x,y,z,w το οποίο δίνει στην μια και μοναδική έξοδο του "1" εαν ο αριθμός που σχηματίζεται είναι "πρώτος", διαφορετικά δίνει "0". Να σημειωθεί ότι "πρώτος" ονομάζεται ένας αριθμός που έχει δύο αποκλειστικά διαιρέτες: τον εαυτό του και τη μονάδα. (Σημείωση: οι αριθμοί 0 και 1 **δεν** είναι πρώτοι)

Α. Να συμπληρωθεί ο σχετικός πίνακας αληθείας

Β. Να συμπληρωθεί πίνακας Karnaugh και να γίνει εξαγωγή της συνάρτησης

Γ. Να σχεδιασθεί κύκλωμα χρησιμοποιώντας μόνο πύλες 2 εισόδων και αντιστροφείς.

Λύση

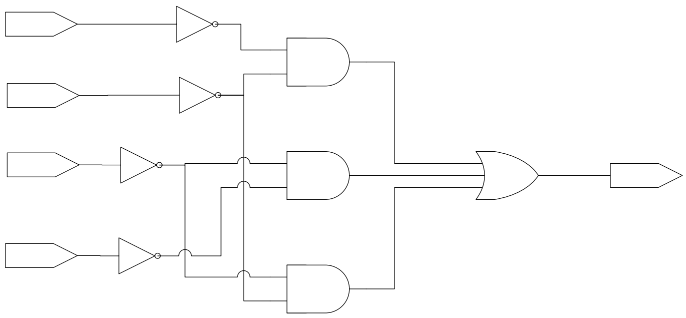
Α.

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

Β) Ο αντίστοιχος χάρτης Karnaugh έχει ως εξής

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

$G = A'C' + B'D' + B'C'$



Β.

xy \ zw	00	01	11	10
00			1	1
01		1	1	
11		1		
10			1	

F=yz'w+x'y'z+x'zw+y'zw

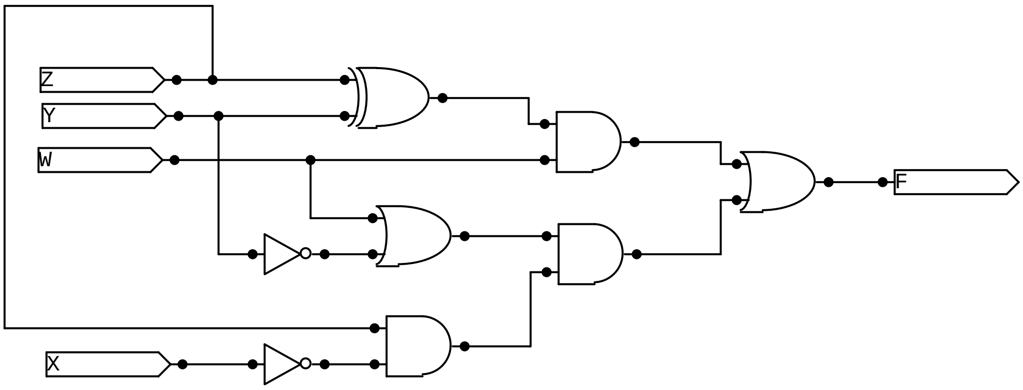
A

Γ. Για να χρησιμοποιήσουμε μόνο πύλες 2 εισόδων, πρέπει να απλοποιήσουμε ακόμα πιο πολύ τη συνάρτηση. Μια ενδεικτική απλοποίηση είναι η εξής:
F=yz'w+x'y'z+x'zw+y'zw=w(y xor z)+x'z(y'+w)
Επομένως το κύκλωμα που προκύπτει είναι το ακόλουθο.

C

B

D



ΑΣΚΗΣΗ 61

Έστω οι προσημασμένοι αριθμοί Α και Β, που σε κώδικα συμπληρώματος ως προς 2 αναπαρίστανται ως a_1a_0 και b_1b_0 αντίστοιχα, δηλαδή με 2 δυαδικά ψηφία ο καθένας. Να σχεδιαστεί ψηφιακό κύκλωμα που να υλοποιεί τη συνάρτηση

$F(a_1, a_0, b_1, b_0) = 1$, όταν $|A - B|=2$ και

0, αλλιώς,

δηλαδή στην έξοδό του να δίνει 1 όταν η απόλυτη τιμή της διαφοράς των Α και Β είναι 2. Να υλοποιήσετε το κύκλωμα με τον ελάχιστο αριθμό πυλών AND, XOR και XNOR.

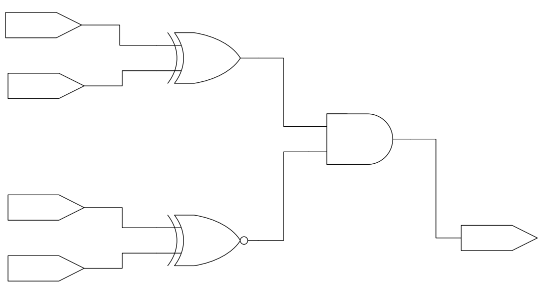
Λύση

Κατασκευάζουμε τον πίνακα αληθείας ως εξής

a_1	a_0	b_1	b_0	$\|A-B\|$	Y
0	0	0	0	0	0
0	0	0	1	1	0
0	0	1	0	2	1
0	0	1	1	1	0
0	1	0	0	1	0
0	1	0	1	0	0
0	1	1	0	3	0
0	1	1	1	2	1
1	0	0	0	2	1
1	0	0	1	3	0
1	0	1	0	0	0
1	0	1	1	1	0
1	1	0	0	1	0
1	1	0	1	2	1
1	1	1	0	1	0
1	1	1	1	0	0

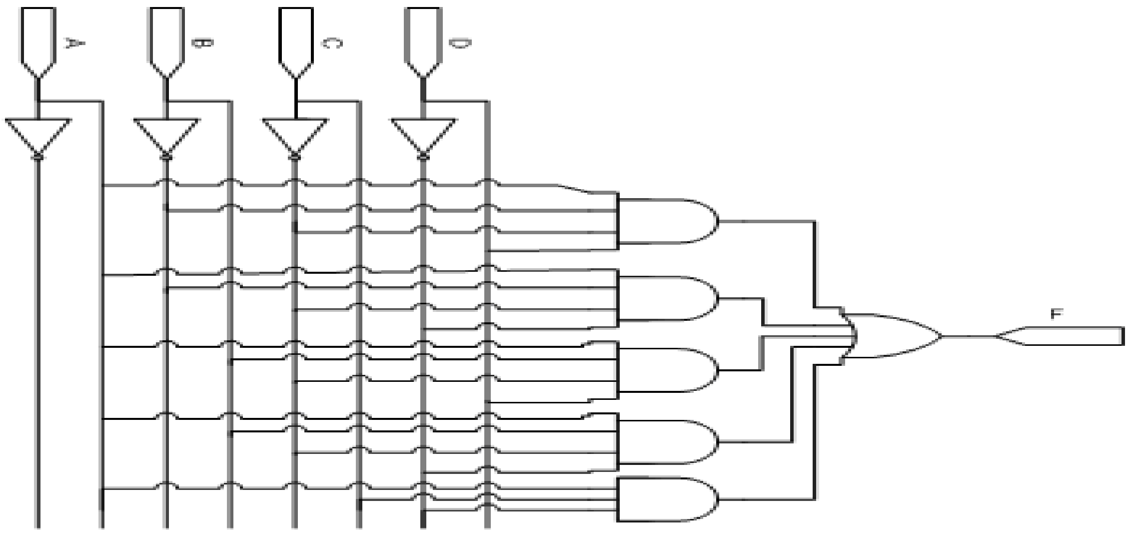
a_1a_0 \ b_1b_0	00	01	11	10
00				1
01			1	
11		1		
10	1			

$$\begin{aligned} Y &= a_1a_0'b_1'b_0' + a_1a_0b_1'b_0 \\ &\quad + a_1'a_0b_1b_0 + a_1'a_0'b_1b_0' \\ &= a_1b_1'\left(a_0 \operatorname{xor} b_0\right)' + a_1'b_1\left(a_0 \operatorname{xor} b_0\right)' \\ &= \left(a_1b_1' + a_1'b_1\right)\left(a_0 \operatorname{xor} b_0\right)' \\ &= \left(a_1 \operatorname{xor} b_1\right)\left(a_0 \operatorname{xnor} b_0\right) \end{aligned}$$



ΑΣΚΗΣΗ 62

Να βρεθεί η απλοποιημένη λογική συνάρτηση του παρακάτω κυκλώματος και να υλοποιηθεί με τη χρήση **ΜΟΝΟ τριών (3)** πυλών NAND 2 εισόδων.



Λύση:

Α' τρόπος:

Από το παραπάνω κύκλωμα προκύπτει η λογική εξίσωση:

F = AB'C'D + AB'C'D' + ABC'D + ABC'D' + ACD'

Η τοποθέτηση των όρων σε πίνακα Karnaugh είναι η παρακάτω:

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

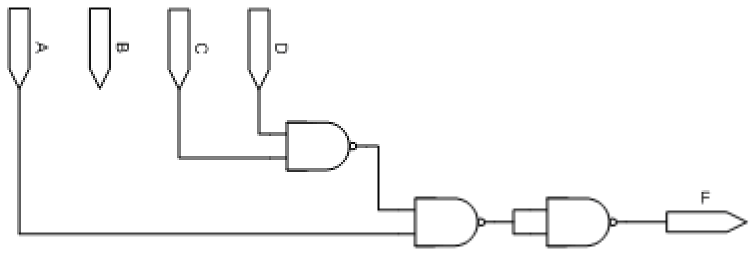
Από τη παραπάνω ομαδοποίηση προκύπτει η λογική εξίσωση:

F = AC' + AD'

Βγάζοντας κοινό παράγοντα το A προκύπτει η παρακάτω μορφή:

F = (AC' + AD') = A (C' + D') = A (C D)' = (A (C D)')''

και τελικά προκύπτει το παρακάτω κύκλωμα με χρήση μόνο 3 πυλών NAND 2 εισόδων:



Β' τρόπος:
Από το παραπάνω κύκλωμα προκύπτει η λογική εξίσωση:

F = AB'C'D + AB'C'D' + ABC'D + ABC'D' + ACD'

Η τοποθέτηση των όρων σε πίνακα Karnaugh είναι όμοια με τον Α΄ τρόπο λύσης και είναι η παρακάτω:

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

Προσπαθώντας να βρούμε την **αντίστροφη συνάρτηση F'**, από τη παραπάνω ομαδοποίηση προκύπτει η λογική εξίσωση:

F' = A' + CD

οπότε η συνάρτηση F μετά και την εφαρμογή του θεωρήματος DeMorgan, δίνει την παρακάτω μορφή που περιέχει μόνο όρους NAND:

F = (A' + CD)' = A (C D)' = (A (C D)')''
που τελικά οδηγεί στο κύκλωμα που σχεδιάστηκε στον Α' τρόπο, με χρήση μόνο 3 πυλών NAND 2 εισόδων.

ΑΣΚΗΣΗ 63

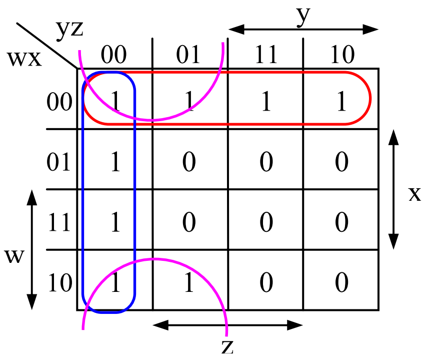
Έστω κύκλωμα το οποίο δέχεται στην είσοδο τέσσερα δυαδικά ψηφία (w,x,y,z) και δίνει στην έξοδο E λογικό '1' όταν το πλήθος των **διαδοχικών** λογικών '0' των τεσσάρων δυαδικών ψηφίων είναι μεγαλύτερο ή ίσο από 2 ενώ δίνει στην έξοδο E λογικό '0' διαφορετικά.
Α. Κατασκευάστε τον πίνακα αλήθειας του κυκλώματος.
Β. Βρείτε την απλοποιημένη συνάρτηση εξόδου χρησιμοποιώντας χάρτη Karnaugh.
Γ. Τροποποιείστε την απλοποιημένη συνάρτηση εξόδου ώστε το κύκλωμα να υλοποιείται χρησιμοποιώντας το πολύ **4 πύλες** των δύο εισόδων η καθεμία (Δεν υπάρχει η δυνατότητα χρήσης πυλών NOT).
Δ. Στην περίπτωση που δεν μας ενδιαφέρει το αποτέλεσμα της εξόδου όταν το πλήθος των διαδοχικών λογικών '0' είναι ίσο με 2, προσπαθήστε να υλοποιήσετε το κύκλωμα χρησιμοποιώντας μόνο μία πύλη δύο εισόδων (Δεν υπάρχει η δυνατότητα χρήσης πυλών NOT).

Λύση

Α. Το κύκλωμα αποτελείται από τέσσερις εισόδους (w,x,y,z) και μία έξοδο Ε. Ο πίνακας αλήθειας έχει 16 γραμμές και είναι ο ακόλουθος:

w	**x**	**y**	**z**	**E**
0	0	0	0	1
0	0	0	1	1
0	0	1	0	1
0	0	1	1	1
0	1	0	0	1
0	1	0	1	0
0	1	1	0	0
0	1	1	1	0
1	0	0	0	1
1	0	0	1	1
1	0	1	0	0
1	0	1	1	0
1	1	0	0	1
1	1	0	1	0
1	1	1	0	0
1	1	1	1	0

Β. Ο χάρτης Karnaugh των τεσσάρων μεταβλητών φαίνεται παρακάτω. Οι γειτονιές ελαχιστόρων φαίνονται σε διάφορα χρώματα. Η συνάρτηση εξόδου και τα γινόμενα που δίνουν φαίνονται με αντίστοιχα χρώματα στο πλαίσιο.

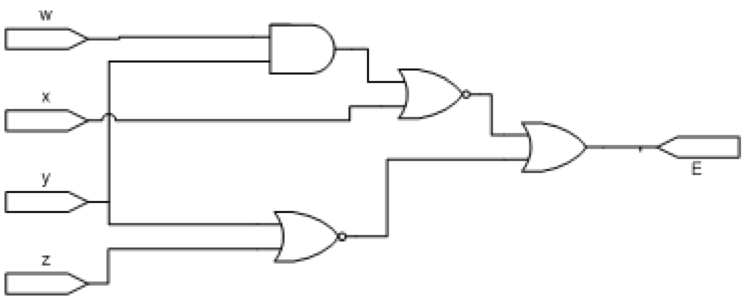


E = $w'x' + x'y' + y'z'$

Γ. Για την υλοποίηση του κυκλώματος με τον μικρότερο δυνατό αριθμό πυλών δύο εισόδων η συνάρτηση εξόδου μπορεί να τροποποιηθεί ως εξής:

$$
\begin{aligned}
E &= w'x' + x'y' + y'z' \\
&= x'(w' + y') + (y + z)' \\
&= x'(w\,y)' + (y + z)' \\
&= (x + (w\,y))' + (y + z)'
\end{aligned}
$$

Η υλοποίηση φαίνεται στο παρακάτω σχήμα:



Δ. Αφού δεν μας ενδιαφέρει το αποτέλεσμα της εξόδου όταν το πλήθος των διαδοχικών λογικών '0' είναι ίσο με 2, σε αυτές τις περιπτώσεις προκύπτουν αδιάφοροι όροι. Ο χάρτης Karnaugh των τεσσάρων μεταβλητών με τους αδιάφορους όρους για την συνάρτηση E φαίνεται παρακάτω. Η γειτονιά ελαχιστόρων με την εκμετάλλευση των αδιάφορων όρων και η συνάρτηση εξόδου E με το γινόμενο που δίνει φαίνεται στο πλαίσιο.

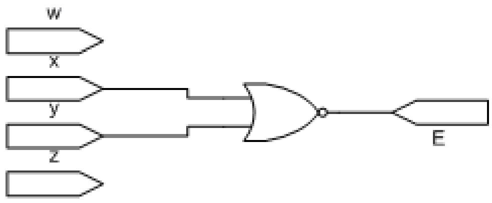
wx \ yz	00	01	11	10
00	1	1	X	X
01	X	0	0	0
11	X	0	0	0
10	1	X	0	0

E = $x'y'$

Για την υλοποίηση του κυκλώματος μόνο με μία πύλη δύο εισόδων η συνάρτηση εξόδου πρέπει να τροποποιηθεί ως εξής:

$E = x'y' = (x+y)'$

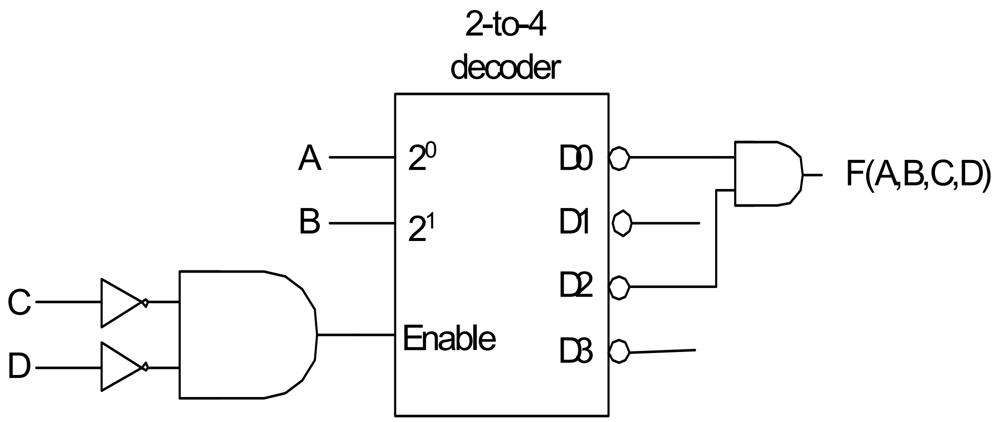
Τώρα η υλοποίηση με μόνο μία πύλη NOR φαίνεται στο παρακάτω σχήμα:



V. Αποκωδικοποιητές – Πολυπλέκτες - Αθροιστές

ΑΣΚΗΣΗ 64

Ποια είναι η δυαδική συνάρτηση F που υλοποιεί το λογικό κύκλωμα του σχήματος; Να εκφραστεί στην απλούστερη δυνατή μορφή.



Λύση:

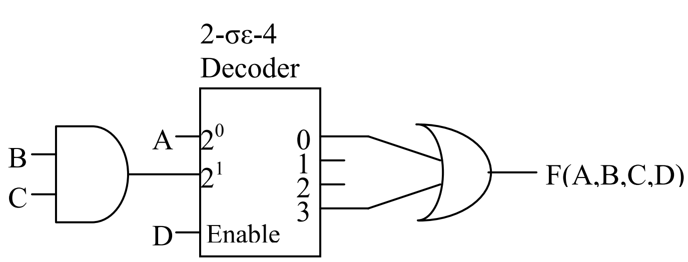
Έστω D_0 και D_2 οι συναρτήσεις των εξόδων 0 και 2 του αποκωδικοποιητή, αντίστοιχα.

Τότε: D_0 = [(C’·D’)·A’B’]’ =[A’·B’·C’·D’]’
και: D_2 = [(C’·D’)·A’B]’ =[A’·B·C’·D’]’

Συνεπώς: $F = D_0\, D_2$ = [A’·B’·C’·D’]’ · [A’·B·C’·D’]’ = [A’·B’·C’·D’+ A’·B·C’·D’]’ = [A’·C’·D’(B’+B)]’=[A’·C’·D’·1]’ = [A’·C’·D’]’ = A+C+D

ΑΣΚΗΣΗ 65

Να βρείτε τη δυαδική συνάρτηση F που υλοποιεί το παρακάτω λογικό κύκλωμα. Η συνάρτηση να απλοποιηθεί και να εκφραστεί σε μορφή αθροίσματος γινομένων.

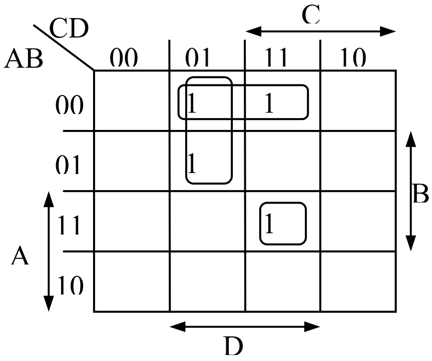


Λύση:

Έστω F_0 και F_3 οι συναρτήσεις των εξόδων 0 και 3 του αποκωδικοποιητή, αντίστοιχα.
Τότε: F_0 = D·[(B·C)’ · A’] = D·[(B’+C’) · A’] = A’·B’·D + A’·C’·D
και: F_3 = D·[(B·C) · A] = A·B·C·D.

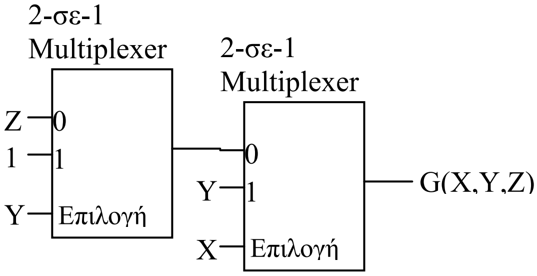
Συνεπώς: $F = F_0 + F_3$ = A’·B’·D + A’·C’·D + A·B·C·D

Με χάρτη Karnaugh εύκολα επιβεβαιώνουμε ότι η παραπάνω είναι και η απλοποιημένη μορφή της συνάρτησης *F*.



ΑΣΚΗΣΗ 66

Να βρείτε τη δυαδική συνάρτηση *G(X,Y,Z)* που υλοποιεί το παρακάτω λογικό κύκλωμα, χρησιμοποιώντας δύο πολυπλέκτες 2-σε-1. Η συνάρτηση να απλοποιηθεί και να εκφραστεί σε μορφή γινομένου αθροισμάτων.

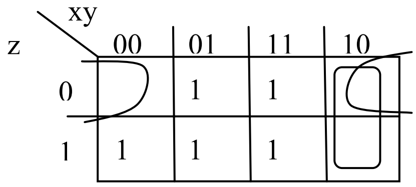


Λύση:

Έστω F(Y,Z) η συνάρτηση εξόδου του πρώτου πολυπλέκτη. Τότε ισχύει: $F = Y\cdot 1 + Y'\cdot Z = Y\cdot(1+Z) + Y'\cdot Z = Y + Y\cdot Z + Y'\cdot Z = Y + (Y+Y')\cdot Z = Y+Z$ (εναλλακτικά: $F=Y1+Y'Z = Y+Y'Z = (Y+Y')(Y+Z) = 1(Y+Z) = Y+Z$).

Αντίστοιχα, για τη συνάρτηση εξόδου του δεύτερου πολυπλέκτη ισχύει: $G = X\cdot Y + X'\cdot F = X\cdot Y + X'\cdot(Y+Z) = X\cdot Y + X'\cdot Y + X'\cdot Z = (X+X')Y+X'Z = Y + X'\cdot Z$

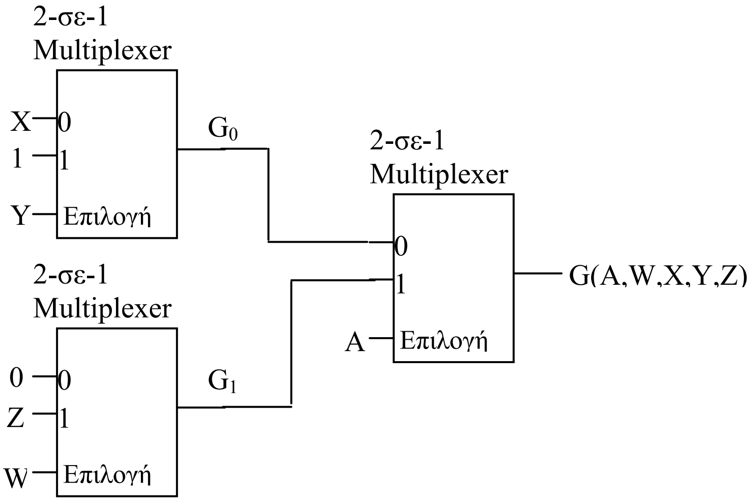
Χρησιμοποιώντας την επιμεριστική ιδιότητα, προκύπτει ότι $G = (Y+X')\ (Y+Z)$. Το ίδιο αποτέλεσμα μπορεί να προκύψει αρχικά κατασκευάζοντας το χάρτη Karnaugh που αντιστοιχεί στην *G* και στη συνέχεια χρησιμοποιώντας τον με δυικό τρόπο σε σχέση με το χάρτη Karnaugh, για αθροίσμα γινομένων, ελαχιστοποιείται η παράσταση σε μορφή γινομένου αθροισμάτων.



Άρα η *G* είναι $G = (y+x')\ (y+z)$.

ΑΣΚΗΣΗ 67

Να βρείτε τη δυαδική συνάρτηση *G(A,W,X,Y,Z)* που υλοποιεί το παρακάτω λογικό κύκλωμα, χρησιμοποιώντας 3 πολυπλέκτες 2-σε-1. Η συνάρτηση να απλοποιηθεί και να εκφραστεί σε μορφή γινομένου αθροισμάτων.



Λύση:

Για έναν πολυπλέκτη 2-σε-1 με εισόδους I_0, I_1, έξοδο F και επιλογή (select) S, ισχύει:
$F = I_0 \cdot S' + I_1 \cdot S$
Συνεπώς:
Για τον 1^o πολυπλέκτη (επάνω αριστερά): $G_0 = X \cdot Y' + 1 \cdot Y = X \cdot Y' + Y = X + Y$
Για τον 2^o πολυπλέκτη (κάτω αριστερά): $G_1 = 0 \cdot W' + Z \cdot W = W \cdot Z$
Για τον 3^o πολυπλέκτη (δεξιά):
$G = G_0 \cdot A' + G_1 \cdot A = (X+Y) \cdot A' + (W \cdot Z) \cdot A = A' \cdot X + A' \cdot Y + A \cdot W \cdot Z$
Για να εκφράσουμε τη συνάρτηση G σε μορφή γινομένου αθροισμάτων, αρχικά υπολογίζουμε την G'(A,W,X,Y,Z), (π.χ. αποτυπώνοντας τη G σε χάρτη Karnaugh και ομαδοποιώντας τα '0'):

A'

WX \ YZ	00	01	11	10
00	0	0	1	1
01	1	1	1	1
11	1	1	1	1
10	0	0	1	1

A

WX \ YZ	00	01	11	10
00	0	0	0	0
01	0	0	0	0
11	0	1	1	0
10	0	1	1	0

Άρα: $G' = AW' + AZ' + A'X'Y'$.
Εφαρμόζοντας διαδοχικά το νόμο De Morgan στην παραπάνω συνάρτηση, έχουμε:
$G = (AW' + AZ' + A'X'Y')' = (AW')'(AZ')'(A'X'Y')' = (A'+W)(A'+Z)(A+X+Y)$.

ΑΣΚΗΣΗ 68

Να υλοποιήσετε τη συνάρτηση

g(w,x,y) = w·y + x’ ·y + w’·x·y’

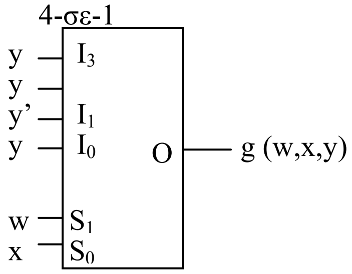
χρησιμοποιώντας μόνο έναν πολυπλέκτη 4-σε-1. Να υποθέσετε ότι οι είσοδοι ελέγχου είναι: S_0=*x*, S_1=*w* και ότι τα συμπληρώματα των μεταβλητών *w,x,y* είναι διαθέσιμα.

Λύση:
Αρχικά υπολογίζουμε τον πίνακα αλήθειας της συνάρτησης *g(w,x,y)*, (βλ. παρακάτω) και με βάση αυτόν, υπολογίζουμε τις τιμές των εισόδων του πολυπλέκτη I_0-I_3, συναρτήσει του *y*.

Πρώτος τρόπος

w	x	y	g(w,x,y)	Είσοδοι
0	0	0	0	I_0=y
0	0	1	1	
0	1	0	1	I_1=y’
0	1	1	0	
1	0	0	0	I_2=y
1	0	1	1	
1	1	0	0	I_3=y
1	1	1	1	

Πολυπλέκτης
4-σε-1



Δεύτερος Τρόπος

Η λογική συνάρτηση την οποία υλοποιεί ο πολυπλέκτης είναι η ακόλουθη:

$$O = S_1'S_0'I_0 + S_1'S_0I_1 + S_1S_0'I_2 + S_1S_0I_3$$
$$= w'x'I_0 + w'xI_1 + wx'I_2 + wxI_3$$

το οποίο σημαίνει ότι:

$O = I_0$, μόνο όταν $w = 0$ και $x = 0$, δηλ. $w'x' = 1$, άρα $I_0 = g\left(0,0,y\right)$

$O = I_1$, μόνο όταν $w = 0$ και $x = 1$, δηλ. $w'x = 1$, άρα $I_1 = g\left(0,1,y\right)$

$O = I_2$, μόνο όταν $w = 1$ και $x = 0$, δηλ. $wx' = 1$, άρα $I_2 = g\left(1,0,y\right)$

$O = I_3$, μόνο όταν $w = 1$ και $x = 1$, δηλ. $wx = 1$, άρα $I_3 = g\left(1,1,y\right)$.

Έτσι,

$$I_0 = g\left(0,0,y\right) = 0\cdot y + 1\cdot y + 0\cdot y' = y$$
$$I_1 = g\left(0,1,y\right) = 0\cdot y + 0\cdot y + 1\cdot 1\cdot y' = y'$$
$$I_2 = g\left(1,0,y\right) = 1\cdot y + 1\cdot y + 0\cdot 0\cdot y' = y$$
$$I_3 = g\left(1,1,y\right) = 1\cdot y + 0\cdot y + 0\cdot 1\cdot y' = y$$

ΑΣΚΗΣΗ 69

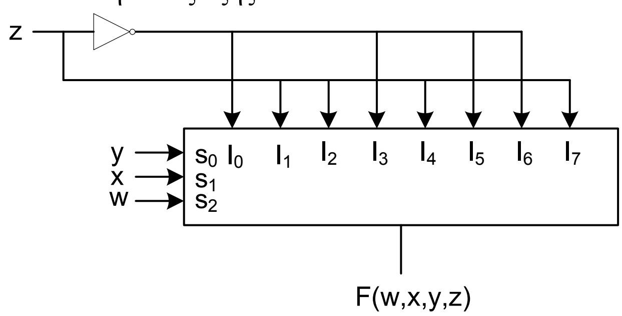
Χρησιμοποιήστε έναν πολυπλέκτη 8-σε-1 και τον ελάχιστο δυνατό αριθμό πυλών για την υλοποίηση της λογικής πράξης *F(w,x,y,z) = XOR (XOR (w',x), XOR (y,z))* $(F(w,x,y,z)=(w'\oplus x)\oplus(y\oplus z))$.

Λύση:
Αρχικά, συμπληρώνουμε τον πίνακα αλήθειας της *F(w,x,y,z)*:

w	x	y	z	w'	XOR (w',x)	XOR (y,z)	F(w,x,y,z)	Είσοδοι
0	0	0	0	1	1	0	1	I_0 = z'
0	0	0	1	1	1	1	0	
0	0	1	0	1	1	1	0	I_1 = z
0	0	1	1	1	1	0	1	
0	1	0	0	1	0	0	0	I_2 = z
0	1	0	1	1	0	1	1	
0	1	1	0	1	0	1	1	I_3 = z'
0	1	1	1	1	0	0	0	
1	0	0	0	0	0	0	0	I_4 = z
1	0	0	1	0	0	1	1	
1	0	1	0	0	0	1	1	I_5 = z'
1	0	1	1	0	0	0	0	
1	1	0	0	0	1	0	1	I_6 = z'
1	1	0	1	0	1	1	0	
1	1	1	0	0	1	1	0	I_7 = z
1	1	1	1	0	1	0	1	

Επιλέγουμε (αυθαίρετα) και χρησιμοποιούμε τις μεταβλητές *w, x, y* (*w*=ΠΣΨ) ως μεταβλητές επιλογής εισόδου του πολυπλέκτη. Στη συνέχεια, συγκρίνοντας τις στήλες "*z*" και "*F*", υπολογίζουμε τις συναρτήσεις των εισόδων $I_0 - I_7$, σε σχέση με τη μεταβλητή *z* (βλ. τελευταία στήλη του παραπάνω πίνακα).

Το ζητούμενο κύκλωμα θα υλοποιηθεί ως εξής:

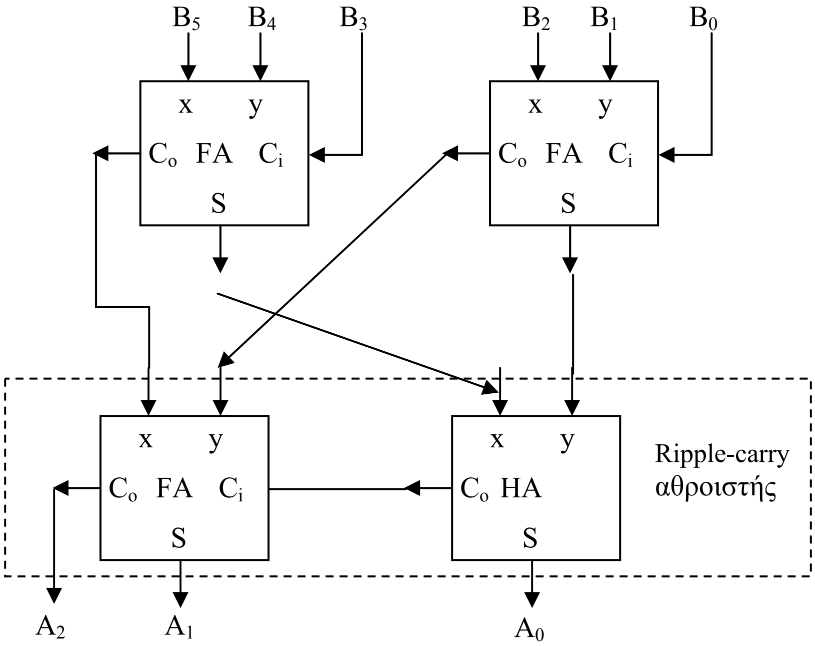


ΑΣΚΗΣΗ 70

Να σχεδιάσετε ένα ψηφιακό κύκλωμα με έξι εισόδους του ενός bit η καθεμία, το οποίο να υπολογίζει το πλήθος των «1» των εισόδων του και να το δίνει στην έξοδό του ως τριψήφιο δυαδικό αριθμό. Στο σχεδιασμό σας χρησιμοποιήστε μόνο 3 πλήρεις αθροιστές και 1 ημιαθροιστή (χωρίς επιπλέον πύλες).

Λύση:

Από τον πίνακα αληθείας του πλήρους αθροιστή προκύπτει ότι οι δύο έξοδοι δίνουν το πλήθος των άσσων που εμφανίζονται στις εισόδους του αθροιστή. Επομένως αν οδηγήσουμε τις εισόδους ανά τρεις σε δύο πλήρεις αθροιστές θα πάρουμε δύο λέξεις των δύο δυαδικών ψηφίων, τις οποίες μπορούμε να προσθέσουμε χρησιμοποιώντας έναν ripple-carry αθροιστή. Αυτός υλοποιείται με έναν πλήρη αθροιστή και έναν ημιαθροιστή, συνδεμένους όπως φαίνεται στο επόμενο κύκλωμα:



ΑΣΚΗΣΗ 71

Να κατασκευάσετε έναν πλήρη αθροιστή BCD 4 ψηφίων. Έχετε στη διάθεσή σας μόνο δύο πλήρεις δυαδικούς αθροιστές 4 ψηφίων και πύλες NAND.

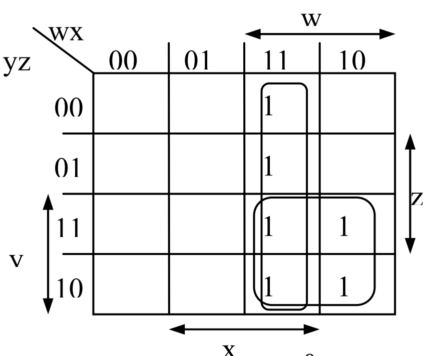
Λύση:

Ένας δυαδικός αθροιστής 4 bit υπολογίζει δυαδικά αθροίσματα από 0000+0000+(0)=0000 έως 1111+1111+(1)=(1)1111 (ο αριθμός σε παρένθεση είναι το «κρατούμενο»). Ένας αθροιστής BCD, αντίστοιχα υπολογίζει αθροίσματα μέχρι 1001+1001+(1)=(1)1001. Στις περιπτώσεις που το άθροισμα είναι μεταξύ (0)0000 έως (0)1001, ο δυαδικός αθροιστής δίνει το σωστό BCD αποτέλεσμα.

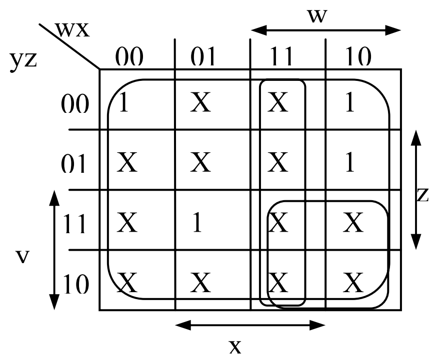
Ένας BCD αθροιστής των 4 δυαδικών ψηφίων παίρνει ως εισόδους τα A, B, Cin, με $0 \leq A, B \leq 9$, άρα αν χρησιμοποιηθεί ένας δυαδικός αθροιστής των 4 δυαδικών ψηφίων θα δίνει τιμές από 00000 έως 10011=(9+9+1). Η σχέση μεταξύ του δυαδικού αθροίσματος δύο ακεραίων και της BCD αναπαράστασης του αθροίσματος παρουσιάζεται αναλυτικά στον ακόλουθο πίνακα:

Ακέραια τιμή αθροίσματος (1)	Έξοδος δυαδικού αθροιστή (κρατούμενο και τέσσερα δυαδικά ψηφία αθροίσματος) (2)	BCD αναπαράσταση (BCD κρατούμενο και BCD άθροισμα) (3)	Ακέραιος που αντιστοιχεί στη BCD αναπαράσταση (4)	Διαφορά (4) – (1)
	(c)wxyz			
0	(0)0000	(0)0000	0	0
1	(0)0001	(0)0001	1	0
2	(0)0010	(0)0010	2	0
3	(0)0011	(0)0011	3	0
4	(0)0100	(0)0100	4	0
5	(0)0101	(0)0101	5	0
6	(0)0110	(0)0110	6	0
7	(0)0111	(0)0111	7	0
8	(0)1000	(0)1000	8	0
9	(0)1001	(0)1001	9	0
10	(0)1010	(1)0000	16	6
11	(0)1011	(1)0001	17	6
12	(0)1100	(1)0010	18	6
13	(0)1101	(1)0011	19	6
14	(0)1110	(1)0100	20	6
15	(0)1111	(1)0101	21	6
16	(1)0000	(1)0110	22	6
17	(1)0001	(1)0111	23	6
18	(1)0010	(1)1000	24	6
19	(1)0011	(1)1001	25	6

Από τον πίνακα προκύπτει ότι όταν το άθροισμα είναι μεγαλύτερο από 9, θα πρέπει να προστίθεται ο αριθμός έξι σε αυτό. Έστω *d* μια δυαδική μεταβλητή η οποία δηλώνει ότι το άθροισμα είναι μεγαλύτερο από το 9. Χρησιμοποιώντας τον πιο πάνω πίνακα, καταστρώνουμε δύο χάρτες Karnaugh τεσσάρων μεταβλητών, έναν για την τιμή c=0 και έναν για την τιμή c=1. Στο δεύτερο πίνακα χρησιμοποιούμε αδιάφορους όρους *X* γιατί οι συνδυασμοί από 19 έως 31 δεν είναι δυνατό να προκύψουν.



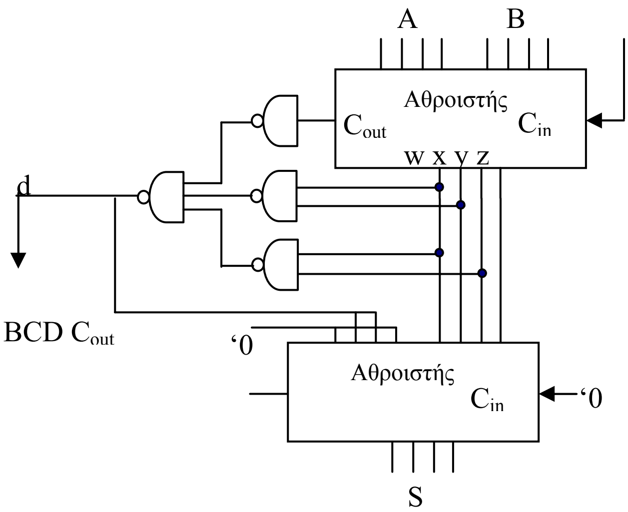
$c = 0$



$c = 1$

Από τους πίνακες αυτούς προκύπτει ότι $d = wx+wy+c$. Με εφαρμογή DeMorgan παίρνουμε $d = wx + wy + c = ((wx)'(wy)'c')' = ((wx)'(wy)'(cc)')'$

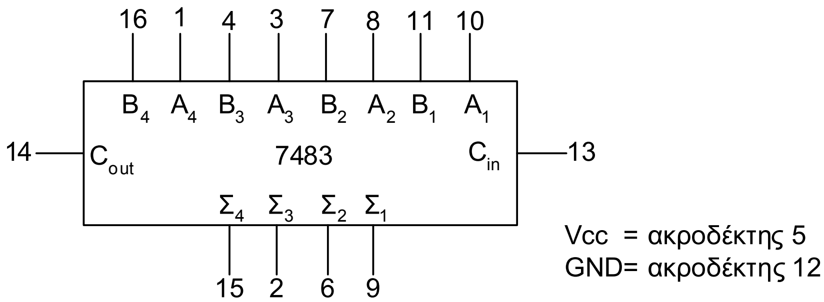
Άρα το ζητούμενο κύκλωμα είναι.



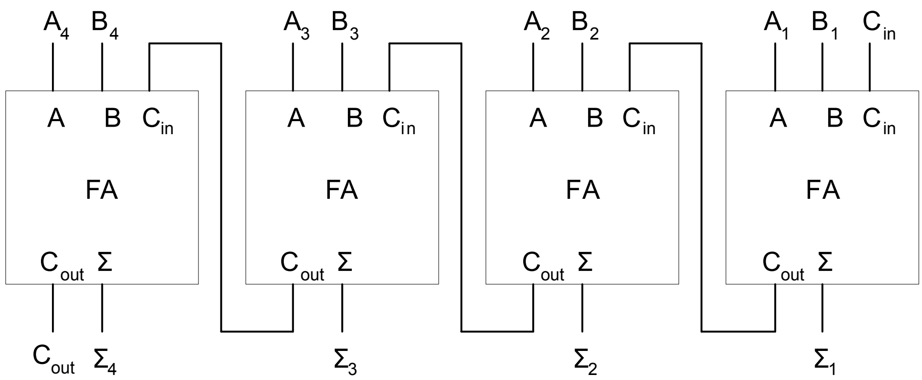
ΑΣΚΗΣΗ 72

Το ολοκληρωμένο κύκλωμα 7483 είναι ένας δυαδικός αθροιστής των 4 bits. Το λογικό σύμβολο και το διάγραμμα βαθμίδων (block diagram) φαίνονται στα παρακάτω σχήματα.

Με τη βοήθεια τέτοιων ολοκληρωμένων κυκλωμάτων και κατάλληλων πυλών να σχεδιάσετε έναν αθροιστή/αφαιρέτη των 8 bits. Η αφαίρεση να γίνεται με χρήση του συμπληρώματος ως προς 2. Να σημειώσετε πάνω στο κύκλωμα που σχεδιάσατε τις τιμές των εισόδων και εξόδων για την πράξη $(33)_{10}$- $(18)_{10}$.

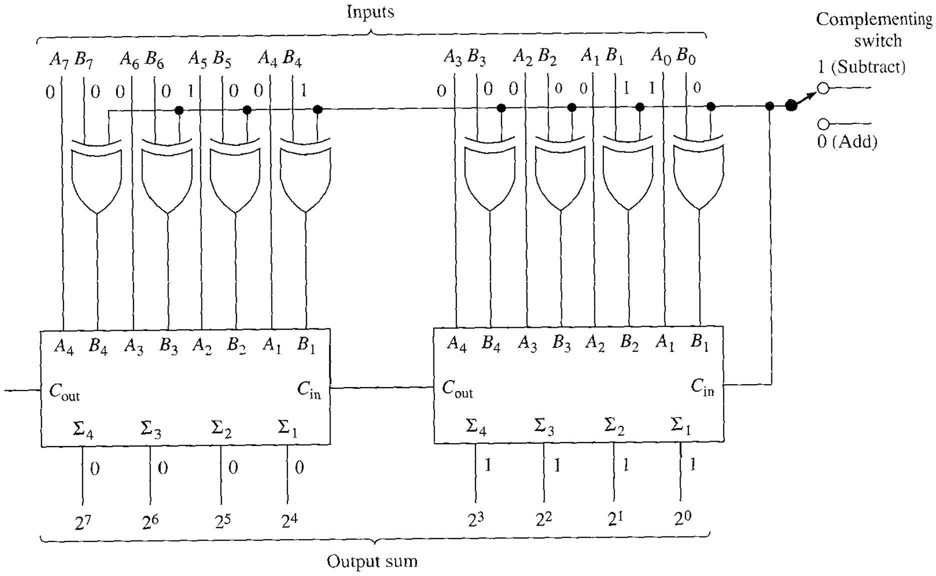


Σχ. 3.1: Λογικό σύμβολο του 4-bit δυαδικού αθροιστή 7483. Στις εισόδους A_1 έως A_4 και B_1 έως B_4 εφαρμόζονται οι προς πρόσθεση τετραψήφιοι δυαδικοί αριθμοί, ενώ συις εξόδους Σ_1 έως Σ_4 εμφανίζεται το αποτέλεσμα της πρόσθεσης. C_{in} είναι η είσοδος του κρατουμένου το οποίο προστίθεται στα A_1, B_1 και C_{out} η έξοδος του κρατουμένου του τελευταίου πλήρους αθροιστή των A_4, B_4. Οι αριθμοί του σχήματος δείχνουν τους αντίστοιχους ακροδέκτες του ολοκληρωμένου κυκλώματος.



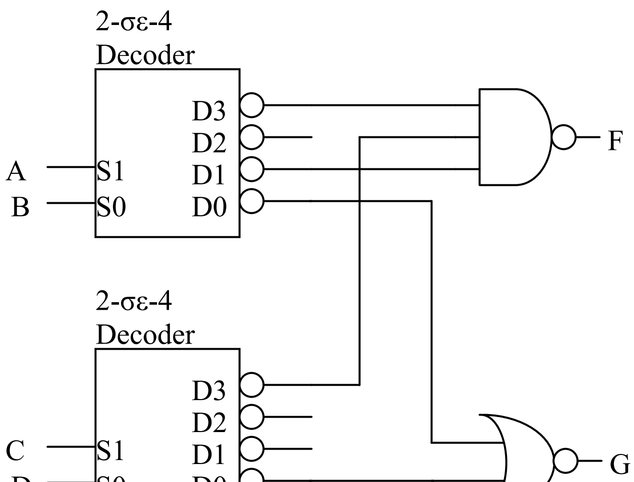
Σχ. 3.2: Διάγραμμα βαθμίδων δυαδικού αθροιστή των 4 bits.

Λύση:



ΑΣΚΗΣΗ 73

Να γράψετε τις συναρτήσεις *F(A,B,C,D)* και *G(A,B,C,D),* που υλοποιεί το παρακάτω κύκλωμα σε ελάχιστη μορφή αθροίσματος γινομένων καθώς και σε κανονική μορφή αθροίσματος ελάχιστων όρων (sum of minterms). (Το ψηφίο Α είναι το ΠΣΨ).



Λύση:

Από τη λειτουργία του αποκωδικοποιητή με ανεστραμμένες (active low) εξόδους έχουμε:

Για τον επάνω decoder: D0 = (A'·B')', D1 = (A'·B)', D2 = (A·B')', D3 = (A·B)'.
Για τον κάτω decoder: D0 = (C'·D')', D1 = (C'·D)', D2 = (C·D')', D3 = (C·D)'.

Συνεπώς, οι F και G υπολογίζονται σαν (ελάχιστα) αθροίσματα ως εξής:

$$F = ((A\cdot B)' \cdot (A'\cdot B)' \cdot (C\cdot D)')' = (A\cdot B)'' + (A'\cdot B)'' + (C\cdot D)'' =$$
$$A\cdot B + A'\cdot B + C\cdot D = (A+A')\cdot B + C\cdot D = 1\cdot B + C\cdot D = B + C\cdot D.$$
$$G = ((A'\cdot B')' + (C'\cdot D'))' = (A'\cdot B')'' \cdot (C'\cdot D')'' = A'\cdot B'\cdot C'\cdot D'.$$

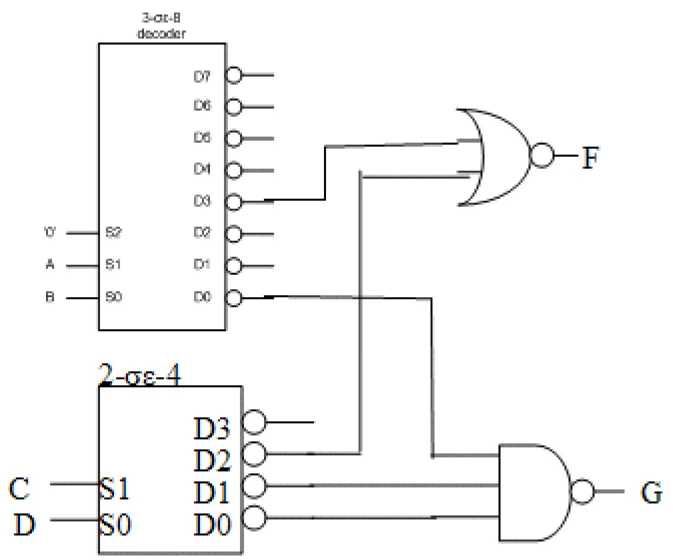
Ο όρος B περιλαμβάνει τους minterms $\{m_4, m_5, m_6, m_7, m_{12}, m_{13}, m_{14}, m_{15}\}$ $(=m_{X1XX})$. Ο όρος C·D περιλαμβάνει τους minterms $\{m_3, m_7, m_{11}, m_{15}\}$ $(=m_{XX11})$.

Ο όρος A'·B'·C'·D' περιλαμβάνει μόνο τον minterm m_0 $(=m_{0000})$.

Με βάση τις παραπάνω σχέσεις, οι συναρτήσεις F και G γράφονται σε κανονική μορφή ελαχίστων όρων (sum of minterms) ως εξής:
$F = \Sigma\ \{3,4,5,6,7,11,12,13,14,15\}$ $\qquad$ $G = \Sigma\ \{0\}$.

ΑΣΚΗΣΗ 74

Να γράψετε τις συναρτήσεις *F(A,B,C,D)* και *G(A,B,C,D),* που υλοποιεί το παρακάτω κύκλωμα σε ελάχιστη μορφή αθροίσματος γινομένων καθώς και σε κανονική μορφή αθροίσματος ελάχιστων όρων (sum of minterms). (Το ψηφίο A είναι το ΠΣΨ). Διευκρίνιση: Η είσοδος S2 του πρώτου αποκωδικοποιητή βρίσκεται μόνιμα στο λογικό μηδέν ("0").



Λύση:

Από τη λειτουργία του αποκωδικοποιητή με ανεστραμμένες (active low) εξόδους και λαμβάνοντας υπόψη ότι η είσοδος S2 είναι μόνιμα στο λογικό 0 (και άρα μόνο οι τέσσερις λιγότερο σημαντικές έξοδοι D3, D2, D1, D0 θα επηρεάζονται από τις εισόδους A, B), έχουμε:

Για τον 3-σε-8 decoder: $D0 = (A'\cdot B')'$, $D1 = (A'\cdot B)'$, $D2 = (A\cdot B')'$, $D3 = (A\cdot B)'$.

Για τον 2-σε-4 decoder: $D0 = (C'\cdot D')'$, $D1 = (C'\cdot D)'$, $D2 = (C\cdot D')'$, $D3 = (C\cdot D)'$.

Συνεπώς, οι F και G υπολογίζονται σαν (ελάχιστα) αθροίσματα ως εξής:

$$F = ((A\cdot B)' + (C\cdot D')')' = (A\cdot B)'' \cdot (C\cdot D')'' = A\cdot B\cdot C\cdot D'$$

$$G = ((A'\cdot B')' \cdot (C'\cdot D)' \cdot (C'D')')' = (A'\cdot B')'' + (C'\cdot D)'' + (C'D')'' =$$

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

Ο όρος $A\cdot B\cdot C\cdot D'$ περιλαμβάνει μόνο τον minterm m14 ($=m_{1110}$).
Ο όρος C' περιλαμβάνει τους minterms {m0, m1, m4, m5, m8, m9, m12, m13} ($=m_{XX0X}$).
Ο όρος $A'\cdot B'$ περιλαμβάνει τους minterms {m0, m1, m2, m3} ($=m_{00XX}$).

Με βάση τις παραπάνω σχέσεις, οι συναρτήσεις F και G γράφονται σε κανονική μορφή ελαχίστων όρων (sum of minterms) ως εξής:

$F_{(A,B,C,D)} = \Sigma\ \{14\}$ $\quad G_{(A,B,C,D)} = \Sigma\ \{0,1,2,3,4,5,8,9,12,13\}$.

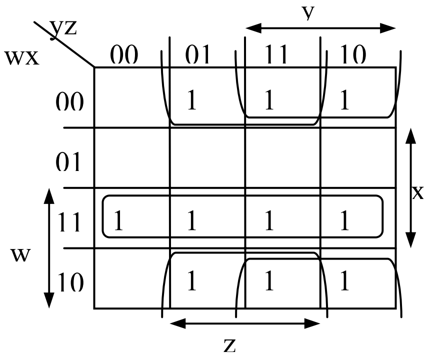
ΑΣΚΗΣΗ 75

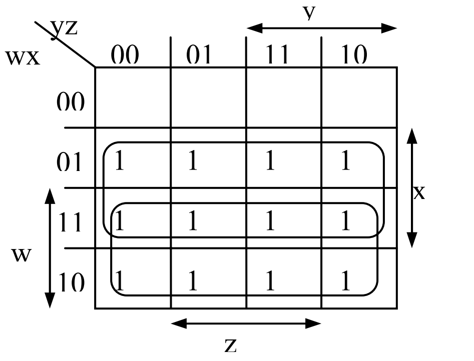
Δίνεται η συνάρτηση f:
$f(n) = \mathrm{trunc}(\sqrt{n})$
που υπολογίζει το ακέραιο μέρος (= με αποκοπή) της τετραγωνικής ρίζας του ακέραιου θετικού αριθμού n. Υποθέτοντας ότι το n είναι αριθμός 4 bit ($0000_{[2]}$-$1111_{[2]}$), τότε το αποτέλεσμα της συνάρτησης είναι ένας αριθμός 2 bit ($00_{[2]}$-$11_{[2]}$).

α) Να κατασκευαστεί ο πίνακας αλήθειας της συνάρτησης f για το παραπάνω πεδίο ορισμού.
β) Να απλοποιηθούν οι δύο δυαδικές συναρτήσεις f_0, f_1, που αποτελούν το ΕΣΨ και ΠΣΨ της f, κάνοντας χρήση χαρτών Karnaugh.
γ) Να υλοποιηθεί η συνάρτηση f (δηλ. οι f_0, f_1) με χρήση μόνο πυλών NAND.
δ) Να υλοποιηθεί η συνάρτηση f_0 με έναν πολυπλέκτη “8-σε-1”.

Λύση:
α)

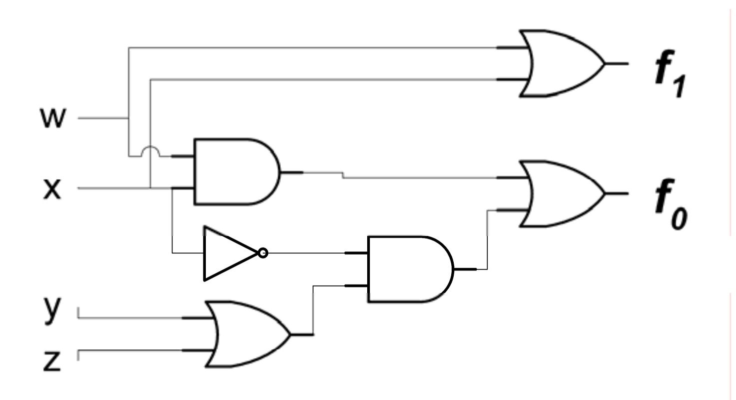
w	x	y	z	n	f(n)= trunc(√n)	f_1	f_0
0	0	0	0	0	0	0	0
0	0	0	1	1	1	0	1
0	0	1	0	2	1	0	1
0	0	1	1	3	1	0	1
0	1	0	0	4	2	1	0
0	1	0	1	5	2	1	0
0	1	1	0	6	2	1	0
0	1	1	1	7	2	1	0
1	0	0	0	8	2	1	0
1	0	0	1	9	3	1	1
1	0	1	0	10	3	1	1
1	0	1	1	11	3	1	1
1	1	0	0	12	3	1	1
1	1	0	1	13	3	1	1
1	1	1	0	14	3	1	1
1	1	1	1	15	3	1	1



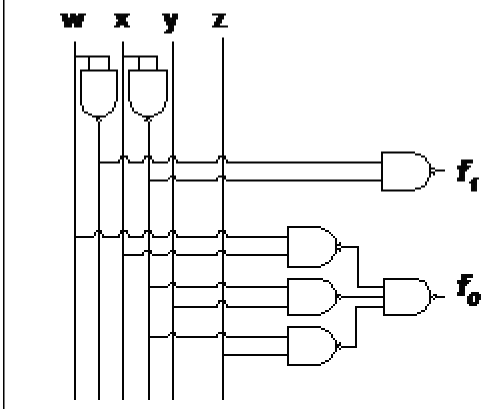


β) $f_0 = wx + x'y + x'z = wx + x'(y+z)$ $f_1 = w + x$

γ) Η υλοποίηση των παραπάνω εξισώσεων μπορεί να γίνει με τα παρακάτω κυκλώματα:



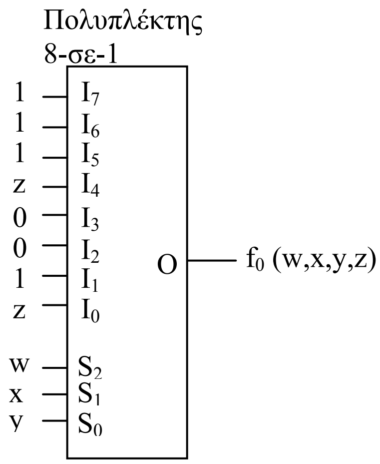
$f_1 = w + x$ $f_0 = wx + x'(y+z)$
Υλοποίηση με διάφορα είδη πυλών (6 πύλες)



$f_1 = w + x$ $f_0 = wx + x'y + x'z$
Υλοποίηση με πύλες NAND (7 πύλες)

δ) Για την υλοποίηση της f_0 με χρήση ενός πολυπλέκτη 8-σε-1, χρησιμοποιούμε τα 3 ΠΣΨ σαν μεταβλητές επιλογής (γραμμές select) και στις γραμμές εισόδου I_0-I_7 συνδέουμε είτε λογικό 0, είτε 1, είτε z, είτε z', συγκρίνοντας τις στήλες f_0 και z, ανά 2 γραμμές κάθε φορά.

w	x	y	z	f_0	Είσοδοι
0	0	0	0	0	I_0=z
0	0	0	1	1	
0	0	1	0	1	I_1=1
0	0	1	1	1	
0	1	0	0	0	I_2=0
0	1	0	1	0	
0	1	1	0	0	I_3=0
0	1	1	1	0	
1	0	0	0	0	I_4=z
1	0	0	1	1	
1	0	1	0	1	I_5=1
1	0	1	1	1	
1	1	0	0	1	I_6=1
1	1	0	1	1	
1	1	1	0	1	I_7=1
1	1	1	1	1	

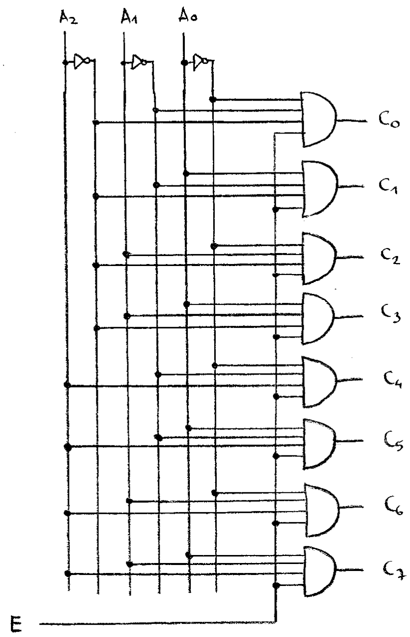


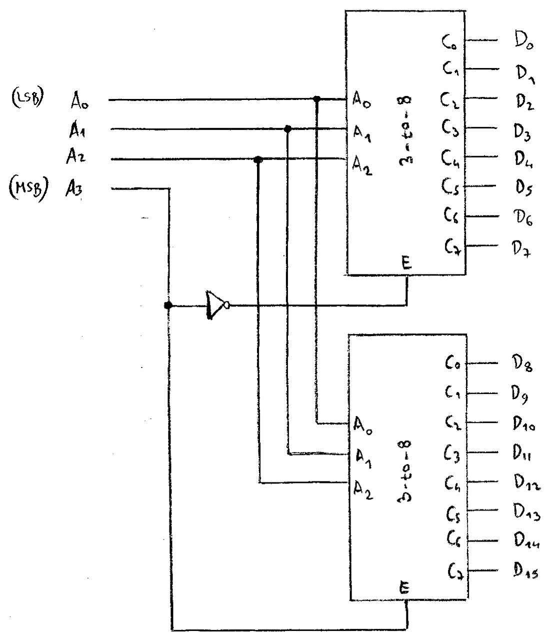
ΑΣΚΗΣΗ 76

Καταστρώστε τον πίνακα αληθείας και σχεδιάστε το κύκλωμα ενός αποκωδικοποιητή 3-σε-8, κατ' αντιστοιχία με το Σχήμα 6.13 του βιβλίου σας. Στη συνέχεια σχεδιάστε το λογικό κύκλωμα ενός αποκωδικοποιητή 4-σε-16 βασισμένου σε βαθμίδες αποκωδικοποιητών 3-σε-8.

Λύση:

E	A_2	A_1	A_0	C_7	C_6	C_5	C_4	C_3	C_2	C_1	C_0
0	x	x	x	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	1
1	0	0	1	0	0	0	0	0	0	1	0
1	0	1	0	0	0	0	0	0	1	0	0
1	0	1	1	0	0	0	0	1	0	0	0
1	1	0	0	0	0	0	1	0	0	0	0
1	1	0	1	0	0	1	0	0	0	0	0
1	1	1	0	0	1	0	0	0	0	0	0
1	1	1	1	1	0	0	0	0	0	0	0





ΑΣΚΗΣΗ 77

Ποιος είναι ο ελάχιστος αριθμός πολυπλεκτών 4-σε-1 (εύρους 1 bit), που απαιτούνται για τη σύνθεση των εξής πολυπλεκτών:
α) 16-σε-1, β) 12-σε-1, γ) 4-σε-1, εύρους 4 bit, δ) 64-σε-1.

Λύση:
α) Για τις 16 εισόδους χρησιμοποιούμε 4 πολυπλέκτες (4 είσοδοι ανά πολυπλέκτη, κοινές γραμμές επιλογής και για τους 4 πολυπλέκτες), τις εξόδους των οποίων συνδέουμε σε έναν επιπλέον πολυπλέκτη. Άρα, συνολικά απαιτούνται *5 πολυπλέκτες* με συνολικό αριθμό 4 γραμμών επιλογής.

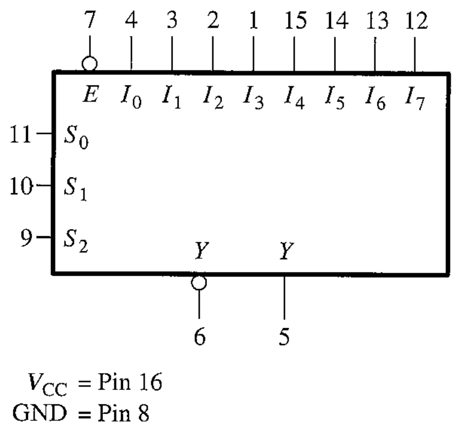
β) Για τις 12 εισόδους χρησιμοποιούμε 3 πολυπλέκτες (4 είσοδοι ανά πολυπλέκτη, κοινές γραμμές επιλογής και για τους 3 πολυπλέκτες), τις εξόδους των οποίων συνδέουμε σε έναν επιπλέον πολυπλέκτη. Άρα, συνολικά απαιτούνται *4 πολυπλέκτες* με συνολικό αριθμό 4 γραμμών επιλογής.

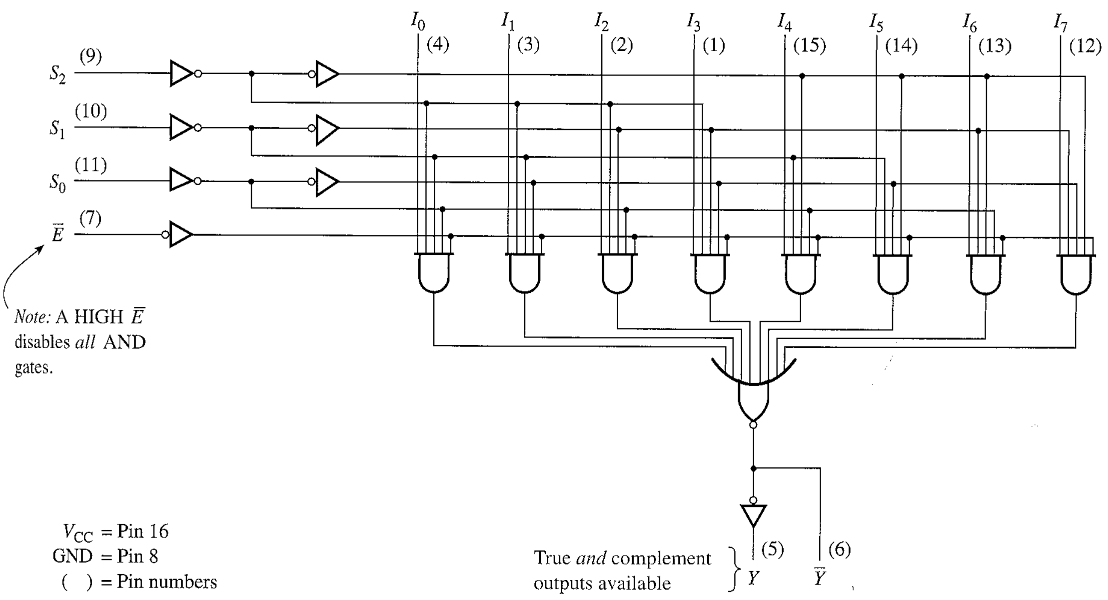
γ) Χρησιμοποιούμε σε παράλληλη συνδεσμολογία *4 πολυπλέκτες* με κοινές εισόδους και γραμμές επιλογής.

δ) Ανάλογα με τα (α) και (β), για τις 64 εισόδους χρησιμοποιούμε τρία επίπεδα πολυπλεκτών, με 16, 4 και 1 πολυπλέκτες αντίστοιχα. Χρησιμοποιούμε από 2 κοινές γραμμές επιλογής ανά επίπεδο. Άρα, συνολικά απαιτούνται *21 πολυπλέκτες* με συνολικό αριθμό 6 γραμμών επιλογής.

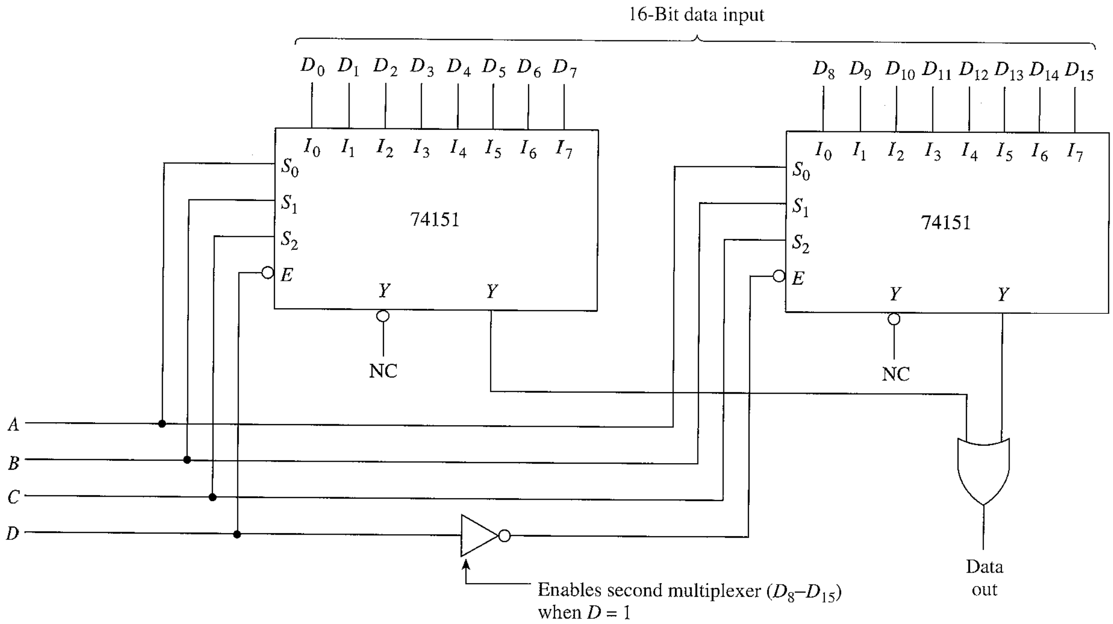
ΑΣΚΗΣΗ 78

Το λογικό σύμβολο και το λογικό διάγραμμα του πολυπλέκτη 8 γραμμών σε 1 (74151) φαίνονται στο παρακάτω σχήμα Χρησιμοποιώντας δύο τέτοια κυκλώματα και κατάλληλες πύλες, σχεδιάστε εναν πολυπλέκτη 16 γραμμών σε 1. Δώστε τις αντίστοιχες επεξηγήσεις.





Λύση:



ΑΣΚΗΣΗ 79

Μπορεί να υλοποιηθεί η συνάρτηση $F(a, b, c, d) = \Sigma(4, 6, 7, 8, 10, 11, 12, 14, 15)$ χρησιμοποιώντας μόνο αποκωδικοποιητές 2-σε-4 (με κανονικές εξόδους και χωρίς είσοδο επίτρεψης);

Λύση:

Κάθε αποκωδικοποιητής 2-σε-4 υλοποιεί όλους τους ελαχιστόρους 2 μεταβλητών. Με επιπλέον χρήση εξωτερικών πυλών (πχ. πύλες OR), μπορούμε να υλοποιήσουμε οποιαδήποτε συνάρτηση 2 μεταβλητών. Θα προσπαθήσουμε αρχικά να αναλύσουμε την παραπάνω συνάρτηση 4 μεταβλητών σε συναρτήσεις 2 μεταβλητών, έτσι ώστε κάθε μία από αυτές να υλοποιηθεί από έναν αποκωδικοποιητή και πιθανώς εξωτερικές πύλες. Αρχικά θα απλοποιήσουμε τη συνάρτηση. Ο χάρτης Karnaugh της συνάρτησης είναι ο ακόλουθος:

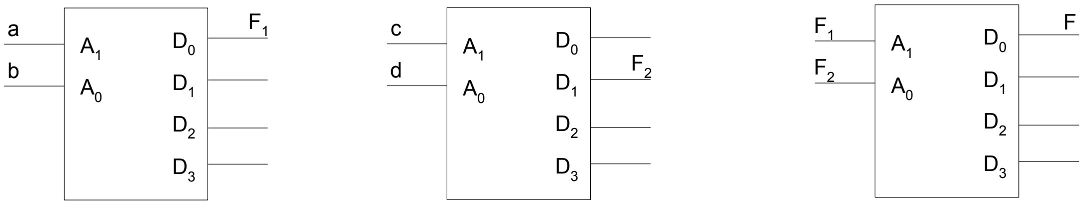
ab \ cd	00	01	11	10
00	0	0	0	0
01	1	0	1	1
11	1	0	1	1
10	1	0	1	1

$$F = bc+ac+bd'+ad' = (a+b)c+(a+b)d' = (a+b)(c+d')$$

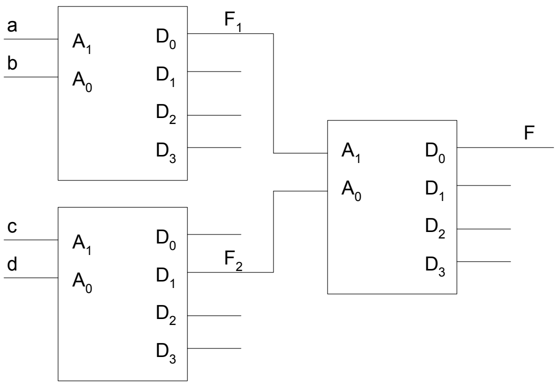
Τώρα με εφαρμογή του θεωρήματος DeMorgan η παραπάνω σχέση γίνεται

$F=(a'b')'(c'd)'=F_1'F_2'$, όπου $F_1= a'b'$ και $F_2= c'd$.

Οι συναρτήσεις F_1, F_2 είναι συναρτήσεις 2 μεταβλητών, και μπορούν να υλοποιηθούν από αποκωδικοποιητές, χωρίς τη χρήση εξωτερικών πυλών, καθώς κάθε μία από αυτές αποτελείται μόνο από έναν ελαχιστόρο. Άρα η F_1 είναι ο ελαχιστόρος 0, και υλοποιείται από την έξοδο D_0 του αποκωδικοποιητή όταν οι γραμμές επιλογής είναι $(A_1,A_0)=(a,b)$ ενώ αντίστοιχα η F_2 είναι ο ελαχιστόρος 1, και υλοποιείται από την έξοδο D_1 του αποκωδικοποιητή όταν οι γραμμές επιλογής είναι $(A_1,A_0)=(c,d)$. Με τον ίδιο τρόπο, εάν θεωρήσουμε τις F_1, F_2 ως μεταβλητές της F, παρατηρούμε ότι ο όρος $F_1'F_2'$ είναι ουσιαστικά ο ελαχιστόρος 0, και μπορεί να υλοποιηθεί από την έξοδο D_0 ενός αποκωδικοποιητή όταν οι γραμμές επιλογής είναι $(A_1,A_0)=(F_1,F_2)$ όπως φαίνεται στο σχήμα:



Άρα η συνάρτηση F μπορεί να υλοποιηθεί μόνο με αποκωδικοποιητές 2-σε-4, και το ζητούμενο κύκλωμα είναι το ακόλουθο:



ΑΣΚΗΣΗ 80

Να σχεδιαστεί το μικρότερο δυνατό κύκλωμα που να υλοποιεί τη συνάρτηση $F=A+3B+2$ όπου $A=A_3A_2A_1A_0$ είναι ένας δυαδικός αριθμός των 4-bits, $B=B_1B_0$ είναι ένας δυαδικός αριθμός 2 bit. Έχετε στη διάθεση σας μόνο πλήρεις αθροιστές, ημιαθροιστές και πολυπλέκτες 2 σε 1 (χωρίς είσοδο ενεργοποίησης). Κάθε είσοδος είναι διαθέσιμη στην κανονική και συμπληρωματική της μορφή.
(Υπόδειξη: Σχεδιάστε αρχικά το κύκλωμα που υλοποιεί την πράξη $C=3B+2$, και κατόπιν το κύκλωμα που υλοποιεί την πράξη $F=A+C$).

Λύση:

Θα σχεδιάσουμε αρχικά το κύκλωμα που υπολογίζει την τιμή $C=3B+2$. Για να βρούμε από πόσα δυαδικά ψηφία αποτελείται ο αριθμός C, θα πρέπει να βρούμε το πεδίο τιμών του αριθμού. Εφόσον ο αριθμός B αποτελείται από δύο δυαδικά ψηφία, θα παίρνει τις τιμές 0, 1, 2 και 3 (στο δεκαδικό). Έτσι ο αριθμός C θα παίρνει τις τιμές 2, 5, 8, 11 (στο δεκαδικό). Άρα ο αριθμός C θα αποτελείται από 4 δυαδικά ψηφία, τα οποία αρκούν για την αναπαράσταση των παραπάνω αριθμών (0010, 0101, 1000, 1011 αντίστοιχα). Ο παρακάτω πίνακας δείχνει την αντιστοίχιση των αριθμών C στις διάφορες τιμές του B (στο δυαδικό σύστημα).

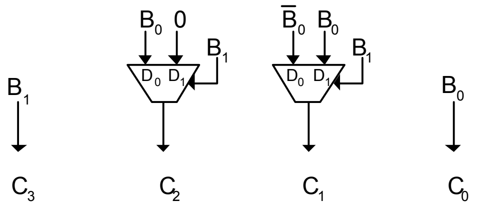
B_1B_0	$C_3C_2C_1C_0$
00	0010
01	0101
10	1000
11	1011

Πρέπει λοιπόν να σχεδιάσουμε ένα συνδυαστικό κύκλωμα που να υπολογίζει τις τιμές του C με βάση τις τιμές του B. Για την σχεδίαση του κυκλώματος αυτού θα χρησιμοποιήσουμε (όπου είναι απαραίτητο) τους πολυπλέκτες. Από τον παραπάνω πίνακα παρατηρούμε ότι $C_3=B_1$, $C_0=B_0$. Τα C_2, C_1 θα υπολογιστούν ως συναρτήσεις του αριθμού B, και θα υλοποιηθούν με χρήση πολυπλεκτών 2 σε 1. Οι παρακάτω πίνακες δείχνουν τις συναρτήσεις αυτές.

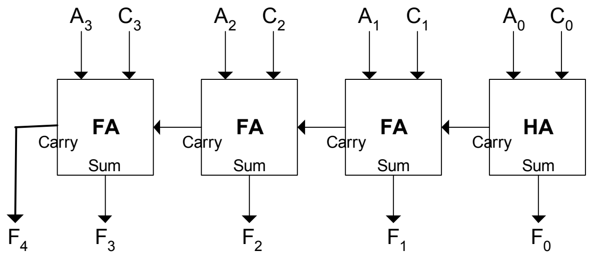
B_1B_0	C_2	
00	0	$D_0=B_0$
01	1	
10	0	$D_1=0$
11	0	

B_1B_0	C_1	
00	1	$D_0=B_0'$
01	0	
10	0	$D_1=B_0$
11	1	

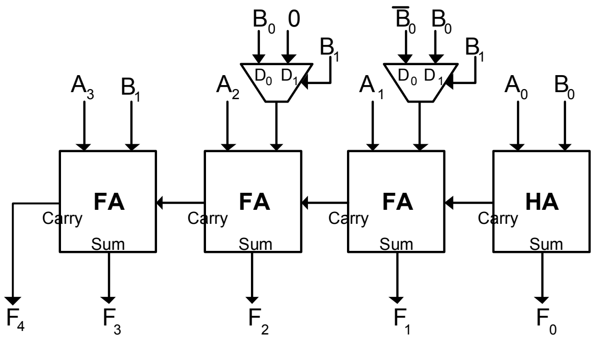
Κάθε μία από τις παραπάνω συναρτήσεις υλοποιείται από έναν πολυπλέκτη 2 σε 1 με γραμμή επιλογής B_1, και εισόδους δεδομένων D_0, D_1 που καθορίζονται από τους παραπάνω πίνακες. Το κύκλωμα υπολογισμού της τιμής C φαίνεται στο ακόλουθο σχήμα:



Τώρα θα πρέπει να σχεδιάσουμε έναν αθροιστή, ο οποίος θα προσθέτει στο A το C. Για την πρόσθεση θα χρησιμοποιηθούν πλήρεις αθροιστές/ημιαθροιστές, όπως φαίνεται στο παρακάτω σχήμα.



Για την πρόσθεση των ψηφίων A_0, C_0 χρησιμοποιείται ημιαθροιστής καθώς σύμφωνα με τη ζητούμενη συνάρτηση F δεν υπάρχει κρατούμενο εισόδου. Επίσης παρατηρούμε ότι ο αριθμός F αποτελείται από 5 δυαδικά ψηφία. Αυτό οφείλεται στην πρόσθεση δύο αριθμών των τεσσάρων δυαδικών ψηφίων ο κάθε ένας. Το 5^{ο} δυαδικό ψηφίο του αποτελέσματος (F_4) είναι ουσιαστικά το κρατούμενο εξόδου του αθροιστή. Τελικά το ζητούμενο κύκλωμα είναι το ακόλουθο:



ΑΣΚΗΣΗ 81

Υλοποιήστε τη συνάρτηση $F(a,b,c)=ab+a'bc+ac$ χρησιμοποιώντας μόνο πολυπλέκτες 2 σε 1 (χωρίς είσοδο ενεργοποίησης).
(Υπόδειξη: Προσπαθήστε να αναπτύξετε τη συνάρτηση F χρησιμοποιώντας το θεώρημα Shannon).

Λύση:

Ένας πολυπλέκτης 2 σε 1 με δύο εισόδους D_0, D_1, μία γραμμή επιλογής S και μία έξοδο G υλοποιεί τη συνάρτηση $G=s'D_0+sD_1$. Θα πρέπει να εκφράσουμε τη συνάρτηση F σε αυτή τη μορφή εφαρμόζοντας επαναληπτικά το θεώρημα του Shannon. Έτσι έχουμε:

$F(a,b,c) = a' \cdot F(0,b,c) + a \cdot F(1,b,c)$ όπου $F(0,b,c) = bc$ και $F(1,b,c) = b+c$.

Τότε η F μπορεί να υλοποιηθεί με έναν πολυπλέκτη με γραμμή επιλογής a και εισόδους $D_0= F(0,b,c)$ και $D_1= F(1,b,c)$. Τώρα αναδρομικά για τις $F(0,b,c)$ και $F(1,b,c)$ εφαρμόζουμε το θεώρημα του Shannon οπότε έχουμε:

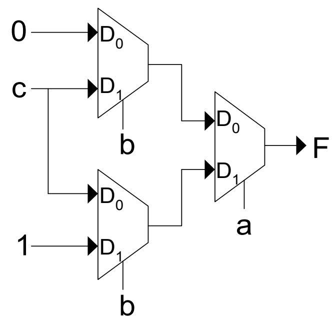
$F(0,b,c) = b' \cdot F(0,0,c) + b \cdot F(0,1,c)$ όπου $F(0,0,c)=0$ και $F(0,1,c)=c$

(πολυπλέκτης με γραμμή επιλογής b και εισόδους $D_0=0$ και $D_1=c$).

$F(1,b,c)= b' \cdot F(1,0,c)+ b \cdot F(1,1,c)$ όπου $F(1,0,c)=c$ και $F(1,1,c)=1$

(πολυπλέκτης με γραμμή επιλογής b και εισόδους $D_0=c$ και $D_1=1$).

Το ζητούμενο κύκλωμα είναι το ακόλουθο:



ΑΣΚΗΣΗ 82

Σχεδιάστε ένα συνδυαστικό κύκλωμα που δέχεται δύο μη προσημασμένους δυαδικούς αριθμούς Α και Β των δύο bit ο κάθε ένας, $A=a_1a_0$ και $B=b_1b_0$, και εκτελεί την πράξη A^B. Υλοποιήστε το κύκλωμα χρησιμοποιώντας αποκωδικοποιητές 3 σε 8 με κανονικές εξόδους και είσοδο επίτρεψης ενεργή στο λογικό 1, καθώς και οποιαδήποτε πύλη θέλετε. (Υπενθύμιση: $0^0=1$).

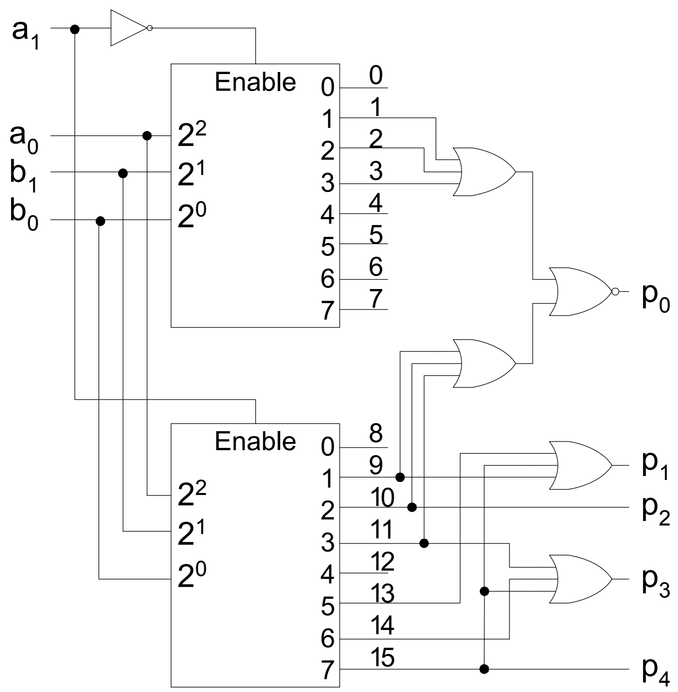
Λύση

Το αποτέλεσμα της πράξης απαιτεί 5 bits, αφού το μεγαλύτερο δυνατό αποτέλεσμα είναι το $3^3=27$, το οποίο απαιτεί 5 bits για την αναπαράσταση του. Αρχικά θα φτιάξουμε τον πίνακα αλήθειας:

a_1 a_0	b_1 b_0	p_4 p_3 p_2 p_1 p_0
0 0	0 0	0 0 0 0 1
0 0	0 1	0 0 0 0 0
0 0	1 0	0 0 0 0 0
0 0	1 1	0 0 0 0 0
0 1	0 0	0 0 0 0 1
0 1	0 1	0 0 0 0 1
0 1	1 0	0 0 0 0 1
0 1	1 1	0 0 0 0 1
1 0	0 0	0 0 0 0 1
1 0	0 1	0 0 0 1 0
1 0	1 0	0 0 1 0 0
1 0	1 1	0 1 0 0 0
1 1	0 0	0 0 0 0 1
1 1	0 1	0 0 0 1 1
1 1	1 0	0 1 0 0 1
1 1	1 1	1 1 0 1 1

Θεωρούμε εισόδους των συναρτήσεων p4, p3, p2, p1, p0 τους αριθμούς Α, Β όπως φαίνονται στον παραπάνω πίνακα. Οι 5 συναρτήσεις θα υλοποιηθούν με χρήση δύο αποκωδικοποιητών 3 σε 8. Ο ένας αποκωδικοποιητής θα υλοποιεί τους ελαχιστόρους 0…7, και ο άλλος τους ελαχιστόρους 8…15. Τότε οι παραπάνω συναρτήσεις υλοποιούνται ως αθροίσματα ελαχιστόρων:

$p_4=\Sigma(15)$, $p_3=\Sigma(11,14,15)$, $p_2=\Sigma(10)$, $p_1=\Sigma(9,13,15)$, $p_0=\Sigma(0,4,5,6,7,8,12,13,14,15)$
Παρατηρούμε ότι η συνάρτηση p_0 μπορεί να γραφεί και ως $p_0=(\Sigma(1,2,3,9,10,11))'$, που υλοποιείται απλούστερα. Το τελικό κύκλωμα φαίνεται στο παρακάτω σχήμα:



ΑΣΚΗΣΗ 83

Υλοποιήστε μία απλή αριθμητική μονάδα η οποία θα λαμβάνει ως εισόδους δύο προσημασμένους αριθμούς Α, Β των 4 δυαδικών ψηφίων που είναι σε παράσταση συμπληρώματος ως προς 2 και θα εκτελεί τις πράξεις που δίνονται στον ακόλουθο Πίνακα. Έχετε στη διάθεση σας μόνο 4 πλήρεις αθροιστές, έναν αποκωδικοποιητή 2 σε 4 με είσοδο ενεργοποίησης και πύλες XOR 2 εισόδων (σημείωση: η αριθμητική μονάδα θα δίνει στην έξοδο της τα 4 δυαδικά ψηφία του αποτελέσματος καθώς και το κρατούμενο εξόδου).

Σήματα Ελέγχου g_2	g_1	g_0	**Αριθμητική Πράξη**
0	X	X	A+B
1	0	0	A+B+1
1	0	1	A-B
1	1	0	B-A
1	1	1	-A-B-1

Λύση

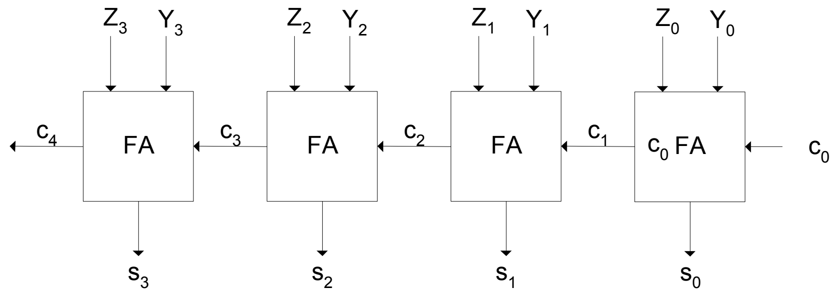
Γνωρίζουμε ότι στην παράσταση συμπληρώματος ως προς 2 η αφαίρεση Z-Y ισοδυναμεί με την πρόσθεση $Z+\overline{Y}+1$ ($\overline{Y}$ είναι το συμπλήρωμα ως προς 1 του Y). Έτσι οι παραπάνω πράξεις αφαίρεσης ισοδυναμούν με τις ακόλουθες πράξεις πρόσθεσης:

$$A\text{-}B=A+\overline{B}+1,\ B\text{-}A=\overline{A}+B+1,\ \text{-}A\text{-}B\text{-}1=\overline{A}+1+\overline{B}+1\text{-}1=\overline{A}+\overline{B}+1$$

οπότε ο πίνακας διαμορφώνεται όπως φαίνεται ακόλουθα

Σήματα Ελέγχου g_2	g_1	g_0	**Αριθμητική Πράξη**
0	X	X	A+B
1	0	0	A+B+1
1	0	1	$A+\overline{B}+1$
1	1	0	$\overline{A}+B+1$
1	1	1	$\overline{A}+\overline{B}+1$

Για τις παραπάνω πράξεις θα χρησιμοποιήσουμε ένα κυματικό αθροιστή των 4 δυαδικών ψηφίων όπως φαίνεται στο ακόλουθο σχήμα.

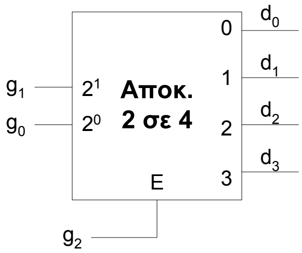


Για να υλοποιήσουμε τις ζητούμενες πράξεις θα πρέπει να σχεδιάσουμε ένα κύκλωμα το οποίο ανάλογα με τις τιμές των σημάτων ελέγχου ($g_2g_1g_0$) θα τροφοδοτεί την είσοδο $Z_{3...0}$ με το $A_{3..0}$ ή το $\overline{A_{3...0}}$ και την είσοδο $Y_{3...0}$ με το $B_{3..0}$ ή το $\overline{B_{3...0}}$ σύμφωνα με την ακόλουθη λογική:

Συμπλήρωση του A όταν $g_2g_1g_0$ = 110, 111

Συμπλήρωση του B όταν $g_2g_1g_0$ = 101, 111

Για την αποκωδικοποίηση αυτών των καταστάσεων θα χρησιμοποιήσουμε τον αποκωδικοποιητή που φαίνεται στο ακόλουθο σχήμα:



Τα σήματα αποκωδικοποίησης d_1, d_2, d_3 είναι στο λογικό ‘1’ όταν τα σήματα ελέγχου της αριθμητικής μονάδας $g_2g_1g_0$ έχουν τις τιμές 101, 110, 111 αντίστοιχα. Οπότε οι συμπληρώσεις των *A, B* γίνονται ως εξής:

Συμπλήρωση του A όταν $d,=1$ *ή* $d_3=1$

Συμπλήρωση του B όταν $d_1=1$ *ή* $d_3=1$

Λαμβάνοντας υπόψη ότι το πολύ ένα από τα σήματα d_1, d_2, d_3 είναι στο λογικό ‘1’ σε κάθε χρονική στιγμή οι συνθήκες συμπλήρωσης των *A, B* γίνονται:

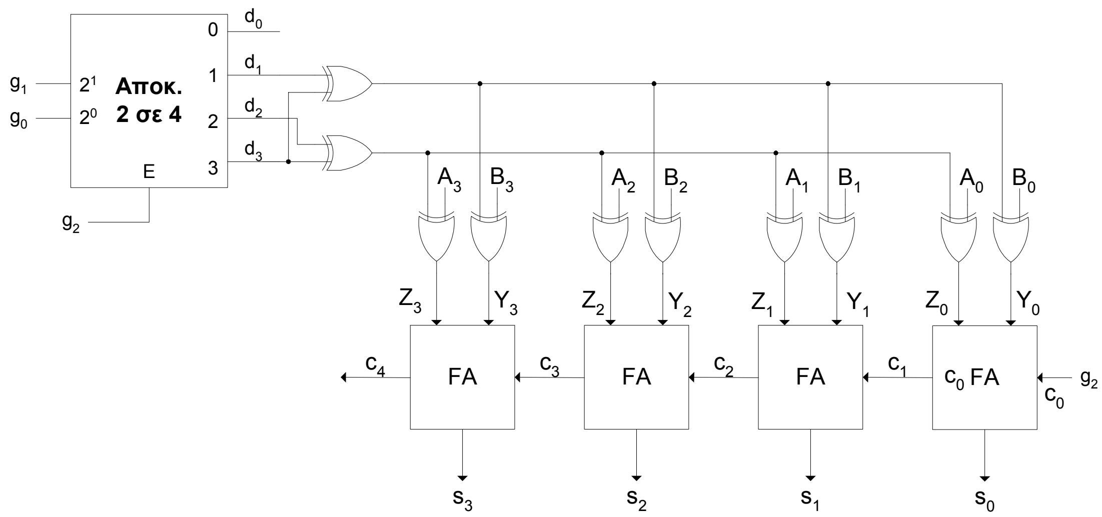
1. Συμπλήρωση του A όταν $d_2=1$ *ή* $d_3=1$, ισοδύναμα όταν d_2d_3=01 ή d_2d_3=10, ισοδύναμα όταν $\overline{d_2}d_3+d_2\overline{d_3}=1$, ισοδύναμα όταν $d_2 \oplus d_3$=1.
2. Συμπλήρωση του B όταν $d_1=1$ *ή* $d_3=1$, ισοδύναμα όταν d_1d_3=01 ή d_1d_3=10, ισοδύναμα όταν $\overline{d_1}d_3+d_1\overline{d_3}=1$, ισοδύναμα όταν $d_1 \oplus d_3$=1.

Γνωρίζουμε ότι το συμπλήρωμα ως προς 1 ενός αριθμού μπορεί να υλοποιηθεί με τη χρήση πυλών XOR. Αυτό μπορεί να φανεί εύκολα με τον πίνακα λειτουργίας της πύλης XOR όπως φαίνεται ακόλουθα:

p	**q**	**p⊕q**
0	0	0
0	1	1
1	0	1
1	1	0

Πράγματι παρατηρούμε ότι όταν $p=1$ τότε $p \oplus q=\overline{q}$ ενώ όταν $p=0$ τότε $p \oplus q=q$. Αντίστοιχα ισχύει ότι όταν $q=1$ τότε $p \oplus q=\overline{p}$ ενώ όταν $q=0$ τότε $p \oplus q=p$. Με άλλα λόγια η πύλη XOR μπορεί να χρησιμοποιηθεί ώστε να περνάει την μία είσοδο της στην έξοδο κανονική ή ανεστραμμένη, ανάλογα με την τιμή της άλλης εισόδου. Αυτή την τεχνική θα χρησιμοποιήσουμε για να τροφοδοτήσουμε την είσοδο $Z_{3...0}$ του αθροιστή με το $A_{3..0}$ ή το $\overline{A_{3...0}}$ και την είσοδο $Y_{3...0}$ με το $B_{3..0}$ ή το $\overline{B_{3...0}}$. Έτσι με βάση τις συνθήκες συμπλήρωσης των *A* και *B* προκύπτει ότι $Z_i=A_i \oplus (d_2 \oplus d_3)$ και $Y_i=B_i \oplus (d_1 \oplus d_3)$ όπου i∈(0…3). Παρατηρήστε ότι όταν τα σήματα ελέγχου $g_2g_1g_0$ έχουν τις τιμές 0XX και 100 τα σήματα αποκωδικοποίησης d_1, d_2, d_3 είναι όλα στο 0 οπότε τα *A, B* περνούν στις εισόδους του αθροιστή χωρίς να αντιστραφούν.

Τέλος από τον πίνακα των αριθμητικών πράξεων φαίνεται ότι στις περιπτώσεις που $g_2g_1g_0$ = 100, 101, 110, 111 πρέπει να προσθέσουμε και τη μονάδα. Για τον λόγο αυτό χρησιμοποιούμε το κρατούμενο εισόδου c_0 του αθροιστή. Παρατηρούμε από τον πίνακα ότι τη μονάδα τη χρειαζόμαστε όταν $g_2=1$ οπότε θέτουμε $c_0=g_2$. Το συνολικό κύκλωμα φαίνεται στο ακόλουθο σχήμα.



ΑΣΚΗΣΗ 84

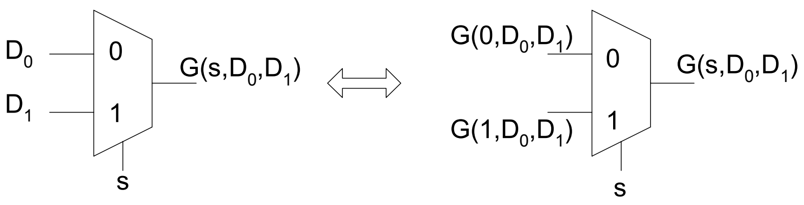
Σχεδιάστε έναν πολυπλέκτη 8 σε 1 χρησιμοποιώντας <u>μόνο</u> πολυπλέκτες 2 σε 1. Δείξτε αναλυτικά την διαδικασία σχεδίασης χρησιμοποιώντας άλγεβρα Boole (θεώρημα Shannon).

Λύση

Ένας πολυπλέκτης 2 σε 1 με δύο εισόδους D_0, D_1, μία γραμμή επιλογής S και μία έξοδο G υλοποιεί τη συνάρτηση $G(s,D_0,D_1)=\overline{s}\,D_0+sD_1$. Γνωρίζουμε ότι ο πολυπλέκτης υλοποιεί το θεώρημα του Shannon ως εξής:

$$G(s,D_0,D_1)=\overline{s}\,D_0+sD_1=\overline{s}\;G(0,D_0,D_1)+s\;G(1,D_0,D_1)$$

αφού $G(0,D_0,D_1)=D_0$ και $G(1,D_0,D_1)=D_1$. Το παρακάτω σχήμα δείχνει αυτή τη λειτουργία του πολυπλέκτη.



Ένας πολυπλέκτης 8 σε 1 με οκτώ εισόδους I_0, I_1, I_2, I_3, I_4, I_5, I_6, I_7, τρεις γραμμές επιλογής $s_2s_1s_0$ και μία έξοδο F υλοποιεί τη συνάρτηση

$F(s_2, s_1, s_0, I_0, I_1, I_2, I_3, I_4, I_5, I_6, I_7)=$

$=\overline{s_2}\,\overline{s_1}\,\overline{s_0}I_0+\overline{s_2}\,\overline{s_1}s_0I_1+\overline{s_2}s_1\overline{s_0}I_2+\overline{s_2}s_1s_0I_3+s_2\overline{s_1}\,\overline{s_0}I_4+s_2\overline{s_1}s_0I_5+s_2s_1\overline{s_0}I_6+s_2s_1s_0I_7$.

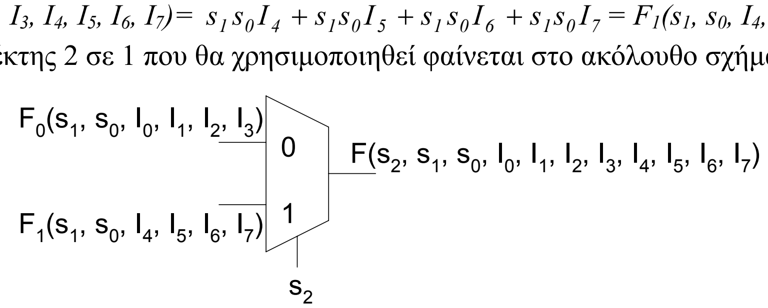
Εφαρμόζουμε στη συνάρτηση F το θεώρημα Shannon ως προς τη μεταβλητή s_2:

$F(s_2,s_1,s_0,I_0,I_1,I_2,I_3,I_4,I_5,I_6,I_7)=\overline{s_2}\;F(0,s_1,s_0,I_0,I_1,I_2,I_3,I_4,I_5,I_6,I_7)+s_2F(1,s_1,s_0,I_0,I_1,I_2,I_3,I_4,I_5,I_6,I_7)$ με

$F(0,s_1,s_0,I_0,I_1,I_2,I_3,I_4,I_5,I_6,I_7)=\overline{s_1}\,\overline{s_0}I_0+\overline{s_1}s_0I_1+s_1\overline{s_0}I_2+s_1s_0I_3=F_0(s_1, s_0, I_0, I_1, I_2, I_3,)$ και

$F(1, s_1, s_0, I_0, I_1, I_2, I_3, I_4, I_5, I_6, I_7)=\overline{s_1}\,\overline{s_0}I_4+\overline{s_1}s_0I_5+s_1\overline{s_0}I_6+s_1s_0I_7=F_1(s_1, s_0, I_4, I_5, I_6, I_7,)$.

Ο πρώτος πολυπλέκτης 2 σε 1 που θα χρησιμοποιηθεί φαίνεται στο ακόλουθο σχήμα.



Εφαρμόζουμε ξανά το θεώρημα Shannon στις συναρτήσεις F_0 και F_1 ως προς τη μεταβλητή s_1:

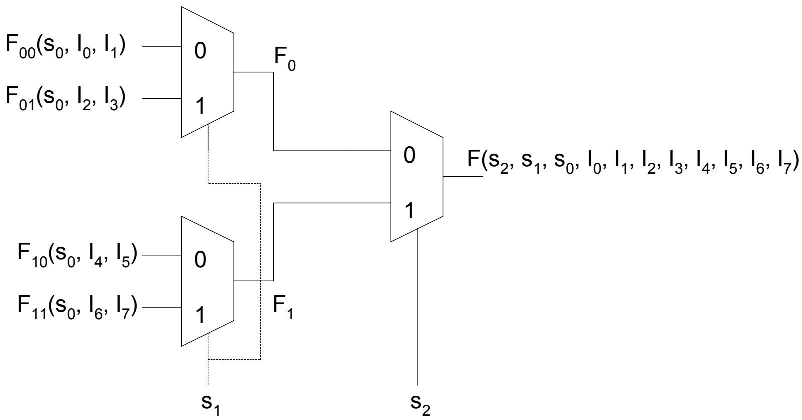
$F_0(0, s_0, I_0, I_1, I_2, I_3,) = \overline{s_0}I_0 + s_0I_1 = F_{00}(s_0, I_0, I_1)$

$F_0(1, s_0, I_0, I_1, I_2, I_3,) = \overline{s_0}I_2 + s_0I_3 = F_{01}(s_0, I_2, I_3)$

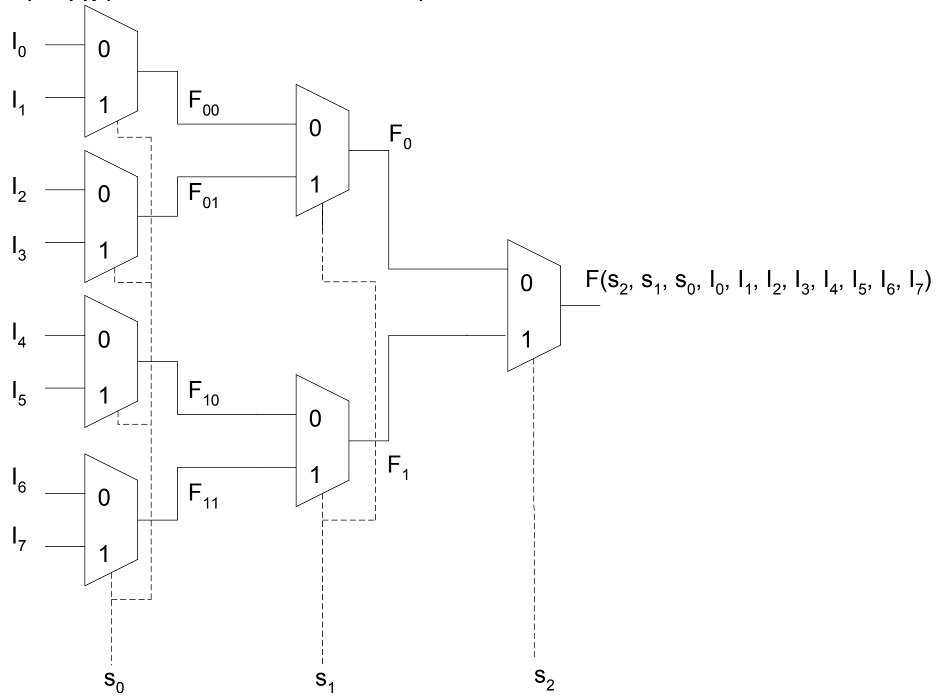
$F_1(0, s_0, I_4, I_5, I_6, I_7,) = \overline{s_0}I_4 + s_0I_5 = F_{10}(s_0, I_4, I_5)$

$F_1(1, s_0, I_4, I_5, I_6, I_7,) = \overline{s_0}I_6 + s_0I_7 = F_{11}(s_0, I_6, I_7)$

Στο ακόλουθο σχήμα φαίνεται το κύκλωμα του πολυπλέκτη όπως έχει διαμορφωθεί ως τώρα



Οι συναρτήσεις F_{00}, F_{01}, F_{10} και F_{11} υλοποιούνται κατευθείαν από πολυπλέκτες 2 σε 1 όπως δείξαμε στην αρχή. Έτσι το συνολικό κύκλωμα είναι το ακόλουθο:



Εναλλακτικά θα μπορούσαμε να χρησιμοποιήσουμε το θεώρημα Shannon ξεκινώντας με τη μεταβλητή s_1 (ή με τη μεταβλητή s_0) και μετά με τις υπόλοιπες μεταβλητές, οπότε θα καταλήγαμε σε ένα παρόμοιο (αλλά όχι ίδιο) με το παραπάνω κύκλωμα με 7 πολυπλέκτες 2 σε 1

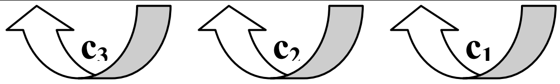
ΑΣΚΗΣΗ 85

Σχεδιάστε ένα συνδυαστικό κύκλωμα το οποίο να δέχεται δύο μη προσημασμένους δυαδικούς αριθμούς A και B των τριών και δύο bit αντίστοιχα ο κάθε ένας, (δηλαδή $A=a_2a_1a_0$ και $B=b_1b_0$), και να υπολογίζει το γινόμενό τους $P=A\cdot B$. Έχετε στην διάθεση σας μόνο πύλες ΚΑΙ (AND), πλήρεις αθροιστές και ημιαθροιστές.

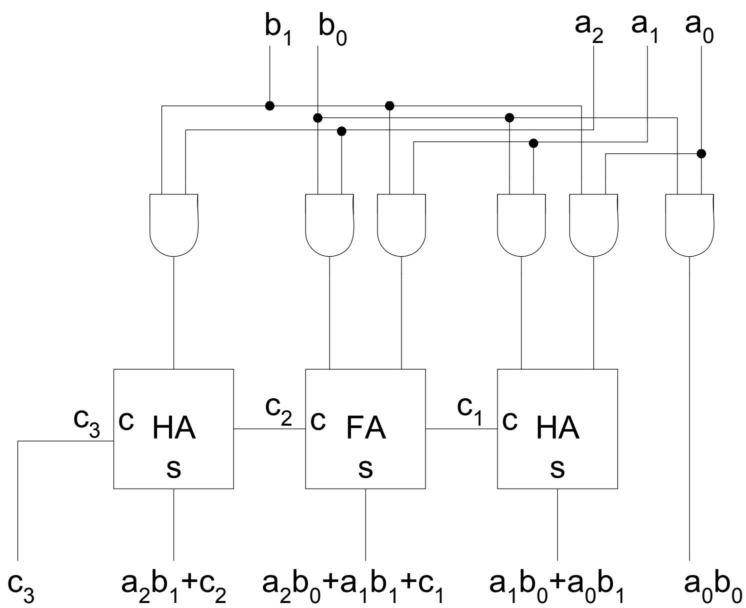
Λύση

Αρχικά αναλύουμε την πράξη του πολλαπλασιασμού ως εξής:

		a_2	a_1	a_0
		x	b_1	b_0
		a_2b_0	a_1b_0	a_0b_0
+	a_2b_1	a_1b_1	a_0b_1	
$\mathbf{c_3}$	$a_2b_1+\mathbf{c_2}$	$a_2b_0+a_1b_1+\mathbf{c_1}$	$a_1b_0+a_0b_1$	a_0b_0

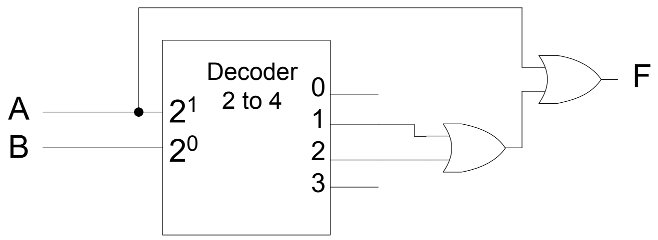


Στον παραπάνω πολλαπλασιασμό τα σύμβολα c_1, c_2, c_3 είναι τα κρατούμενα που προκύπτουν από τις προσθέσεις $a_1b_0+a_0b_1$, $a_2b_0+a_1b_1+\mathbf{c_1}$, και $a_2b_1+\mathbf{c_2}$ αντίστοιχα. Τα μερικά γινόμενα a_ib_j υλοποιούνται με την λογική πράξη ΚΑΙ αφού πρόκειται για μερικά γινόμενα δύο δυαδικών ψηφίων το κάθε ένα. Άρα θα πρέπει να χρησιμοποιήσουμε 6 πύλες ΚΑΙ δύο εισόδων για τα μερικά γινόμενα, δύο ημιαθροιστές για τις προσθέσεις $a_1b_0+a_0b_1$και $a_2b_1+\mathbf{c_2}$, και τέλος έναν πλήρη αθροιστή για την πρόσθεση $a_2b_0+a_1b_1+\mathbf{c_1}$. Το κύκλωμα φαίνεται στο παρακάτω σχήμα.



ΑΣΚΗΣΗ 86

Σε ποια γνωστή λογική πύλη αντιστοιχεί η έξοδος (F) του παρακάτω κυκλώματος;



Λύση

F=(A′B+AB′)+A=A′B+A(B′+1)= A′B+A=A+B, η οποία είναι η πύλη OR.

ΑΣΚΗΣΗ 87

Έχετε έναν αποκωδικοποιητή (decoder) **4 σε 16** καθώς και πύλες **OR δύο εισόδων**. Σχεδιάστε κύκλωμα το οποίο να δέχεται ως είσοδο δύο ακέραιους αριθμούς **A** και **B** σε παράσταση **συμπληρώματος του 2,** των **δύο δυαδικών** ψηφίων ο καθένας, και να παρέχει μία έξοδο G τέτοια ώστε: **G = 1 αν A >B**, και **G = 0 διαφορετικά**.

Λύση

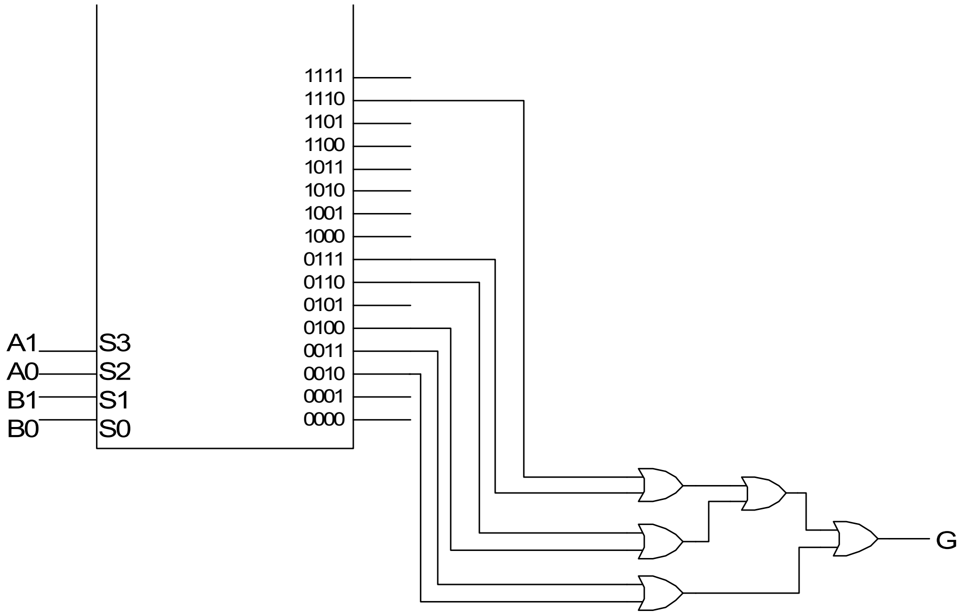
Έστω $A=A_1A_0$ και $B=B_1B_0$. Τότε, δεδομένου ότι σύμφωνα με την αναπαράσταση συμπληρώματος του 2 ισχύει ότι

10	-2
11	-1
00	0
01	1

Καταστρώνουμε τον ακόλουθο πίνακα αληθείας:

$A_1A_0B_1B_0$	G(A>B)
0000	0
0001	0
0010	1
0011	1
0100	1
0101	0
0110	1
0111	1
1000	0
1001	0
1010	0
1011	0
1100	0
1101	0
1110	1
1111	0

Σχηματίζοντας τα αθροίσματα των ελαχιστόρων της συνάρτησης, όπως αυτά παράγεται από τον αποκωδικοποιητή, προκύπτει το ακόλουθο σχήμα:

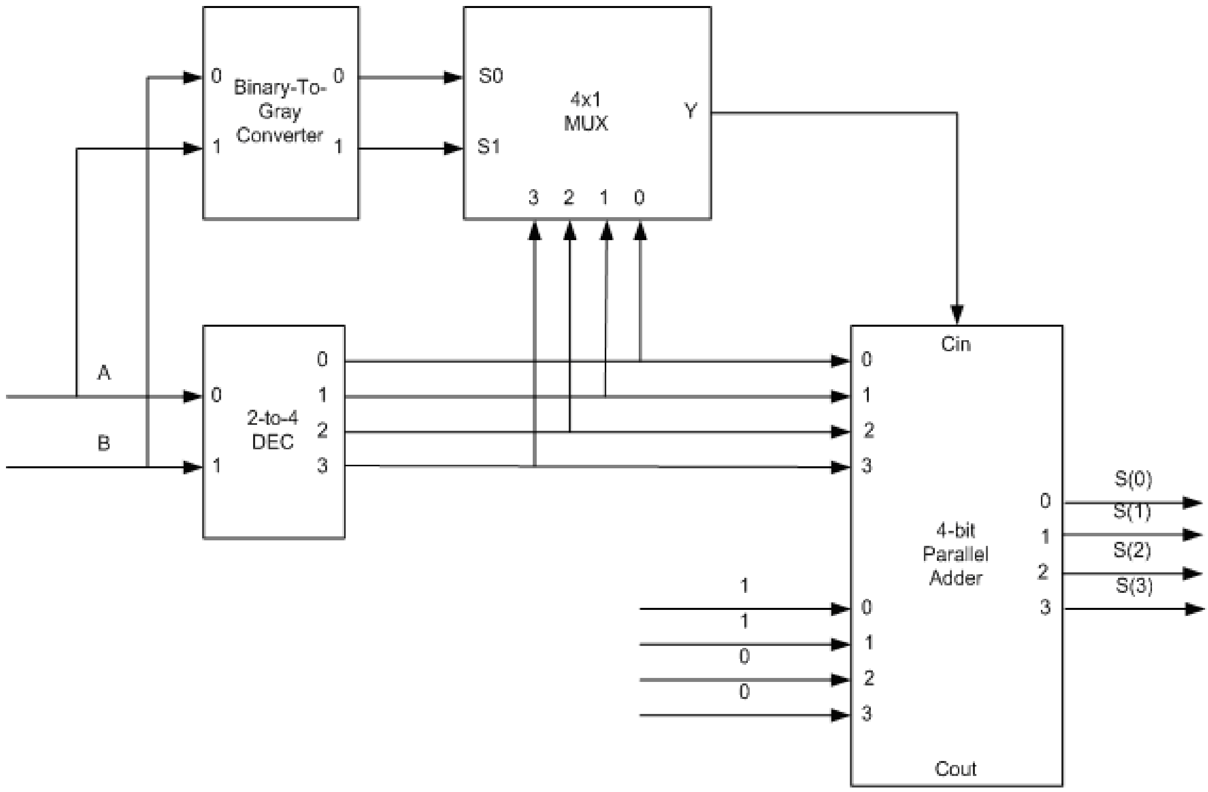


ΑΣΚΗΣΗ 88

Να συμπληρωθεί η δεύτερη και τρίτη γραμμή του πίνακα αλήθειας του κυκλώματος. Να δοθούν οι κατάλληλες επεξηγήσεις όπου αυτό κρίνεται απαραίτητο.

Σημείωση: Για τον πλήρη αθροιστή η είσοδος ή έξοδος '0' αντιστοιχεί στο λιγότερο σημαντικό ψηφίο (LSB), ενώ η είσοδος ή έξοδος '3' αντιστοιχεί στο περισσότερο σημαντικό ψηφίο (MSB).

A	B	Cout	S(3)	S(2)	S(1)	S(0)	Cin
0	0	0	0	1	0	1	1
0	1						
1	0						
1	1	0	1	0	1	1	0



Λύση:

A	B	Cout	S(3)	S(2)	S(1)	S(0)	Cin
0	0	0	0	1	0	1	1
0	1	0	0	1	1	0	1
1	0	0	0	1	1	1	0
1	1	0	1	0	1	1	0

ΑΣΚΗΣΗ 89

Έχετε στη διάθεσή σας μόνο 3 ημιαθροιστές. Υλοποιήστε με αυτούς συγχρόνως όλες τις παρακάτω συναρτήσεις Boole:

$D = A'B'C+A'BC'+AB'C'+ABC$

$E = A'BC + AB'C$

$F = ABC' + A'C + B'C$

$G = ABC$

Λύση

Ο ημιαθροιστής έχει 2 εισόδους (έστω x και y) και 2 εξόδους (έστω s η έξοδος του αθροίσματος και c η έξοδος του κρατουμένου). Το κύκλωμα του ημιαθροιστή υλοποιεί τις συναρτήσεις: $s = x \oplus y$ και $c = xy$. Επομένως πρέπει να μετασχηματίσουμε τις συναρτήσεις που δίνονται ώστε να περιέχουν μόνο λογικές πράξεις XOR και λογικές πράξεις AND.

Έτσι έχουμε:

$D \quad = A'B'C + A'BC' + AB'C' + ABC$

$\quad\;\; = A'(B'C + BC') + A(B'C' + BC)$

$= A'(B \oplus C) + A(B \oplus C)'$

$= A \oplus (B \oplus C) = (A \oplus B) \oplus C$

$= H_1 \oplus C$, *όπου* $H_1 = A \oplus B$

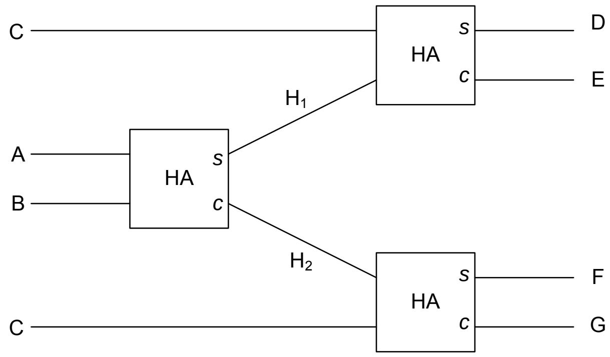
$E \quad = A'BC + AB'C$
$= (A'B + AB')C$
$= (A \oplus B)\, C = H_1\, C$
$F \quad = ABC' + A'C + B'C$
$= ABC' + (A' + B')C$
$= (AB)C' + (AB)'C$
$= (AB) \oplus C = H_2 \oplus C$, *όπου* $H_2 = AB$
$G \quad = ABC$
$= (AB)\, C = H_2\, C$

Επομένως πρέπει να χρησιμοποιήσουμε:

(α) έναν ημιαθροιστή με εισόδους A και B και εξόδους $A \oplus B$ και AB,

(β) έναν ημιαθροιστή με εισόδους $A \oplus B$ και C και εξόδους τις συναρτήσεις D και E, και

(γ) έναν ημιαθροιστή με εισόδους AB και C και εξόδους τις συναρτήσεις F και G, όπως φαίνεται στο παρακάτω σχήμα:



ΑΣΚΗΣΗ 90

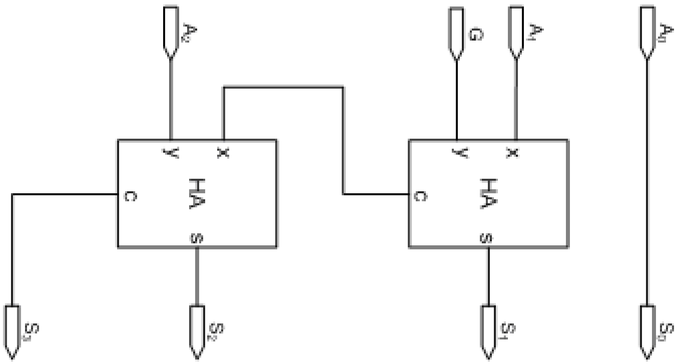
Για τις ανάγκες ενός ψηφιακού συστήματος απαιτείται ένα συνδυαστικό κύκλωμα που θα δέχεται στην είσοδο ένα μη προσημασμένο αριθμό των 3 δυαδικών ψηφίων ($A=A_2A_1A_0$) και θα παράγει στην έξοδο S είτε τον αριθμό A είτε το αποτέλεσμα της αριθμητικής πράξης A+2, ανάλογα με την τιμή μίας εισόδου επιλογής G ($G=0 \rightarrow S=A$, $G=1 \rightarrow S=A+2$). Σχεδιάστε το συνδυαστικό κύκλωμα με τρεις διαφορετικούς τρόπους χρησιμοποιώντας:
(α) μόνο ημιαθροιστές.
(β) αποκωδικοποιητές 2-σε-4 χωρίς είσοδο επίτρεψης και λογικές πύλες OR 2 εισόδων.
(γ) πολυπλέκτες 2-σε-1 χωρίς είσοδο επίτρεψης και λογικές πύλες NOT.
Σε κάθε μία από τις παραπάνω περιπτώσεις θα εκτιμηθεί η χρήση του κατά το δυνατόν μικρότερου αριθμού πυλών ή/και βασικών κυκλωμάτων (ημιαθροιστών, αποκωδικοποιητών, πολυπλεκτών).

Λύση

(α) Αφού ο αριθμός Α μπορεί να πάρει τις τιμές 0 (000) έως και 7 (111), η έξοδος του κυκλώματος πρέπει να εμφανίζει τους αριθμούς 0 (0000) έως και 9 (1001). Επομένως η έξοδος S πρέπει να έχει 4 δυαδικά ψηφία (έστω $S_3S_2S_1S_0$). Όταν $G=1$, τότε πρέπει να προστεθεί στον αριθμό A ο δυαδικός αριθμός 010 ενώ όταν $G=0$, τότε πρέπει να εμφανιστεί στην έξοδο ο αριθμός A ή ισοδύναμα να προστεθεί στον αριθμό A ο δυαδικός αριθμός 000. Με άλλα λόγια αρκεί σε κάθε περίπτωση να προσθέτουμε στον αριθμό A τον δυαδικό αριθμό 0G0. Η πράξη της πρόσθεσης γίνεται όπως φαίνεται παρακάτω:

$$\begin{array}{cccc} & C_2 & C_1 & 0 \\ & A_2 & A_1 & A_0 \\ + & 0 & G & 0 \\ \hline S_3 & S_2 & S_1 & S_0 \end{array}$$

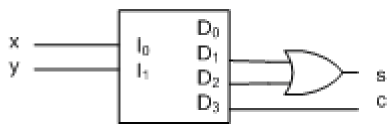
Μπορούμε επομένως να υλοποιήσουμε το κύκλωμα με έναν παράλληλο αθροιστή 3 δυαδικών ψηφίων ο οποίος αποτελείται από 3 πλήρεις αθροιστές με εισόδους $(0,A_0,0)$, (G,A_1,C_1) και $(0,A_2,C_2)$. Παρατηρούμε τα εξής: Ο πρώτος πλήρης αθροιστής εκτελεί την πράξη A_0+0+0. Επομένως η έξοδος αθροίσματός του θα είναι $S_0=A_0$ και η έξοδος κρατουμένου $C_1=0$ και επομένως μπορεί να παραληφθεί. Ο δεύτερος πλήρης αθροιστής εκτελεί την πράξη $G+A_1+C_1$ = $G+A_1$ (αφού $C_1=0$) και μπορεί να αντικατασταθεί από έναν ημιθροιστή με εισόδους τις τιμές G και A_1. Ο τρίτος πλήρης αθροιστής εκτελεί την πράξη $0+A_2+C_2 = A_2+C_2$ και μπορεί να αντικατασταθεί από έναν ημιαθροιστή με εισόδους τις τιμές A_2 και C_2.
Επομένως χρειάζονται μόνο 2 ημιαθροιστές για την υλοποίηση του κυκλώματος το οποίο παρατίθεται στη συνέχεια:



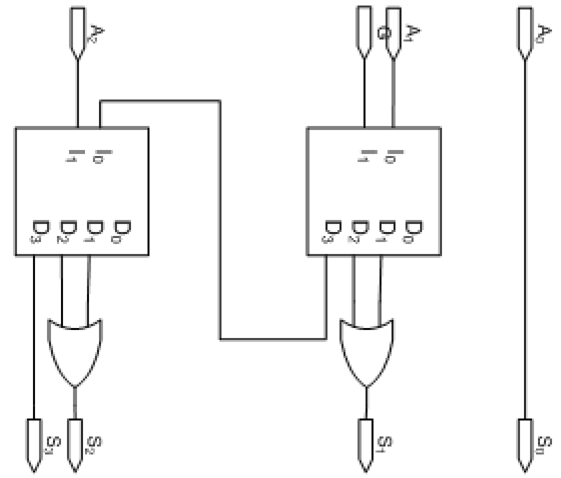
(β) Με έναν αποκωδικοποιητή 2-σε-4 και πύλες OR μπορούμε να υλοποιήσουμε οποιαδήποτε συνάρτηση 2 μεταβλητών, όπως για παράδειγμα την κάθε έξοδο ενός ημιαθροιστή. Άρα, το ζητούμενο κύκλωμα μπορεί να βρεθεί αντικαθιστώντας τον κάθε ημιαθροιστή του κυκλώματος του υποερωτήματος (α) με το αντίστοιχό του κύκλωμα με αποκωδικοποιητή 2-σε-4.
Το πρόβλημά μας ανάγεται στην υλοποίηση ενός ημιαθροιστή με αποκωδικοποιητή. Ο πίνακας αλήθειας του ημιαθροιστή είναι ο παρακάτω:

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

Με βάση τον πίνακα αλήθειας, ο ημιαθροιστής (συνάρτηση 2 μεταβλητών) μπορεί να υλοποιηθεί με έναν αποκωδικοποιητή 2-σε-4 ο οποίος παράγει και τους 4 ελαχιστόρους (D_0, D_1, D_2, D_3) της συνάρτησης 2 μεταβλητών και μία πύλη OR 2 εισόδων η οποία παράγει την έξοδο *s*. Η έξοδος *c* του ημιαθροιστή παράγεται απευθείας από την έξοδο D_3 του αποκωδικοποιητή όπως φαίνεται στο παρακάτω σχήμα.



Επομένως το ζητούμενο κύκλωμα μπορεί να υλοποιηθεί με 2 αποκωδικοποιητές 2-σε-4 και 2 πύλες OR 2 εισόδων όπως φαίνεται στο παρακάτω σχήμα:

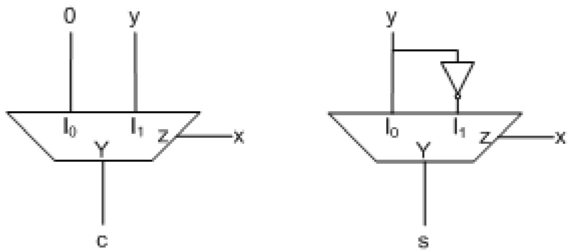


(γ) Κάθε συνάρτηση Boole ν μεταβλητών μπορεί να υλοποιηθεί με έναν πολυπλέκτη $2^{\nu-1}$-σε-1 συνδέοντας κατάλληλα τις ν-1 μεταβλητές στις εισόδους επιλογής και τη ν-οστή μεταβλητή στις υπόλοιπες εισόδους του πολυπλέκτη. Αφού έχουμε στη διάθεσή μας πολυπλέκτες 2-σε-1, μπορούμε να υλοποιήσουμε οποιαδήποτε συνάρτηση Boole 2 μεταβλητών.
Γνωρίζουμε πως ένας πολυπλέκτης 2-σε-1 με εισόδους I_0 και I_1 και είσοδο επιλογής z υλοποιεί τη συνάρτηση $Y = z'I_0 + zI_1$. Οι έξοδοι s και c ενός ημιαθροιστή με εισόδους x και y δίνονται από τις συναρτήσεις:

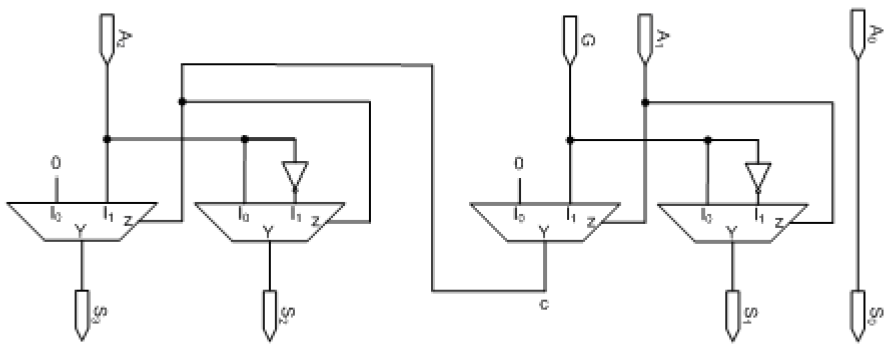
$$s = x \oplus y = x'y + xy' \; (I_0 = y,\ I_1 = y',\ z = x) \text{ και}$$

$$c = xy = x'0 + xy \; (I_0 = 0,\ I_1 = y,\ z = x).$$

Επομένως, ένας ημιαθροιστής μπορεί να υλοποιηθεί με δύο πολυπλέκτες 2^1-σε-1 όπως φαίνεται στο παρακάτω κύκλωμα:



Το ζητούμενο κύκλωμα μπορεί να υλοποιηθεί με πολυπλέκτες 2-σε-1 εάν υλοποιήσουμε κάθε ημιαθροιστή στο κύκλωμα του υποερωτήματος (α) με πολυπλέκτες. Επομένως το ζητούμενο κύκλωμα είναι αυτό που φαίνεται στο παρακάτω σχήμα:



Συμπερασματικά καταλήγουμε ότι η σχεδίαση ενός κυκλώματος μπορεί να γίνει με πάρα πολλούς τρόπους ανάλογα με τα είδη των βασικών κυκλωμάτων που είναι διαθέσιμα (ημιαθροιστές, αποκωδικοποιητές, πολυπλέκτες, κ.λπ) και τα κυκλώματα που προκύπτουν μπορεί να είναι λιγότερο ή περισσότερο πολύπλοκα.

ΑΣΚΗΣΗ 91

Υλοποιήστε την συνάρτηση F(x,y,z,w) = Σ(1,2,5,7,10) χρησιμοποιώντας έναν πολυπλέκτη 4 σε 1 και τον ελάχιστο δυνατό αριθμό λογικών πυλών.

Λύση

Γνωρίζουμε ότι $F(x,y,z,w)=\Sigma(1,2,5,7,10)=\overline{x}\,\overline{y}\,\overline{z}w+\overline{x}\,\overline{y}z\overline{w}+\overline{x}y\overline{z}w+\overline{x}yzw+x\overline{y}z\overline{w}$. Εάν εφαρμόσουμε το θεώρημα του Shannon στην συνάρτηση F για την μεταβλητή x έχουμε

$$F(x,y,z,w)=\overline{x}\cdot F(0,y,z,w)+x\cdot F(1,y,z,w)$$

όπου $F(0,y,z,w)=\overline{y}\,\overline{z}w+\overline{y}z\overline{w}+y\overline{z}w+yzw$, και $F(1,y,z,w)=\overline{y}z\overline{w}$. Εάν εφαρμόσουμε το θεώρημα του Shannon στις συναρτήσεις F(0,y,z,w) και F(1,y,z,w) για την μεταβλητή y έχουμε

$$F(0,y,z,w)=\overline{y}\cdot F(0,0,z,w)+y\cdot F(0,1,z,w) \text{ και } F(1,y,z,w)=\overline{y}\cdot F(1,0,z,w)+y\cdot F(1,1,z,w) \text{ όπου}$$

$$F(0,0,z,w)=\overline{z}w+z\overline{w}=z\oplus w \text{ και } F(0,1,z,w)=w$$

$$F(1,0,z,w)=z\overline{w} \text{ και } F(1,1,z,w)=0$$

Έτσι η συνάρτηση F μπορεί να γραφεί ως

$$F(x,y,z,w)=\overline{x}\,\overline{y}\cdot F(0,0,z,w)+\overline{x}y\cdot F(0,1,z,w)+x\overline{y}\cdot F(1,0,z,w)+xy\cdot F(1,1,z,w)$$

Ένας πολυπλέκτης 4 σε 1 διαθέτει τέσσερις εισόδους δεδομένων D_0, D_1, D_2, D_3, δύο γραμμές επιλογής s_1s_0 και μία έξοδο G, και υλοποιεί τη συνάρτηση

$$G(s_1,s_0,D_0,D_1,D_2,D_3)=\overline{s_1}\,\overline{s_0}D_0+\overline{s_1}s_0D_1+s_1\overline{s_0}D_2+s_1s_0D_3$$

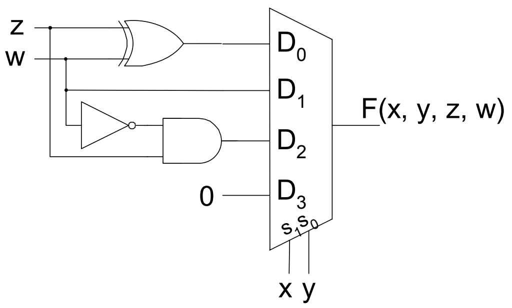
Θέτοντας $s_1=x$, $s_0=y$ η παραπάνω συνάρτηση γίνεται

$$G(x,y,D_0,D_1,D_2,D_3)=\overline{x}\,\overline{y}D_0+\overline{x}yD_1+x\overline{y}D_2+xyD_3$$

Άρα μπορούμε να χρησιμοποιήσουμε τον πολυπλέκτη για να υλοποιήσουμε την συνάρτηση F εάν αντικαταστήσουμε τις εισόδους D_0, D_1, D_2, D_3 από τις συναρτήσεις F(0,0,z,w), F(0,1,z,w), F(1,0,z,w), F(1,1,z,w) αντίστοιχα οπότε έχουμε

$$D_0=z\oplus w$$
$$D_1=w$$
$$D_2=z\overline{w}$$
$$D_3=0$$

Το κύκλωμα που υλοποιεί την συνάρτηση F φαίνεται ακόλουθα

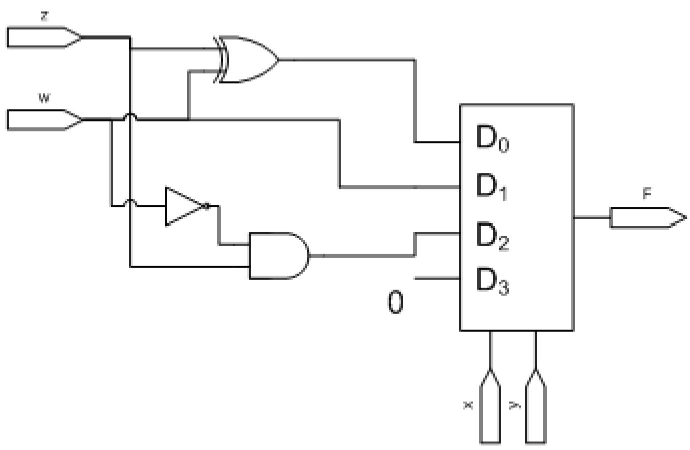


Παρατήρηση: σε ανάλογο αποτέλεσμα θα οδηγηθούμε εάν χρησιμοποιήσουμε διαφορετικό συνδυασμό των εισόδων x, y, z, w για τις εισόδους δεδομένων-επιλογής του πολυπλέκτη.

Άλλος τρόπος

	xyzw	F	
0	0000		$z \oplus w$
1	0001	1	
2	0010	1	
3	0011		
4	0100		w
5	0101	1	
6	0110		
7	0111	1	
8	1000		zw'
9	1001		
10	1010	1	
11	1011		
12	1100		0
13	1101		
14	1110		
15	1111		

Χρησιμοποιώντας ως σήματα επιλογής τα x και y, προκύπτουν για κάθε μια από τις τέσσερεις δυνατές περιπτώσεις, οι εκφράσεις της τελευταίας στήλης. Η σχετική υλοποίηση έχει ως εξής:



ΑΣΚΗΣΗ 92

A) Υλοποιήστε έναν αποκωδικοποιητή **4-σε-16** με είσοδο επίτρεψης χρησιμοποιώντας μόνο αποκωδικοποιητές **2-σε-4** με εισόδους επίτρεψης.

B) Χρησιμοποιώντας τον αποκωδικοποιητή που μόλις σχεδιάσατε, καθώς και λογικές πύλες **2 ή/και 3 εισόδων** η καθεμία, υλοποιήστε τη συνάρτηση $\mathbf{F(w,x,y,z)=xy+yz+wx}\bar{y}$

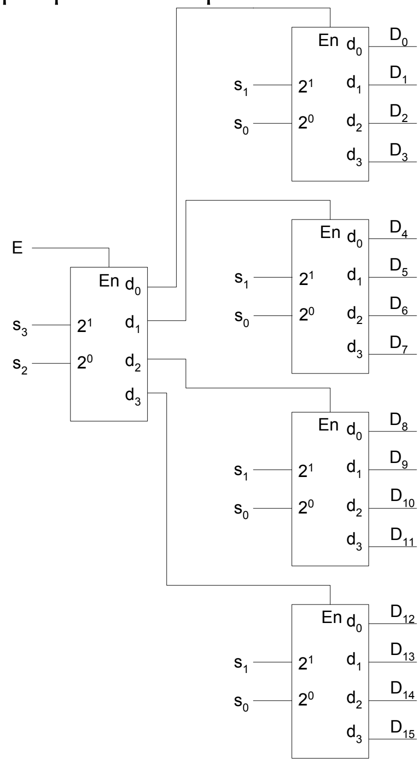
Απάντηση

A) Το ζητούμενο κύκλωμα του αποκωδικοποιητή 4 σε 16 με είσοδο επίτρεψης E, εισόδους επιλογής $s_3s_2s_1s_0$ και εξόδους $D_0 \ldots D_{15}$ έχει τον ακόλουθο πίνακα αλήθειας:

E	s_3s_2	s_1s_0	$D_0D_1D_2D_3$	$D_4D_5D_6D_7$	$D_8D_9D_{10}D_{11}$	$D_{12}D_{13}D_{14}D_{15}$
0	x x	x x	0 0 0 0	0 0 0 0	0 0 0 0	0 0 0 0
1	0 0	0 0	1 0 0 0	0 0 0 0	0 0 0 0	0 0 0 0
		0 1	0 1 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 0 0 0	0 0 0 0
		1 1	0 0 0 1	0 0 0 0	0 0 0 0	0 0 0 0
	0 1	0 0	0 0 0 0	1 0 0 0	0 0 0 0	0 0 0 0
		0 1	0 0 0 0	0 1 0 0	0 0 0 0	0 0 0 0
		1 0	0 0 0 0	0 0 1 0	0 0 0 0	0 0 0 0
		1 1	0 0 0 0	0 0 0 1	0 0 0 0	0 0 0 0
	1 0	0 0	0 0 0 0	0 0 0 0	1 0 0 0	0 0 0 0
		0 1	0 0 0 0	0 0 0 0	0 1 0 0	0 0 0 0
		1 0	0 0 0 0	0 0 0 0	0 0 1 0	0 0 0 0
		1 1	0 0 0 0	0 0 0 0	0 0 0 1	0 0 0 0

	1 1	0 0	0 0 0 0	0 0 0 0	0 0 0 0	1 0 0 0
		0 1	0 0 0 0	0 0 0 0	0 0 0 0	0 1 0 0
		1 0	0 0 0 0	0 0 0 0	0 0 0 0	0 0 1 0
		1 1	0 0 0 0	0 0 0 0	0 0 0 0	0 0 0 1

Από τον παραπάνω πίνακα παρατηρούμε ότι κάθε σκιασμένο μπλοκ μπορεί να υλοποιηθεί από έναν αποκωδικοποιητή 2 σε 4 με γραμμές επιλογής $\mathbf{s_1s_0}$. Η είσοδος επίτρεψης En του κάθε αποκωδικοποιητή 2 σε 4 ενεργοποιείται από τον κατάλληλο συνδυασμό των εισόδων $\mathbf{s_3s_2}$. Έτσι ο αποκωδικοποιητής που αντιστοιχεί στις εξόδους $D_0 \ldots D_3$ ενεργοποιείται όταν s_3s_2=00, ο αποκωδικοποιητής που αντιστοιχεί στις εξόδους $D_4 \ldots D_7$ ενεργοποιείται όταν s_3s_2=01, ο αποκωδικοποιητής που αντιστοιχεί στις εξόδους $D_8 \ldots D_{11}$ ενεργοποιείται όταν s_3s_2=10, ο αποκωδικοποιητής που αντιστοιχεί στις εξόδους $D_{12} \ldots D_{15}$ ενεργοποιείται όταν s_3s_2=11. Ο κάθε αποκωδικοποιητής ενεργοποιείται από την αντίστοιχη είσοδο επίτρεψης En οπότε μπορεί να χρησιμοποιηθεί ένας αποκωδικοποιητής 2 σε 4 με εισόδους επιλογής s_3s_2, και εξόδους E_0, E_1, E_2, E_3 οι οποίες οδηγούν τις επιμέρους εισόδους επίτρεψης En των αποκωδικοποιητών. Άρα το ζητούμενο κύκλωμα είναι το ακόλουθο:



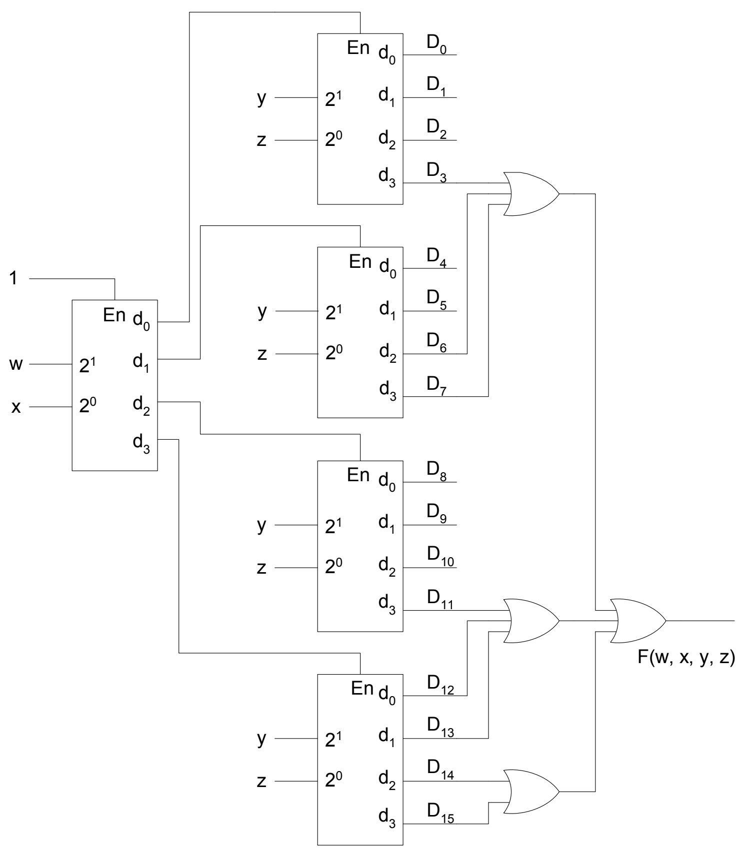
Β) Όπως γνωρίζουμε ένας αποκωδικοποιητής 4 σε 16 μπορεί να υλοποιήσει οποιαδήποτε συνάρτηση 4 μεταβλητών εάν η είσοδος ενεργοποίησης του τεθεί μόνιμα στο 1. Έτσι συνδέουμε τις εισόδους w, x, y, z της συνάρτησης στις εισόδους s_3, s_2, s_1, s_0 αντίστοιχα και μετατρέπουμε την συνάρτηση F σε άθροισμα ελαχιστόρων:

$$F(w,x,y,z)=xy+yz+wx\bar{y} =$$

$$\bar{w}xy\bar{z} + \bar{w}xyz+wxy\bar{z}+wxyz+\bar{w}\bar{x}yz+\bar{w}xyz+w\bar{x}yz+wxyz+wx\bar{y}\bar{z}+wx\bar{y}z=$$

$$\Sigma(3, 6, 7, 11, 12, 13, 14, 15)$$

Την παραπάνω συνάρτηση υλοποιούμε χρησιμοποιώντας 3 πύλες OR τριών εισόδων και μία πύλη OR δύο εισόδων οι οποίες διασυνδέονται στις κατάλληλες εξόδους του αποκωδικοποιητή. Το κύκλωμα φαίνεται στο παρακάτω σχήμα



ΑΣΚΗΣΗ 93

Έστω προσημασμένος αριθμός X σε παράσταση συμπληρώματος ως προς 2 με 4 δυαδικά ψηφία ($X_3X_2X_1X_0$). Να σχεδιαστεί κύκλωμα που να υπολογίζει το $|X|$ (απόλυτη τιμή του αριθμού). Έχετε στη διάθεσή σας μόνο ημιαθροιστές, πολυπλέκτες 2-σε-1 και πύλες NOT.

Λύση:

Ένας προσημασμένος αριθμός X σε παράσταση συμπληρώματος ως προς 2 με n δυαδικά ψηφία αναπαριστά τους ακέραιους αριθμούς που βρίσκονται στο διάστημα $[-2^{n-1}, +2^{n-1}-1]$ (για n=4 το διάστημα γίνεται [-8, +7]). Το περισσότερο σημαντικό ψηφίο (X_3) του δυαδικού αριθμού δηλώνει το πρόσημο του X (0 = +, 1 = –).

Η απόλυτη τιμή ενός αριθμού X, συμβολίζεται με $|X|$ και δίνεται από τη σχέση:

$$|X| = \begin{cases} X & X \geq 0 \\ -X & X < 0 \end{cases}$$

Η έξοδος του κυκλώματος που υπολογίζει την απόλυτη τιμή ενός αριθμού X με 4 δυαδικά ψηφία είναι ένας αριθμός στο διάστημα [0, 8] και επομένως απαιτούνται 4 δυαδικά ψηφιά για την αναπαράστασή του (έστω $Z=Z_3Z_2Z_1Z_0$ η έξοδος του κυκλώματος).

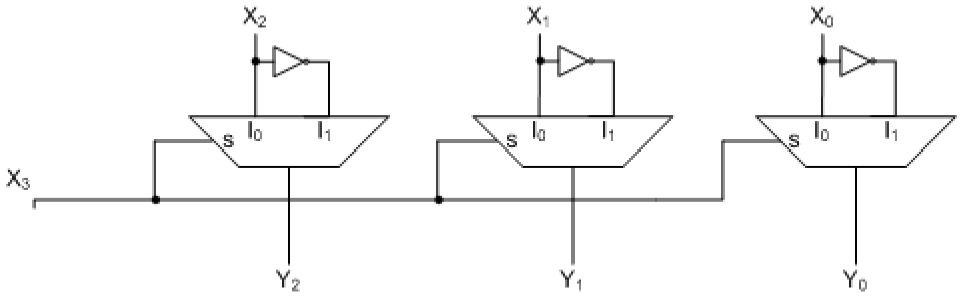
Λύση (Α)

Εάν ο αριθμός X ($=X_3X_2X_1X_0$) είναι αρνητικός, τότε $X_3 = 1$ και η απόλυτη τιμή του αριθμού είναι ίση με το $-X$ ή με άλλα λόγια με το συμπλήρωμα ως προς 2 του αριθμού X. Το συμπλήρωμα ως προς 2 προκύπτει από την πρόσθεση του δυαδικού ψηφίου 1 στα αντεστραμμένα δυαδικά ψηφία του αριθμού (συμπλήρωμα ως προς 1). Επομένως το αποτέλεσμα μπορεί να υπολογιστεί από το άθροισμα $Z = \overline{1}\ \overline{X_2}\ \overline{X_1}\ \overline{X_0} + 1 = \overline{X_2}\ \overline{X_1}\ \overline{X_0} + X_3$.

Εάν ο αριθμός X ($=X_3X_2X_1X_0$) είναι θετικός, τότε $X_3 = 0$ και η απόλυτη τιμή του αριθμού είναι ίση με τον αριθμό X, δηλαδή $Z = 0\ X_2\ X_1\ X_0$. Σε αντιστοιχία με την προηγούμενη περίπτωση το αποτέλεσμα μπορεί να υπολογιστεί από το άθροισμα $Z = X_2\ X_1\ X_0 + 0 = X_2\ X_1\ X_0 + X_3$.

Σε κάθε περίπτωση επομένως η έξοδος του κυκλώματος μπορεί να προκύψει από το άθροισμα ενός αριθμού τριών δυαδικών ψηφίων (έστω $Y_2\ Y_1\ Y_0$) με το δυαδικό ψηφίο X_3.

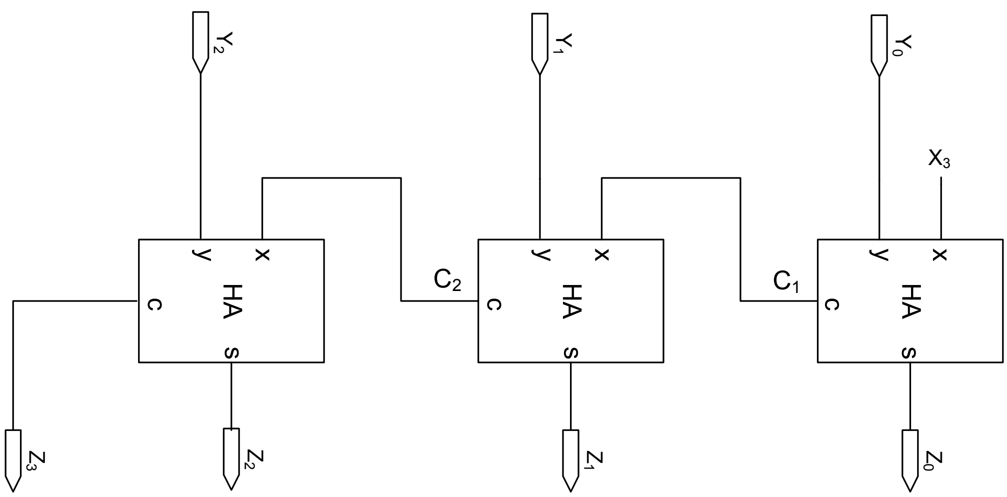
Ισχύει $Y = X_2\ X_1\ X_0$ όταν X_3=0 και $Y = \overline{X_2}\ \overline{X_1}\ \overline{X_0}$ όταν X_3=1 . Επομένως, ο αριθμός Y μπορεί να δημιουργηθεί από έναν πολυπλέκτη 2-σε-1 των 3 δυαδικών ψηφίων. Ο πολυπλέκτης αυτός σχεδιάζεται με 3 πολυπλέκτες 2-σε-1 με κοινή είσοδο επιλογής το X_3 όπως φαίνεται στο παρακάτω σχήμα:



Το κύκλωμα της πρόσθεσης κάνει την ακόλουθη πράξη:

$$\begin{array}{ccccc} & & C_2 & C_1 & \\ & & Y_2 & Y_1 & Y_0 \\ + & & & & X_3 \\ \hline & Z_3 & Z_2 & Z_1 & Z_0 \end{array}$$

Μπορεί επομένως να δημιουργηθεί με τρεις ημιαθροιστές όπως φαίνεται στο παρακάτω σχήμα:



<u>Λύση (Β)</u>

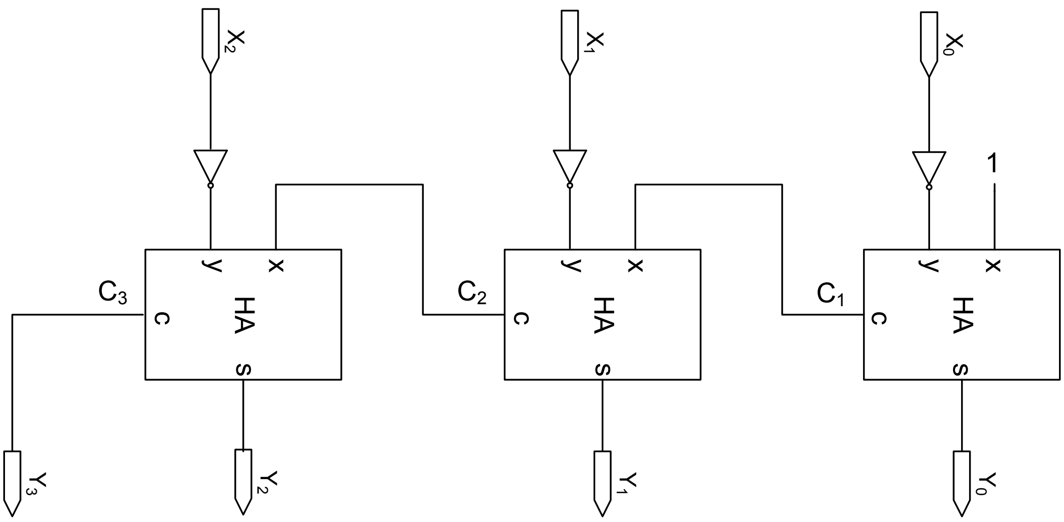
Μία διαφορετική λύση (η οποία όμως χρησιμοποιεί έναν επιπλέον πολυπλέκτη 2-σε-1 σε σχέση με τη λύση (A)) μπορεί να προκύψει εάν ακολουθήσουμε το παρακάτω σκεπτικό:

Με βάση τον ορισμό της απόλυτης τιμής, το ζητούμενο κύκλωμα θα πρέπει να επιλέγει και να εμφανίζει στην έξοδό του τον αριθμό X ή τον αριθμό $-X$ ανάλογα με το αν ο αριθμός είναι θετικός ή αρνητικός. Απαιτείται επομένως: (i) ένα υποκύκλωμα με είσοδο X που να υπολογίζει το $-X$ και (ii) ένα υποκύκλωμα με εισόδους X, $-X$ και X_3 για την επιλογή της τελικής εξόδου.

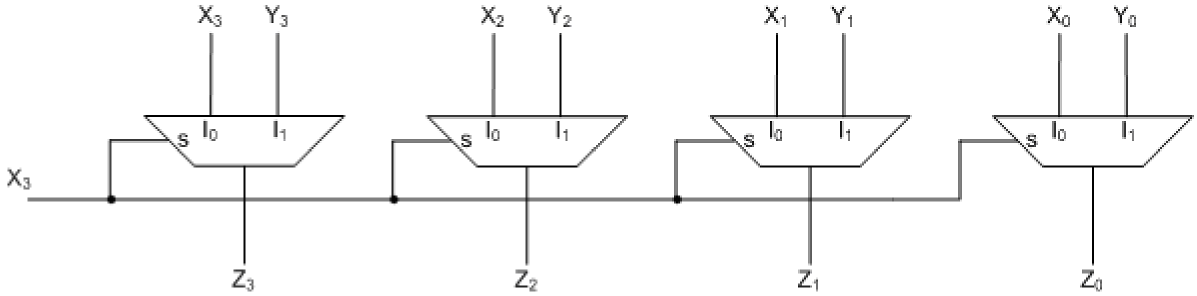
Το υποκύκλωμα (i) υλοποιείται ως εξής: Έστω Y ($=Y_3Y_2Y_1Y_0$) η έξοδος του ζητούμενου υποκυκλώματος. Ο αριθμός $-X$ προκύπτει ως το συμπλήρωμα ως προς 2 του αριθμού X. Το συμπλήρωμα ως προς 2 προκύπτει από την πρόσθεση του δυαδικού ψηφίου 1 στα αντεστραμμένα δυαδικά ψηφία του αριθμού (συμπλήρωμα ως προς 1):

$$\begin{array}{ccccc} & C_3 & C_2 & C_1 & \\ & \overline{X}_3 & \overline{X}_2 & \overline{X}_1 & \overline{X}_0 \\ + & & & & 1 \\ \hline & Y_3 & Y_2 & Y_1 & Y_0 \end{array}$$

Επομένως το ζητούμενο κύκλωμα αποτελείται από 4 πύλες NOT και 4 ημιαθροιστές. Όμως, επειδή το $-X$ απαιτείται μόνο στην περίπτωση που το X είναι αρνητικός αριθμός και επομένως το X_3 έχει την τιμή 1, ο ημιαθροιστής που αθροίζει τα C_3 και $\overline{X}_3$ είναι περιττός (δηλαδή $Y_3 = C_3$) και επομένως αρκούν μόνο 3 ημιαθροιστές και 3 πύλες NOT. Το υποκύκλωμα (i) φαίνεται στο σχήμα που ακολουθεί:



Το υποκύκλωμα (ii) πρέπει να εμφανίζει στην έξοδο το δυαδικό αριθμό X ή το δυαδικό αριθμό Y ανάλογα με την τιμή του ψηφίου X_3. Απαιτείται επομένως ένας πολυπλέκτης 2-σε-1 των 4 δυαδικών ψηφίων. Ο πολυπλέκτης αυτός σχεδιάζεται με 4 πολυπλέκτες 2-σε-1 με κοινή είσοδο επιλογής X_3. Ο κάθε πολυπλέκτης δέχεται στις εισόδους του ένα ζεύγος δυαδικών ψηφίων X_i, Y_i και παράγει στην έξοδο το αντίστοιχο ψηφίο Z_i (i=0…3) όπως φαίνεται στο παρακάτω σχήμα:



ΑΣΚΗΣΗ 94

Έστω $A = a_6\, a_5\, a_4\, a_3\, a_2\, a_1\, a_0$ και $B = b_6\, b_5\, b_4\, b_3\, b_2\, b_1\, b_0$ δύο αριθμοί των 7 δυαδικών ψηφίων. Με τα ψηφία των δύο αριθμών σχηματίζουμε 7 ζεύγη $G_i = (a_i, b_i)$, i=0,…,6.

(Α) Να σχεδιαστεί κύκλωμα που να μετρά το πλήθος των ζευγών G_i στα οποία ισχύει $a_i=b_i$ και το πλήθος των ζευγών G_i στα οποία ισχύει $a_i \neq b_i$.
[Παράδειγμα: Έστω Α = 0001111 και Β = 0010111. Υπάρχουν 5 ζεύγη με την τιμή (0, 0) ή (1, 1) και 2 ζεύγη με την τιμή (0, 1) ή (1,0)].
Έχετε στη διάθεσή σας <u>μόνο</u> αποκωδικοποιητές 2-σε-4, πλήρεις αθροιστές και πύλες OR 2 εισόδων. Ζητείται το κύκλωμα με το ελάχιστο πλήθος αυτών.

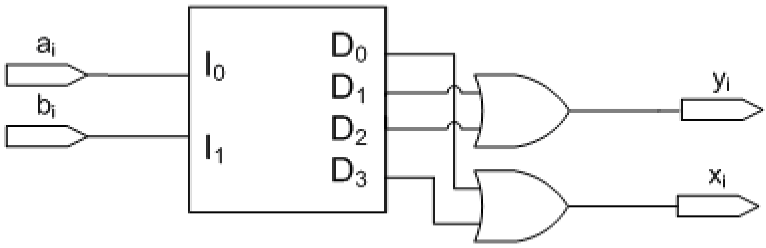
(Β) Εάν είχατε στη διάθεσή σας εκτός των άλλων και πύλες NOT, θα μπορούσατε να απλοποιήσετε περισσότερο το κύκλωμα του ερωτήματος (Α); Αν ναι, με ποιόν τρόπο;

Λύση:

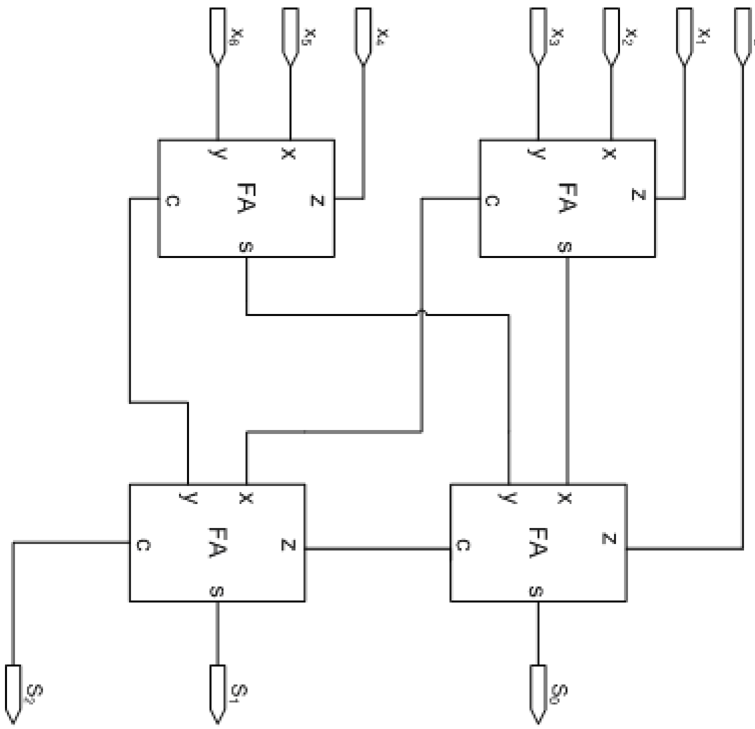
(Α) Για κάθε περίπτωση τιμών ((0,0) ή (0,1) ή (1,0) ή (1,1)) ενός ζεύγους G_i χρειαζόμαστε:
(i) ένα υποκύκλωμα που να δέχεται στην είσοδο τα δύο δυαδικά ψηφία του ζεύγους, a_i και b_i, και να παράγει δύο εξόδους. Η πρώτη έξοδος (έστω x_i) θα παίρνει την τιμή 1 τότε και μόνο τότε όταν τα ψηφία a_i και b_i του ζεύγους G_i είναι όμοια ($a_i=b_i$). Η δεύτερη έξοδος (έστω y_i) θα παίρνει την τιμή 1 τότε και μόνο τότε όταν τα ψηφία a_i και b_i του ζεύγους G_i είναι διαφορετικά ($a_i \neq b_i$).
(ii) ένα υποκύκλωμα που να μετρά το πλήθος των x_i (αντίστοιχα y_i) με τιμή 1.

Θα εξετάσουμε πρώτα το υποκύκλωμα (i). Η λογική συνάρτηση που δίνει την τιμή 1 όταν $a_i=b_i$ (δηλαδή $G_i = (0,0)$ ή $G_i = (1,1)$) και την τιμή 0 όταν $a_i \neq b_i$ (δηλαδή $G_i = (0,1)$ ή $G_i = (1,0)$) είναι η $x_i = a_i' \cdot b_i' + a_i \cdot b_i$ (συνάρτηση XNOR), δηλαδή το άθροισμα των ελαχιστόρων 0 και 3 μίας συνάρτησης δύο μεταβλητών. Η λογική συνάρτηση που δίνει την τιμή 1 όταν $a_i \neq b_i$ (δηλαδή $G_i = (0,1)$ ή $G_i = (1,0)$) και την τιμή 0 όταν $a_i=b_i$ (δηλαδή $G_i = (0,0)$ ή $G_i = (1,1)$) είναι η $y_i = a_i' \cdot b_i + a_i \cdot b_i'$ (συνάρτηση XOR), δηλαδή το άθροισμα των ελαχιστόρων 1 και 2 μίας συνάρτησης δύο μεταβλητών. Προφανώς ισχύει $y_i = x_i'$ και $x_i = y_i'$. Αφού δεν έχουμε στη διάθεσή μας λογικές πύλες εκτός από πύλες OR, θα πρέπει να υλοποιήσουμε τις παραπάνω λογικές συναρτήσεις είτε με ένα κύκλωμα αποκωδικοποιητή 2-σε-4 είτε με ένα κύκλωμα πλήρη αθροιστή. Είναι γνωστό ότι ένας αποκωδικοποιητής 2-σε-4 παράγει στις εξόδους του όλους τους ελαχιστόρους μίας συνάρτησης δύο μεταβλητών τους οποίους μπορούμε να αθροίσουμε με τη βοήθεια μίας πύλης OR 2 εισόδων.

Επομένως το υποκύκλωμα (i) είναι ένας αποκωδικοποιητής 2-σε-4 με εισόδους a_i και b_i. Οι έξοδοι 0 και 3 του αποκωδικοποιητή οδηγούνται σε μία πύλη OR και δημιουργούν την έξοδο x_i ενώ οι έξοδοι 1 και 2 του αποκωδικοποιητή οδηγούνται σε μία πύλη OR και δημιουργούν την έξοδο y_i. Χρησιμοποιούμε 7 τέτοια κυκλώματα, ένα για κάθε ζεύγος G_i, και παράγουμε τα ψηφία x_i και y_i, i=0,...,6.



Πρέπει τώρα να μετρήσουμε το πλήθος των ψηφίων x_i (αντίστοιχα y_i) με τιμή 1 (υποκύκλωμα (ii)). Για το σκοπό αυτό θα χρησιμοποιήσουμε πλήρεις αθροιστές. Ένας πλήρης αθροιστής με εισόδους p, q και r παράγει στην έξοδο του έναν 2-bit αριθμό που αντιστοιχεί στο άθροισμα $p+q+r$. Αφού έχουμε 7 δυαδικά ψηφία, αθροίζουμε τα 6 από αυτά (έστω τα x_6, x_5, x_4, x_3, x_2, x_1) με 2 πλήρεις αθροιστές. Το αποτέλεσμα είναι δύο 2-bit αριθμοί που αντιστοιχούν στο άθροισμα $x_6+x_5+x_4$ και $x_3+x_2+x_1$ αντίστοιχα. Για να βρούμε το συνολικό πλήθος των ψηφίων με τιμή 1 θα πρέπει να αθροίσουμε τους δύο αυτούς αριθμούς μαζί με την τιμή του ψηφίου x_0. Αυτό μπορεί να γίνει πολύ εύκολα με έναν παράλληλο δυαδικό αθροιστή των 2-bit που αποτελείται από δύο πλήρεις αθροιστές με διάδοση κρατουμένου ανάμεσα τους. Επομένως, το υποκύκλωμα (ii), για την περίπτωση της πρόσθεσης των ψηφίων x_i, είναι αυτό που φαίνεται στο σχήμα που ακολουθεί:



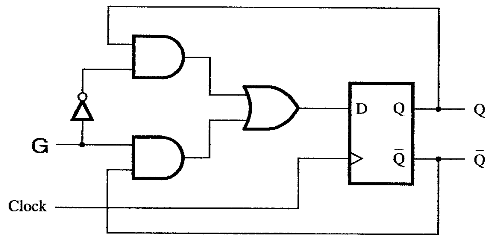
Το συνολικό κύκλωμα περιέχει δύο τέτοια υποκυκλώματα, ένα για τα ψηφία x_i και ένα για τα ψηφία y_i. Όμως, το άθροισμα του πλήθους των όμοιων και διαφορετικών ζευγών των δύο αριθμών A και B είναι ίσο με την τιμή 7 (=111_2). Επομένως εάν γνωρίζουμε το πλήθος των ζευγών με όμοια ψηφία (έστω $S_2S_1S_0$), τότε το πλήθος των ζευγών με διαφορετικά ψηφία είναι $7–S$=111-$S_2S_1S_0$, ή με άλλα λόγια είναι ο δυαδικός αριθμός που προκύπτει από τον S αν αντιστρέψουμε τις τιμές των δυαδικών του ψηφίων. Δυστυχώς όμως δεν έχουμε πύλες NOT στη διάθεσή μας. Μπορούμε όμως να δημιουργήσουμε μία πύλη NOT με έναν αποκωδικοποιητή 2-σε-4. Εάν συνδέσουμε στην είσοδο I_0 του αποκωδικοποιητή το S_i και στην είσοδο I_1 την τιμή 1, τότε η έξοδος D_2 δίνει $D_2 = I_1 \cdot I_0' = 1 \cdot S_i' = S_i'$. Επομένως το υποκύκλωμα (ii) για τα ψηφία y_i μπορεί να αντικατασταθεί από 3 αποκωδικοποιητές 2-σε-4 που λειτουργούν ως πύλες NOT με εισόδους τις εξόδους του υποκυκλώματος (ii) για τα ψηφία x_i. Επιπλέον, στην περίπτωση αυτή, οι πύλες OR που παράγουν τα y_i είναι περιττές και μπορούν να αφαιρεθούν.

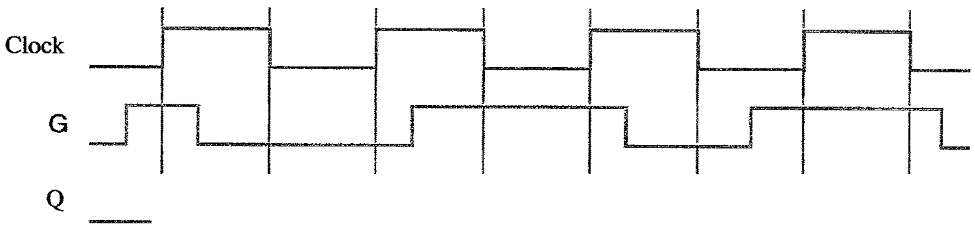
(Β) Μπορούμε να αντικαταστήσουμε τους 3 αποκωδικοποιητές που παράγουν το άθροισμα του πλήθους των διαφορετικών ζευγών των δύο αριθμών με τις αντίστοιχες πύλες NOT και να δημιουργήσουμε ένα απλούστερο κύκλωμα.

VI. Ακολουθιακά κυκλώματα

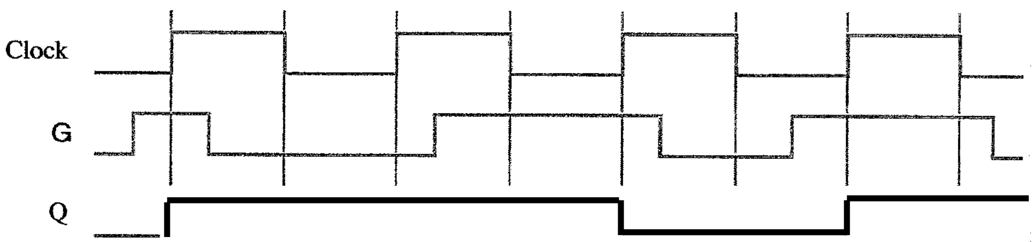
ΑΣΚΗΣΗ 95

Σχεδιάστε την κυματομορφή της εξόδου Q του παρακάτω κυκλώματος. Θεωρείστε ότι αρχικά έχουμε Q=0. Για τι κύκλωμα πρόκειται;





Λύση:



$Q_{t+1} = D = G'Q_t + GQ_t'$. Δηλ., το Q_{t+1} είναι το αποτέλεσμα του «αποκλειστικού-Ή» (XOR) των G και Q_t. Έτσι, όταν η είσοδος G=1, τότε D= Q_t', οπότε λειτουργεί σαν Toggle Flip-Flop, ενώ όταν G=0, τότε $D=Q_t$ και το flip-flop δεν αλλάζει κατάσταση.

ΑΣΚΗΣΗ 96

Έστω Σύγχρονο Ακολουθιακό Κύκλωμα (ΣΑΚ) με 3 D flip flops Q_A, Q_B, Q_C. Αν οι είσοδοι των flip flops είναι αντίστοιχα $D_A=Q'_C$, $D_B=Q_A$, $D_C=(Q_A+Q_C)Q_B$, να σχεδιάσετε τις κυματομορφές των εξόδων του κυκλώματος σε σχέση με τους ωρολογιακούς παλμούς που εφαρμόζονται σ' αυτό. Για τι κύκλωμα πρόκειται; Ποια η χρήση ενός τέτοιου κυκλώματος;

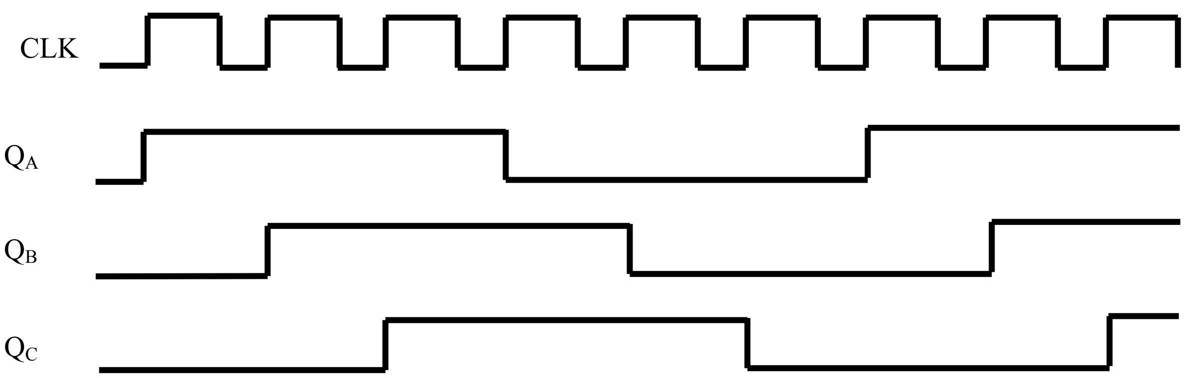
Λύση:
Οι είσοδοι των τριών D flip-flops είναι: $D_A=Q'_C$, $D_B=Q_A$, $D_C=(Q_A+Q_C)Q_B$.

Συνεπώς, έχουμε τον παρακάτω πίνακα μετάβασης καταστάσεων:

Προηγούμενη Κατάσταση			Επόμενη Κατάσταση		
Q_A	Q_B	Q_C	$Q_A=D_A$	$Q_B=D_B$	$Q_C=D_C$
0	0	0	1	0	0
0	0	1	0	0	0
0	1	0	1	0	0
0	1	1	0	0	1
1	0	0	1	1	0
1	0	1	0	1	0
1	1	0	1	1	1
1	1	1	0	1	1

Με βάση τον παραπάνω πίνακα, ξεκινώντας από την αρχική κατάσταση $Q_AQ_BQ_C$=000, η προκύπτουσα αλληλουχία καταστάσεων είναι: 000 → 100 → 110 → 111 → 011 → 001 → 000 → ... . Οι υπόλοιπες καταστάσεις δημιουργούν την αλληλουχία: 101 → 010 → 100 → ....

Το χρονοδιάγραμμα των εξόδων έχει ως εξής:



Το κύκλωμα αυτό είναι ο γνωστός απαριθμητής Johnson, ο οποίος χρησιμοποιείται για τη διαίρεση συχνότητας τετραγωνικών παλμών και κυρίως για τη δημιουργία σημάτων χρονισμού με διαφορές φάσης.

ΑΣΚΗΣΗ 97

Με flip-flops τύπου **JK θετικής ακμής πυροδότησης** να σχεδιάσετε σύγχρονο ακολουθιακό κύκλωμα το οποίο να διατρέχει τις καταστάσεις **0, 4, 6, 3, 1**, 0, 4, 6, ... Υποθέστε ότι όλες οι καταστάσεις που δεν ανήκουν στη ζητούμενη αλληλουχία οδηγούν στην κατάσταση 0. Σχεδιάστε τις κυματομορφές των εξόδων του κυκλώματος σε σχέση με αυτή των ωρολογιακών παλμών.

Λύση:
Καταστρώνουμε τον πίνακα καταστάσεων:

Παρούσα Κατάσταση			Επόμενη Κατάσταση			ΕΙΣΟΔΟΙ των FLIP-FLOPS					
Q_A	Q_B	Q_C	Q_A	Q_B	Q_C	J_A	K_A	J_B	K_B	J_C	K_C
0	0	0	1	0	0	1	X	0	X	0	X
0	0	1	0	0	0	0	X	0	X	X	1
0	1	0	0	0	0	0	X	X	1	0	X
0	1	1	0	0	1	0	X	X	1	X	0
1	0	0	1	1	0	X	0	1	X	0	X

1	0	1	0	0	0	X	1	0	X	X	1
1	1	0	0	1	1	X	1	X	0	1	X
1	1	1	0	0	0	X	1	X	1	X	1

Προχωρώ σε απλοποίηση των J_A, K_A, J_B, K_B, J_C, K_C, με χρήση χαρτών Karnaugh:

$Q_A \backslash Q_B Q_C$	00	01	11	10
0	1			
1	X	X	X	X

$J_A = Q_B'Q_C' = (Q_B + Q_C)'$

$Q_A \backslash Q_B Q_C$	00	01	11	10
0	X	X	X	X
1		1	1	1

$K_A = Q_B + Q_C$

$Q_A \backslash Q_B Q_C$	00	01	11	10
0			X	X
1	1		X	X

$J_B = Q_AQ_C' = (Q_A' + Q_C)'$

$Q_A \backslash Q_B Q_C$	00	01	11	10
0	X	X	1	1
1	X	X	1	

$K_B = Q_A' + Q_C$

$Q_A \backslash Q_B Q_C$	00	01	11	10
0		X	X	
1		X	X	1

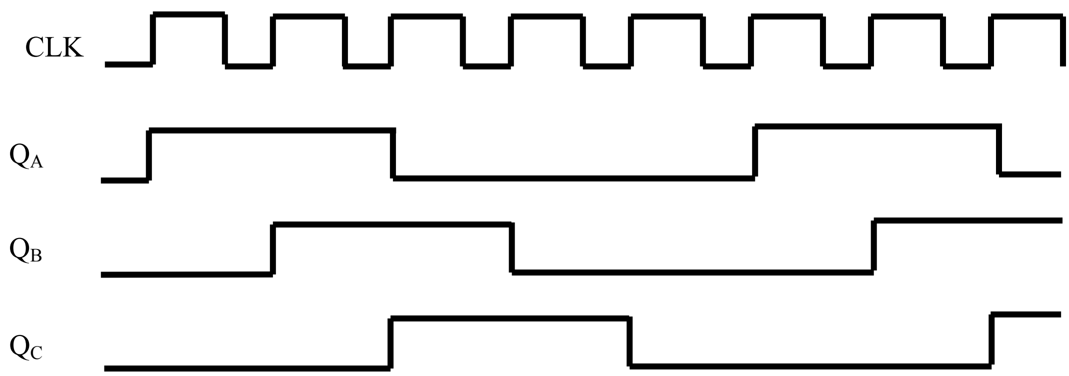
$J_C = Q_AQ_B$

$Q_A \backslash Q_B Q_C$	00	01	11	10
0	X	1		X
1	X	1	1	X

$K_C = Q_A + Q_B'$

Στη συνέχεια, από τις παραπάνω απλοποιημένες εξισώσεις για τα J_A, K_A, J_B, K_B, J_C, K_C, σχεδιάζω το ζητούμενο κύκλωμα.

Οι κυματομορφές των εξόδων του κυκλώματος αυτού φαίνονται παρακάτω:



ΑΣΚΗΣΗ 98

Με flip-flops τύπου **D αρνητικής ακμής πυροδότησης** να σχεδιάσετε σύγχρονο ακολουθιακό κύκλωμα το οποίο να διατρέχει τις καταστάσεις **0,1,2,3,4,5,6,7**,0,1,2,...

Λύση:
Καταστρώνουμε τον πίνακα καταστάσεων:

Παρούσα Κατάσταση			Επόμενη Κατάσταση			Είσοδοι FF		
Q_A	Q_B	Q_C	Q_A	Q_B	Q_C	D_A	D_B	D_C
0	0	0	0	0	1	0	0	1
0	0	1	0	1	0	0	1	0
0	1	0	0	1	1	0	1	1
0	1	1	1	0	0	1	0	0
1	0	0	1	0	1	1	0	1
1	0	1	1	1	0	1	1	0
1	1	0	1	1	1	1	1	1
1	1	1	0	0	0	0	0	0

Στη συνέχεια, προχωρούμε σε απλοποίηση των D_A, D_B, D_C, με χρήση χαρτών Karnaugh:

Q_A \ $Q_B Q_C$	00	01	11	10
0			1	
1	1	1		1

$D_A = Q_A'Q_BQ_C + Q_A(Q_B'+Q_C') =$
$= Q_A'(Q_BQ_C) + Q_A(Q_BQ_C)' =$
$= XOR\ (Q_A, Q_BQ_C)$

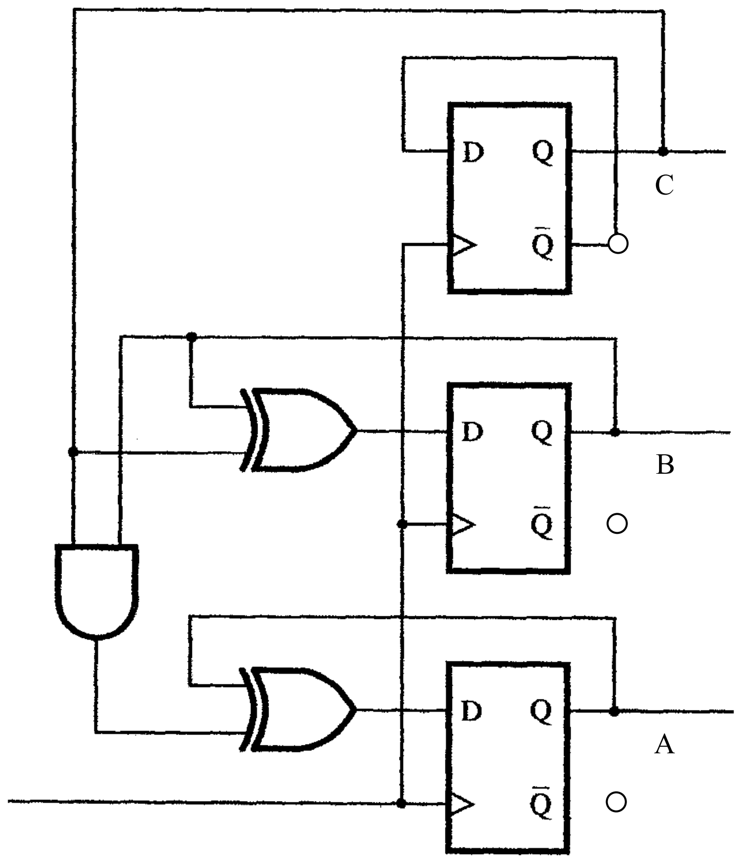
Q_A \ $Q_B Q_C$	00	01	11	10
0		1		1
1		1		1

$D_B = XOR\ (Q_B, Q_C)$

Q_A \ $Q_B Q_C$	00	01	11	10
0	1			1
1	1			1

$D_C = Q_C'$

Το κύκλωμα ακολουθεί παρακάτω:



ΑΣΚΗΣΗ 99

Α) Μετατρέψτε ένα D flip-flop θετικής ακμής πυροδότησης σε ένα T flip-flop τοποθετώντας πύλη/πύλες στην είσοδο του D flip-flop.

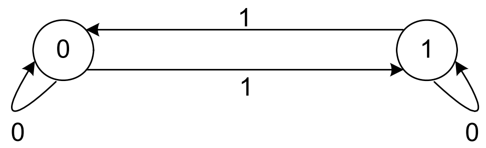
Β) Σχεδιάστε ένα JK flip-flop χρησιμοποιώντας ένα D flip-flop θετικής ακμής πυροδότησης και ένα πολυπλέκτη 2-σε-1 (χωρίς είσοδο ενεργοποίησης). Κάθε είσοδος είναι διαθέσιμη στην κανονική και συμπληρωματική της μορφή.

Λύση:

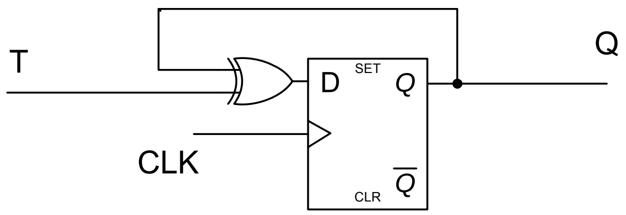
Α) $1^{ος}$ τρόπος: Από τον πίνακα λειτουργίας του T flip-flop παρατηρούμε ότι η επόμενη κατάστασή του προκύπτει από τη σχέση: $Q(t+1) = T'Q(t) + TQ'(t)$. Αντίστοιχα η επόμενη κατάσταση ενός D flip-flop προκύπτει από τη σχέση: $Q(t+1) = D$. Επομένως για να μετατρέψουμε ένα D flip-flop σε ένα T flip-flop θα πρέπει να συνδέσουμε στην είσοδο του D flip-flop την έξοδο της συνάρτησης: $D = T'Q(t) + TQ'(t) = T \oplus Q(t)$, δηλαδή μία πύλη XOR με εισόδους την έξοδο Q του D flip-flop και την είσοδο T.

$2^{ος}$ τρόπος: Ακολουθούμε τη γενική διαδικασία σχεδιασμού ενός σύγχρονου ακολουθιακού κυκλώματος. Κατασκευάζουμε τον πίνακα και το διάγραμμα καταστάσεων, προσδιορίζουμε τη συνάρτηση εισόδου του D flip-flop χρησιμοποιώντας τον πίνακα διέγερσης αυτού, απλοποιούμε τη συνάρτηση αυτή και σχεδιάζουμε το λογικό κύκλωμα:

Παρούσα κατάσταση	**Είσοδος**	*Επόμενη κατάσταση*	*Είσοδος D-FF*
Q	***T***	***Q***	***D***
0	0	0	0
0	1	1	1
1	0	1	1
1	1	0	0



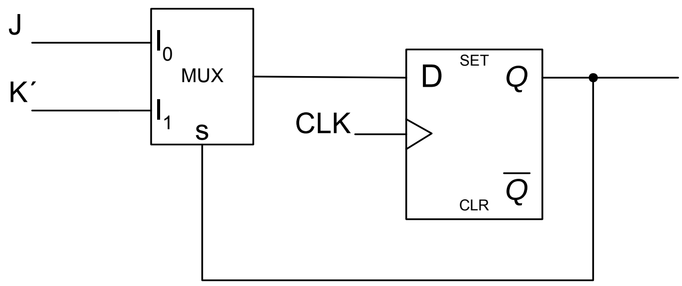
Από τον πίνακα καταστάσεων προκύπτει ότι $D = T \oplus Q$.



B) Από τον πίνακα λειτουργίας του JK flip-flop

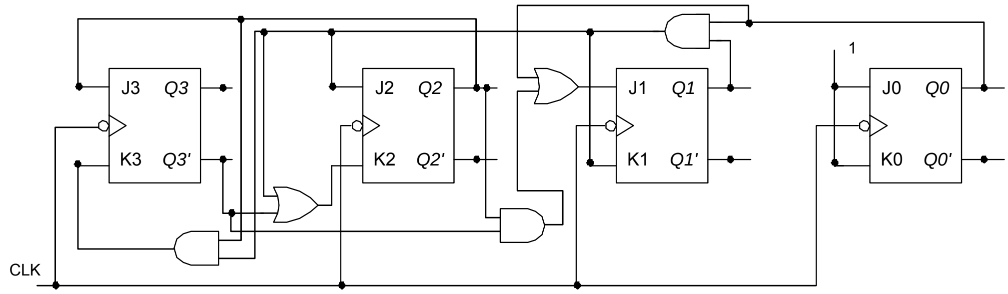
J	*K*	*Q(t)*	*Q(t+1)*
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

παρατηρούμε ότι η επόμενη κατάστασή του προκύπτει από τη σχέση: $Q(t+1) = JQ'(t) + K'Q(t)$. Αντίστοιχα η επόμενη κατάσταση ενός D flip-flop προκύπτει από τη σχέση: $Q(t+1) = D$. Επομένως για να μετατρέψουμε ένα D flip-flop σε ένα JK flip-flop θα πρέπει να συνδέσουμε στην είσοδο του D flip-flop την έξοδο της συνάρτησης: $D = JQ'(t) + K'Q(t)$, δηλαδή την έξοδο ενός πολυπλέκτη 2-σε-1 με τις δύο εισόδους του I_0 και I_1 συνδεδεμένες στα J και K' αντίστοιχα και την είσοδο επιλογής s συνδεδεμένη στην έξοδο Q του D flip-flop όπως φαίνεται στο παρακάτω σχήμα.



ΑΣΚΗΣΗ 100

A) Να προσδιοριστεί ο τρόπος λειτουργίας του Σύγχρονου Ακολουθιακού Κυκλώματος (ΣΑΚ) με FFs τύπου JK αρνητικής ακμής πυροδότησης, που ακολουθεί.



Β) Μπορεί το παραπάνω κύκλωμα να χαρακτηριστεί ως κύκλωμα «Αυτόματης Διόρθωσης» ;

Λύση:

Α) Το πρώτο βήμα είναι να γράψουμε τις συναρτήσεις εισόδου των FF's.

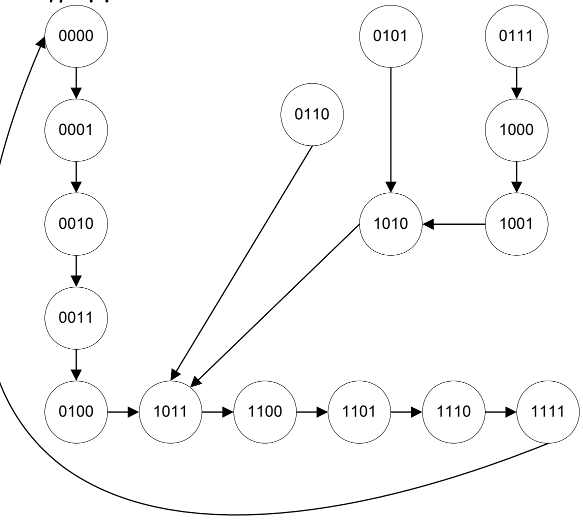
$J_0 = 1$ $\quad J_1 = (Q_2\, Q_3{'}) + Q_0 \quad J_2 = Q_0\, Q_1 \quad J_3 = Q_2$

$K_0 = 1$ $\quad K_1 = Q_0\, Q_1 \quad K_2 = (Q_0\, Q_1) + Q_3{'} \quad K_3 = (Q_0\, Q_1)\, Q_2$

Στη συνέχεια καταστρώνουμε τον Πίνακα Καταστάσεων. Εφόσον το κύκλωμα αποτελείται από 4 FFs οι δυνατές καταστάσεις είναι $2^4 = 16$ και συνεπώς κάθε τμήμα του Πίνακα Καταστάσεων θα αποτελείται από 16 γραμμές.

ΠΑΡΟΥΣΑ ΚΑΤΑΣΤΑΣΗ				***ΕΙΣΟΔΟΙ FFs***				***ΕΠΟΜΕΝΗ ΚΑΤΑΣΤΑΣΗ***			
Q_3	Q_2	Q_1	Q_0	J_3K_3	J_2K_2	J_1K_1	J_0K_0	Q_3	Q_2	Q_1	Q_0
0	0	0	0	00	01	00	11	0	0	0	1
0	0	0	1	00	01	10	11	0	0	1	0
0	0	1	0	00	01	00	11	0	0	1	1
0	0	1	1	00	11	11	11	0	1	0	0
0	1	0	0	10	01	10	11	1	0	1	1
0	1	0	1	10	01	10	11	1	0	1	0
0	1	1	0	10	01	10	11	1	0	1	1
0	1	1	1	11	11	11	11	1	0	0	0
1	0	0	0	00	00	00	11	1	0	0	1
1	0	0	1	00	00	10	11	1	0	1	0
1	0	1	0	00	00	00	11	1	0	1	1
1	0	1	1	00	11	11	11	1	1	0	0
1	1	0	0	10	00	00	11	1	1	0	1
1	1	0	1	10	00	10	11	1	1	1	0
1	1	1	0	10	00	00	11	1	1	1	1
1	1	1	1	11	11	11	11	0	0	0	0

Στη συνέχεια σχεδιάζουμε το διάγραμμα καταστάσεων.



Βλέπουμε ότι το κύκλωμα διατρέχει διαδοχικά τις καταστάσεις $Q_3Q_2Q_1Q_0$: 0, 1, 2, 3, 4, 11, 12, 13, 14, 15 (απαριθμητής 10 καταστάσεων σε κωδικοποίηση με βάρη 2421).

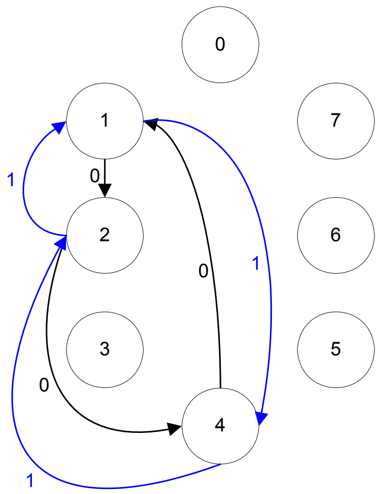
Β) Από το διάγραμμα καταστάσεων διαπιστώνουμε ότι ακόμα και αν το κύκλωμα βρεθεί σε μια μη έγκυρη κατάσταση (5, 6, 7, 8, 9, 10), μετά από τέσσερις ωρολογιακούς παλμούς (στη χειρότερη των περιπτώσεων) μεταβαίνει σε έγκυρη κατάσταση και συνεχίζει το κύκλο λειτουργίας του κανονικά. Ως εκ τούτου το κύκλωμα μπορεί να χαρακτηριστεί ως κύκλωμα «Αυτόματης Διόρθωσης».

ΑΣΚΗΣΗ 101

Α) Σχεδιάστε ένα Σύγχρονο Ακολουθιακό Κύκλωμα (ΣΑΚ) με 3 FFs τύπου D αρνητικής ακμής πυροδότησης, το οποίο να διατρέχει τις καταστάσεις 001, 010, 100, 001 ... ή 100, 010, 001, 100 ... ανάλογα με το αν μια εξωτερική είσοδος *X* είναι 0 ή 1 αντίστοιχα.

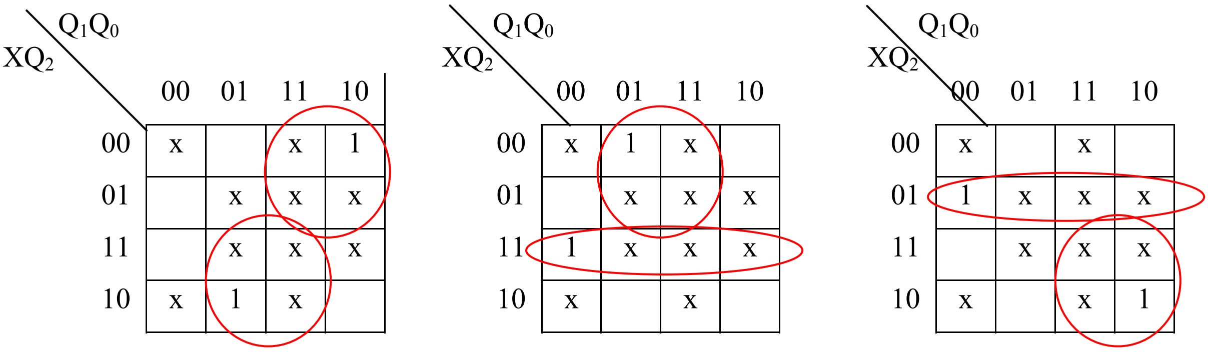
Β) Επαληθεύστε τη σωστή λειτουργία του κυκλώματος που σχεδιάσατε στο (Α) (για παράδειγμα αν η λειτουργία του κυκλώματος ξεκινήσει από μια μη έγκυρη κατάσταση, θα μπορέσει το κύκλωμα, μετά από κάποιο αριθμό ωρολογιακών παλμών, να μεταβεί σε κάποια έγκυρη κατάσταση και να συνεχίσει το κύκλο λειτουργίας του κανονικά ;). Αν διαπιστωθεί πρόβλημα στη λειτουργία να αντιμετωπιστεί τροποποιώντας κατάλληλα την αρχική σχεδίαση.

Λύση:
Α) Το πρώτο βήμα είναι να σχεδιάσουμε το διάγραμμα καταστάσεων και στη συνέχεια να καταστρώσουμε τον πίνακα καταστάσεων.
Αν και το πλήθος των καταστάσεων που διατρέχει το ζητούμενο κύκλωμα είναι 3, επειδή χρειάζεται να αποτυπώνεται η κατάσταση 100, καταλαβαίνουμε γιατί το κύκλωμά μας θα πρέπει να αποτελείται από 3 FFs τύπου D. Καθώς μας είναι αδιάφορο σε ποια κατάσταση θα βρεθεί το κύκλωμα μετά την κατάσταση 0, 3, 5, 6 και 7, το διάγραμμα καταστάσεων θα είναι όπως φαίνεται δίπλα. Οι αριθμοί στα βέλη του διαγράμματος καταστάσεων υποδηλώνουν τις αντίστοιχες τιμές της εισόδου *X*. Η επόμενη κατάσταση προκύπτει από το διάγραμμα καταστάσεων, ενώ οι τιμές των εισόδων των FFs προκύπτουν από τις τιμές της παρούσας και επόμενης κατάστασης καθενός FF σε συνδυασμό με τον πίνακα διέγερσης του FF. Ο Πίνακας καταστάσεων ακολουθεί.



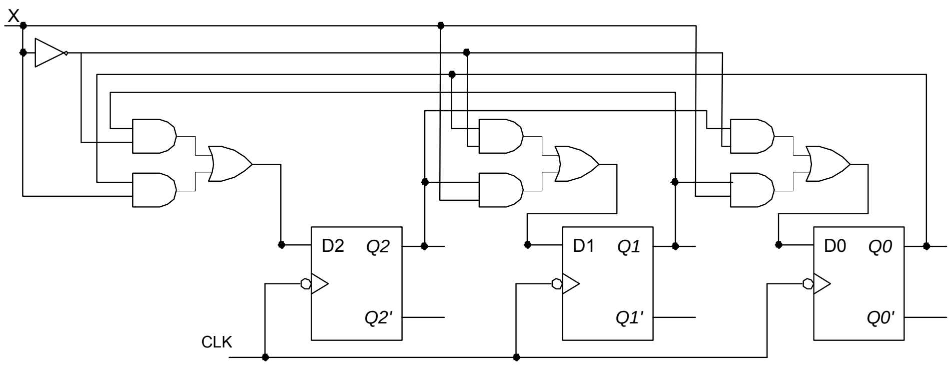
ΕΙΣΟΔΟΣ και ΠΑΡΟΥΣΑ ΚΑΤΑΣΤΑΣΗ				*ΕΠΟΜΕΝΗ ΚΑΤΑΣΤΑΣΗ*			*ΕΙΣΟΔΟΙ FFs*		
X	Q_2	Q_1	Q_0	Q_2	Q_1	Q_0	D_2	D_1	D_0
0	0	0	0	x	x	x	x	x	x
0	0	0	1	0	1	0	0	1	0
0	0	1	0	1	0	0	1	0	0
0	0	1	1	x	x	x	x	x	x
0	1	0	0	0	0	1	0	0	1
0	1	0	1	x	x	x	x	x	x
0	1	1	0	x	x	x	x	x	x
0	1	1	1	x	x	x	x	x	x
1	0	0	0	x	x	x	x	x	x
1	0	0	1	1	0	0	1	0	0
1	0	1	0	0	0	1	0	0	1
1	0	1	1	x	x	x	x	x	x
1	1	0	0	0	1	0	0	1	0
1	1	0	1	x	x	x	x	x	x
1	1	1	0	x	x	x	x	x	x
1	1	1	1	x	x	x	x	x	x

Τώρα πρέπει να προσδιορίσουμε τις απλοποιημένες συναρτήσεις εισόδου των FFs με χρήση χαρτών Karnaugh, όπως φαίνεται στη συνέχεια.



$D_2 = X'\,Q_1 + X\,Q_0$ $\qquad$ $D_1 = X'\,Q_0 + X\,Q_2$ $\qquad$ $D_0 = X'\,Q_2 + X\,Q_1$

Γνωρίζοντας πλέον τις συναρτήσεις της εισόδου κάθε FF, προχωρούμε στη σχεδίαση του λογικού κυκλώματος, που είναι και το ζητούμενο κύκλωμα.



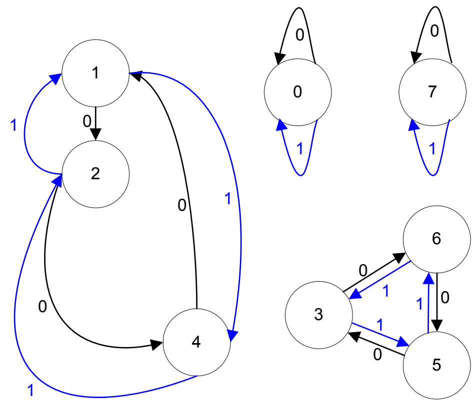
Β) Για να επαληθεύσουμε τη λειτουργία του κυκλώματος που μόλις σχεδιάσαμε θα πρέπει να το «αναλύσουμε».Το πρώτο βήμα είναι να γράψουμε τις συναρτήσεις εισόδου των FF’s.

$D_2 = X'\,Q_1 + X\,Q_0$ $\qquad$ $D_1 = X'\,Q_0 + X\,Q_2$ $\qquad$ $D_0 = X'\,Q_2 + X\,Q_1$

Στη συνέχεια καταστρώνουμε τον Πίνακα Καταστάσεων. Εφόσον το κύκλωμα αποτελείται από 3 FFs και 1 εξωτερική είσοδο οι δυνατές καταστάσεις είναι $2^4 = 16$ και συνεπώς κάθε τμήμα του Πίνακα Καταστάσεων θα αποτελείται από 16 γραμμές.

ΕΙΣΟΔΟΣ και ΠΑΡΟΥΣΑ ΚΑΤΑΣΤΑΣΗ				*ΕΙΣΟΔΟΙ FFs*			*ΕΠΟΜΕΝΗ ΚΑΤΑΣΤΑΣΗ*		
X	Q_2	Q_1	Q_0	D_2	D_1	D_0	Q_2	Q_1	Q_0
0	0	0	0	0	0	0	0	0	0
0	0	0	1	0	1	0	0	1	0
0	0	1	0	1	0	0	1	0	0
0	0	1	1	1	1	0	1	1	0
0	1	0	0	0	0	1	0	0	1
0	1	0	1	0	1	1	0	1	1
0	1	1	0	1	0	1	1	0	1
0	1	1	1	1	1	1	1	1	1
1	0	0	0	0	0	0	0	0	0
1	0	0	1	1	0	0	1	0	0
1	0	1	0	0	0	1	0	0	1
1	0	1	1	1	0	1	1	0	1
1	1	0	0	0	1	0	0	1	0
1	1	0	1	1	1	0	1	1	0
1	1	1	0	0	1	1	0	1	1
1	1	1	1	1	1	1	1	1	1

Συνεπώς το διάγραμμα καταστάσεων του κυκλώματος είναι:

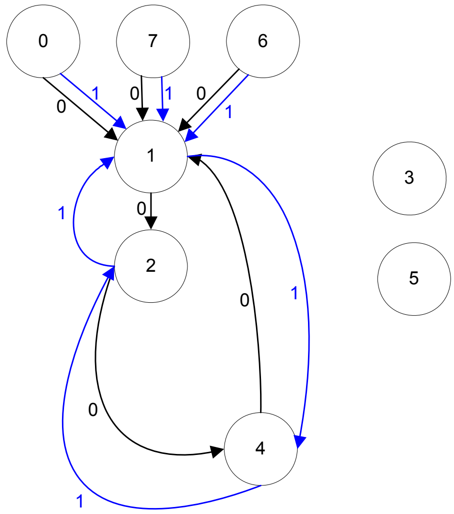


Από το διάγραμμα καταστάσεων προκύπτει ότι αν το κύκλωμα βρεθεί σε μια από τι μη έγκυρες καταστάσεις (0, 3, 5, 6, 7) τότε δεν μπορεί να επανέλθει στον έγκυρο κύκλο λειτουργίας του (ανεξάρτητα από την τιμή της εξωτερικής εισόδου).

Για να λυθεί το πρόβλημα θα πρέπει να παρέμβουμε στο διάγραμμα καταστάσεων που σχεδιάσαμε στο (Α) μέρος της άσκησης, αναγκάζοντας το κύκλωμα:

- Μετά την κατάσταση 0 να μεταβεί στη κατάσταση 1 (ή οποιαδήποτε άλλη έγκυρη κατάσταση), ανεξάρτητα από την τιμή της εξωτερικής εισόδου X
- Μετά την κατάσταση 7 να μεταβεί στη κατάσταση 1 (ή οποιαδήποτε άλλη έγκυρη κατάσταση), ανεξάρτητα από την τιμή της εξωτερικής εισόδου X
- Μετά την κατάσταση 6 (ή εναλλακτικά την κατάσταση 3 ή 5) να μεταβεί στη κατάσταση 1 (ή οποιαδήποτε άλλη έγκυρη κατάσταση), ανεξάρτητα από την τιμή της εξωτερικής εισόδου X

Συνεπώς το διάγραμμα καταστάσεων γίνεται:



Ο νέος Πίνακας Καταστάσεων του υπό σχεδίαση κυκλώματος γίνεται:

ΕΙΣΟΔΟΣ και ΠΑΡΟΥΣΑ ΚΑΤΑΣΤΑΣΗ				***ΕΠΟΜΕΝΗ ΚΑΤΑΣΤΑΣΗ***			***ΕΙΣΟΔΟΙ FFs***		
X	Q_2	Q_1	Q_0	Q_2	Q_1	Q_0	D_2	D_1	D_0
0	0	0	0	0	0	1	0	0	1
0	0	0	1	0	1	0	0	1	0
0	0	1	0	1	0	0	1	0	0
0	0	1	1	x	x	x	x	x	x
0	1	0	0	0	0	1	0	0	1
0	1	0	1	x	x	x	x	x	x
0	1	1	0	0	0	1	0	0	1
0	1	1	1	0	0	1	0	0	1
1	0	0	0	0	0	1	0	0	1
1	0	0	1	1	0	0	1	0	0
1	0	1	0	0	0	1	0	0	1
1	0	1	1	x	x	x	x	x	x
1	1	0	0	0	1	0	0	1	0
1	1	0	1	x	x	x	x	x	x
1	1	1	0	0	0	1	0	0	1
1	1	1	1	0	0	1	0	0	1

Τέλος, πρέπει να επαναπροσδιορίσουμε, σύμφωνα με το νέο Πίνακα Καταστάσεων, τις απλοποιημένες συναρτήσεις εισόδου των FFs με χρήση καρτών Karnaugh, όπως φαίνεται στη συνέχεια.

XQ_2 \ Q_1Q_0	00	01	11	10
00			x	1
01		x		
11		x		
10		1	x	

$D_2 = X'\,Q_2'\,Q_1 + X\,Q_2'\,Q_0$

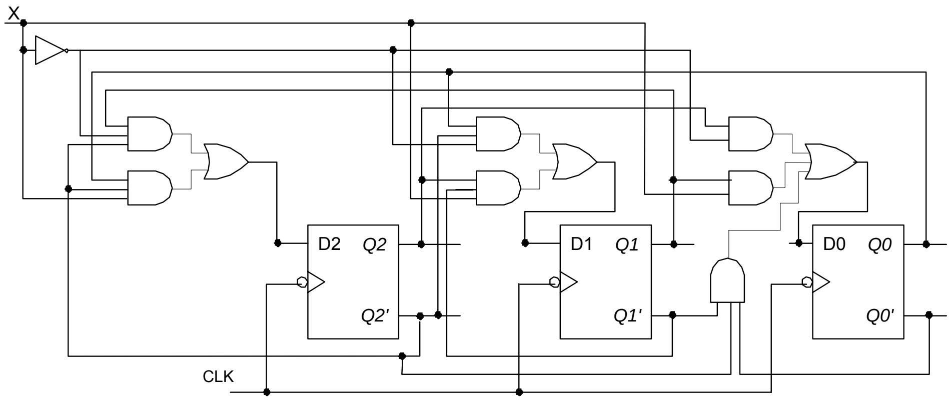
XQ_2 \ Q_1Q_0	00	01	11	10
00		1	x	
01		x		
11	1	x		
10			x	

$D_1 = X'\,Q_2'\,Q_0 + X\,Q_2\,Q_1'$

XQ_2 \ Q_1Q_0	00	01	11	10
00	1		x	
01	1	x	1	1
11		x	1	1
10	1		x	1

$D_0 = X'\,Q_2 + X\,Q_1 + Q_2'\,Q_1'\,Q_0'$

Με τις νέες («διορθωμένες») συναρτήσεις εισόδου για κάθε FF, προχωρούμε στη σχεδίαση του ζητούμενου λογικού κυκλώματος.



Με επανάληψη της διαδικασίας της «ανάλυσης» για το «διορθωμένο» λογικό κύκλωμα μπορούμε πλέον να διαπιστώσουμε ότι αυτό έχει τη δυνατότητα «Αυτόματης Διόρθωσης».

ΑΣΚΗΣΗ 102

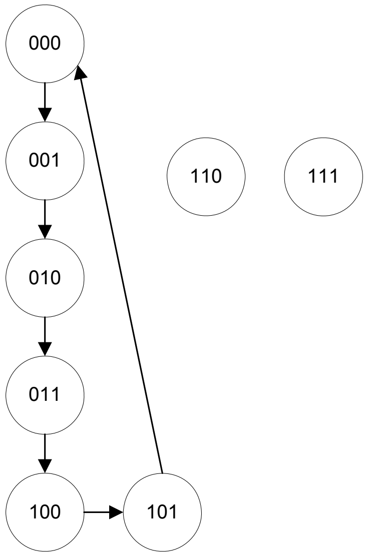
Α) Με FFs τύπου JK αρνητικής ακμής πυροδότησης, να σχεδιάσετε ένα Σύγχρονο Ακολουθιακό Κύκλωμα (ΣΑΚ) το οποίο να διατρέχει διαδοχικά τις καταστάσεις 0, 1, 2, 3, 4, 5, 0, ... (απαριθμητής mod-6). Ζητείται να σχεδιάσετε το απλούστερο δυνατό κύκλωμα.

Β) Το ΣΑΚ που σχεδιάσατε στο (Α) μπορεί να χαρακτηριστεί ως κύκλωμα «Αυτόματης Διόρθωσης»; Τεκμηριώστε την απάντηση σας.

Λύση

Α) Το πλήθος των καταστάσεων που απαιτείται να διατρέχει το ζητούμενο κύκλωμα είναι 6. Συνεπώς το κύκλωμά μας θα αποτελείται από 3 FFs τύπου JK και θα μπορεί να βρεθεί σε 8 διαφορετικές καταστάσεις (0-7).

Αρχικά θα σχεδιάσουμε το διάγραμμα καταστάσεων και στη συνέχεια θα καταστρώσουμε τον πίνακα καταστάσεων. Αν το κύκλωμα βρεθεί στη κατάσταση 6 ή στη κατάσταση 7 μας είναι αδιάφορο σε ποια κατάσταση θα μεταβεί. Συνεπώς το διάγραμμα καταστάσεων διαμορφώνεται όπως φαίνεται παρακάτω:



Η επόμενη κατάσταση προκύπτει από το διάγραμμα καταστάσεων, ενώ οι τιμές των εισόδων των FFs προκύπτουν από τις τιμές της παρούσας και επόμενης κατάστασης καθενός FF σε συνδυασμό με τον πίνακα διέγερσης του FF. Ο Πίνακας καταστάσεων ακολουθεί.

ΠΑΡΟΥΣΑ ΚΑΤΑΣΤΑΣΗ			***ΕΠΟΜΕΝΗ ΚΑΤΑΣΤΑΣΗ***			***ΕΙΣΟΔΟΙ FFs***		
Q_2	Q_1	Q_0	Q_2	Q_1	Q_0	J_2K_2	J_1K_1	J_0K_0
0	0	0	0	0	1	0 x	0 x	1 x
0	0	1	0	1	0	0 x	1 x	x 1
0	1	0	0	1	1	0 x	x 0	1 x
0	1	1	1	0	0	1 x	x 1	x 1
1	0	0	1	0	1	x 0	0 x	1 x
1	0	1	0	0	0	x 1	0 x	x 1
1	1	0	x	x	x	x x	x x	x x
1	1	1	x	x	x	x x	x x	x x

Τώρα πρέπει να προσδιορίσουμε τις απλοποιημένες συναρτήσεις εισόδου των FFs με χρήση χαρτών Karnaugh, όπως φαίνεται στη συνέχεια.

Q_2 \ Q_1Q_0	00	01	11	10
0			1	
1	x	x	x	x

$J_2 = Q_1 Q_0$

Q_2 \ Q_1Q_0	00	01	11	10
0		1	x	x
1			x	x

$J_1 = Q_2' Q_0$

Q_2 \ Q_1Q_0	00	01	11	10
0	1	x	x	1
1	1	x	x	x

$J_0 = 1$

Q_2 \ Q_1Q_0	00	01	11	10
0	x	x	x	x
1		1	x	x

$K_2 = Q_0$

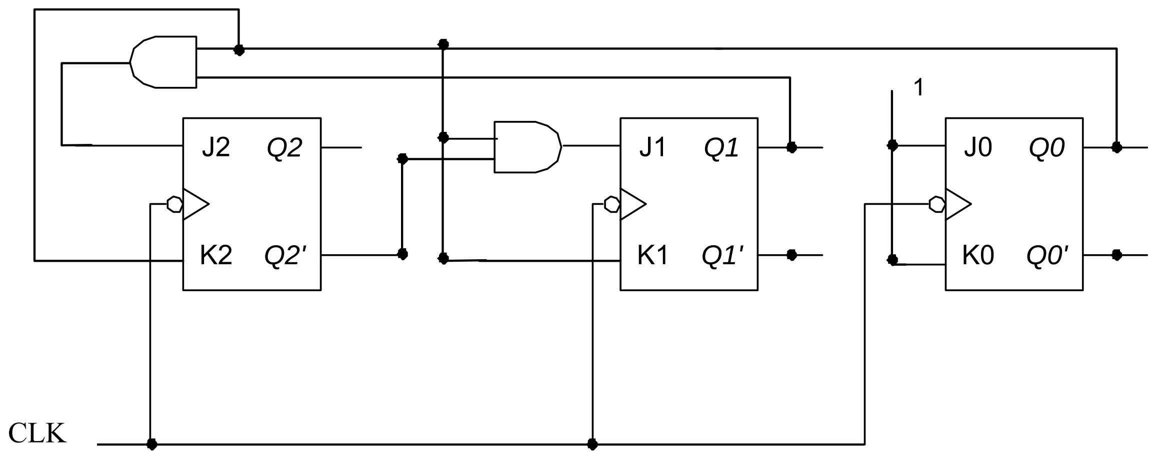
Q_2 \ Q_1Q_0	00	01	11	10
0	x	x	1	
1	x	x	x	x

$K_1 = Q_0$

Q_2 \ Q_1Q_0	00	01	11	10
0	x	1	1	x
1	x	1	x	x

$K_0 = 1$

Γνωρίζοντας πλέον τις συναρτήσεις της εισόδου κάθε FF, προχωρούμε στη σχεδίαση του λογικού κυκλώματος, που είναι και το ζητούμενο κύκλωμα.



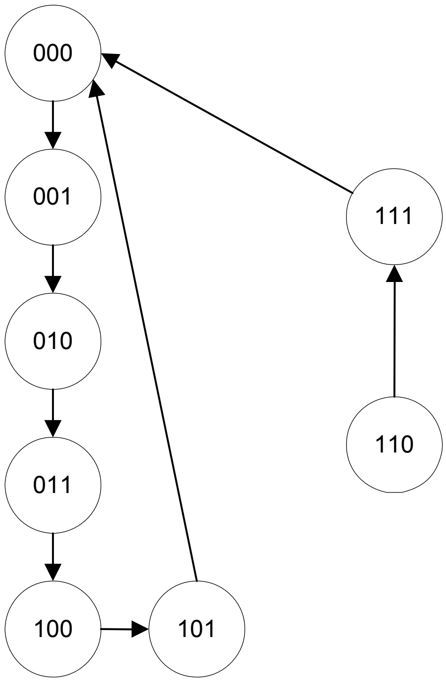
B) Για να διαπιστώσουμε αν το κύκλωμα που σχεδιάσαμε παραπάνω μπορεί να χαρακτηριστεί ως κύκλωμα αυτόματης διόρθωσης θα πρέπει να το «*αναλύσουμε*».
Το πρώτο βήμα είναι να γράψουμε τις συναρτήσεις εισόδου των FF's.

$J_2 = Q_1 Q_0$ $\qquad J_1 = Q_2' Q_0$ $\qquad J_0 = 1$
$K_2 = Q_0$ $\qquad K_1 = Q_0$ $\qquad K_0 = 1$

Στη συνέχεια καταστρώνουμε τον Πίνακα Καταστάσεων. Εφόσον το κύκλωμα αποτελείται από 3 FFs οι δυνατές καταστάσεις είναι $2^3 = 8$ και συνεπώς κάθε τμήμα του Πίνακα Καταστάσεων θα αποτελείται από 8 γραμμές.

ΠΑΡΟΥΣΑ ΚΑΤΑΣΤΑΣΗ			***ΕΙΣΟΔΟΙ FFs***			***ΕΠΟΜΕΝΗ ΚΑΤΑΣΤΑΣΗ***		
Q_2	Q_1	Q_0	J_2K_2	J_1K_1	J_0K_0	Q_2	Q_1	Q_0
0	0	0	0 0	0 0	1 1	0	0	1
0	0	1	0 1	1 1	1 1	0	1	0
0	1	0	0 0	0 0	1 1	0	1	1
0	1	1	1 1	1 1	1 1	1	0	0
1	0	0	0 0	0 0	1 1	1	0	1
1	0	1	0 1	0 1	1 1	0	0	0
1	1	0	0 0	0 0	1 1	1	1	1
1	1	1	1 1	0 1	1 1	0	0	0

Συνεπώς το διάγραμμα καταστάσεων του κυκλώματος είναι:

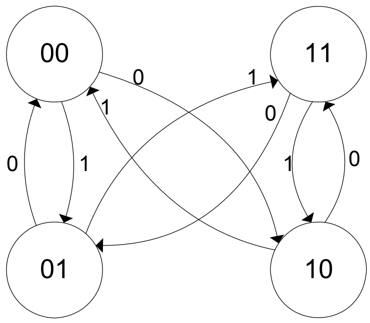


Από το διάγραμμα καταστάσεων προκύπτει ότι αν το κύκλωμα βρεθεί σε μια από τις μη έγκυρες καταστάσεις (6 ή 7) το πολύ μετά από δύο ωρολογιακούς παλμούς μεταβαίνει σε έγκυρη κατάσταση και συνεχίζει το κύκλο λειτουργίας του κανονικά. Ως εκ τούτου το κύκλωμα μπορεί να χαρακτηριστεί ως κύκλωμα «Αυτόματης Διόρθωσης».

ΑΣΚΗΣΗ 103

Α) Με FFs τύπου T αρνητικής ακμής πυροδότησης να σχεδιάσετε ένα σύγχρονο ακολουθιακό κύκλωμα το οποίο για E=0 να διατρέχει διαδοχικά τις καταστάσεις **0, 2, 3, 1,** 0, 2, 3, 1, 0, ... ενώ για E=1 να διατρέχει διαδοχικά τις καταστάσεις **0, 1, 3, 2,** 0, 1, 3, 2, 0, ... Ζητείται να σχεδιάσετε το απλούστερο δυνατό κύκλωμα, με τον μικρότερο αριθμό πυλών.

Λύση
Το ζητούμενο κύκλωμα θα αποτελείται από δύο FFs.
Βήμα Σ1: Διάγραμμα καταστάσεων



Βήμα Σ2:

Είσοδος	**Παρ. Κατάσταση**	**Επόμενη Κατάσταση**	**Είσοδοι FFs**
E	**Q_1 Q_0**	**Q_1 Q_0**	**T_1 T_0**
0	0 0	1 0	1 0
0	0 1	0 0	0 1

0	1 0	1 1	0 1
0	1 1	0 1	1 0
1	0 0	0 1	0 1
1	0 1	1 1	1 0
1	1 0	0 0	1 0
1	1 1	1 0	0 1

E \ Q_1Q_0	00	01	11	10
0	1		1	
1		1		1

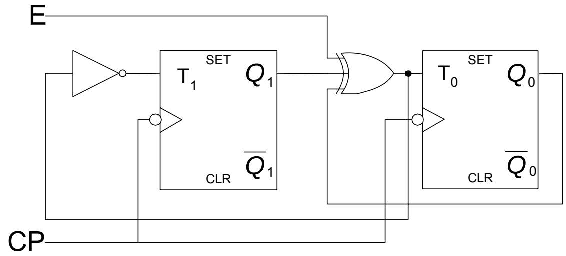
T_1

E \ Q_1Q_0	00	01	11	10
0		1		1
1	1		1	

T_0

$$T_0 = \overline{E}\,\overline{Q_1}Q_0 + \overline{E}Q_1\overline{Q_0} + E\overline{Q_1}\,\overline{Q_0} + EQ_1Q_0 = \overline{E}\cdot\left(Q_1 \oplus Q_0\right) + E\cdot\overline{Q_1 \oplus Q_0} = E \oplus Q_1 \oplus Q_0$$
$$T_1 = \overline{E}\,\overline{Q_1}\,\overline{Q_0} + \overline{E}Q_1Q_0 + E\overline{Q_1}Q_0 + EQ_1\overline{Q_0} = \overline{E}\cdot\overline{Q_1 \oplus Q_0} + E\cdot Q_1 \oplus Q_0 = \overline{E \oplus Q_1 \oplus Q_0} = \overline{T_0}$$

Βήμα Σ3:



ΑΣΚΗΣΗ 104

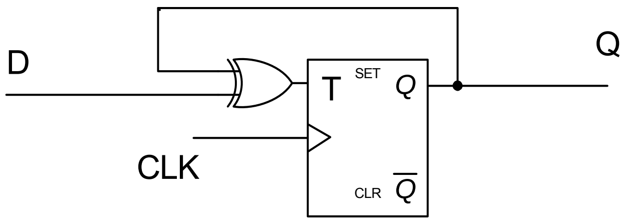
Α) Σχεδιάστε ένα D flip-flop χρησιμοποιώντας ένα T flip-flop θετικής ακμής πυροδότησης. Επιπλέον μπορείτε να χρησιμοποιήσετε λογικές πύλες (AND, OR, NOT, NAND, NOR, XOR, XNOR). Ζητείται το απλούστερο δυνατό κύκλωμα.
Β) Σχεδιάστε ένα JK flip-flop χρησιμοποιώντας μόνο ένα πολυπλέκτη και ένα T flip-flop θετικής ακμής πυροδότησης. *Επισημαίνεται ότι δεν μπορείτε να χρησιμοποιήσετε λογικές πύλες.*

Λύση

Α) Κατασκευάζουμε τον πίνακα καταστάσεων και προσδιορίζουμε τη συνάρτηση εισόδου του T flip-flop χρησιμοποιώντας τον πίνακα διέγερσης αυτού.

Παρούσα κατάσταση	**Είσοδος**	***Επόμενη κατάσταση***	***Είσοδος T-FF***
Q	***D***	***Q***	***T***
0	0	0	0
0	1	1	1
1	0	0	1
1	1	1	0

Από τον πίνακα καταστάσεων προκύπτει ότι $T = D \oplus Q$.



Β) Κατασκευάζουμε τον πίνακα καταστάσεων και προσδιορίζουμε τη συνάρτηση εισόδου του T flip-flop χρησιμοποιώντας τον πίνακα διέγερσης αυτού.

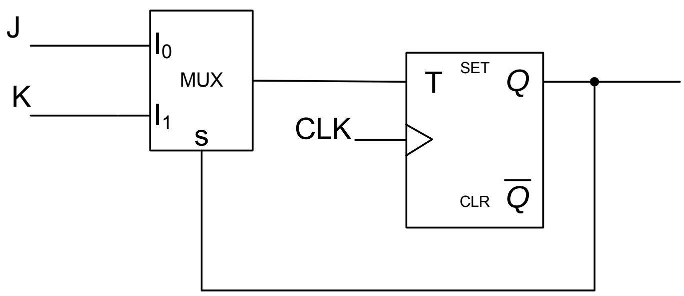
Παρούσα κατάσταση	**Είσοδοι**		***Επόμενη κατάσταση***	***Είσοδος FF***
Q	***J***	***K***	***Q***	***T***
0	0	0	0	0
0	0	1	0	0
0	1	0	1	1
0	1	1	1	1
1	0	0	1	0
1	0	1	0	1
1	1	0	1	0
1	1	1	0	1

Στη συνέχεια απλοποιούμε τη συνάρτηση της εισόδου του T flip-flop με τη βοήθεια του χάρτη Karnough τριών μεταβλητών.

Q \ *JK*	00	01 (*K*)	11 (*K*, *J*)	10 (*J*)
0			1	1
Q 1		1	1	

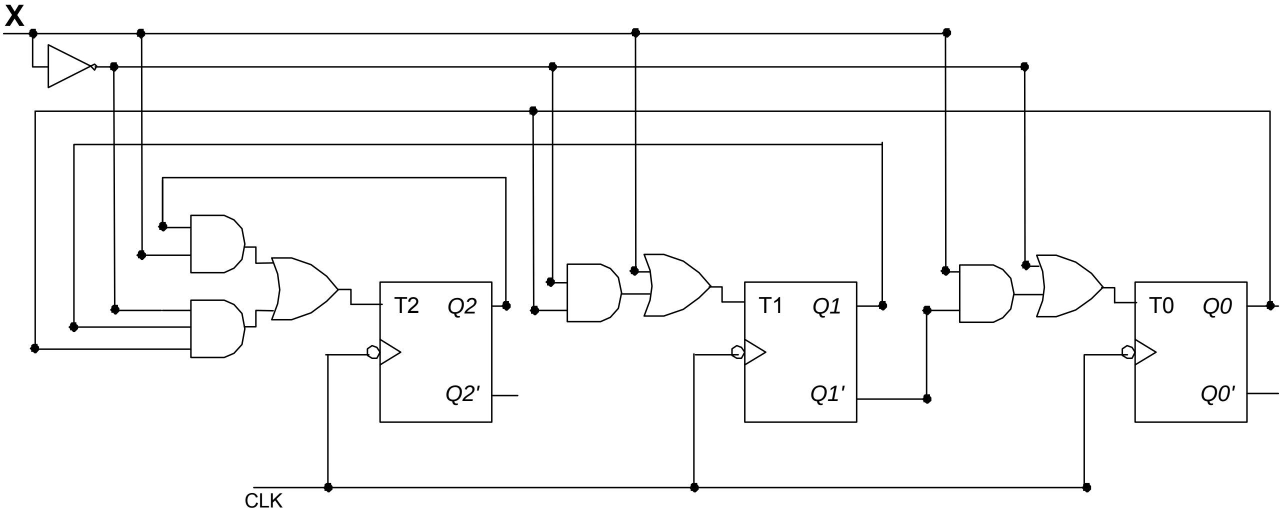
Η απλοποιημένη συνάρτηση που περιγράφει την είσοδο T είναι η: $T = JQ' + KQ$.

Επομένως για να μετατρέψουμε ένα T flip-flop σε ένα JK flip-flop θα πρέπει να συνδέσουμε στην είσοδο του T flip-flop την έξοδο ενός πολυπλέκτη 2-σε-1 με τις δύο εισόδους του I_0 και I_1 συνδεδεμένες στα J και K αντίστοιχα και την είσοδο επιλογής s στην έξοδο Q του T flip-flop όπως φαίνεται στο παρακάτω σχήμα.



ΑΣΚΗΣΗ 105

Α) Να σχεδιαστεί το «*Διάγραμμα Καταστάσεων*» του Σύγχρονου Ακολουθιακού Κυκλώματος (ΣΑΚ) με FFs τύπου T αρνητικής ακμής πυροδότησης που ακολουθεί. Διευκρινίζεται ότι το σήμα X είναι μια εξωτερική είσοδος του κυκλώματος.



Β) Σχολιάστε τη λειτουργία του παραπάνω σύγχρονου ακολουθιακού κυκλώματος. Εξετάστε ξεχωριστά την περίπτωση που η εξωτερική είσοδος είναι X=0 και την περίπτωση που είναι X=1.

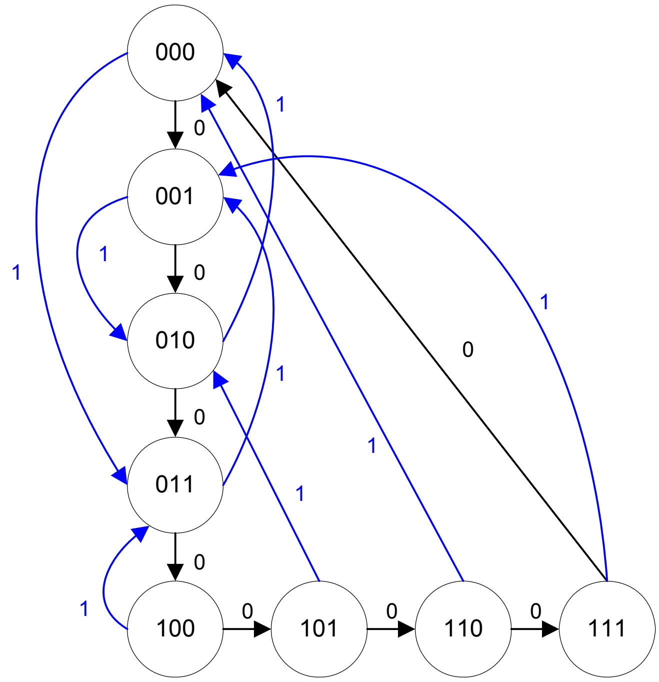
Λύση:
Α) Το πρώτο βήμα είναι να γράψουμε τις συναρτήσεις εισόδου των FF's.

$$T_2 = X\,Q_2 + X'\,Q_1\,Q_0 \qquad T_1 = X + X'\,Q_0 = X + Q_0 \qquad T_0 = X' + X\,Q_1' = X' + Q_1'$$

Στη συνέχεια καταστρώνουμε τον Πίνακα Καταστάσεων. Εφόσον το κύκλωμα αποτελείται από 3 FFs και υπάρχει και μια εξωτερική είσοδος οι δυνατές καταστάσεις είναι 2^4 = 16 και συνεπώς κάθε τμήμα του Πίνακα Καταστάσεων θα αποτελείται από 16 γραμμές.

ΠΑΡΟΥΣΑ ΚΑΤΑΣΤΑΣΗ				*ΕΙΣΟΔΟΙ FFs*			*ΕΠΟΜΕΝΗ ΚΑΤΑΣΤΑΣΗ*		
X	Q_2	Q_1	Q_0	T_2	T_1	T_0	Q_2	Q_1	Q_0
0	0	0	0	0	0	1	0	0	1
0	0	0	1	0	1	1	0	1	0
0	0	1	0	0	0	1	0	1	1
0	0	1	1	1	1	1	1	0	0
0	1	0	0	0	0	1	1	0	1
0	1	0	1	0	1	1	1	1	0
0	1	1	0	0	0	1	1	1	1
0	1	1	1	1	1	1	0	0	0
1	0	0	0	0	1	1	0	1	1
1	0	0	1	0	1	1	0	1	0
1	0	1	0	0	1	0	0	0	0
1	0	1	1	0	1	0	0	0	1
1	1	0	0	1	1	1	0	1	1
1	1	0	1	1	1	1	0	1	0
1	1	1	0	1	1	0	0	0	0
1	1	1	1	1	1	0	0	0	1

Συνεπώς το διάγραμμα καταστάσεων του κυκλώματος είναι.



Β) Από το διάγραμμα καταστάσεων διαπιστώνουμε ότι για:

- για Χ=0 σχηματίζεται ένας κύκλος 0→1→2→3→4→5→6→7→0→1… οπότε σε οποιαδήποτε από τις καταστάσεις αυτές και αν βρεθεί το κύκλωμα πηγαίνει στην επόμενη κατάσταση που ορίζεται από τον κύκλο. Συνεπώς το κύκλωμα λειτουργεί ως απαριθμητής με 8 καταστάσεις (mod-8)

- για Χ=1
  - παρατηρούμε ότι σχηματίζεται ένας κύκλος 0→3→2→1→0…, επομένως σε οποιαδήποτε από τις καταστάσεις αυτές και αν βρεθεί το κύκλωμα πηγαίνει στην επόμενη κατάσταση που ορίζεται από τον κύκλο.
  - εάν το κύκλωμα βρεθεί σε κάποια από τις καταστάσεις 4, 5, 6 και 7 θα πάει αντίστοιχα στην κατάσταση 3, 2, 0 και 1 οι οποίες όπως μπορούμε να δούμε είναι καταστάσεις του κύκλου 0→3→1→2→0…. Επομένως στον επόμενο χρονικό παλμό το κύκλωμα θα μεταβεί στην επόμενη κατάσταση του κύκλου.

ΑΣΚΗΣΗ 106

Α) Με FFs τύπου JK αρνητικής ακμής πυροδότησης, να σχεδιάσετε ένα Σύγχρονο Ακολουθιακό Κύκλωμα (ΣΑΚ) με δύο εξωτερικές εισόδους X και Y, το οποίο για

X=0, Y=0: διατρέχει διαδοχικά τις καταστάσεις **3, 2, 1, 0**, 3, 2, ...

X=0, Y=1: διατρέχει διαδοχικά τις καταστάσεις **0, 1, 2, 3**, 0, 1, ...

X=1, Y=0: διατρέχει διαδοχικά τις καταστάσεις **0, 2,** 0, 2, ...

X=1, Y=1: διατρέχει διαδοχικά τις καταστάσεις **1, 3,** 1, 3, ...

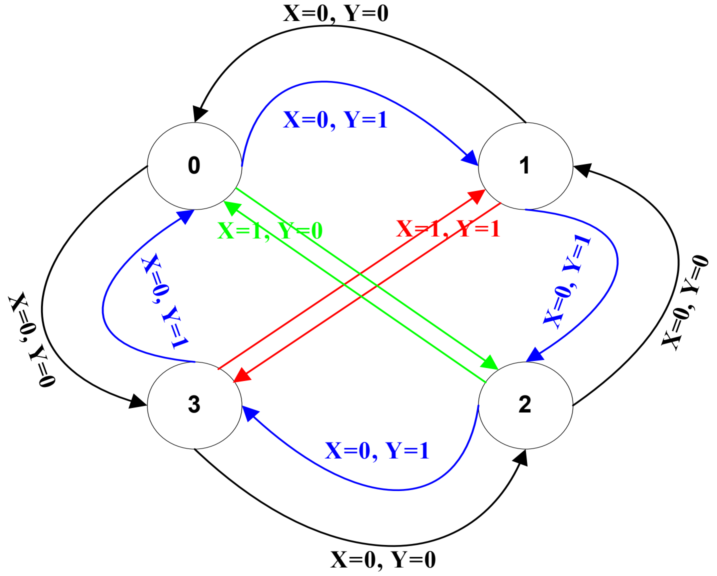
Ζητείται το απλούστερο δυνατό κύκλωμα.

Β) Το ΣΑΚ που σχεδιάσατε στο (Α) μπορεί να χαρακτηριστεί ως κύκλωμα «Αυτόματης Διόρθωσης»; Τεκμηριώστε την απάντηση σας.

Λύση:
Α) Το πρώτο βήμα είναι να σχεδιάσουμε το διάγραμμα καταστάσεων και στη συνέχεια να καταστρώσουμε τον πίνακα καταστάσεων.
Το πλήθος των καταστάσεων που απαιτείται να διατρέχει το ζητούμενο κύκλωμα είναι 4. Συνεπώς το κύκλωμα μας θα αποτελείται από 2 FFs τύπου JK.

Καθώς μας είναι αδιάφορο σε ποια κατάσταση θα βρεθεί το κύκλωμα:
- μετά την κατάσταση 1 και 3 όταν X=1 και Y=0
- μετά την κατάσταση 0 και 2 όταν X=1 και Y=1

το διάγραμμα καταστάσεων θα είναι όπως φαίνεται παρακάτω.



Η επόμενη κατάσταση προκύπτει από το διάγραμμα καταστάσεων, ενώ οι τιμές των εισόδων των FFs προκύπτουν από τις τιμές της παρούσας και επόμενης κατάστασης καθενός FF σε συνδυασμό με τον πίνακα διέγερσης του FF. Ο πίνακας καταστάσεων ακολουθεί:

ΕΙΣΟΔΟΙ και ΠΑΡΟΥΣΑ ΚΑΤΑΣΤΑΣΗ				***ΕΠΟΜΕΝΗ ΚΑΤΑΣΤΑΣΗ***		***ΕΙΣΟΔΟΙ FFs***	
X	Y	Q_1	Q_0	Q_1	Q_0	J_1K_1	J_0K_0
0	0	0	0	1	1	1 x	1 x
0	0	0	1	0	0	0 x	x 1
0	0	1	0	0	1	x 1	1 x
0	0	1	1	1	0	x 0	x 1
0	1	0	0	0	1	0 x	1 x
0	1	0	1	1	0	1 x	x 1
0	1	1	0	1	1	x 0	1 x
0	1	1	1	0	0	x 1	x 1
1	0	0	0	1	0	1 x	0 x
1	0	0	1	x	x	x x	x x
1	0	1	0	0	0	x 1	0 x
1	0	1	1	x	x	x x	x x
1	1	0	0	x	x	x x	x x
1	1	0	1	1	1	1 x	x 0
1	1	1	0	x	x	x x	x x
1	1	1	1	0	1	x 1	x 0

Τώρα πρέπει να προσδιορίσουμε τις απλοποιημένες συναρτήσεις εισόδου των FFs με χρήση χαρτών Karnaugh, όπως φαίνεται στη συνέχεια.

XY \ Q_1Q_0	00	01	11	10
00	1		x	x
01		1	x	x
11	x	1	x	x
10	1	x	x	x

$J_1 = Y\,Q_0 + Y'\,Q_0' = (Y \oplus Q_0)'$

XY \ Q_1Q_0	00	01	11	10
00	x	x		1
01	x	x	1	
11	x	x	1	x
10	x	x	x	1

$K_1 = Y\,Q_0 + Y'\,Q_0' = (Y \oplus Q_0)'$

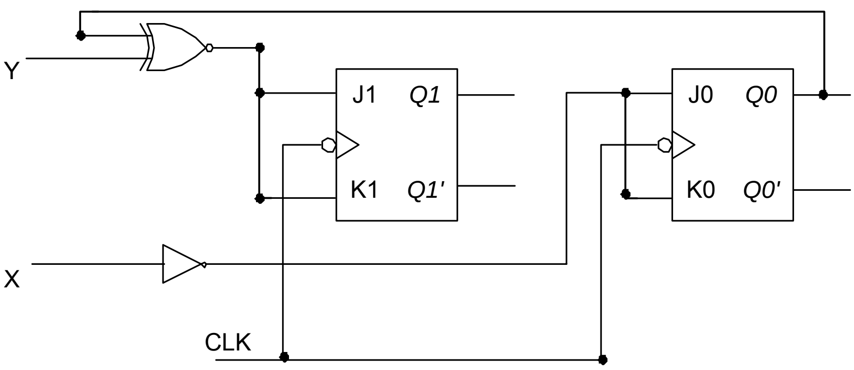
XY \ Q_1Q_0	00	01	11	10
00	1	x	x	1
01	1	x	x	1
11	x	x	x	x
10		x	x	

$J_0 = X'$

XY \ Q_1Q_0	00	01	11	10
00	x	1	1	x
01	x	1	1	x
11	x			x
10	x	x	x	x

$K_0 = X'$

Γνωρίζοντας πλέον τις συναρτήσεις της εισόδου κάθε FF, προχωρούμε στη σχεδίαση του λογικού κυκλώματος, που είναι και το ζητούμενο κύκλωμα.



Β) Για να διαπιστώσουμε αν το κύκλωμα που σχεδιάσαμε μπορεί να χαρακτηριστεί ως κύκλωμα αυτόματης διόρθωσης θα πρέπει να το «αναλύσουμε»

Το πρώτο βήμα είναι να γράψουμε τις συναρτήσεις εισόδου των FF's.

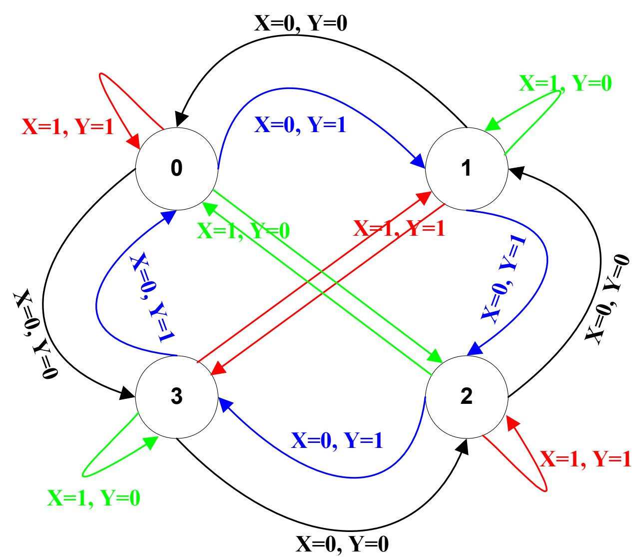
$J_1 = Y\,Q_0 + Y'\,Q_0'$ $\qquad$ $K_1 = Y\,Q_0 + Y'\,Q_0'$

$J_0 = X'$ $\qquad$ $K_0 = X'$

Στη συνέχεια καταστρώνουμε τον Πίνακα Καταστάσεων. Εφόσον το κύκλωμα αποτελείται από 2 FFs και 2 εξωτερικές εισόδους οι δυνατές καταστάσεις είναι $2^4 = 16$ και συνεπώς κάθε τμήμα του Πίνακα Καταστάσεων θα αποτελείται από 16 γραμμές.

ΕΙΣΟΔΟΙ και ΠΑΡΟΥΣΑ ΚΑΤΑΣΤΑΣΗ				***ΕΙΣΟΔΟΙ FFs***		***ΕΠΟΜΕΝΗ ΚΑΤΑΣΤΑΣΗ***	
X	*Y*	Q_1	Q_0	J_1K_1	J_0K_0	Q_1	Q_0
0	0	0	0	1 1	1 1	1	1
0	0	0	1	0 0	1 1	0	0
0	0	1	0	1 1	1 1	0	1
0	0	1	1	0 0	1 1	1	0
0	1	0	0	0 0	1 1	0	1
0	1	0	1	1 1	1 1	1	0
0	1	1	0	0 0	1 1	1	1
0	1	1	1	1 1	1 1	0	0
1	0	0	0	1 1	0 0	1	0
1	0	0	1	0 0	0 0	0	1
1	0	1	0	1 1	0 0	0	0
1	0	1	1	0 0	0 0	1	1
1	1	0	0	0 0	0 0	0	0
1	1	0	1	1 1	0 0	1	1
1	1	1	0	0 0	0 0	1	0
1	1	1	1	1 1	0 0	0	1

Συνεπώς το διάγραμμα καταστάσεων του κυκλώματος είναι:



Από το διάγραμμα καταστάσεων προκύπτει ότι αν το κύκλωμα βρεθεί σε μια από τις μη έγκυρες καταστάσεις :

- κατάσταση 1 ή κατάσταση 3 όταν X=1 και Y=0 ή
- κατάσταση 0 ή κατάσταση 2 όταν X=1 και Y=1

τότε δεν μπορεί να επανέλθει στον έγκυρο κύκλο λειτουργίας του. Συνεπώς το κύκλωμα δεν μπορεί να χαρακτηριστεί «Κύκλωμα Αυτόματης Διόρθωσης».

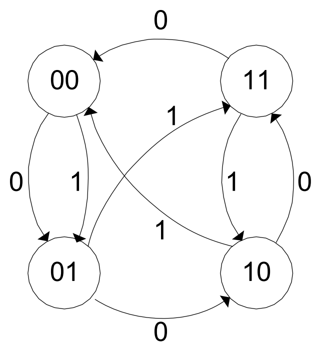
ΑΣΚΗΣΗ 107

Με **FFs** τύπου **T** αρνητικής ακμής πυροδότησης, να σχεδιάσετε σύγχρονο ακολουθιακό κύκλωμα απαριθμητή των **2 δυαδικών ψηφίων,** ο οποίος να μετρά ως εξής: για είσοδο **E=0** να μετρά προς τα πάνω ως κανονικός **δυαδικός** απαριθμητής, ενώ για είσοδο **E=1** να μετρά σύμφωνα με τον ανακλαστικό κώδικα **Gray**. Ζητείται το απλούστερο δυνατό κύκλωμα.

<u>Απάντηση</u>

Μας ζητείται να σχεδιάσουμε έναν σύγχρονο απαριθμητή των 2 bits η λειτουργία του οποίου να ελέγχεται από μία είσοδο E. Ακολουθούμε τα γνωστά βήματα σχεδίασης Σ.Α.Κ.

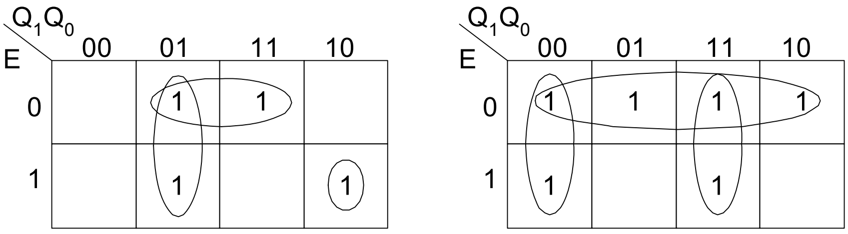
Βήμα Σ1. Σχεδιάζουμε το διάγραμμα καταστάσεων.



Βήμα Σ2. Καταστρώνουμε τον πίνακα καταστάσεων.

Είσοδος	Παρούσα Κατάσταση	Επόμενη Κατάσταση	Είσοδοι FFs
E	**Q_1 Q_0**	**Q_1 Q_0**	**T_1 T_0**
0	0 0	0 1	0 1
0	0 1	1 0	1 1
0	1 0	1 1	0 1
0	1 1	0 0	1 1
1	0 0	0 1	0 1
1	0 1	1 1	1 0
1	1 0	0 0	1 0
1	1 1	1 0	0 1

Βήμα Σ3. Απλοποιούμε τις συναρτήσεις εισόδου με χρήση χαρτών karnaugh.

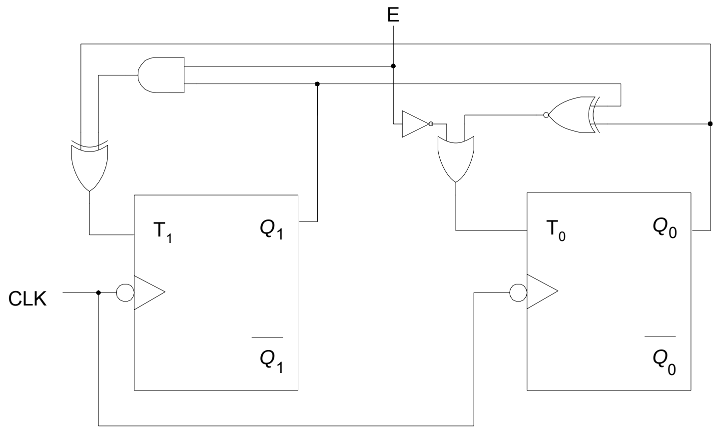


T_1 T_0

$$T_1 = Q_1{}' Q_0 + E' Q_0 + E Q_1 Q_0{}' = \left(Q_1{}' + E'\right) Q_0 + \left(E Q_1\right) Q_0{}' = \left(E Q_1\right)' Q_0 + \left(E Q_1\right) Q_0{}' = \left(E Q_1\right) \oplus Q_0$$

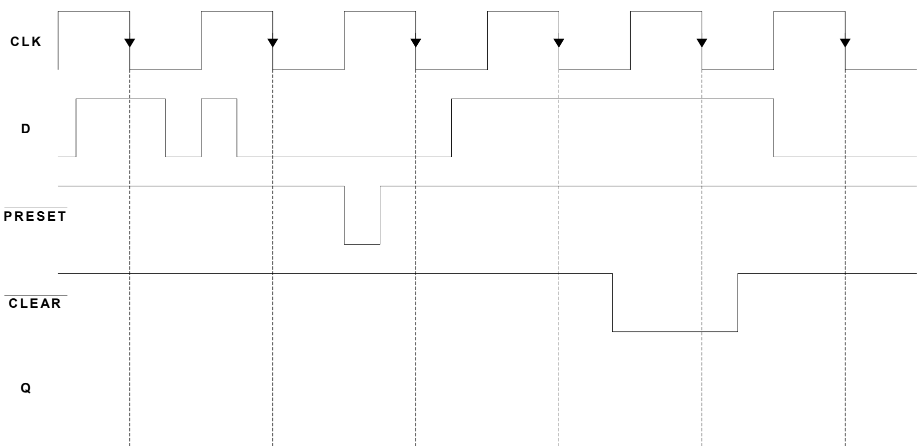
$$T_0 = E' + Q_1{}' Q_0{}' + Q_1 Q_0 = E' + \left(Q_1 \oplus Q_0\right)'$$

Βήμα Σ4. Σχεδιάζουμε το κύκλωμα.



ΑΣΚΗΣΗ 108

α) Σχεδιάστε την κυματομορφή της εξόδου Q ενός αρνητικά ακμοπυροδοτούμενου D Flip Flop όταν στην είσοδο D και στις ασύγχρονες εισόδους PRESET και CLEAR αυτού εφαρμόζονται οι παλμοί που φαίνονται στο σχήμα που ακολουθεί. Θεωρήστε ότι η αρχική τιμή της εξόδου Q είναι το λογικό 0 και ότι τα σήματα PRESET και CLEAR ενεργοποιούνται κατά τη μετάβαση στο λογικό 0.



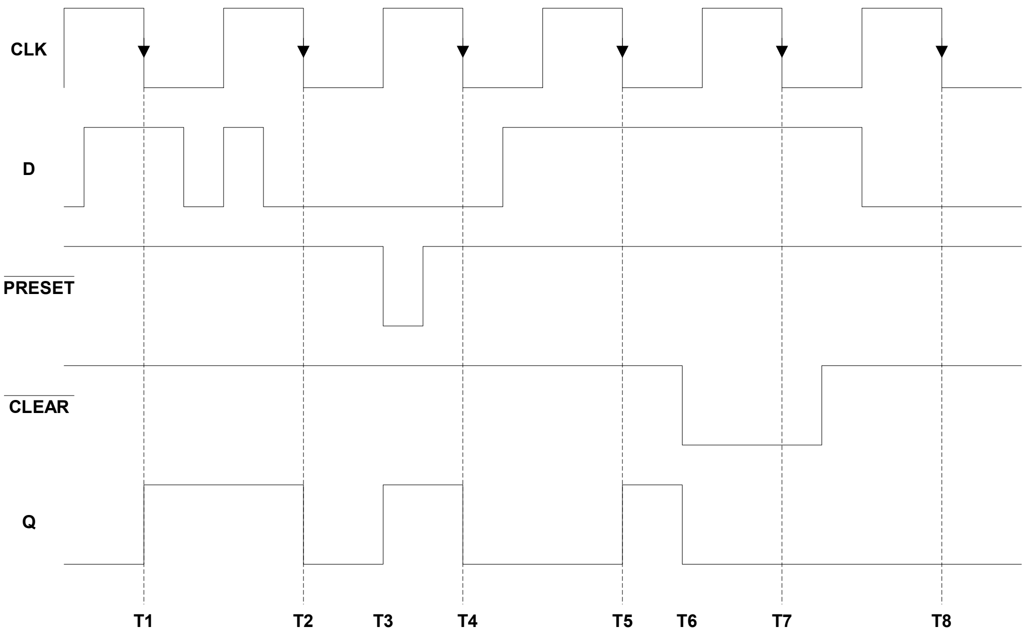
β) Έστω ότι έχετε στη διάθεσή σας ένα και μόνο ένα αρνητικά ακμοπυροδοτούμενο JK Flip Flop. Προσδιορίστε τι πρέπει να συνδέσετε στις εισόδους J και K του Flip Flop ώστε η έξοδός του να σηματοδοτεί (δηλαδή να λαμβάνει το λογικό 1) τις αρνητικές ακμές του ρολογιού κατά τις οποίες μία εξωτερική είσοδος D και η έξοδος Q του Flip Flop είναι διαφορετικές.

Λύση:

α) Πριν προχωρήσουμε στον προσδιορισμό της κυματομορφής της εξόδου Q του FF ας θυμηθούμε πως οι ασύγχρονες είσοδοι $\overline{\text{PRESET}}$ και $\overline{\text{CLEAR}}$ επηρεάζουν τη λειτουργία του.

PRESET	***CLEAR***	***Λειτουργία FF***
0	1	Q = 1
1	0	Q = 0
1	1	Κανονική Σύγχρονη Λειτουργία

Συνεπώς, η κυματομορφή της εξόδου Q θα είναι όπως φαίνεται στο σχήμα που ακολουθεί.



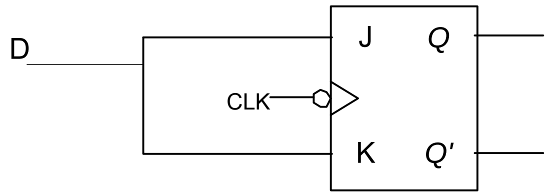
Συγκεκριμένα, σε κάθε χρονική στιγμή έχουμε:
T1: $\overline{\text{PRESET}}$ = $\overline{\text{CLEAR}}$ = 1, άρα η έξοδος Q θα λάβει τη τιμή της εισόδου D που είναι 1.
T2: $\overline{\text{PRESET}}$ = $\overline{\text{CLEAR}}$ = 1, άρα η έξοδος Q θα λάβει τη τιμή της εισόδου D που είναι 0.
T3: $\overline{\text{PRESET}}$ = 0, $\overline{\text{CLEAR}}$ = 1, άρα η έξοδος Q θα γίνει 1 (ασύγχρονη λειτουργία).
T4: $\overline{\text{PRESET}}$ = $\overline{\text{CLEAR}}$ = 1, άρα η έξοδος Q θα λάβει τη τιμή της εισόδου D που είναι 0.
T5: $\overline{\text{PRESET}}$ = $\overline{\text{CLEAR}}$ = 1, άρα η έξοδος Q θα λάβει τη τιμή της εισόδου D που είναι 1.
T6: $\overline{\text{PRESET}}$ = 1, $\overline{\text{CLEAR}}$ = 0, άρα η έξοδος Q θα γίνει 0 (ασύγχρονη λειτουργία).
T7: $\overline{\text{PRESET}}$ = 1, $\overline{\text{CLEAR}}$ = 0, άρα η έξοδος Q θα παραμείνει 0 (ασύγχρονη λειτουργία).
T8: $\overline{\text{PRESET}}$ = $\overline{\text{CLEAR}}$ = 1, άρα η έξοδος Q θα λάβει τη τιμή της εισόδου D που είναι 0.

β) **$1^{ος}$ Τρόπος.** Στο συγκεκριμένο ερώτημα το ζητούμενο είναι η έξοδος του JK Flip Flop να αποτυπώνει το αποτέλεσμα $D \oplus Q$. Συνεπώς αν κατά την αρνητική ακμή του ρολογιού τα D και Q είναι διαφορετικά τότε το JK Flip Flop θα λαμβάνει την τιμή «1», ενώ αν είναι τα ίδια θα λαμβάνει την τιμή «0».

Κατασκευάζουμε τον πίνακα καταστάσεων και προσδιορίζουμε τη συνάρτηση για τις εισόδους του JK Flip Flop χρησιμοποιώντας τον πίνακα διέγερσης αυτού.

Παρούσα κατάσταση	**Είσοδος**	***Επόμενη κατάσταση***	***Είσοδοι JK-FF***	
Q	***D***	***Q***	***J***	***K***
0	0	0	0	X
0	1	1	1	X
1	0	1	X	0
1	1	0	X	1

Από τον πίνακα καταστάσεων προκύπτει ότι $J = K = D$.



2^{ος} Τρόπος. Γνωρίζουμε ότι σε ένα JK ff η επόμενη κατάσταση δίνεται από τη λογική εξίσωση :

$$Q(t+1) = J\, Q'(t) + K'\, Q(t) \qquad (1)$$

Σύμφωνα με την εκφώνηση επιθυμούμε η επόμενη έξοδος του κυκλώματός να είναι 1 όταν $D \neq Q(t)$ ή ισοδύναμα όταν D XOR Q(t) = 1, δηλαδή θέλουμε

$$Q(t+1) = D \text{ XOR } Q(t) = D\, Q'(t) + D'\, Q(t) \qquad (2)$$

Από τις (1) και (2) βλέπουμε ότι αυτό επιτυγχάνεται θέτοντας J = K = D.

ΑΣΚΗΣΗ 109

Α) Με Flip Flops τύπου JK θετικής ακμής πυροδότησης και το μικρότερο δυνατό αριθμό λογικών πυλών 2 εισόδων, να σχεδιάσετε ένα Σύγχρονο Ακολουθιακό Κύκλωμα (ΣΑΚ) με δύο εξωτερικές εισόδους X_1 και X_0, το οποίο από την τρέχουσα κατάσταση του $[A_1A_0]_t$ θα μεταβαίνει στην επόμενη κατάσταση $[A_1A_0]_{t+1}$ σύμφωνα με την ακόλουθη σχέση:

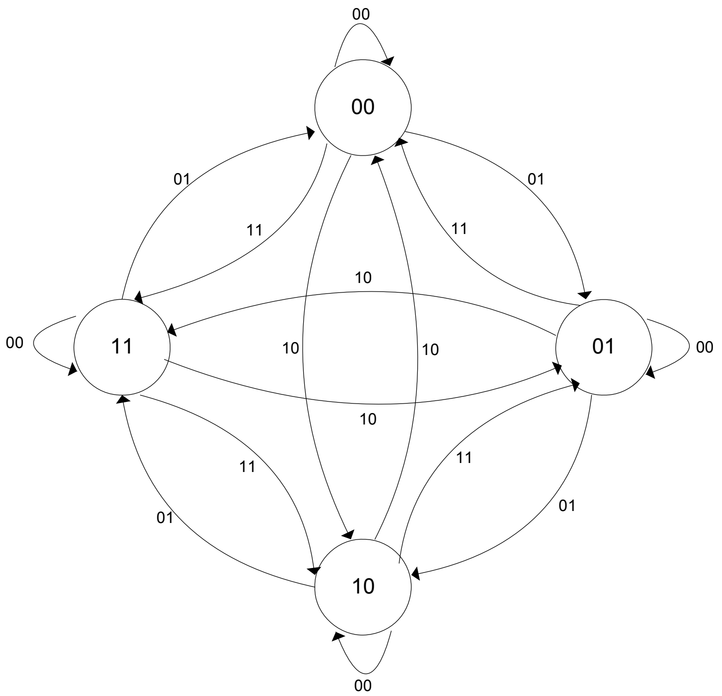
$$[A_1A_0]_{t+1} = [A_1A_0]_t + X_1X_0$$

(τόσο η τρέχουσα κατάσταση $[A_1A_0]_t$ όσο και η επόμενη $[A_1A_0]_{t+1}$ καθώς και οι είσοδοι X_1X_0 θεωρούνται διψήφιοι μη προσημασμένοι δυαδικοί αριθμοί ενώ το κρατούμενο που προκύπτει από την πρόσθεση των δύο παραπάνω διψήφιων δυαδικών αριθμών αγνοείται).

Β) Σχεδιάστε το ίδιο κύκλωμα χρησιμοποιώντας ημιαθροιστές, πλήρεις αθροιστές και D Flip Flops θετικής ακμής πυροδότησης. Δεν μπορείτε να χρησιμοποιήσετε λογικές πύλες.

Λύση:

(Α) Αρχικά θα σχεδιάσουμε το διάγραμμα καταστάσεων. Αφού το ακολουθιακό κύκλωμα απαρτίζεται από δύο Flip Flops οι καταστάσεις στις οποίες μπορεί να βρεθεί είναι οι ακόλουθες τέσσερις: 00, 01, 10, 11. Οι μεταβάσεις που συμβαίνουν εξαρτώνται από τις τιμές των εξωτερικών εισόδων X_1X_0 και παρουσιάζονται στο ακόλουθο διάγραμμα καταστάσεων:



Με βάση το παραπάνω διάγραμμα καταστάσεων προκύπτει ο πίνακας καταστάσεων. Οι τιμές των εισόδων των FFs προκύπτουν από τις τιμές της παρούσας και επόμενης κατάστασης καθενός FF σε συνδυασμό με τον πίνακα διέγερσης του FF. Ο πίνακας καταστάσεων ακολουθεί:

ΕΙΣΟΔΟΙ και ΠΑΡΟΥΣΑ ΚΑΤΑΣΤΑΣΗ				***ΕΠΟΜΕΝΗ ΚΑΤΑΣΤΑΣΗ***		***ΕΙΣΟΔΟΙ FFs***	
X_1	X_0	A_1	A_0	A_1	A_0	JA_1KA_1	JA_0KA_0
0	0	0	0	0	0	0 x	0 x
0	0	0	1	0	1	0 x	x 0
0	0	1	0	1	0	x 0	0 x
0	0	1	1	1	1	x 0	x 0
0	1	0	0	0	1	0 x	1 x
0	1	0	1	1	0	1 x	x 1
0	1	1	0	1	1	x 0	1 x
0	1	1	1	0	0	x 1	x 1
1	0	0	0	1	0	1 x	0 x
1	0	0	1	1	1	1 x	x 0
1	0	1	0	0	0	x 1	0 x
1	0	1	1	0	1	x 1	x 0
1	1	0	0	1	1	1 x	1 x
1	1	0	1	0	0	0 x	x 1
1	1	1	0	0	1	x 1	1 x
1	1	1	1	1	0	x 0	x 1

Το επόμενο βήμα είναι ο προσδιορισμός των απλοποιημένων συναρτήσεων εισόδου των FFs με χρήση χαρτών karnaugh όπως φαίνεται ακόλουθα:

JA_1

X_1X_0 \ A_1A_0	00	01	11	10
00			x	x
01		1	x	x
11	1		x	x
10	1	1	x	x

KA_1

X_1X_0 \ A_1A_0	00	01	11	10
00	x	x		
01	x	x	1	
11	x	x		1
10	x	x	1	1

JA_0

X_1X_0 \ A_1A_0	00	01	11	10
00		x	x	
01	1	x	x	1
11	1	x	x	1
10		x	x	

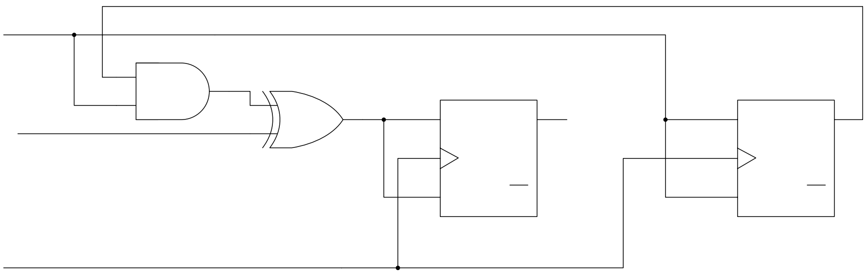
KA_0

X_1X_0 \ A_1A_0	00	01	11	10
00	x			x
01	x	1	1	x
11	x	1	1	x
10	x			x

$$JA_1=KA_1=X_1X_0'+X_1A_0'+X_1'X_0A_0=X_1(X_0'+A_0')+X_1'(X_0A_0)=X_1(X_0A_0)'+X_1'(X_0A_0)=X_1\oplus(X_0A_0)$$

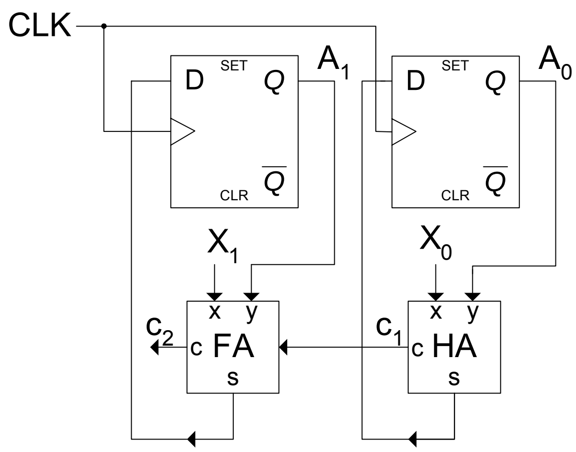
$$JA_0=KA_0=X_0$$

Χρησιμοποιώντας τις παραπάνω λογικές συναρτήσεις σχεδιάζουμε το ζητούμενο κύκλωμα το οποίο είναι το ακόλουθο:



(Β) Εφόσον η επόμενη κατάσταση $[A_1A_0]_{t+1}$ προκύπτει από το άθροισμα της δυαδικής τιμής της παρούσας κατάστασης $[A_1A_0]_t$ με τη δυαδική τιμή X_1X_0 καταλαβαίνουμε εύκολα ότι οι αθροιστές/ημιαθροιστές θα χρησιμοποιηθούν για να υπολογίζουν σε κάθε κύκλο την επόμενη κατάσταση. Η πρόσθεση που πρέπει να υλοποιηθεί φαίνεται ακόλουθα:

Το κρατούμενο c_1 προκύπτει από την πρόσθεση του A_0 με το X_0 και το κρατούμενο c_2 από την πρόσθεση $c_1+A_1+X_1$. Σύμφωνα με την εκφώνηση το κρατούμενο c_2 αγνοείται. Άρα απαιτείται ένας πλήρης αθροιστής και ένας ημιαθροιστής και το ζητούμενο κύκλωμα είναι το ακόλουθο:

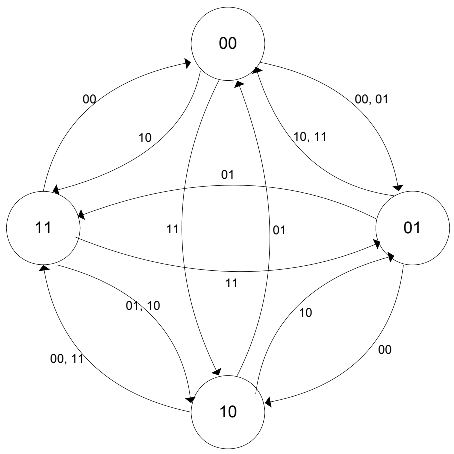


ΑΣΚΗΣΗ 110

Υλοποιήστε ακολουθιακό κύκλωμα το οποίο διαθέτει δύο εισόδους D, G και δύο JK flip flops A_1, A_0. Το κύκλωμα μετράει προς τα πάνω όταν D=0 και προς τα κάτω όταν D=1. Η μέτρηση γίνεται στο δυαδικό σύστημα όταν G=0 και σε κώδικα Gray όταν G=1. Ως εξόδους του κυκλώματος θεωρήστε τις κανονικές εξόδους Q των δύο flip flops. Χρησιμοποιήστε τον ελάχιστο δυνατό αριθμό λογικών πυλών.

<u>Απάντηση</u>

Αρχικά θα σχεδιάσουμε το διάγραμμα καταστάσεων του ακολουθιακού κυκλώματος. Πάνω από κάθε βέλος μετάβασης τοποθετούμε τις αντίστοιχες τιμές των εισόδων DG.



Με βάση το παραπάνω διάγραμμα καταστάσεων προκύπτει ο πίνακας καταστάσεων. Οι τιμές των εισόδων των FFs προκύπτουν από τις τιμές της παρούσας και επόμενης κατάστασης καθενός FF σε συνδυασμό με τον πίνακα διέγερσης του FF. Ο πίνακας καταστάσεων ακολουθεί:

ΕΙΣΟΔΟΙ και ΠΑΡΟΥΣΑ ΚΑΤΑΣΤΑΣΗ				*ΕΠΟΜΕΝΗ ΚΑΤΑΣΤΑΣΗ*		*ΕΙΣΟΔΟΙ FFs*	
D	*G*	*A_1*	*A_0*	*A_1*	*A_0*	*JA_1KA_1*	*JA_0KA_0*
0	0	0	0	0	1	0 x	1 x
0	0	0	1	1	0	1 x	x 1

0	0	1	0	1	1	x	0	1	x
0	0	1	1	0	0	x	1	x	1
0	1	0	0	0	1	0	x	1	x
0	1	0	1	1	1	1	x	x	0
0	1	1	0	0	0	x	1	0	x
0	1	1	1	1	0	x	0	x	1
1	0	0	0	1	1	1	x	1	x
1	0	0	1	0	0	0	x	x	1
1	0	1	0	0	1	x	1	1	x
1	0	1	1	1	0	x	0	x	1
1	1	0	0	1	0	1	x	0	x
1	1	0	1	0	0	0	x	x	1
1	1	1	0	1	1	x	0	1	x
1	1	1	1	0	1	x	1	x	0

Το επόμενο βήμα είναι ο προσδιορισμός των απλοποιημένων συναρτήσεων εισόδου των FFs με χρήση χαρτών karnaugh όπως φαίνεται ακόλουθα:

JA_1

DG \ A_1A_0	00	01	11	10
00		1	x	x
01		1	x	x
11	1		x	x
10	1		x	x

KA_1

DG \ A_1A_0	00	01	11	10
00	x	x	1	
01	x	x		1
11	x	x	1	
10	x	x		1

JA_0

DG \ A_1A_0	00	01	11	10
00	1	x	x	1
01	1	x	x	
11		x	x	1
10	1	x	x	1

KA_0

DG \ A_1A_0	00	01	11	10
00	x	1	1	x
01	x		1	x
11	x	1		x
10	x	1	1	x

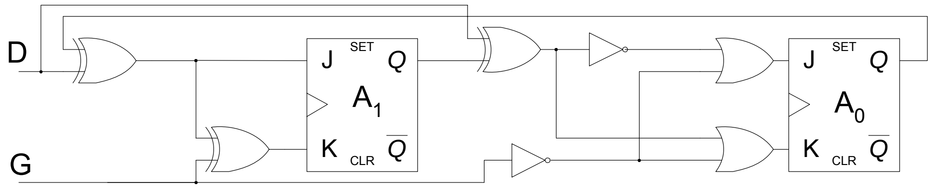
$$JA_1= DA_0'+ D'A_0=D\oplus A_0$$

$$KA_1= DG'A_0'+ DGA_0+ D'GA_0'+ D'G'A_0= (D\oplus G) A_0'+(D\oplus G)'A_0= D\oplus G\oplus A_0= G\oplus JA_1$$

$$JA_0=G'+D'A_1'+DA_1= G'+(D\oplus A_1)'$$

$$KA_0= G'+DA_1'+D'A_1= G'+(D\oplus A_1)$$

Χρησιμοποιώντας τις παραπάνω λογικές συναρτήσεις σχεδιάζουμε το ζητούμενο κύκλωμα το οποίο είναι το ακόλουθο:



ΑΣΚΗΣΗ 111

Έστω αθροιστής δύο αριθμών σε παράσταση συμπληρώματος ως προς 2 ο οποίος εκτελεί μία πρόσθεση σε κάθε περίοδο ρολογιού. Έστω *A*, *B* τα περισσότερο σημαντικά δυαδικά ψηφία των δύο αριθμών και *S* το περισσότερο σημαντικό δυαδικό ψηφίο του αποτελέσματος της πρόσθεσης.

(Α) Σχεδιάστε ένα συνδυαστικό κύκλωμα με μία έξοδο που να λαμβάνει την τιμή 1 αν η πρόσθεση προκάλεσε υπερχείλιση και την τιμή 0 σε αντίθετη περίπτωση. Χρησιμοποιήστε μόνο λογικές πύλες NOR και NOT.

(Β) Χρησιμοποιήστε ένα T-FF αρνητικής ακμής πυροδότησης και μόνο λογικές πύλες XOR για να σχεδιάσετε ένα κύκλωμα αποθήκευσης της τιμής της εξόδου του κυκλώματος του ερωτήματος (Α). Συγκεκριμένα, η έξοδος του FF πρέπει να λαμβάνει την τιμή 1 αν κατά τη διάρκεια της προηγούμενης περιόδου του ρολογιού συνέβη υπερχείλιση και την τιμή 0 σε αντίθετη περίπτωση.
(Υπόδειξη: περισσότερες πληροφορίες για την υπερχείλιση μπορείτε να βρείτε στην ενότητα 3.1.2. του βιβλίου της Αρχιτεκτονικής Υπολογιστών Ι.)

Λύση:

(Α) Γνωρίζουμε ότι εφόσον οι δυαδικοί αριθμοί που προστίθενται είναι σε μορφή συμπληρώματος ως προς 2, το περισσότερο σημαντικό ψηφίο υποδηλώνει το πρόσημό τους. Συγκεκριμένα αν το περισσότερο σημαντικό ψηφίο είναι 0 ο αριθμός είναι θετικός ενώ αν είναι 1 ο αριθμός είναι αρνητικός. Υπερχείλιση μπορούμε να έχουμε μόνο στη περίπτωση πρόσθεσης δύο ομόσημων αριθμών και συγκεκριμένα όταν το αποτέλεσμα έχει πρόσημο διαφορετικό από αυτό των αριθμών που προσθέσαμε (π.χ. προσθέτω δύο θετικούς αριθμούς και έχω αποτέλεσμα που είναι αρνητικός αριθμός). Άρα έχουμε τον παρακάτω πίνακα αληθείας:

A	***B***	***S***	***ΥΠΕΡΧΕΙΛΙΣΗ (V)***
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	0

Με τη βοήθεια του χάρτη Karnaugh τριών μεταβλητών που ακολουθεί

$A \backslash BS$	00	01	11	10
0	0		0	0
1	0	0	0	

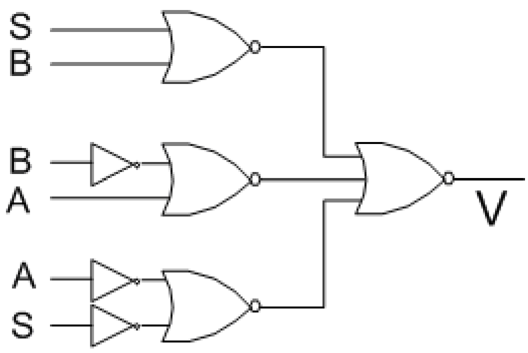
βρίσκουμε ότι $V' = B'S' + AS + A'B$

Συνεπώς *V* = [B΄S΄ + AS + A΄B]΄ =
(B΄S΄)΄ (AS)΄ (A΄B)΄ = (B+S) (A΄+S΄) (A+B΄)

Στη συνέχεια με διπλή αντιστροφή έχουμε:

V ΄΄ = V = [(B+S) (A΄+S΄) (A+B΄)]΄΄ = [(B+S)΄ + (A΄+S΄)΄ + (A+B΄)΄] ΄

Άρα το ζητούμενο κύκλωμα είναι το παρακάτω:

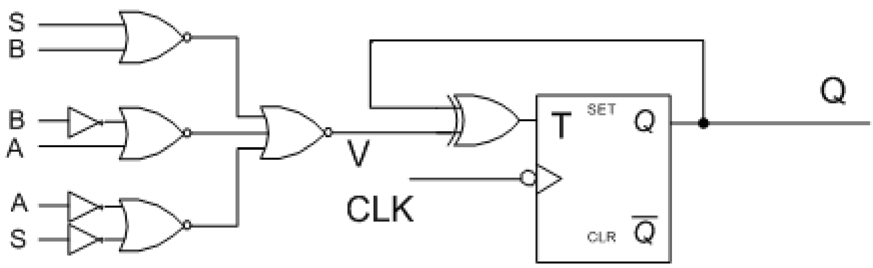


(Β) Χρειάζεται τώρα να σχεδιάσουμε ένα κύκλωμα, που θα αποτελείται από ένα T flip flop και λογικές πύλες XOR, το οποίο θα αποθηκεύει την τιμή της υπερχείλισης V. Συγκεκριμένα, η έξοδος V του κυκλώματος που σχεδιάσαμε στο προηγούμενο ερώτημα θα αξιοποιηθεί ως είσοδος στο νέο κύκλωμα ενώ η έξοδος του νέου κυκλώματος πρέπει να λαμβάνει την τιμή 1 αν κατά τη διάρκεια της προηγούμενης περιόδου του ρολογιού συνέβη υπερχείλιση (*V*=1) και την τιμή 0 αν δε συνέβη υπερχείλιση (*V*=0).

Κατασκευάζουμε τον πίνακα καταστάσεων και προσδιορίζουμε τη συνάρτηση εισόδου του T flip-flop χρησιμοποιώντας τον πίνακα διέγερσης αυτού.

Παρούσα κατάσταση	**Είσοδος**	***Επόμενη κατάσταση***	***Είσοδος T-FF***
Q	***V***	***Q***	***T***
0	0	0	0
0	1	1	1
1	0	0	1
1	1	1	0

Από τον πίνακα καταστάσεων προκύπτει ότι $T = V \oplus Q$.
(Να σημειωθεί ότι με τη συγκεκριμένη συνδεσμολογία το T flip flop λειτουργεί ως ένα D flip flop για το οποίο ισχύει ότι D=V).



Στο ίδιο αποτέλεσμα καταλήγουμε (με περισσότερο κόπο όμως) εάν ακολουθήσουμε εξαρχής τη διαδικασία σχεδίασης ενός σύγχρονου ακολουθιακού κυκλώματος, δηλαδή (α) δημιουργήσουμε τον πίνακα καταστάσεων του κυκλώματος με τρεις εισόδους A, B, S και ένα T flip-flop, (β) απλοποιήσουμε τη συνάρτηση της εισόδου T του flip-flop με χάρτη Karnaugh και (γ) εκτελέσουμε αλγεβρικούς μετασχηματισμούς ώστε να υλοποιήσουμε την απλοποιημένη συνάρτηση με λογικές πύλες XOR, NOR και NOT.

ΑΣΚΗΣΗ 112

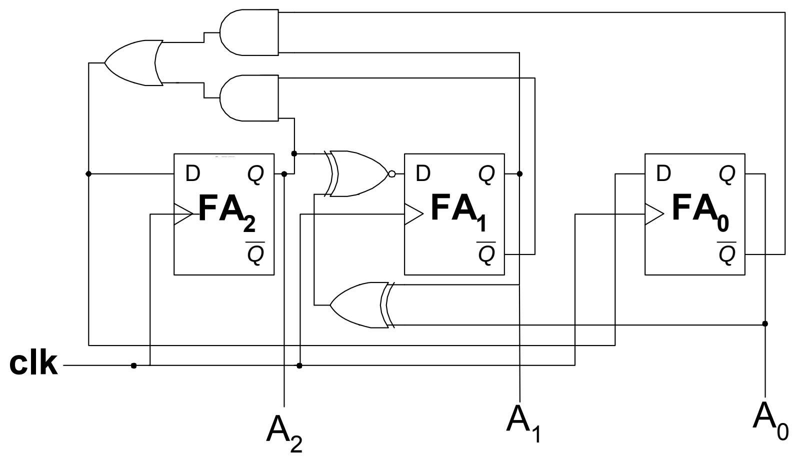
Δίνεται το σύγχρονο ακολουθιακό κύκλωμα με 3 D flip flops του σχήματος που ακολουθεί. Έστω ότι το κύκλωμα ξεκινά πάντα από την κατάσταση $A_2A_1A_0$=000.

(A) Να επανασχεδιαστεί το κύκλωμα χρησιμοποιώντας 3 flip flop τύπου JK θετικής ακμής πυροδότησης.

(B) Να επανασχεδιαστεί το κύκλωμα χρησιμοποιώντας 3 flip flop τύπου T θετικής ακμής πυροδότησης.

(Γ) Μπορείτε να σχεδιάσετε το παραπάνω κύκλωμα χρησιμοποιώντας μόνο 2 flip flop;

Σε κάθε περίπτωση σχεδιάστε τα απλούστερα δυνατά κυκλώματα (αν είναι δυνατόν χωρίς τη χρήση επιπλέον λογικών πυλών) λαμβάνοντας υπόψη ότι τα ζητούμενα κυκλώματα δεν απαιτείται να είναι αυτόματης διόρθωσης.



Λύση:

Η σχεδίαση του κυκλώματος πρέπει να γίνει σε δύο στάδια. Αρχικά θα αναλύσουμε το κύκλωμα που δίνεται και θα δημιουργήσουμε το διάγραμμα καταστάσεων. Κατόπιν θα σχεδιάσουμε το ακολουθιακό κύκλωμα χρησιμοποιώντας JK flip flop και T flip flop.

Ανάλυση Κυκλώματος

Οι λογικές συναρτήσεις για τις εισόδους των τριών D flip flops είναι οι ακόλουθες:

$$DA_2 = A_2A_1' + A_1A_0'$$

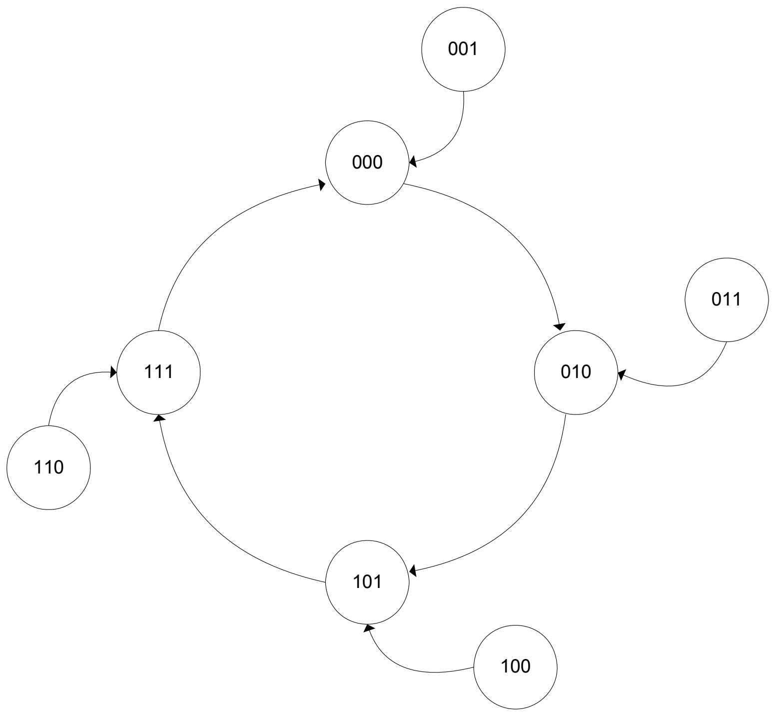
$$DA_1 = (A_2 \oplus A_1 \oplus A_0)'$$

$$DA_0 = A_2A_1' + A_1A_0'$$

Με βάση τις παραπάνω εξισώσεις, ο πίνακας καταστάσεων είναι ο ακόλουθος

Παρούσα Κατάσταση			**Είσοδοι Flip Flop**			**Επόμενη Κατάσταση**		
A_2	A_1	A_0	DA_2	DA_1	DA_0	A_2	A_1	A_0
0	0	0	0	1	0	0	1	0
0	0	1	0	0	0	0	0	0
0	1	0	1	0	1	1	0	1
0	1	1	0	1	0	0	1	0
1	0	0	1	0	1	1	0	1
1	0	1	1	1	1	1	1	1
1	1	0	1	1	1	1	1	1
1	1	1	0	0	0	0	0	0

Με βάση τον παραπάνω πίνακα φτιάχνουμε το ακόλουθο διάγραμμα καταστάσεων



Εφόσον το κύκλωμα βρίσκεται στην κατάσταση 000 κατά την αρχικοποίηση καταλαβαίνουμε ότι το κύκλωμα θα εκτελεί πάντα τον κύκλο 000→010→101→111→000. Επιπλέον παρατηρούμε ότι είναι αυτόματης διόρθωσης εφόσον εάν βρεθεί σε οποιαδήποτε άκυρη κατάσταση (κατάσταση στην οποία δεν μπορεί να βρεθεί υπό κανονικές συνθήκες) επανέρχεται αμέσως σε μία έγκυρη. Για να μπορέσουμε να απλοποιήσουμε το κύκλωμα όσο το δυνατόν περισσότερο θα πρέπει να εκμεταλλευτούμε το γεγονός ότι το τελικό κύκλωμα δεν χρειάζεται να είναι αυτόματης διόρθωσης. Έτσι η επόμενη κατάσταση μετά από κάθε άκυρη κατάσταση είναι αδιάφορη.

(A) Σχεδίαση Κυκλώματος με 3 JK flip flops

Με βάση τα παραπάνω προκύπτει ο πίνακας καταστάσεων. Οι τιμές των εισόδων των FFs προκύπτουν από τις τιμές της παρούσας και επόμενης κατάστασης καθενός FF σε συνδυασμό με τον πίνακα διέγερσης του FF. Ο πίνακας καταστάσεων ακολουθεί:

Παρούσα Κατάσταση			**Επόμενη Κατάσταση**			**Είσοδοι Flip Flop**					
A_2	**A_1**	**A_0**	**A_2**	**A_1**	**A_0**	**JA_2**	**KA_2**	**JA_1**	**KA_1**	**JA_0**	**KA_0**
0	0	0	0	1	0	0	x	1	x	0	x
0	0	1	x	x	x	x	x	x	x	X	x
0	1	0	1	0	1	1	x	x	1	1	x
0	1	1	x	x	x	x	x	x	x	X	x
1	0	0	x	x	x	x	x	x	x	X	x
1	0	1	1	1	1	x	0	1	x	X	0
1	1	0	x	x	x	x	x	x	x	X	x
1	1	1	0	0	0	x	1	x	1	X	1

Με μία πρώτη ματιά στον πίνακα καταστάσεων παρατηρούμε ότι JA_1=KA_1=1 με αντικατάσταση όλων των αδιάφορων όρων με τη μονάδα στις αντίστοιχες στήλες. Επίσης παρατηρούμε ότι JA_2=JA_0 και KA_2=KA_0. Θα χρησιμοποιήσουμε χάρτη Karnaugh για τις εισόδους JA_2 και KA_2.

JA_2

A_2 \ A_1A_0	00	01	11	10
0		x	x	1
1	x	x	x	x

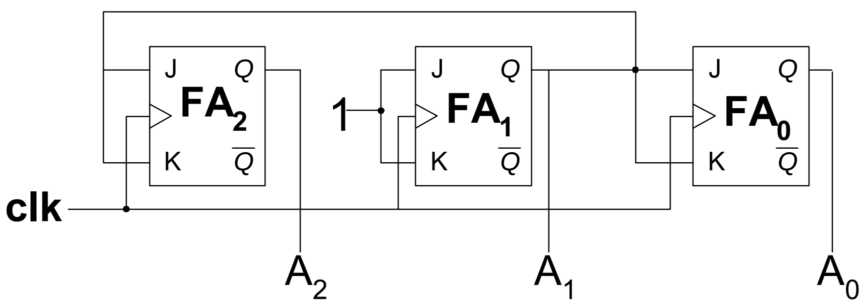
JA_2=A_1

KA_2

A_2 \ A_1A_0	00	01	11	10
0	x	x	x	x
1	x		1	x

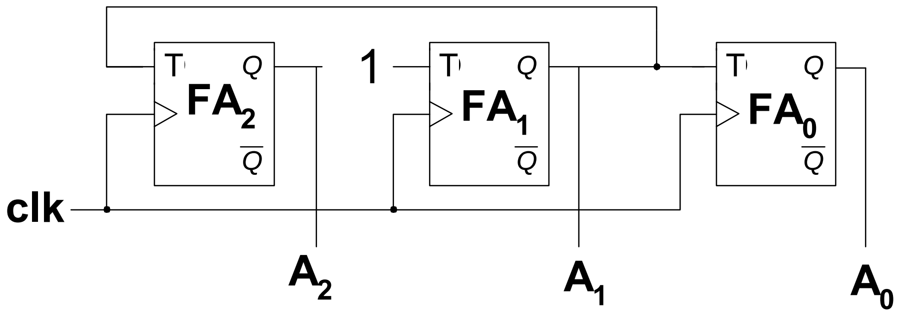
KA_2=A_1

Χρησιμοποιώντας τις παραπάνω λογικές συναρτήσεις σχεδιάζουμε το ζητούμενο κύκλωμα το οποίο είναι το ακόλουθο:



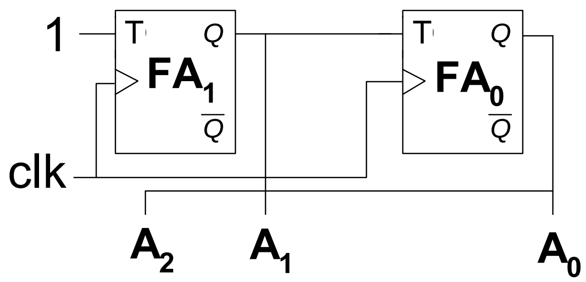
(Β) Σχεδίαση κυκλώματος με 3 T flip flops

Για το (Β) ερώτημα δεν απαιτείται να επαναλάβουμε τη διαδικασία σχεδίασης καθώς γνωρίζουμε από τη θεωρία ότι ένα T flip flop είναι ουσιαστικά ένα JK flip flop με βραχυκυκλωμένες τις εισόδους J, K. Παρατηρούμε από το παραπάνω κύκλωμα ότι και τα τρία JK flip flop έχουν βραχυκυκλωμένες τις εισόδους τους. Έτσι η χρήση των T flip flop είναι προφανής.



(Γ) Σχεδίαση κυκλώματος με 2 flip flops

Παρατηρώντας τα δύο παραπάνω κυκλώματα βλέπουμε ότι τα flip flop FA_2, FA_0 έχουν τις ίδιες εισόδους, οπότε το ένα από τα δύο είναι περιττό. Άρα το κύκλωμα για την περίπτωση χρήσης T flip flops είναι το ακόλουθο:



Παρατηρήστε ότι το παραπάνω κύκλωμα μοιάζει με έναν απαριθμητή 2 δυαδικών ψηφίων εάν αγνοήσουμε το A_2 και θωρήσουμε ως λιγότερο σημαντικό ψηφίο της μέτρησης το A_1 και ως περισσότερο σημαντικό ψηφίο της μέτρησης το A_0.

VII. Σχεδίαση Καταχωρητών

ΑΣΚΗΣΗ 113

Σχεδιάστε το κύκλωμα ενός καταχωρητή 4-bit με τέσσερα flip-flops τύπου D θετικής ακμής πυροδότησης και τέσσερις πολυπλέκτες 4-σε-1 (χωρίς είσοδο ενεργοποίησης). Ο καταχωρητής θα έχει δύο εισόδους επιλογής τρόπου λειτουργίας s_1 και s_0 και θα λειτουργεί σύμφωνα με τον ακόλουθο πίνακα.

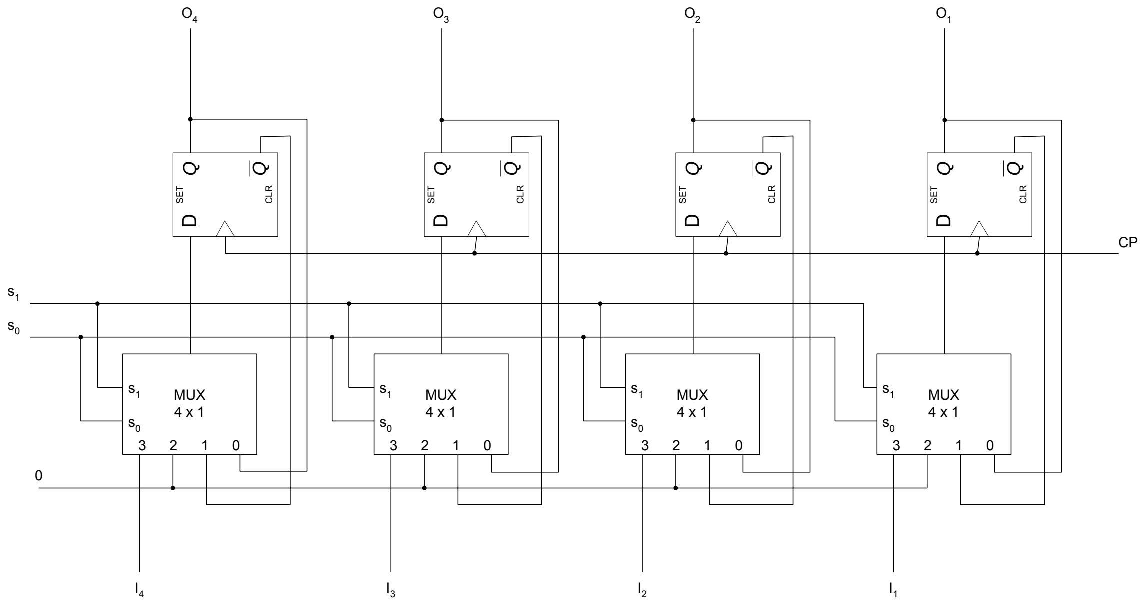
s_1	s_0	Λειτουργία Καταχωρητή
0	0	Καμία αλλαγή
0	1	Αντιστροφή των τεσσάρων εξόδων
1	0	Σύγχρονος μηδενισμός του καταχωρητή
1	1	Παράλληλη φόρτωση δεδομένων από εξωτερικές εισόδους I_1-I_4

Λύση:

Συνδέουμε την είσοδο *D* κάθε ενός D flip-flop με την έξοδο ενός πολυπλέκτη. Συνδέουμε όλες τις εισόδους ρολογιού των 4 D flip-flops σε ένα κοινό σήμα χρονισμού CP και συνδέουμε όλες τις εισόδους επιλογής s_1 (αντίστοιχα s_0) σε ένα κοινό σήμα επιλογής s_1 (αντίστοιχα s_0). Έχουμε τις παρακάτω περιπτώσεις:

- Όταν (s_1,s_0) = 00, τότε θα πρέπει ο καταχωρητής να διατηρεί τα δεδομένα του. Για να διατηρήσει ο καταχωρητής τα δεδομένα του θα πρέπει να συνδέσουμε την είσοδο *D* κάθε flip-flop στην έξοδο *Q* του ίδιου flip-flop. Αυτό επιτυγχάνεται συνδέοντας την έξοδο *Q* κάθε flip-flop στην είσοδο 0 του πολυπλέκτη που αντιστοιχεί στο flip-flop αυτό καθώς όταν (s_1,s_0) = 00 η έξοδος του κάθε πολυπλέκτη είναι ίση με την είσοδο 0.
- Όταν (s_1,s_0) = 01, τότε θα πρέπει ο καταχωρητής να αντιστρέφει τα δεδομένα του. Για να αντιστρέψει ο καταχωρητής τα δεδομένα του θα πρέπει να συνδέσουμε την είσοδο *D* κάθε flip-flop στην έξοδο *Q΄* του ίδιου flip-flop. Αυτό επιτυγχάνεται συνδέοντας την έξοδο *Q΄* κάθε flip-flop στην είσοδο 1 του πολυπλέκτη που αντιστοιχεί στο flip-flop αυτό καθώς όταν (s_1,s_0) = 01 η έξοδος του κάθε πολυπλέκτη είναι ίση με την είσοδο 1.
- Όταν (s_1,s_0) = 10, τότε θα πρέπει ο καταχωρητής να μηδενίζεται σύγχρονα με το σήμα χρονισμού. Για να μηδενίσει ο καταχωρητής τα δεδομένα του θα πρέπει να συνδέσουμε την είσοδο *D* κάθε flip-flop στην τιμή 0. Αυτό επιτυγχάνεται συνδέοντας ένα σήμα που έχει μόνιμα την τιμή 0 στην είσοδο 2 όλων των πολυπλεκτών καθώς όταν (s_1,s_0) = 10 η έξοδος του κάθε πολυπλέκτη είναι ίση με την είσοδο 2.
- Όταν (s_1,s_0) = 11, τότε θα πρέπει ο καταχωρητής να φορτώνεται παράλληλα από την τιμή των εξωτερικών εισόδων I_1-I_4. Για να φορτωθεί ο καταχωρητής παράλληλα με νέα δεδομένα θα πρέπει να συνδέσουμε την είσοδο *D* κάθε flip-flop σε μία είσοδο I_i (i=1,…4). Αυτό επιτυγχάνεται συνδέοντας την είσοδο I_i (i=1,…4) στην είσοδο 3 των πολυπλεκτών καθώς όταν (s_1,s_0) = 11 η έξοδος του κάθε πολυπλέκτη είναι ίση με την είσοδο 3.

Επομένως το λογικό διάγραμμα του καταχωρητή είναι αυτό που φαίνεται στο παρακάτω σχήμα.



ΑΣΚΗΣΗ 114

Σχεδιάστε έναν καταχωρητή 4 δυαδικών ψηφίων με δυνατότητες δεξιάς ολίσθησης και παράλληλης φόρτωσης. Ο καταχωρητής περιέχει 2 εισόδους ελέγχου λειτουργίας: shift και load και λειτουργεί σύμφωνα με τον ακόλουθο πίνακα λειτουργιών. Έχετε στη διάθεσή σας 4 D flip-flops θετικής ακμής πυροδότησης καθώς και 4 πολυπλέκτες 4-σε-1.

shift	***load***	***Λειτουργία καταχωρητή***
0	0	Καμία αλλαγή
0	1	Παράλληλη φόρτωση δεδομένων
1	X	Δεξιά ολίσθηση

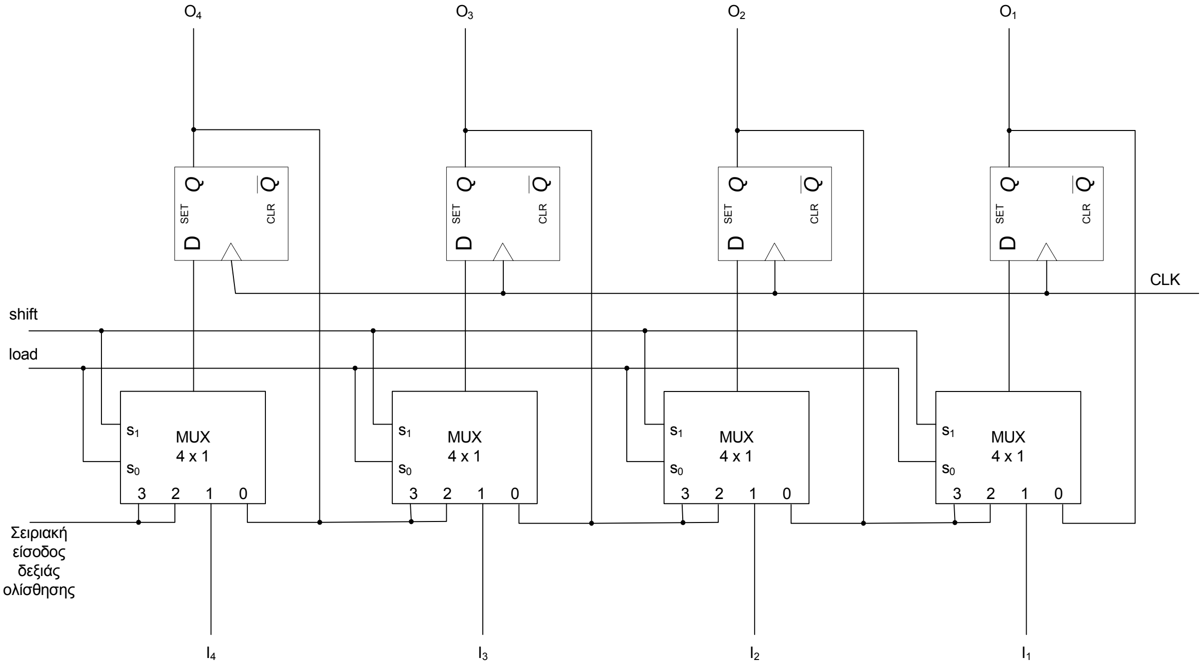
Λύση:

Συνδέουμε την είσοδο *D* κάθε ενός D flip-flop με την έξοδο ενός πολυπλέκτη. Συνδέουμε όλες τις εισόδους ρολογιού των 4 D flip-flops σε ένα κοινό σήμα χρονισμού CLK και συνδέουμε όλες τις εισόδους επιλογής s_1 (αντίστοιχα s_0) των πολυπλεκτών στην είσοδο *shift* (αντίστοιχα *load*). Έχουμε τις παρακάτω περιπτώσεις:

- Όταν (*shift*,*load*) = 00, τότε θα πρέπει ο καταχωρητής να διατηρεί τα δεδομένα του. Για να διατηρήσει ο καταχωρητής τα δεδομένα του θα πρέπει να συνδέσουμε την είσοδο *D* κάθε flip-flop στην έξοδο *Q* του ίδιου flip-flop. Αυτό επιτυγχάνεται συνδέοντας την έξοδο *Q* κάθε flip-flop στην είσοδο 0 του πολυπλέκτη που αντιστοιχεί στο flip-flop αυτό καθώς όταν (*shift*,*load*) = 00 η έξοδος του κάθε πολυπλέκτη είναι ίση με την είσοδο 0.
- Όταν (*shift*,*load*) = 01, τότε θα πρέπει ο καταχωρητής να φορτώνεται παράλληλα από την τιμή των εξωτερικών εισόδων I_1-I_4. Για να φορτωθεί ο καταχωρητής παράλληλα με νέα δεδομένα θα πρέπει να συνδέσουμε την είσοδο *D* κάθε flip-flop σε μία είσοδο I_i (i=1,…4). Αυτό επιτυγχάνεται συνδέοντας την είσοδο I_i (i=1,…4) στην είσοδο 1 των πολυπλεκτών καθώς όταν (*shift*,*load*) = 01 η έξοδος του κάθε πολυπλέκτη είναι ίση με την είσοδο 1.
- Όταν (*shift*,*load*) = 10 ή 11, τότε θα πρέπει ο καταχωρητής να εκτελεί δεξιά ολίσθηση. Αυτό επιτυγχάνεται συνδέοντας την έξοδο Q του κάθε flip-flop στην

είσοδο 2 και 3 του πολυπλέκτη που βρίσκεται δεξιά του ενώ αντίστοιχα συνδέουμε τις εισόδους 2 και 3 του αριστερότερου πολυπλέκτη στην είσοδο σειριακής δεξιάς ολίσθησης. Όταν (*shift*,*load*) = 10 ή 11 η έξοδος του κάθε πολυπλέκτη είναι ίση με την είσοδο 2 ή 3 αντίστοιχα.

Επομένως το λογικό διάγραμμα του καταχωρητή είναι αυτό που φαίνεται στο παρακάτω σχήμα.



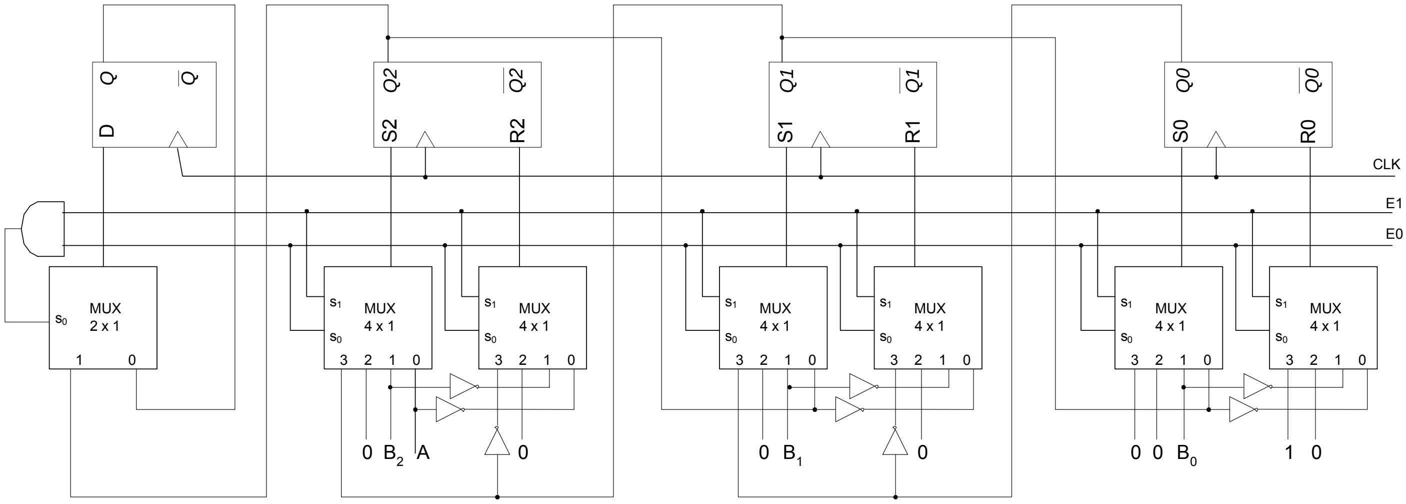
ΑΣΚΗΣΗ 115

Σχεδιάστε το κύκλωμα ενός καταχωρητή 3 δυαδικών ψηφίων με δύο εξωτερικά σήματα E_1 και E_0 ο οποίος εκτελεί τις παρακάτω λειτουργίες:

E_1	E_0	***Λειτουργία καταχωρητή***
0	0	Σειριακή φόρτωση του αριθμού $a_2a_1a_0$, τα bits του οποίου παρέχονται από μια εξωτερική είσοδο *A* (Υποθέστε ότι παρέχεται ένα bit ανά ωρολογιακό παλμό ξεκινώντας από το λιγότερο σημαντικό bit).
0	1	Παράλληλη φόρτωση του αριθμού $b_2b_1b_0$, τα bits του οποίου παρέχονται από τις εξωτερικές εισόδους B_2, B_1 και B_0.
1	0	Καμία αλλαγή.
1	1	Διπλασιασμός της τιμής του καταχωρητή. (Για τη συγκεκριμένη λειτουργία και επειδή το αποτέλεσμα απαιτεί 4 bits για την αποθήκευση του, έχετε στη διάθεση σας ένα D Flip Flop για την αποθήκευση του περισσότερο σημαντικού ψηφίου του αποτελέσματος. Κατά τη διάρκεια όλων των άλλων λειτουργιών του καταχωρητή το συγκεκριμένο D FF διατηρεί αναλλοίωτα τα δεδομένα του.)

Έχετε στη διάθεσή σας 3 SR Flip Flops και 1 D Flip Flop, όλα πυροδοτούμενα στη θετική ακμή, 6 πολυπλέκτες 4-σε-1, 1 πολυπλέκτη 2-σε-1, 1 πύλη AND και όσες πύλες NOT χρειάζεστε.

Λύση:



Συνδέουμε την κάθε είσοδο (S και R) κάθε ενός SR Flip Flop με την έξοδο ενός 4 x 1 πολυπλέκτη, και την είσοδο *D* του D Flip Flop με την έξοδο ενός 2 x 1 πολυπλέκτη. Συνδέουμε όλες τις εισόδους ρολογιού των 4 Flip Flops (3 x SR και 1 x D) σε ένα κοινό σήμα χρονισμού *CLK*. Τέλος συνδέουμε όλες τις εισόδους επιλογής s_1 (αντίστοιχα s_0) των 4 x 1 πολυπλεκτών στην είσοδο *E1* (αντίστοιχα *E0*) και την είσοδο επιλογής s_0 του 2 x 1 πολυπλέκτη στην έξοδο μίας πύλης ΚΑΙ που έχει σαν εισόδους τα E_1 και E_0. Έχουμε τις παρακάτω περιπτώσεις:

- Όταν (E_1,E_0) = 00, τότε θα πρέπει ο καταχωρητής να φορτώνει σειριακά τον αριθμό $a_2a_1a_0$, τα bits του οποίου παρέχονται μέσω της εξωτερικής εισόδου *A* (ένα bit ανά ωρολογιακό παλμό ξεκινώντας από το λιγότερο σημαντικό bit). Για να ολοκληρωθεί η σειριακή αυτή λειτουργία και να φορτωθεί ο αριθμός $a_2a_1a_0$ θα απαιτηθούν 3 ωρολογιακοί παλμοί. Στον πρώτο ωρολογιακό παλμό το πρώτο (από αριστερά) SR Flip Flop θα φορτώσει το bit a_0. Στο δεύτερο ωρολογιακό παλμό το bit a_0 θα μεταφερθεί στο μεσαίο SR Flip Flop (λειτουργία δεξιάς ολίσθησης) ενώ το πρώτο (από αριστερά) SR Flip Flop θα φορτώσει το bit a_1. Με την ίδια λογική κατά τον τρίτο ωρολογιακό παλμό το πρώτο (από αριστερά) SR Flip Flop θα φορτώσει το bit a_2, το μεσαίο SR Flip Flop το bit a_1 (λειτουργία δεξιάς ολίσθησης), και το τελευταίο SR Flip Flop το bit a_0 (λειτουργία δεξιάς ολίσθησης). Λαμβάνοντας υπόψη ότι όταν $E_1 = E_0 = 0$ ως έξοδος των έξι 4 x 1 πολυπλεκτών επιλέγεται η είσοδος 0, παρατηρείτε ότι πράγματι το κύκλωμα λειτουργεί όπως περιγράφεται παραπάνω, καθώς το πρώτο (από αριστερά) SR Flip Flop φορτώνει την τιμή της εξωτερικής εισόδου *A*, το μεσαίο SR Flip Flop την τιμή που είναι αποθηκευμένη στο αριστερά του SR Flip Flop (δεξιά ολίσθηση) και το τελευταίο SR Flip Flop την τιμή που είναι αποθηκευμένη στο μεσαίο SR Flip Flop (δεξιά ολίσθηση).
  Όπως παρατηρείτε στο κύκλωμα, το bit που επιθυμούμε να φορτωθεί σε κάποιο SR Flip Flop τροφοδοτείται (μέσω του ενός πολυπλέκτη) στη είσοδο S του συγκεκριμένου FF, ενώ στην είσοδο R τροφοδοτείται (μέσω του δεύτερου πολυπλέκτη) το αντίστροφο του ιδίου bit. Για παράδειγμα αν το bit που θέλουμε να φορτωθεί σε κάποιο SR Flip Flop είναι 1, τότε S=1, R=0 γεγονός που σύμφωνα με το πίνακα αληθείας του SR Flip Flop σημαίνει ότι η τιμή που θα πάρει η έξοδος Q είναι 1, δηλαδή η τιμή που θέλαμε να φορτώσουμε.

- Όταν (E_1,E_0) = 01, τότε θα πρέπει ο καταχωρητής να φορτώνει παράλληλα τον αριθμό $b_2b_1b_0$, τα bits του οποίου παρέχονται μέσω των εξωτερικών εισόδων B_2, B_1 και B_0. Λαμβάνοντας υπόψη ότι όταν $E_1 = 0$, $E_0 = 1$ ως έξοδος των έξι 4 x 1 πολυπλεκτών επιλέγεται η είσοδος 1, παρατηρείτε ότι πράγματι το κύκλωμα λειτουργεί όπως περιγράφεται παραπάνω, καθώς το πρώτο (από αριστερά) SR Flip Flop φορτώνει την τιμή της εξωτερικής εισόδου B_2, το μεσαίο SR Flip Flop την τιμή της εξωτερικής εισόδου B_1 και το τελευταίο SR Flip Flop την τιμή της εξωτερικής εισόδου B_0.

- Όταν (E_1,E_0) = 10, τότε θα πρέπει ο καταχωρητής να διατηρεί τα δεδομένα του. Λαμβάνοντας υπόψη ότι όταν $E_1 = 1$, $E_0 = 0$ ως έξοδος των έξι 4 x 1 πολυπλεκτών επιλέγεται η είσοδος 2, παρατηρείτε ότι αυτό πράγματι συμβαίνει, καθώς οι είσοδοι *S* και *R* όλων των SR Flip Flops γίνονται 0, γεγονός που σύμφωνα με τον πίνακα αληθείας του SR Flip Flop σημαίνει ότι η τιμή του δεν μεταβάλλεται.

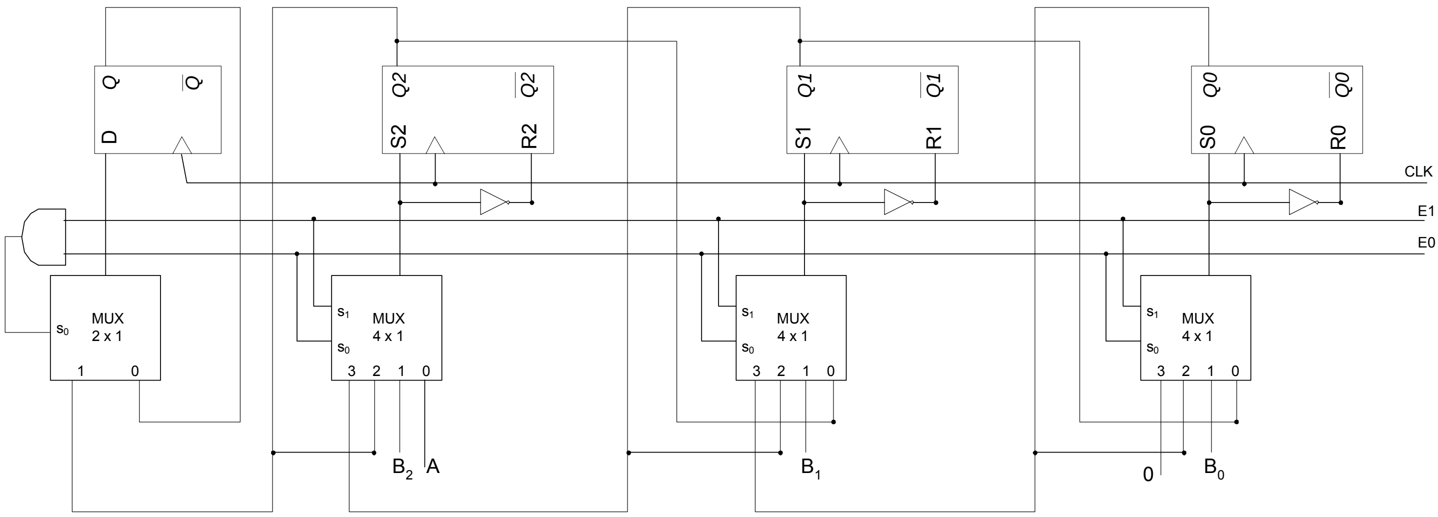
- Όταν (E_1,E_0) = 11, τότε θα πρέπει ο καταχωρητής να διπλασιάζει το περιεχόμενο του. Ο διπλασιασμός αυτός υλοποιείται με αριστερή ολίσθηση των δεδομένων κατά μια θέση. Λαμβάνοντας υπόψη ότι όταν $E_1 = E_0 = 1$ ως έξοδος των έξι 4 x 1 πολυπλεκτών επιλέγεται η είσοδος 3, παρατηρείτε ότι πράγματι το κύκλωμα λειτουργεί όπως περιγράφεται παραπάνω, καθώς το πρώτο (από αριστερά) SR Flip Flop φορτώνει την τιμή του μεσαίου SR Flip Flop (αριστερή ολίσθηση), το μεσαίο SR Flip Flop φορτώνει την τιμή που είναι αποθηκευμένη στο δεξιά του SR Flip Flop (αριστερή ολίσθηση) και το τελευταίο SR Flip Flop φορτώνει την τιμή 0 (καθώς S=0, R=1).

  Σε περίπτωση που τα περιεχόμενα του καταχωρητή είναι αριθμός μεγαλύτερος του 3, το αποτέλεσμα του πολλαπλασιασμού είναι αριθμός μεγαλύτερος από το 7 (που είναι ο μεγαλύτερος αριθμός που μπορεί να αναπαρασταθεί με 3 bits) και συνεπώς απαιτούνται 4 bit για την αποθήκευση του. Το τέταρτο αυτό bit (πλέον σημαντικό ψηφίο του αποτελέσματος) αποθηκεύεται στο D Flip Flop, καθώς για $E_1 = E_0 = 1$ ως έξοδος του 2 x 1 πολυπλέκτη επιλέγεται η είσοδος 1 στην οποία είναι συνδεδεμένη η έξοδος του SR Flip Flop.

Είναι προφανές ότι οι πύλες NOT του παραπάνω σχήματος που χρησιμοποιούνται για την αντιστροφή των εξόδων Q των FFs μπορούν να παραληφθούν καθώς κάθε Flip Flop παρέχει δύο εξόδους Q και Q'. Οι συγκεκριμένες πύλες NOT υπάρχουν στο παραπάνω σχήμα απλά και μόνο για τη διευκόλυνση της «αναγνωσιμότητας» του.

Εναλλακτικά, το κύκλωμα θα μπορούσε να υλοποιηθεί χρησιμοποιώντας μόνο ένα 4 x 1 πολυπλέκτη ανά SR Flip Flop όπως φαίνεται στο σχήμα που ακολουθεί. Προσέξτε ότι η είσοδος *R* σε κάθε SR Flip Flop είναι η αντίστροφη της εισόδου *S*. Η συγκεκριμένη συνδεσμολογία έχει σαν αποτέλεσμα το SR Flip Flop να λειτουργεί σαν D Flip Flop (αν η τιμή της εισόδου *S* είναι 1, τότε S=1, R=0 γεγονός που σύμφωνα με το πίνακα αληθείας του SR Flip Flop σημαίνει ότι η τιμή που θα πάρει η έξοδος *Q* είναι 1. Αν η τιμή της εισόδου *S* είναι 0, τότε S=0, R=1 γεγονός που σύμφωνα με το πίνακα αληθείας του SR Flip Flop σημαίνει ότι η τιμή που θα πάρει η έξοδος *Q* είναι 0 --- δηλαδή το Flip Flop λειτουργεί ακριβώς όπως ένα D Flip Flop όπου $D = S$).

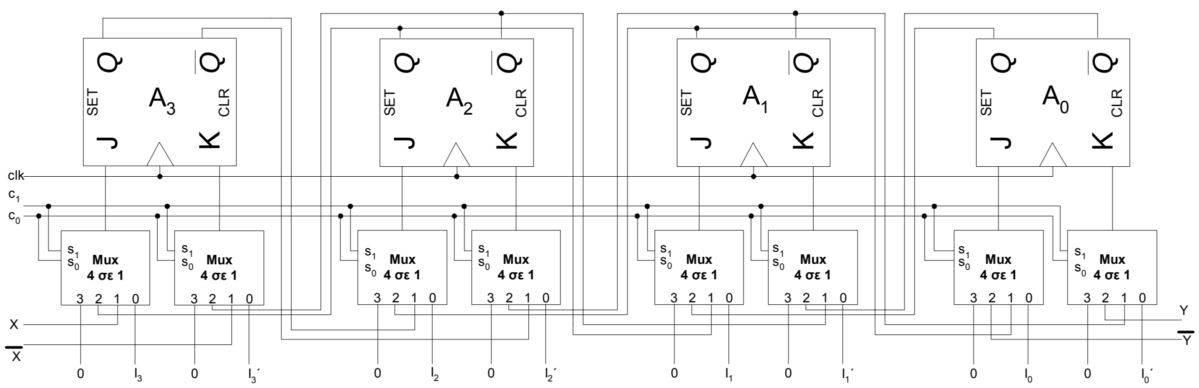
Οι είσοδοι 0, 1 και 3 των πολυπλεκτών δεν διαφοροποιούνται από αυτές της αρχικής λύσης παραπάνω. Διαφορά υπάρχει μόνο στη είσοδο 2 η οποία υποστηρίζει τη λειτουργία της αποθήκευσης (E_1,E_0 = 10 – «Καμία Αλλαγή»). Συγκεκριμένα η είσοδος 2 του κάθε 4 x 1 πολυπλέκτη συνδέεται με την έξοδο *Q* του αντίστοιχου SR Flip Flop, γεγονός που έχει ως αποτέλεσμα τα Flip Flops να διατηρούν σταθερά τα περιεχόμενα τους.



ΑΣΚΗΣΗ 116

Δίνεται το σύγχρονο ακολουθιακό κύκλωμα του σχήματος. Για τί κύκλωμα πρόκειται; Ποια η λειτουργία του όταν οι είσοδοι $\mathbf{c_1c_0}$ παίρνουν αντίστοιχα τις τιμές **00, 01,10, 11**; (Δώστε σύντομες και περιεκτικές απαντήσεις –ενδεικτικά 5 γραμμές ανά περίπτωση- και συμπληρώστε τον συνολικό πίνακα λειτουργίας του κυκλώματος).

$\mathbf{c_1c_0}$	**Λειτουργία**
00	
01	
10	
11	



<u>Απάντηση</u>

Πρόκειται για ένα καταχωρητή των 4 bits με δυνατότητα παράλληλης φόρτωσης, ολίσθησης δεξιά ή αριστερά και διατήρησης των δεδομένων.

Ο πίνακας λειτουργίας ενός JK flip flop είναι ο ακόλουθος:

J	**K**	$\mathbf{Q_{t+1}}$
0	0	Q_t
0	1	0
1	0	1
1	1	Q_t'

Από τον παραπάνω πίνακα παρατηρούμε ότι για να αποθηκευτεί μία τιμή P στο flip flop πρέπει να θέσουμε J=P και K=P' (επιβεβαιώστε το για P=0, 1 από τον παραπάνω πίνακα). Με βάση αυτή την παρατήρηση η λειτουργία του κυκλώματος έχει ως εξής:

α) Για $\mathbf{c_1c_0}$ = 00 γίνεται παράλληλη φόρτωση: στις εισόδους J, K του flip flop A_j ($j\in[0, 3]$) συνδέονται οι παράλληλες είσοδοι I_j και I_j' αντίστοιχα.

β) Για $\mathbf{c_1c_0}$ = 01 γίνεται δεξιά ολίσθηση: στις εισόδους J, K του flip flop A_j ($j\in[0, 2]$) συνδέονται αντίστοιχα η κανονική και συμπληρωματική έξοδος Q, Q' του flip flop A_{j+1} ενώ στις εισόδους J, K του flip flop A_3 συνδέονται η κανονική και συμπληρωματική σειριακή είσοδος X και X' αντίστοιχα.

γ) Για $\mathbf{c_1c_0}$ = 10 γίνεται αριστερή ολίσθηση: στις εισόδους J, K του flip flop A_j ($j\in[1, 3]$) συνδέονται αντίστοιχα η κανονική και συμπληρωματική έξοδος Q, Q' του flip flop A_{j-1} ενώ στις εισόδους J, K του flip flop A_0 συνδέονται η κανονική και συμπληρωματική σειριακή είσοδος Y και Y' αντίστοιχα.

δ) Για $\mathbf{c_1c_0}$ = 11 έχουμε διατήρηση δεδομένων: στις εισόδους J, K του flip flop A_j ($j\in[0, 3]$) συνδέεται το 0.

$\mathbf{c_1c_0}$	**Λειτουργία**
00	Παράλληλη φόρτωση
01	Δεξιά Ολίσθηση
10	Αριστερή Ολίσθηση
11	Διατήρηση δεδομένων

ΑΣΚΗΣΗ 117

Σχεδιάστε το κύκλωμα ενός καταχωρητή 3 δυαδικών ψηφίων με τρία εξωτερικά σήματα E_2, E_1 και E_0 ο οποίος εκτελεί τις παρακάτω λειτουργίες:

E_2	E_1	E_0	*Λειτουργία καταχωρητή*
0	0	0	Παράλληλη φόρτωση του αριθμού $A_2\,A_1\,A_0$ τα bits του οποίου παρέχονται μέσω τριών εξωτερικών εισόδων.
1	0	0	Παράλληλη φόρτωση του **συμπληρώματος ως προς 1** του αριθμού $A_2\,A_1\,A_0$ τα bits του οποίου παρέχονται μέσω τριών εξωτερικών εισόδων.
X	0	1	Δεξιά Κυκλική Ολίσθηση
X	1	X	Καμία αλλαγή.

Έχετε στη διάθεσή σας 3 JK Flip Flops, όλα πυροδοτούμενα στη θετική ακμή, 3 πολυπλέκτες 4-σε-1, 3 πύλες XOR και όσες πύλες NOT χρειάζεστε.

Λύση:

Παρατηρώντας τον πίνακα λειτουργιών του καταχωρητή διαπιστώνουμε ότι οι λειτουργίες που πρέπει να υποστηρίζονται είναι:

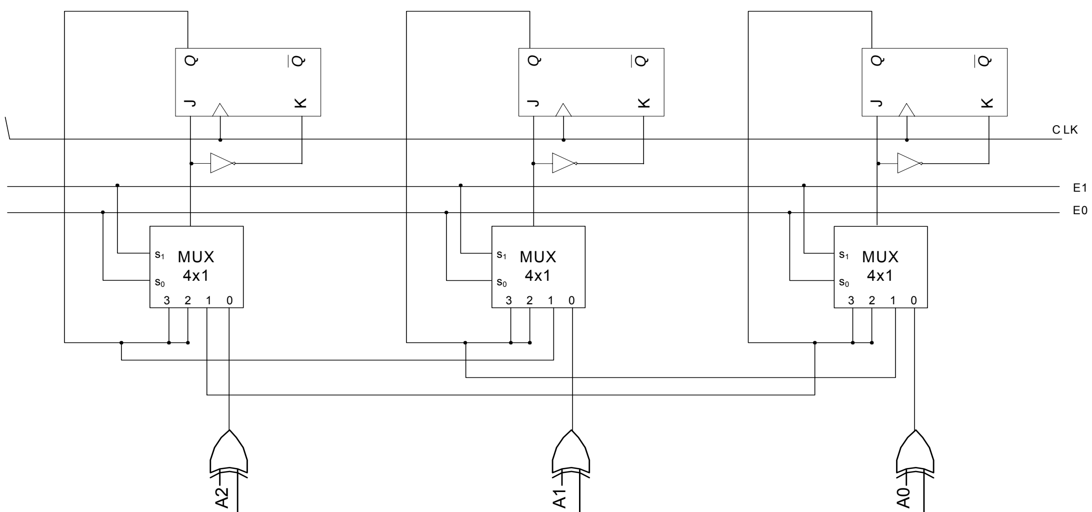
- Παράλληλη φόρτωση ενός αριθμού τριών bits,
- Δεξιά κυκλική ολίσθηση, και
- Αποθήκευση (καμία αλλαγή) των περιεχομένων του.

Για να υποστηριχτούν οι παραπάνω λειτουργίες συνδέουμε την είσοδο J κάθε ενός Flip Flop με την έξοδο ενός 4-σε-1 πολυπλέκτη, και την είσοδο K με την αντίστροφη τιμή (επισημαίνεται ότι η συγκεκριμένη συνδεσμολογία έχει σαν αποτέλεσμα το JK Flip Flop να λειτουργεί σαν D Flip Flop αφού αν η τιμή της εισόδου J είναι 1, τότε J=1, K=0 γεγονός που σύμφωνα με το πίνακα αληθείας του JK Flip Flop σημαίνει ότι η τιμή που θα πάρει η έξοδος Q είναι 1. Αν η τιμή της εισόδου J είναι 0, τότε J=0, K=1 γεγονός που σύμφωνα με το πίνακα αληθείας του JK Flip Flop σημαίνει ότι η τιμή που θα πάρει η έξοδος Q είναι 0 --- δηλαδή το Flip Flop λειτουργεί ακριβώς όπως ένα D Flip Flop όπου D = J).

Η επιλογή της λειτουργίας γίνεται μέσω των εξωτερικών σημάτων E_1 και E_0 (δεν επηρεάζεται από το εξωτερικό σήμα E_2). Συγκεκριμένα, αν $E_1 = E_0 = 0$ ο καταχωρητής πρέπει να εκτελεί παράλληλη φόρτωση, αν $E_1 = 0$ και $E_0 = 1$ τότε ο καταχωρητής πρέπει να εκτελεί δεξιά κυκλική ολίσθηση και τέλος αν $E_1 = 1$ (ανεξαρτήτως από την τιμή του E_0) ο καταχωρητής πρέπει να κρατάει σταθερά τα δεδομένα του. Καταλαβαίνουμε λοιπόν ότι τα σήματα E_1 και E_0 πρέπει να συνδεθούν στις εισόδους επιλογής (s_1 και s_0 αντίστοιχα) των τριών 4-σε-1 πολυπλεκτών που έχουμε διαθέσιμους.

Από τον πίνακα λειτουργίας του καταχωρητή παρατηρούμε ότι η επιλογή των δεδομένων που θα αποθηκευτούν στον καταχωρητή, στην περίπτωση της παράλληλης φόρτωσης, γίνεται μέσω του εξωτερικού σήματος E_2. Συγκεκριμένα αν $E_2 = 0$ θα πρέπει να φορτωθεί ο αριθμός $A_2 A_1 A_0$ ενώ αν $E_2 = 1$ θα πρέπει να φορτωθεί το συμπλήρωμα ως προς 1 του αριθμού $A_2 A_1 A_0$. Μπορούμε λοιπόν να αξιοποιήσουμε το σήμα E_2 ως είσοδο στις τρεις διαθέσιμες πύλες XOR, με δεύτερη είσοδο σε κάθε πύλη τα εξωτερικά σήματα A_2, A_1 και A_0. Με τη συγκεκριμένη συνδεσμολογία επιτυγχάνουμε ελεγχόμενη αντιστροφή των bits A_2, A_1 και A_0. Συγκεκριμένα αν E_2=0, τότε οι έξοδοι των τριών πυλών XOR είναι A_2, A_1 και A_0 ενώ αν E_2=1 τότε οι έξοδοι των τριών πυλών είναι A_2', A_1' και A_0'.

Προκύπτει λοιπόν ότι το ζητούμενο κύκλωμα είναι το παρακάτω.



Η επιλογή των εισόδων των τριών 4-σε-1 πολυπλεκτών έγινε σύμφωνα με το παρακάτω σκεπτικό:

- Όταν (E_1,E_0) = 00, τότε πρέπει ο καταχωρητής να φορτώνει παράλληλα τον αριθμό $A_2\,A_1\,A_0$ (αν E_2 = 0) ή το συμπλήρωμα ως προς 1 του αριθμού $A_2\,A_1\,A_0$ (αν E_2 =1). Λαμβάνοντας υπόψη ότι όταν E_1 = 0, E_0 = 0 ως έξοδος των τριών 4-σε-1 πολυπλεκτών επιλέγεται η είσοδος 0, στην είσοδο 0 του πρώτου (από αριστερά) πολυπλέκτη συνδέουμε την έξοδο της πύλης XOR που αντιστοιχεί στο bit A_2, στην είσοδο 0 του μεσαίου πολυπλέκτη συνδέουμε την έξοδο της πύλης XOR που αντιστοιχεί στο bit A_1, και στην είσοδο 0 του τελευταίου πολυπλέκτη συνδέουμε την έξοδο της πύλης XOR που αντιστοιχεί στο bit A_0.
- Όταν (E_1,E_0) = 01, τότε πρέπει ο καταχωρητής να εκτελεί δεξιά κυκλική ολίσθηση. Λαμβάνοντας υπόψη ότι όταν E_1 = 0, E_0 = 1 ως έξοδος των τριών 4-σε-1 πολυπλεκτών επιλέγεται η είσοδος 1, στην είσοδο 1 του πρώτου (από αριστερά) JK Flip Flop συνδέουμε την έξοδο του τελευταίου FF, στην είσοδο 1 του μεσαίου FF συνδέουμε την έξοδο του πρώτου FF και στην είσοδο 1 του τελευταίου FF συνδέουμε την έξοδο του μεσαίου FF.
- Όταν (E_1,E_0) = 1X, τότε πρέπει ο καταχωρητής να διατηρεί τα δεδομένα του. Λαμβάνοντας υπόψη ότι όταν E_1 = 1, E_0 = 0 ως έξοδος των τριών 4-σε-1 πολυπλεκτών επιλέγεται η είσοδος 2 και ότι όταν E_1 = 1, E_0 = 1 ως έξοδος των τριών 4-σε-1 πολυπλεκτών επιλέγεται η είσοδος 3, θα πρέπει στις εισόδους 2 και 3 όλων των πολυπλεκτών να συνδέσουμε την έξοδο του FF που αντιστοιχεί στον κάθε πολυπλέκτη, ώστε και για τους δύο συνδυασμούς των εξωτερικών σημάτων E_1,E_0 (10 ή 11) το κάθε FF να φορτώνει εκ νέου την τρέχουσα τιμή της εξόδου του Q.