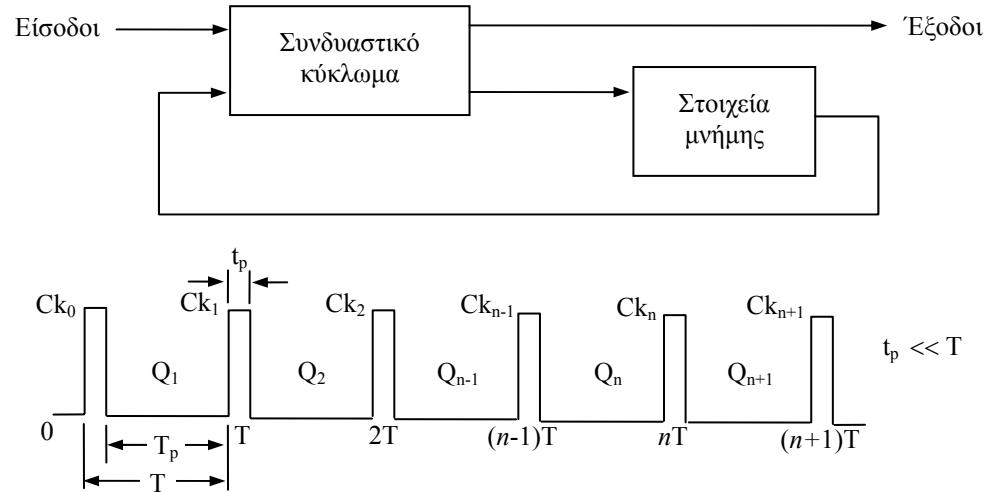


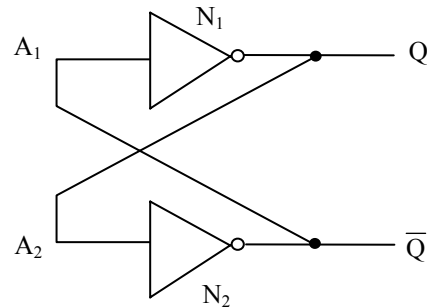
## Ακολουθιακά Κυκλώματα



Στα ακολουθιακά κυκλώματα η αλλαγή καταστάσεως είναι δυνατή μόνο μετά την εφαρμογή σε αυτά του παλμού ρολογιού. Το χρονικό διάστημα μεταξύ δύο παλμών  $T - t_p = T_p$  διατίθεται επομένως για να εκφρασθεί στην έξοδο της πύλης η κατάσταση  $Q_n$  μετά τον  $(n - 1)$  παλμό. Το σύστημα αυτό στο οποίο οι διαδοχικές καταστάσεις  $Q_1, \dots, Q_n$  εμφανίζονται υποχρεωτικά κατά σειρά και μέσα στα χρονικά διαστήματα  $T_1, \dots, T_n$  αντίστοιχα, και επίσης η κατάσταση  $Q_{n+1}$  εξαρτάται από την κατάσταση  $Q_n$  η οποία προηγήθηκε, ονομάζονται ακολουθιακά λογικά κυκλώματα.

## Flip-Flops

Το βασικό ψηφιακό κύκλωμα μνήμης flip-flop επιτυγχάνεται με την ένωση δύο πυλών NOT όπως φαίνεται στο σχήμα.

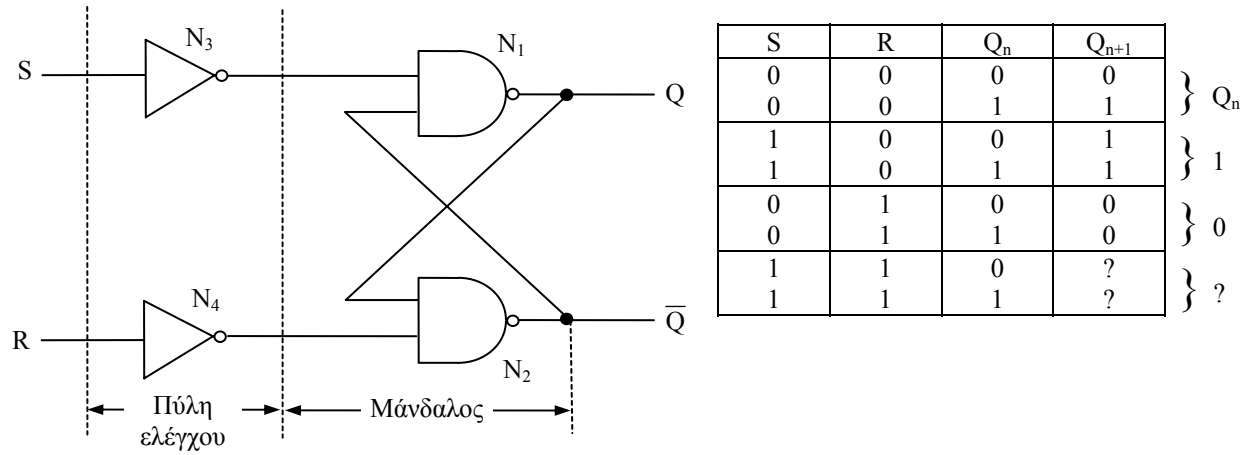


Στο κύκλωμα αυτό η έξοδος της μιας πύλης  $N_1$  συνδέεται στην είσοδο της άλλης και αντιστρόφως.

Η κυριότερη ιδιότητα ενός κυκλώματος αυτού του τύπου είναι, όταν η μία έξοδος  $Q$  βρίσκεται στην κατάσταση “0” η άλλη βρίσκεται υποχρεωτικά στην κατάσταση “1” και αντιστρόφως. Τούτο διότι όταν είναι  $Q = 1$  είναι και  $A_2 = 1$  άρα η έξοδος της  $N_2$  θα είναι 0.

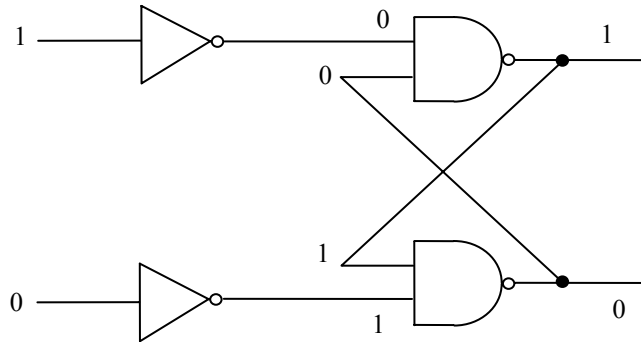
Ομοίως, εάν είναι  $A_1 = 0$  τότε η έξοδος της  $N_1$  δεν είναι δυνατόν να έχει άλλη τιμή από  $Q = 1$  όπως ήταν και η αρχική μας υπόθεση. Παρόμοια απόδειξη γίνεται και εάν θεωρηθεί αρχικά το  $Q = 0$ .

# Flip-Flop S-R



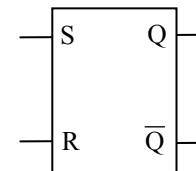
S	R	$Q_{n+1}$
0	0	$Q_n$
1	0	1
0	1	0
1	1	?

(α)

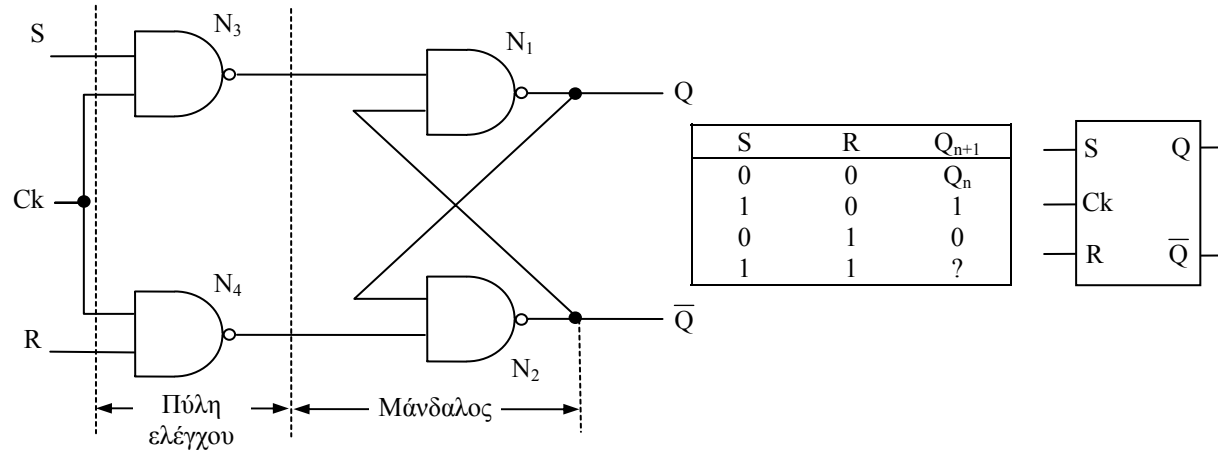


$$Q_{n+1} = S + \bar{R}Q_n$$

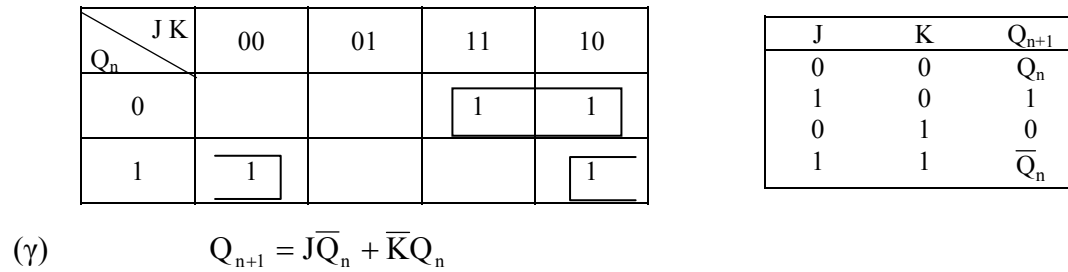
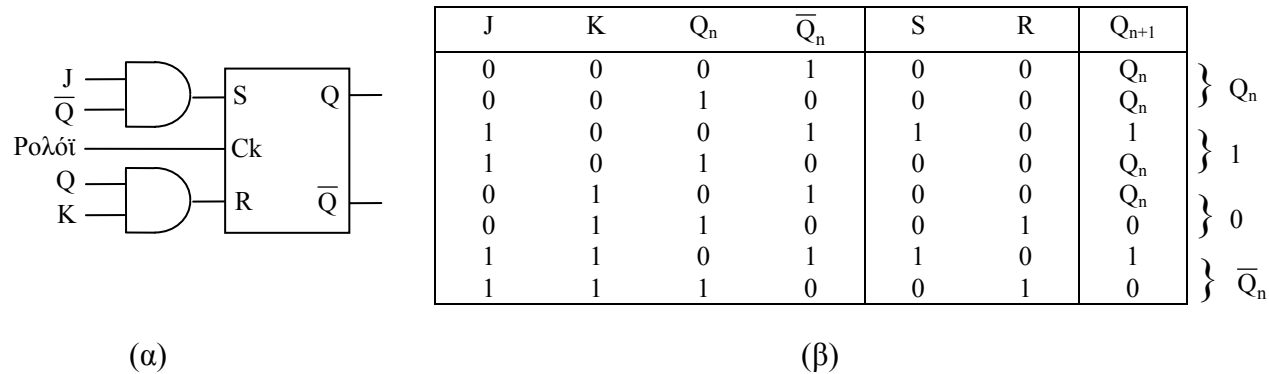
$Q_n \backslash SR$	00	01	11	10
0			X	1
1	1		X	1



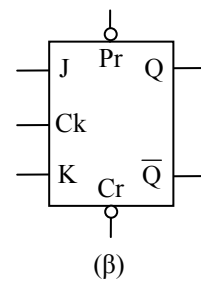
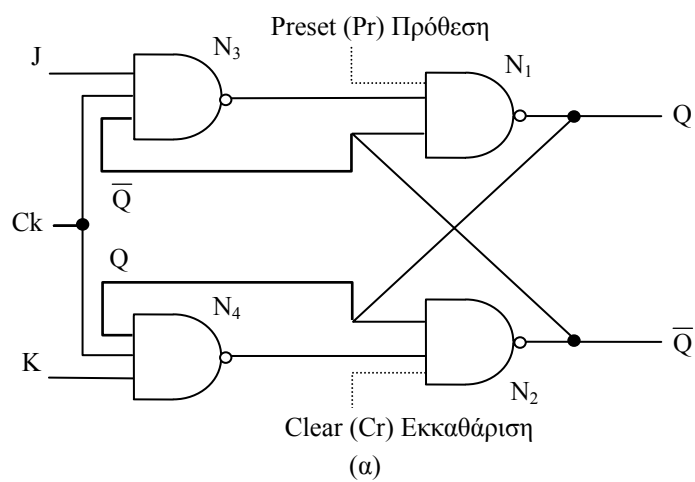
(β)



## Flip-Flop J-K



Στο κύκλωμα του σχήματος έχουν προστεθεί από μία είσοδος σε κάθε πύλη  $N_1$  και  $N_2$ . Οι είσοδοι αυτές φαίνονται στο κύκλωμα με διακεκομμένη γραμμή και με αυτές γίνεται ο προσδιορισμός της αρχικής κατάστασης του flip-flop J-K.

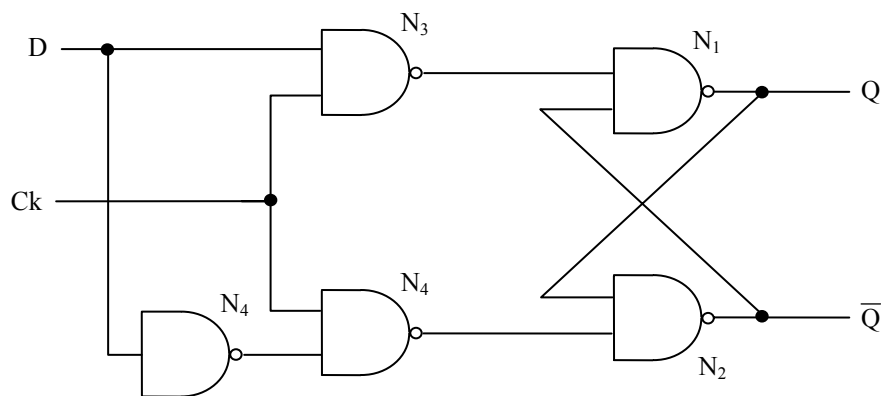


Ck	Cr	Pr	Q
1	1	1	*
0	0	1	0
0	1	0	1

Ελεύθερο  
Εκκαθάριση  
Πρόθεση

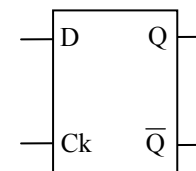
(γ)

## Flip-Flop D



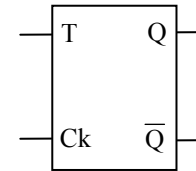
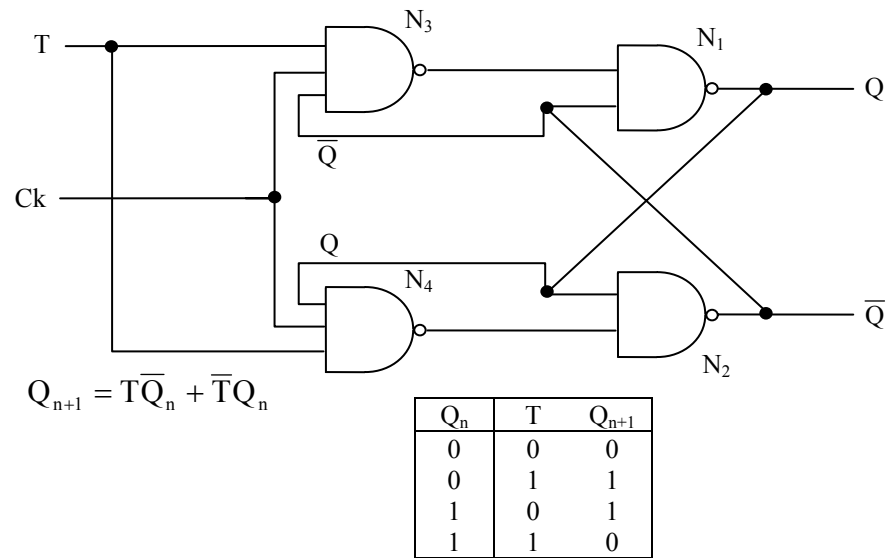
$$Q_{n+1} = D$$

$Q_n$	D	$Q_{n+1}$
0	0	0
0	1	1
1	0	0
1	1	1



	D	0	1
$Q_n$			
0			1
1			1

## Flip-Flop T



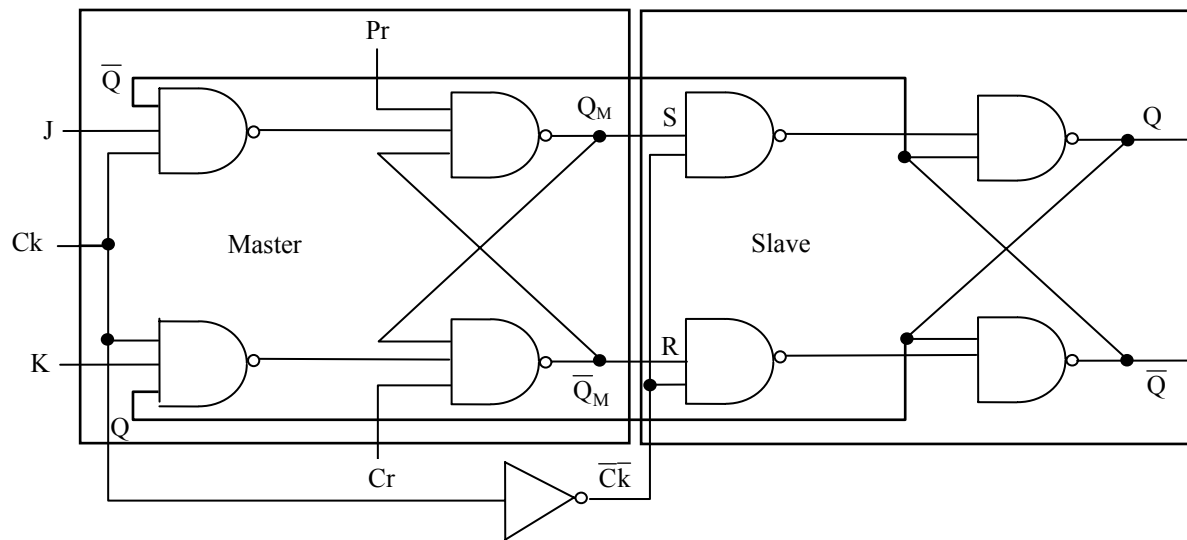
$Q_n \backslash T$	0	1
0		1
1	1	

## Πυροδότηση των Flip-Flops

Το flip-flop J-K, που αναλύσαμε παραπάνω, παρουσιάζει την εξής αδυναμία. Όταν  $Ck = 1$ , και  $J = K = 1$ , τότε αλλάζει κατάσταση. Λόγω της ανατροφοδότησης που υπάρχει, το flip-flop θα συνεχίζει να αλλάζει καταστάσεις για όσο χρονικό διάστημα ο παλμός του ρολογιού βρίσκεται στο 1. Πόσο συχνά θα αλλάζει καταστάσεις; Όση είναι η καθυστέρηση διάδοσης του σήματος μέσα από το flip-flop. Στην προκειμένη περίπτωση αυτή είναι ίση με την καθυστέρηση 3 πυλών. Αυτό σημαίνει πως για τα συνήθη ολοκληρωμένα κυκλώματα στα οποία κάθε πύλη εισάγει καθυστέρηση 10 nsec, η συνολική καθυστέρηση του flip-flop θα είναι 30 nsec. Άρα, μια πρώτη αντιμετώπιση του προβλήματος αυτού θα ήταν οι παλμοί του ρολογιού να έχουν εύρος στο λογικό "1", μικρότερο από την καθυστέρηση διάδοσης του σήματος μέσα από το flip-flop, δηλαδή μικρότερο των 30 nsec.

Αυτή η απαίτηση είναι εξαιρετικά δεσμευτική και δεν χρησιμοποιείται στην πράξη. Το πρόβλημα αυτό μπορεί να αντιμετωπιστεί με δύο τρόπους: (α) με flip-flop J-K τύπου Master-Slave και (β) με flip-flops τα οποία ενεργοποιούνται κατά την άνοδο ή καθοδο του παλμού ρολογιού (ακμοπυροδοτούμενα, edge-triggered) και όχι κατά τη διάρκεια του παλμού. Αυτές τις δύο τεχνικές θα εξετάσουμε αμέσως παρακάτω.

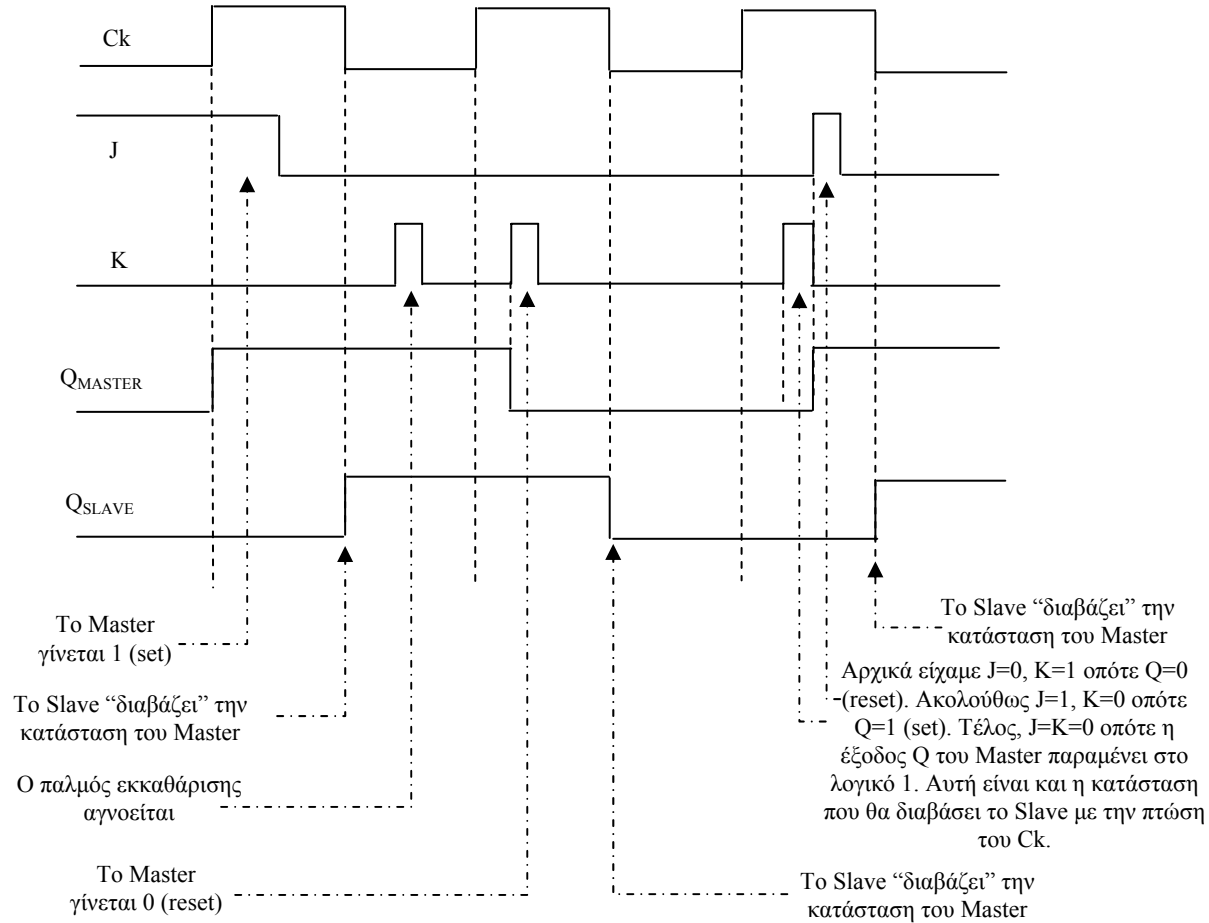
### Master - Slave Flip-Flop



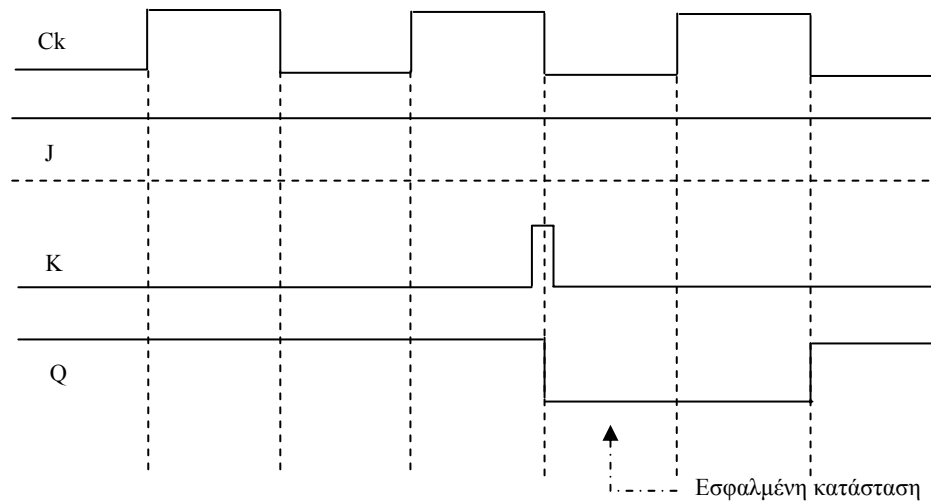


## Παράδειγμα

Σχεδιάστε την κυματομορφή εξόδου Q ενός Flip-Flop J-K Master-Slave του οποίου οι κυματομορφές εισόδου φαίνονται παρακάτω. Θεωρήστε ότι αρχικά έχουμε  $Q = 0$ .



Τα Master-Slave flip-flop ονομάζονται **παλμο-πυροδοτούμενα** ή **επιπεδο-πυροδοτούμενα** (pulse-triggered ή level-triggered) διότι ενεργοποιούνται στη διάρκεια του παλμού ρολογιού (Π) και όχι κατά τη στιγμή που ο παλμός αλλάζει κατάσταση από HIGH ("1") σε LOW ("0") (↓) ή από LOW ("0") σε HIGH (↑) ("1"). Έτσι τα δεδομένα εισόδου διαβάζονται σε όλη τη διάρκεια που ο παλμός ρολογιού είναι HIGH. Το γεγονός αυτό εξακολουθεί να μας δημιουργεί κάποιο πρόβλημα. Έστω, για παράδειγμα, ότι  $J = 1$  και  $K = 0$ , οπότε η έξοδος του Master-Slave JK γίνεται  $Q = 1$ .

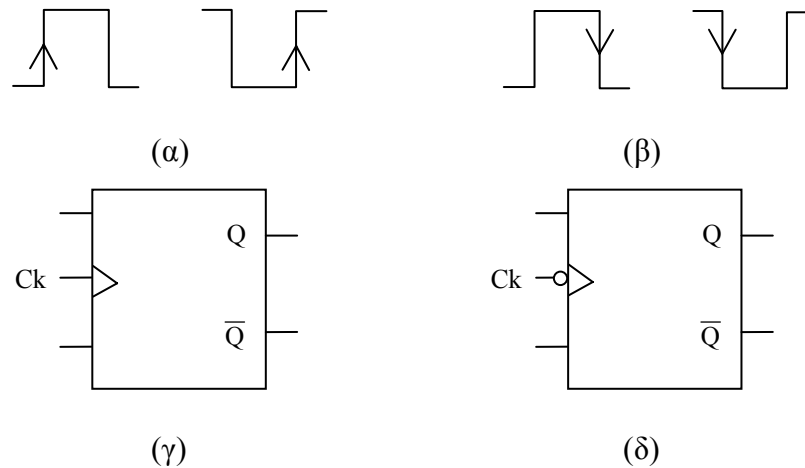


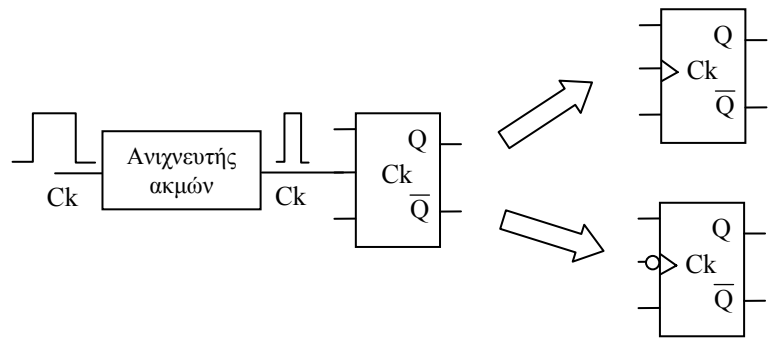
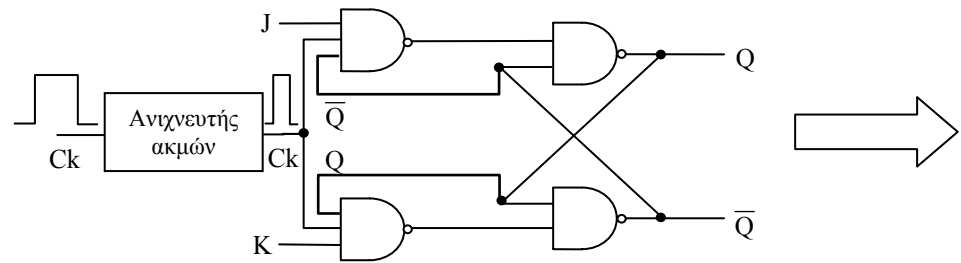
**Ένας ανεπιθύμητος παλμός στην είσοδο K ενός Master-Slave JK flip-flop οδηγεί εσφαλμένα στην αλλαγή κατάστασής του.**

Ένας ανεπιθύμητος θετικός παλμός μικρού εύρους (σπινθήρας, spike) στην είσοδο K κατά τη διάρκεια που ο παλμός ρολογιού είναι HIGH, οδηγεί σε εκκαθάριση του Master ( $Q_M = 0$ ). Αυτή η κατάσταση μεταφέρεται με την πτώση του παλμού ρολογιού στην έξοδο του Slave, οπότε τελικά  $Q = 0$ . Η εμφάνιση ενός θετικού σπινθήρα στην είσοδο J θα οδηγούσε εσφαλμένα στην αλλαγή κατάστασης του flip-flop. Τη λύση σε όλα αυτά τα προβλήματα, μας την δίνει η σχεδίαση flip-flop που να είναι ευαίσθητα κατά τη στιγμή που αλλάζει κατάσταση ο παλμός ρολογιού και όχι όταν αυτός βρίσκεται σε μια συγκεκριμένη λογική στάθμη. Αυτά ονομάζονται ακμοπυροδοτούμενα (edge-triggered) και τα μελετάμε στη συνέχεια.

## Ακμοπυροδοτούμενα Flip-Flop

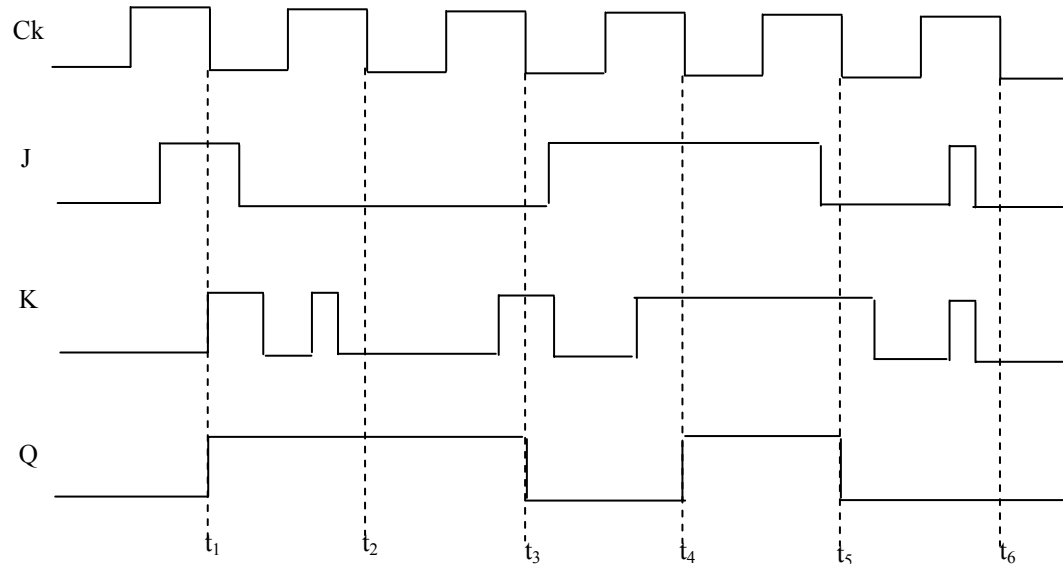
Ακμοπυροδοτούμενα flip-flop είναι αυτά τα οποία είναι έτσι σχεδιασμένα ώστε να είναι ευαίσθητα κατά την αλλαγή της λογικής στάθμης του παλμού ρολογιού. Όταν ο παλμός ρολογιού μεταβαίνει από το “0” στο “1”, τότε λέμε ότι έχουμε μία θετική μετάβαση (Σχ. 6.14), και τα flip-flop που είναι ευαίσθητα κατά τη μετάβαση αυτή ονομάζονται ακμοπυροδοτούμενα (positive edge-triggered flip-flops). Στην αντίθετη περίπτωση, δηλαδή για μετάβαση από “1” σε “0” έχουμε τα αρνητικά ακμοπυροδοτούμενα flip-flop (negative edge-triggered flips-flops). Η είσοδος του παλμού ρολογιού σε ένα flip-flop συμβολίζεται με ένα τρίγωνο. Το τριγωνικό αυτό σύμβολο, Σχ. 6.14γ, δηλώνει ότι το flip-flop πυροδοτείται στη θετική ακμή του παλμού. Η παρουσία ενός μικρού κύκλου έξω από το ορθογώνιο, και αμέσως πριν από το τριγωνικό σύμβολο, δηλώνει ότι το flip-flop πυροδοτείται στην αρνητική ακμή του παλμού, Σχ. 6.14δ.





## Παράδειγμα

Σχεδιάστε την κυματομορφή εξόδου Q ενός αρνητικά ακμοπυροδοτούμενου Flip-Flop J-K του οποίου οι κυματομορφές εισόδου φαίνονται παρακάτω. Θεωρήστε ότι αρχικά έχουμε  $Q = 0$ .



Το flip-flop είναι αρνητικά ακμοπυροδοτούμενο, άρα μας ενδιαφέρει το τι βλέπει στις εισόδους του εκείνη τη χρονική στιγμή. Οι αρνητικές μεταβάσεις των παλμών ρολογιού σημειώνονται ως  $t_1, t_2, \dots, t_6$ . Έτσι έχουμε:

$t_1$  :  $J = 1, K = 0$ , άρα  $Q = 1$ .

$t_2$  :  $J = 0, K = 0$  και συνεπώς αυτό παραμένει στην προηγούμενη κατάσταση του  $Q = 1$ . Παρατηρείστε ότι οι μεταβάσεις της εισόδου K πριν την αρνητική ακμή αγνοήθηκαν.

$t_3$  :  $J = 0, K = 1$  άρα  $Q = 0$ .

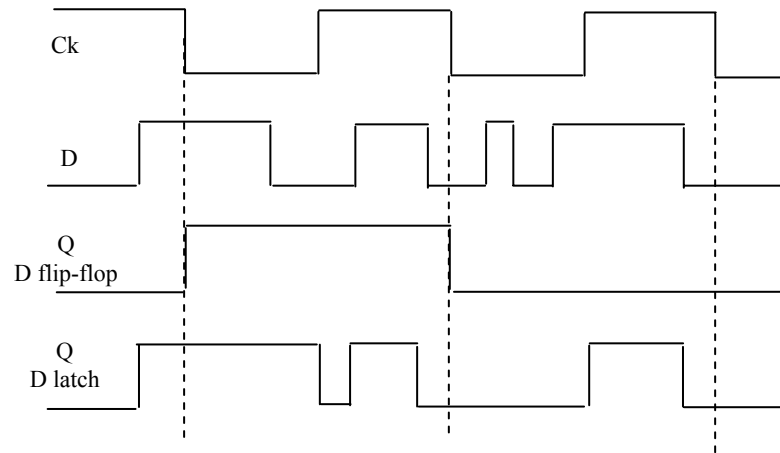
$t_4$  :  $J = 1, K = 1$  άρα  $Q = 1$ .

$t_5$  :  $J = 0, K = 1$  άρα  $Q = 0$ .

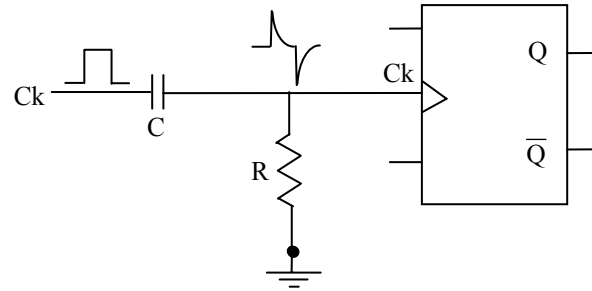
$t_6$  :  $J = 0, K = 0$  άρα  $Q = 0$ , όπως και στην αμέσως προηγούμενη κατάσταση.

## Παράδειγμα

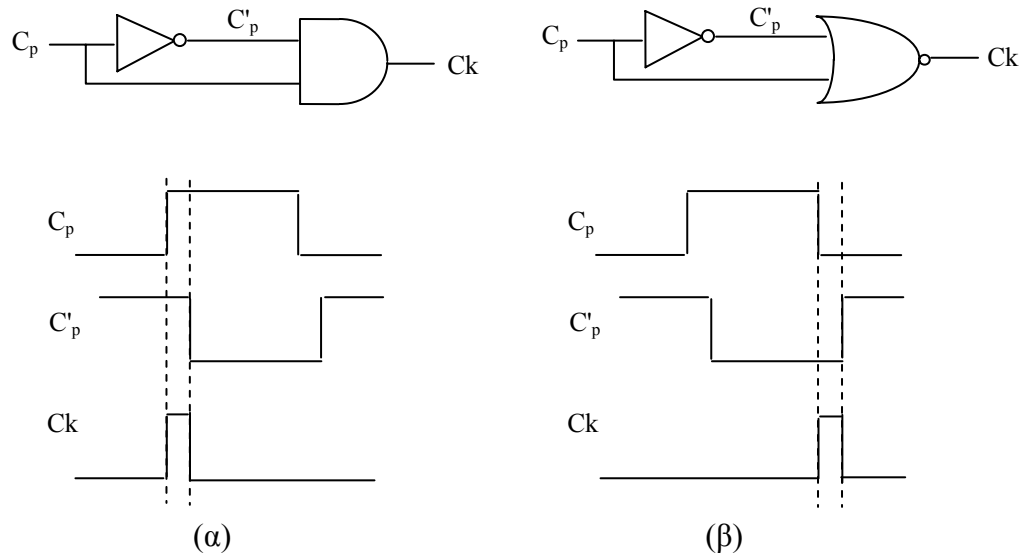
Σχεδιάστε την κυματομορφή εξόδου Q ενός αρνητικά ακμοπυροδοτούμενου flip-flop D του οποίου οι κυματομορφές εισόδου φαίνονται παρακάτω. Συγκρίνετε την κυματομορφή που προκύπτει με την αντίστοιχη του μανταλωτή D (D-latch). Θεωρήστε ότι αρχικά έχουμε  $Q = 0$ .



## Ακμοπυροδοτούμενα Flip-Flop



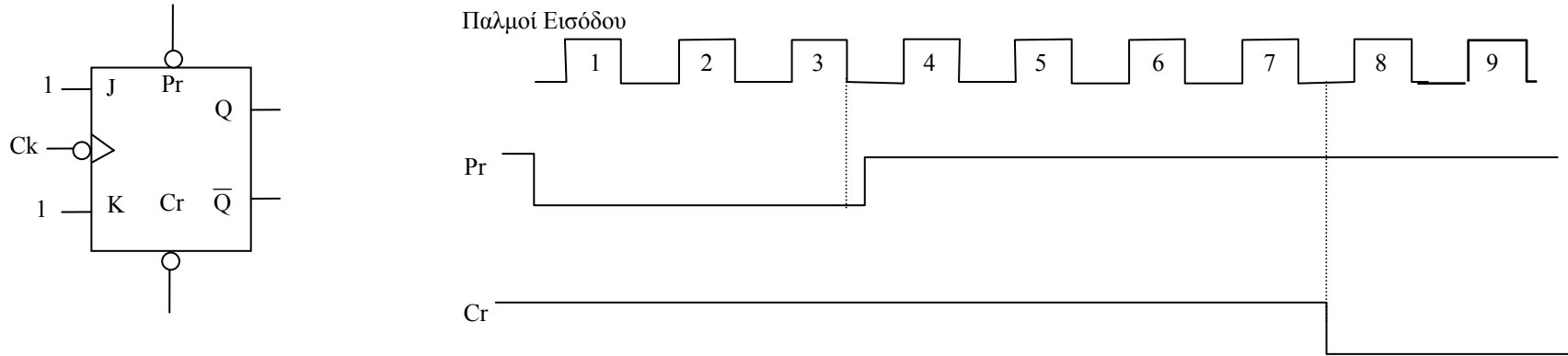
Ανίχνευση ακμών μέσω ενός απλού RC κυκλώματος διαφόρισης.



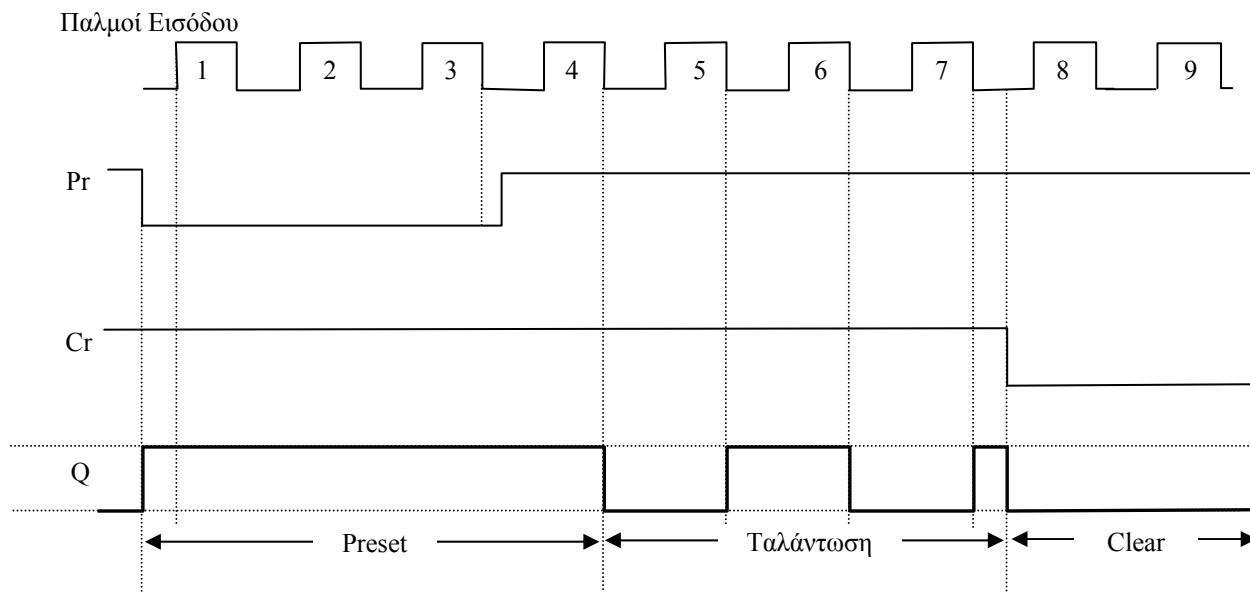
Κυκλώματα ανίχνευσης (α) των θετικών και (β) των αρνητικών ακμών για την πυροδότηση των flip-flops.

## Παράδειγμα

Σχεδιάστε την κυματομορφή εξόδου Q του Flip-Flop J-K του οποίου οι κυματομορφές Clock, Preset και Clear φαίνονται παρακάτω.



## Λύση



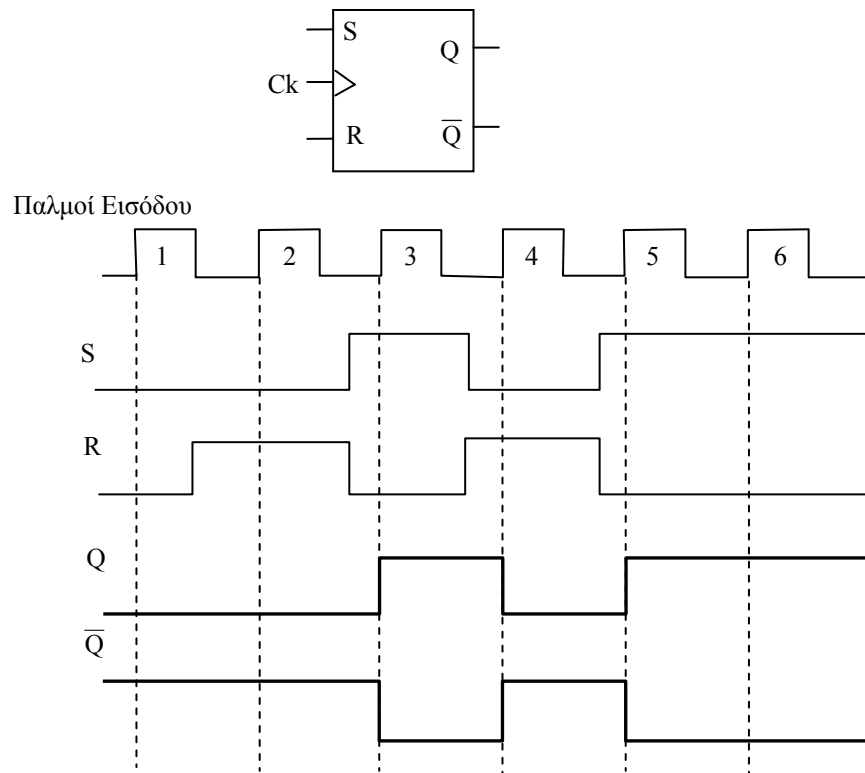


## Παράδειγμα

Σχεδιάστε τις κυματομορφές εξόδου Q και  $\bar{Q}$  του Flip-Flop S-R του οποίου οι κυματομορφές Clock, S και R φαίνονται παρακάτω. Θεωρήστε ότι οι αλλαγές κατάστασης του Flip-Flop συμβαίνουν στη θετική ακμή κάθε παλμού

## Λύση

Με βάση τον πίνακα αληθείας του Flip-Flop S-R θα έχουμε:



### Παράδειγμα 6.6

Να σχεδιαστεί κύκλωμα που υλοποιεί ένα Flip-Flop T με τη βοήθεια Flip-Flop S-R.

#### Λύση

Κατ' αρχήν θα καταστρώσουμε τον πίνακα συσχέτισης των εισόδων ενός Flip-Flop T και των αντίστοιχων εισόδων του Flip-Flop S-R, που έχουν το ίδιο αποτέλεσμα στην κατάσταση  $Q_{n+1}$ . Δηλαδή, θα καταγράψουμε όλους τους δυνατούς συνδυασμούς των εισόδων S, R και της προηγούμενης κατάστασης  $Q_n$  που παράγουν συγκεκριμένο αποτέλεσμα για την επόμενη κατάσταση  $Q_{n+1}$ , μαζί με τους αντίστοιχους συνδυασμούς της εισόδου T που παράγει το ίδιο αποτέλεσμα  $Q_{n+1}$ .

Από τους πίνακες αληθείας των Flip-Flop T και S-R, έχουμε:

**Πίνακας Συσχέτισης**

T	$Q_n$	$Q_{n+1}$	S	R
0	0	0	0	X
0	1	1	X	0
1	0	1	1	0
1	1	0	0	1

Από τον πίνακα συσχέτισης, εξάγουμε τις εξισώσεις των S, R συναρτήσεων των T και  $Q_n$  με τη βοήθεια του χάρτη Karnaugh.

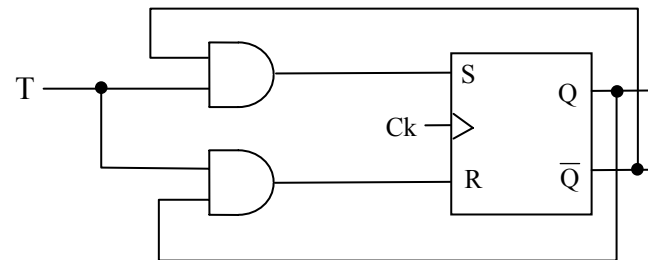
$Q_n \backslash T$	0	1
0	0	1
1	X	0

$$S = T\bar{Q}_n$$

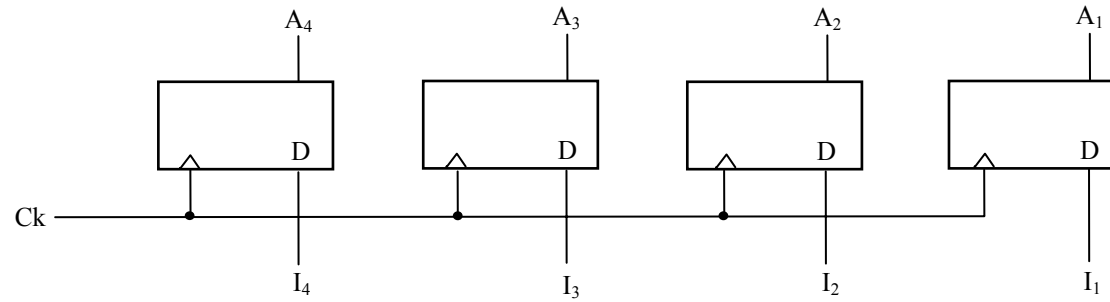
T Q <sub>n</sub>	0	1
0	X	0
1	0	1

$$R = TQ_n$$

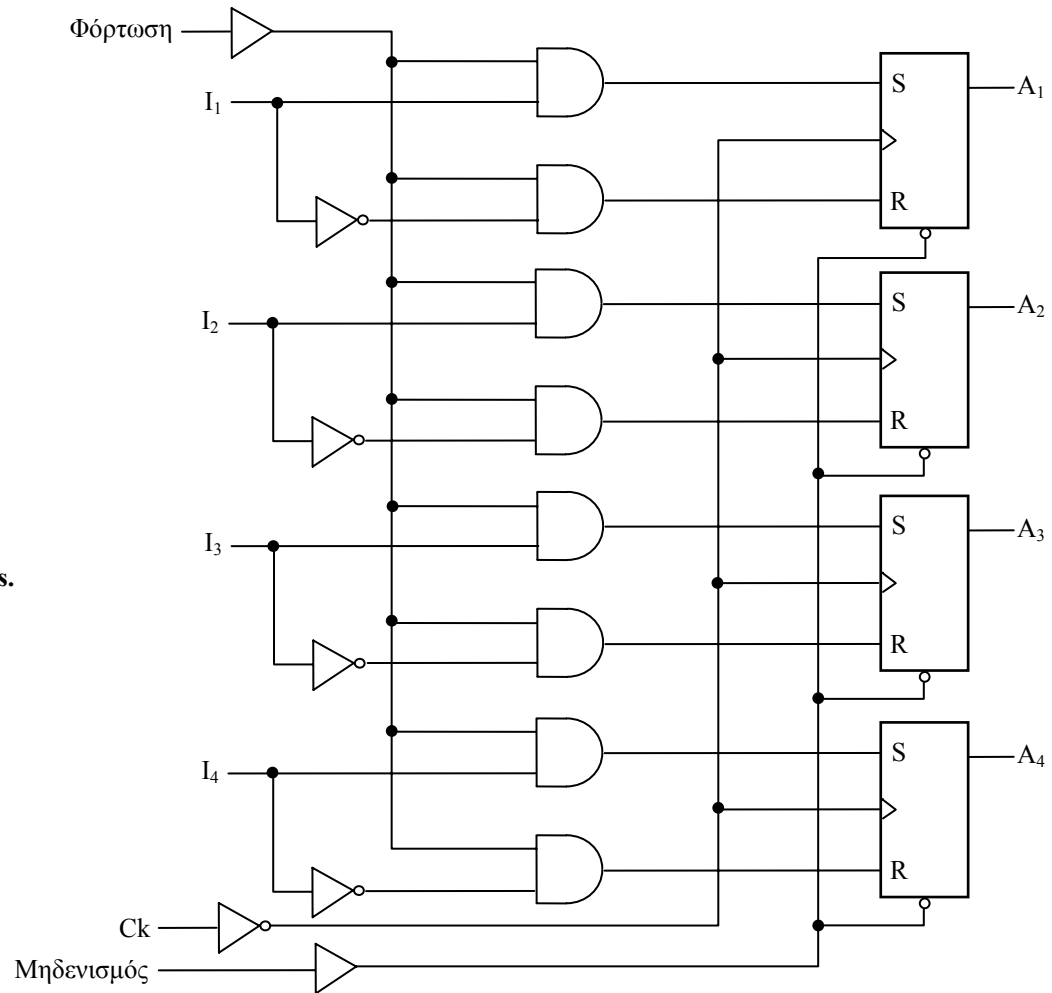
Με βάση τα παραπάνω, σχεδιάζουμε το τελικό κύκλωμα:



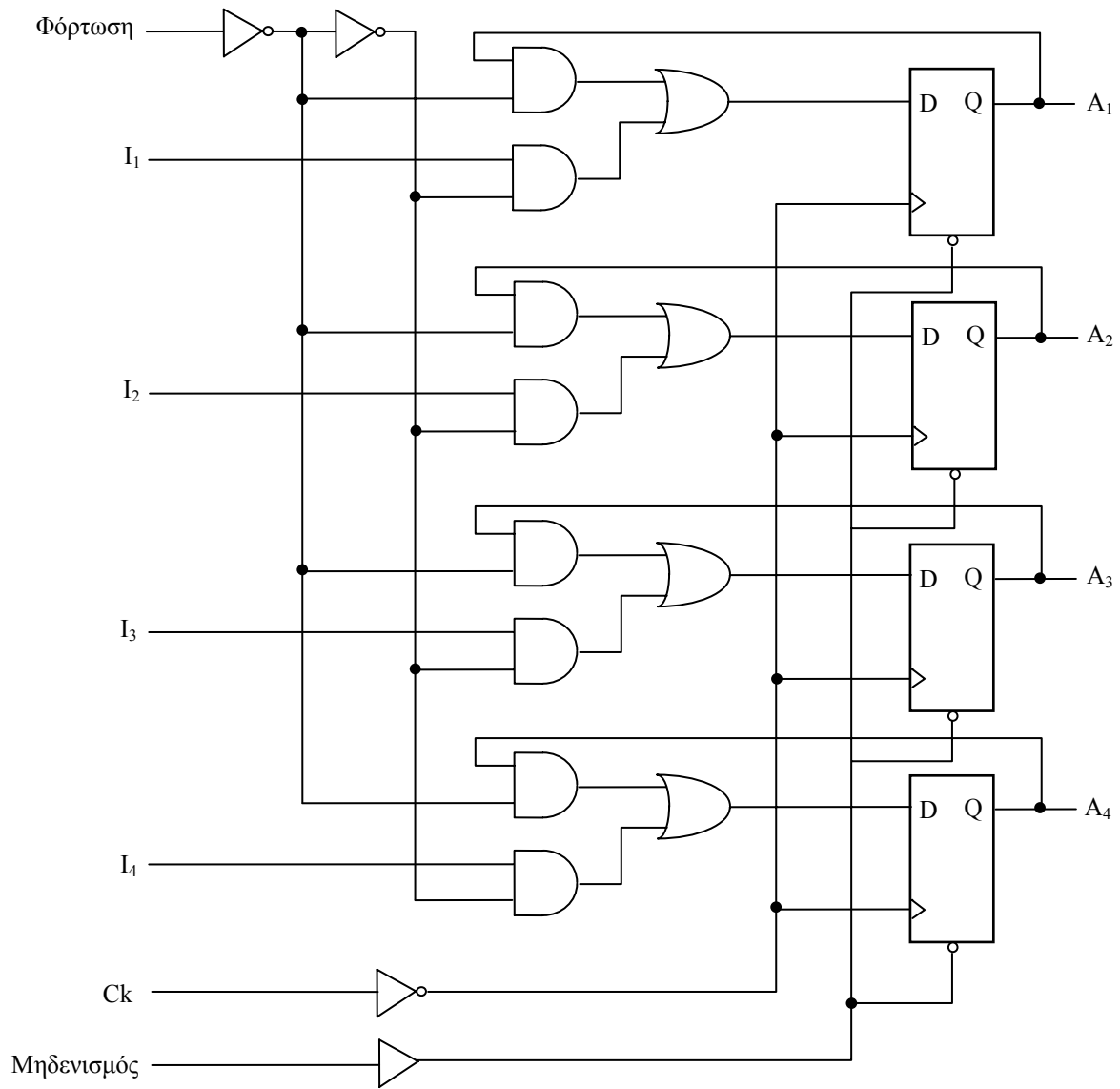
## Καταχωρητές (Registers)



## Καταχωρητές με παράλληλη φόρτωση

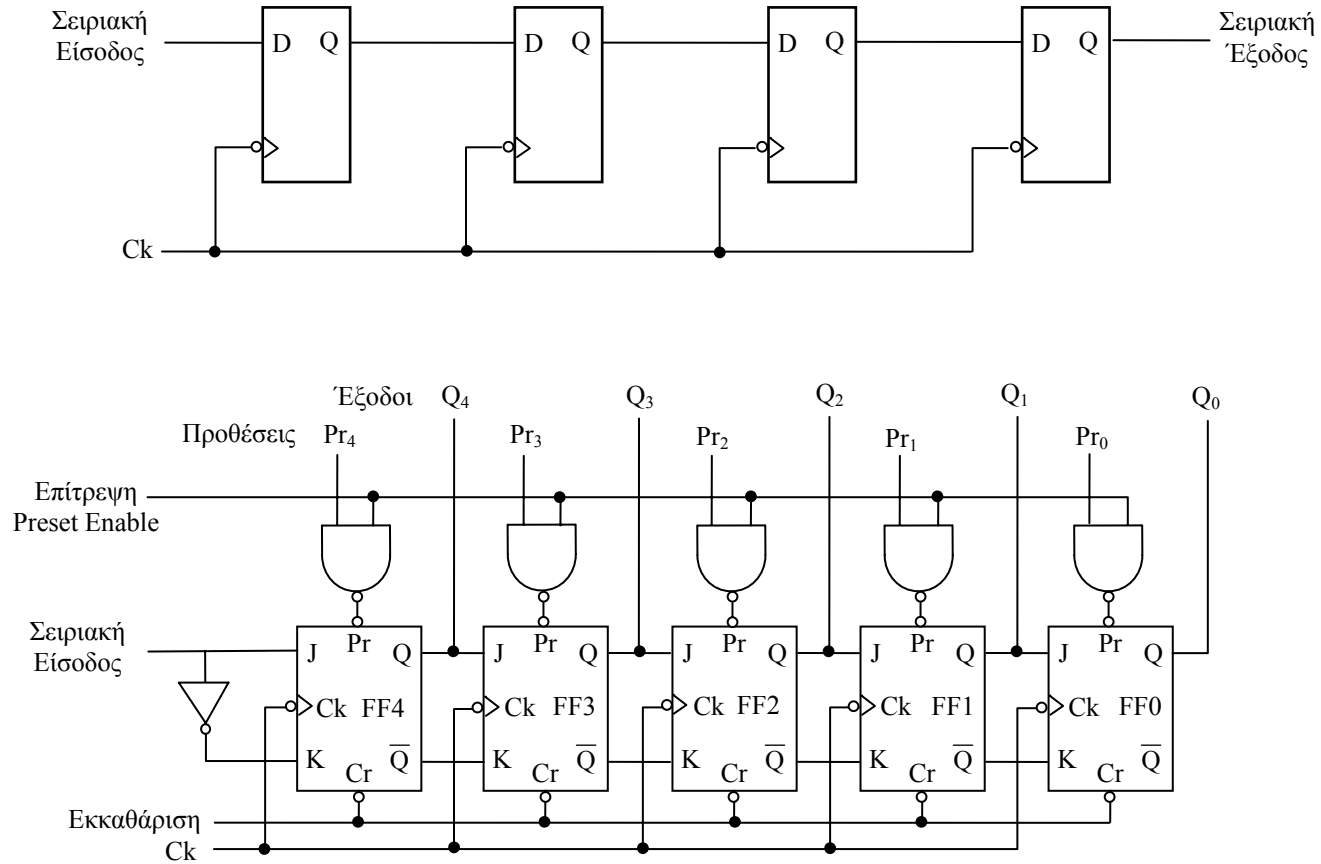


Καταχωρητής παράλληλης φόρτωσης 4 bits.



**Καταχωρητής παράλληλης φόρτωσης με flip-flops τύπου D.**

### Καταχωρητές με σειριακή φόρτωση ή Καταχωρητές ολίσθησης



Καταχωρητής σειριακής και παράλληλης φόρτωσης.

Παλμοί ρολογιού	Ψηφία αριθμού	Q <sub>4</sub>	Q <sub>3</sub>	Q <sub>2</sub>	Q <sub>1</sub>	Q <sub>0</sub>
1	1	1	0	0	0	0
2	0	0	1	0	0	0
3	0	0	0	1	0	0
4	1	1	0	0	1	0
5	1	1	1	0	0	1

Σειριακή καταχώρηση δυαδικού αριθμού.

## Παράδειγμα

Να σχεδιαστεί καταχωρητής τεσσάρων bits ο οποίος όταν το σήμα ελέγχου  $L = 1$  ολισθαίνει μία θέση προς τα αριστερά και αντιστρόφως ολισθαίνει μία θέση προς τα δεξιά όταν  $R = 1$ .

Πώς τροποποιείται το κύκλωμα όταν ο καταχωρητής είναι  $n$  bits και έχουμε  $n$  σήματα ελέγχου  $L_i$  και  $n$  σήματα ελέγχου  $R_i$  για τα οποία ισχύει:  $L_i = 1 \rightarrow$  ολίσθηση κατά  $i$  θέσεις αριστερά,  $R_i = 1 \rightarrow$  ολίσθηση κατά  $i$  θέσεις δεξιά.

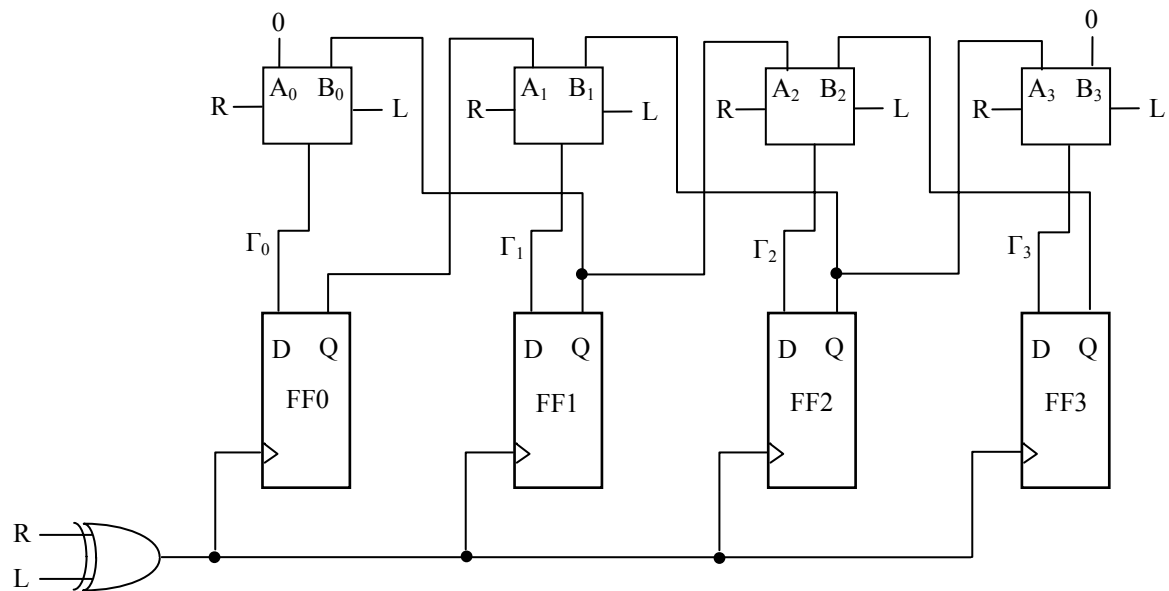
Τροποποιείστε το αρχικό κύκλωμα έτσι ώστε ο καταχωρητής εκτός από τα σήματα ελέγχου  $L$  και  $R$  να ελέγχεται και από τρίτο σήμα  $W$  το οποίο όταν  $W = 1$  επιτρέπει την παράλληλη καταχώρηση από 4 γραμμές δεδομένων. Επίσης, θα πρέπει όταν  $R = L = W = 0$  να μην επηρεάζεται ο καταχωρητής, και όταν εφαρμόζονται ταυτόχρονα περισσότερα από ένα σήματα ελέγχου να μην εφαρμόζεται καμία είσοδος στον καταχωρητή.

## Λύση:

Η κυκλωματική υλοποίηση του καταχωρητή φαίνεται στο σχήμα. Τα συνδυαστικά κυκλώματα στο επάνω μέρος του σχήματος υλοποιούν τις συναρτήσεις

$$\Gamma_i = RA_i + LB_i \quad i = 0, 1, 2, 3$$

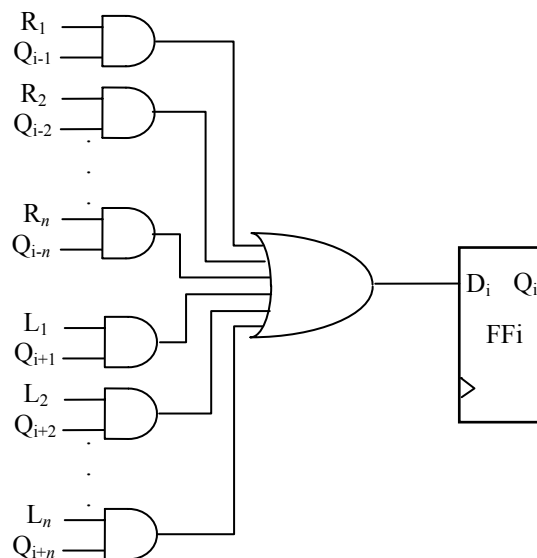
Εύκολα διαπιστώνουμε ότι όταν  $R = 1$ , το περιεχόμενο της εξόδου  $Q$  κάθε flip-flop τύπου D μετατοπίζεται μία θέση προς τα δεξιά. Οι εισοδοι  $R$  και  $L$  συνδέονται στην είσοδο ρολογιού κάθε flip-flop μέσω μιας πύλης XOR ώστε να είναι αμοιβαία αποκλειόμενες μεταξύ τους. Ομοίως, όταν  $L = 1$ , το περιεχόμενο της εξόδου  $Q$  κάθε flip-flop τύπου D μετατοπίζεται μία θέση προς τα αριστερά. Σε κάθε μετατόπιση προς τα αριστερά ή προς τα δεξιά, στη θέση του bit μικρότερης σημαντικότητας τίθεται μέσω του συνδυαστικού κυκλώματος η τιμή 0.





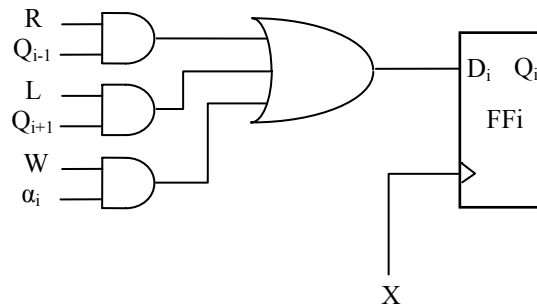
Στην περίπτωση  $n$  σημάτων ελέγχου η κυκλωματική υλοποίηση τροποποιείται για κάθε flip-flop ως ακολούθως:

Το παρακάτω σχήμα αποτελεί ουσιαστικά επέκταση του προηγούμενου σχήματος για καταχωρητή  $n$  bits και για λόγους σχεδιαστικής ευκολίας δίνουμε μόνο την κυκλωματική υλοποίηση ενός flip-flop.



**Κυκλωματική τροποποίηση του καταχωρητή για περισσότερα σήματα ελέγχου.**

Τέλος, δίνεται στο επόμενο σχήμα η τροποποίηση του αρχικού κυκλώματος προκειμένου να μπορούμε να επιβάλλουμε και τρίτο σήμα ελέγχου  $W$ . Για να ικανοποιούνται δε, οι δύο τελευταίες απαιτήσεις της εκφώνησης, επιβάλλεται το σήμα ελέγχου καταχώρησης  $X$ , το οποίο συνδέεται στην είσοδο  $Ck$  κάθε flip-flop, όπως φαίνεται στο επόμενο σχήμα, και προκύπτει από τον ακόλουθο πίνακα αληθείας.



Τροποποίηση του αρχικού κυκλώματος για σήματα ελέγχου W και X.

R	L	W	X
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	0

$$X = \bar{R} \bar{L} W + \bar{R} L \bar{W} + R \bar{L} \bar{W}$$

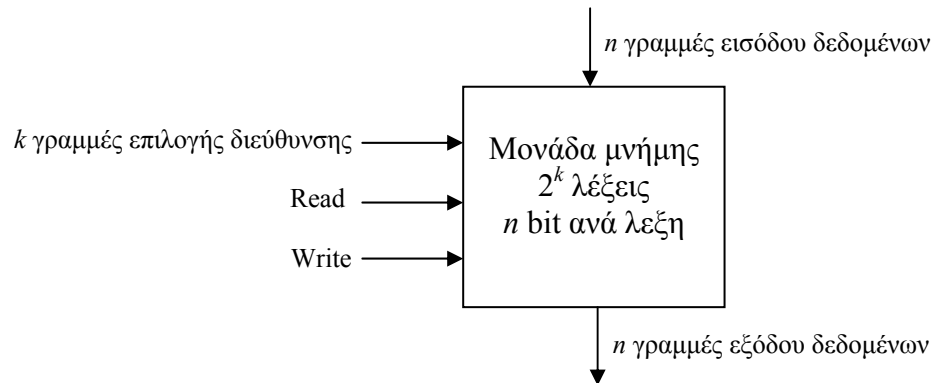
Σύμφωνα με τον πίνακα αυτό, όταν  $R = L = W = 0$  δεν επηρεάζεται ο καταχωρητής διότι  $X = 0$ , και όταν εφαρμόζονται ταυτόχρονα περισσότερα από ένα σήματα ελέγχου, δεν εφαρμόζεται καμία είσοδος στον καταχωρητή διότι και πάλι  $X = 0$ . Για οποιοδήποτε άλλο συνδυασμό το σήμα ελέγχου X έχει τιμή 1, επιτρέποντας την ομαλή λειτουργία των σημάτων ελέγχου R, L, και W.

Τα δεδομένα παράλληλης καταχώρησης μέσω του σήματος ελέγχου W, συμβολίζονται με  $\alpha_i$ . Επίσης, για λόγους σχεδιαστικής ευκολίας παραλείπουμε το συνδυαστικό κύκλωμα υλοποίησης της συνάρτησης X.

### 6.3. Η μνήμη τυχαίας προσπέλασης (RAM)

Μία μονάδα μνήμης είναι ένα σύνολο από κύτταρα αποθήκευσης, μαζί με τα απαραίτητα κυκλώματα για τη μεταφορά των πληροφοριών μέσα και έξω από τη συσκευή. Επειδή μπορεί να γίνει προσπέλαση στα κύτταρα μνήμης, για μεταφορά πληροφορίας από και προς οποιαδήποτε επιθυμητή τυχαία θέση, προέκυψε το όνομα “**μνήμη τυχαίας προσπέλασης**” (**Random Access Memory, RAM**).

Οι λέξεις μνήμης αποτελούνται από bits. Μία ομάδα από 8 bits ονομάζεται byte. Οι περισσότερες μνήμες υπολογιστών χρησιμοποιούν λέξεις που έχουν μήκος πολλαπλάσιο των 8 bits. Η χωρητικότητα μιας μνήμης ορίζεται από το **συνολικό αριθμό των bytes** που μπορεί να αποθηκεύσει. Η επικοινωνία μεταξύ της μονάδας μνήμης και του περιβάλλοντός της επιτυγχάνεται με γραμμές εισόδου και εξόδου δεδομένων, γραμμές επιλογής διεύθυνσης και γραμμές ελέγχου, που καθορίζουν τη διεύθυνση μεταφοράς. Στο Σχ. 6.29 φαίνεται ένα σχηματικό διάγραμμα μνήμης RAM. Οι  $n$  γραμμές εισόδου δεδομένων παρέχουν την πληροφορία που θα αποθηκευθεί στη μνήμη και οι  $n$  γραμμές εξόδου δεδομένων την πληροφορία που εξέρχεται από τη μνήμη. Οι  $k$  γραμμές διεύθυνσης προσδιορίζουν τη συγκεκριμένη λέξη που επιλέχθηκε ανάμεσα στις πολλές διαθέσιμες. Οι 2 είσοδοι ελέγχου καθορίζουν την επιθυμητή κατεύθυνση μεταφοράς: η είσοδος write προκαλεί μεταφορά δυαδικών πληροφοριών προς τη μνήμη, ενώ η είσοδος read μεταφορά δυαδικών πληροφοριών από τη μνήμη.



Σχ. 6.29. Σχηματικό διάγραμμα μνήμης RAM.

Μια μονάδα μνήμης περιγράφεται από τον αριθμό των λέξεων που περιέχει και τον αριθμό των bits ανά λέξη. Οι γραμμές διεύθυνσης επιλέγουν μια συγκεκριμένη λέξη. Κάθε λέξη της μνήμης εφοδιάζεται με έναν αριθμό αναγνώρισης, που καλείται διεύθυνση και ο οποίος ξεκινά από το 0 και συνεχίζει με 1, 2, 3 μέχρι το  $2^k-1$ , όπου  $k$  είναι ο αριθμός των γραμμών διεύθυνσης. Η επιλογή μιας συγκεκριμένης λέξης στη μνήμη γίνεται με την εφαρμογή της δυαδικής διεύθυνσης των  $k$  bits στις γραμμές διεύθυνσης. Ένας αποκωδικοποιητής στο εσωτερικό της μνήμης δέχεται τη διεύθυνση και ανοίγει τους διαύλους που χρειάζονται για την επιλογή της συγκεκριμένης λέξης. Οι μνήμες των υπολογιστών κυμαίνονται από 1024 λέξεις – που απαιτούν διεύθυνση των 10 bits – μέχρι  $2^{32}$  λέξεις – που απαιτούν 32 bits διεύθυνσης. Συνηθίζεται να αναφερόμαστε στον αριθμό των λέξεων (ή bytes) στη μνήμη με ένα από τα γράμματα **K (kilo)**, **M (mega)**, ή **G (giga)**.

Το **K** είναι ισοδύναμο με  $2^{10}$ , το **M** με  $2^{20}$  και το **G** με  $2^{30}$ . Έτσι, έχουμε  $64K = 2^{16}$ ,  $2M = 2^{21}$  και  $4G = 2^{32}$ .

Διεύθυνση μνήμης		Περιεχόμενα μνήμης
Δυαδικό	Δεκαδικό	
0000000000	0	1011010101011100
0000000001	1	1010101110001001
0000000010	2	0000110101000110
.	.	.
.	.	.
.	.	.
.	.	.
.	.	.
1111111101	1021	1001110100010101
1111111110	1022	0000110100011110
1111111111	1023	1101111000100101

Σχ. 6.30. Περιεχόμενα μιας μνήμης RAM 1024×16.

Έστω για παράδειγμα, μια RAM με 1K λέξεων των 16 bits η καθεμία. Αφού  $1K = 1024 = 2^{10}$  και 16 bits συνιστούν 2 bytes, μπορούμε να πούμε ότι η μνήμη χωράει  $2048 = 2K$  bytes. Το Σχ. 6.30 δείχνει το πιθανό περιεχόμενο των 3 πρώτων και των 3 τελευταίων λέξεων της μνήμης. Κάθε λέξη περιέχει 16 bits, που μπορούν να χωριστούν σε 2 bytes. Οι λέξεις αναγνωρίζονται από τη δεκαδική τους διεύθυνση, που κυμαίνεται από 0 έως 1023. Η ισοδύναμη δυαδική διεύθυνση αποτελείται από 10 bits. Η πρώτη διεύθυνση δηλώνεται με δέκα 0, ενώ η τελευταία διεύθυνση με 10 1. Αυτό συμβαίνει, γιατί στο δυαδικό σύστημα το 1023 είναι ισοδύναμο με 1111111111. Μία λέξη της μνήμης επιλέγεται με βάση τη δυαδική της διεύθυνση. Όταν μια λέξη διαβάζεται ή γράφεται, η μνήμη συμπεριφέρεται στα 16 bits σαν να αποτελούν μια ενιαία μονάδα.

Η μνήμη των  $1K \times 16$  του Σχ. 6.30 έχει 10 bits στη διεύθυνση και 16 bits σε κάθε λέξη. Αντίστοιχα, μια μνήμη των  $64K \times 10$  θα έχει 16 bits στη διεύθυνση (αφού  $64K = 2^{16}$ ) και κάθε λέξη θα αποτελείται από 10 bits. Ο αριθμός των bits της διεύθυνσης, που απαιτούνται σε μια μνήμη, εξαρτάται από το συνολικό αριθμό λέξεων που μπορούν να αποθηκευτούν στη μνήμη και είναι ανεξάρτητος από τον αριθμό των bits της κάθε λέξης. Ο αριθμός των bits στη διεύθυνση καθορίζεται από τη σχέση  $2^k = m$ , όπου  $m$  είναι ο συνολικός αριθμός των λέξεων και  $k$  ο αριθμός των bits της διεύθυνσης.

Διεύθυνση μνήμης		Περιεχόμενα μνήμης
Δυαδικό	Δεκαδικό	
0000000000	0	1011010101011100
0000000001	1	1010101110001001
0000000010	2	0000110101000110
.	.	.
.	.	.
.	.	.
.	.	.
.	.	.
1111111101	1021	1001110100010101
1111111110	1022	0000110100011110
1111111111	1023	1101111000100101

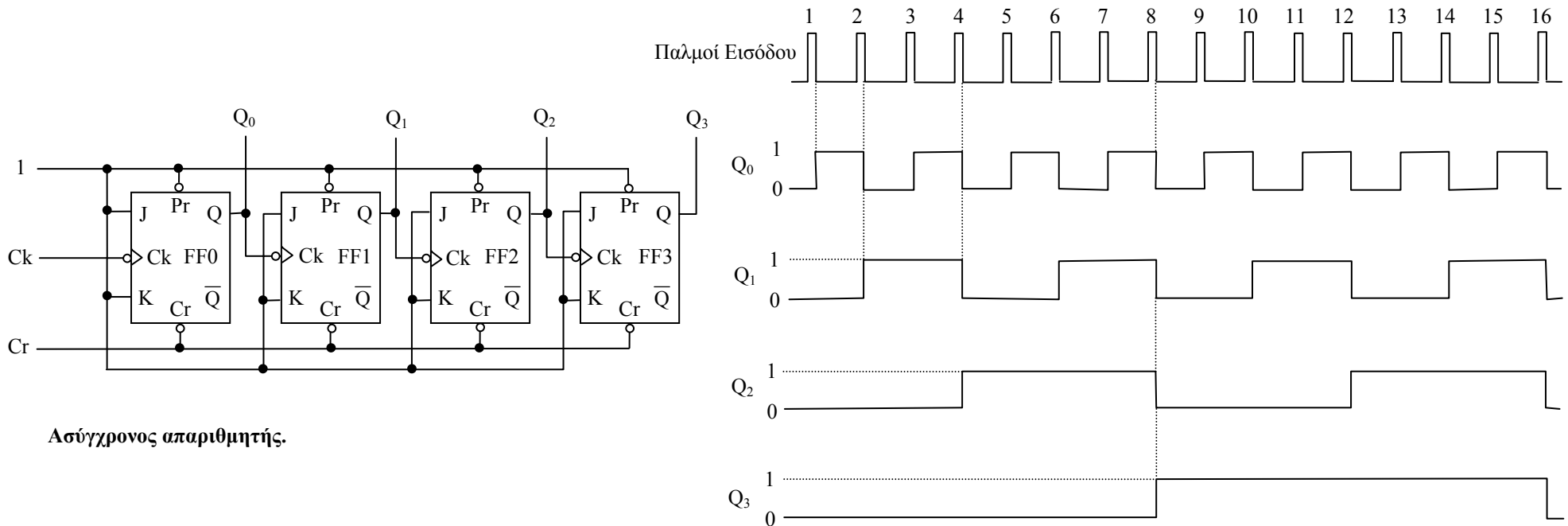
Σχ. 6.30. Περιεχόμενα μιας μνήμης RAM  $1024 \times 16$ .

## Απαριθμητές (Counters)

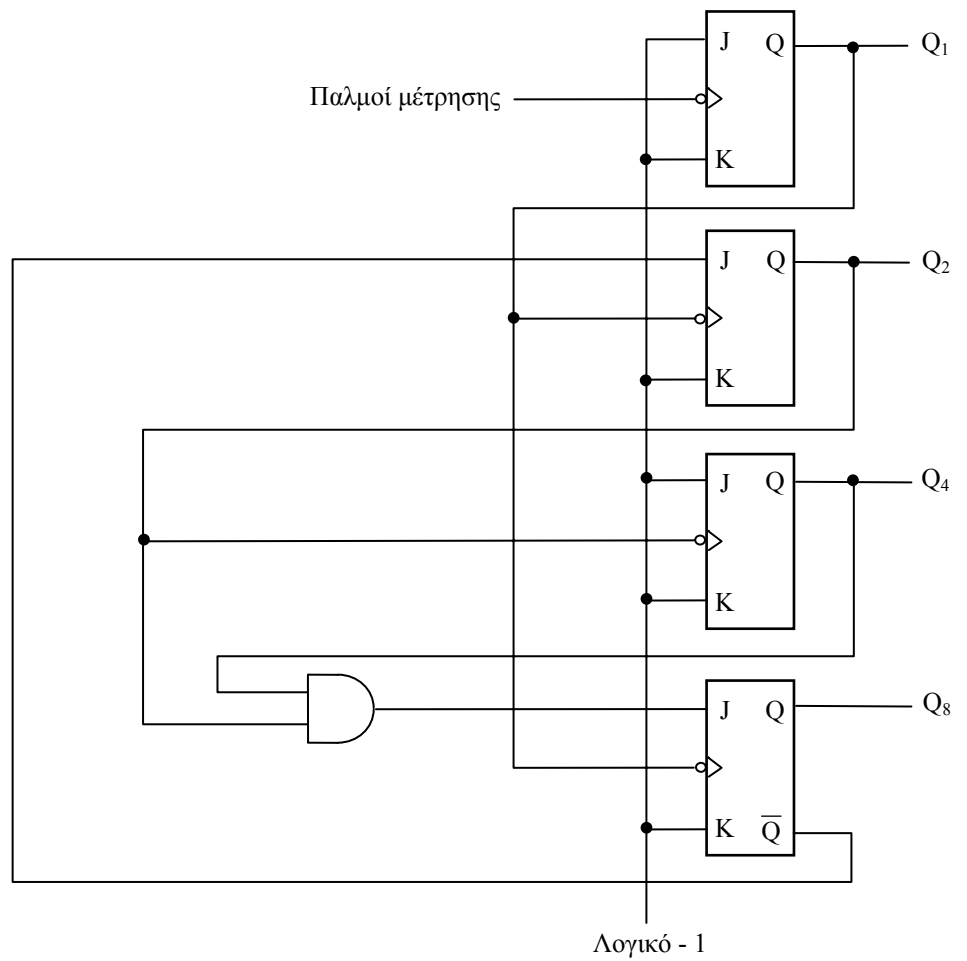
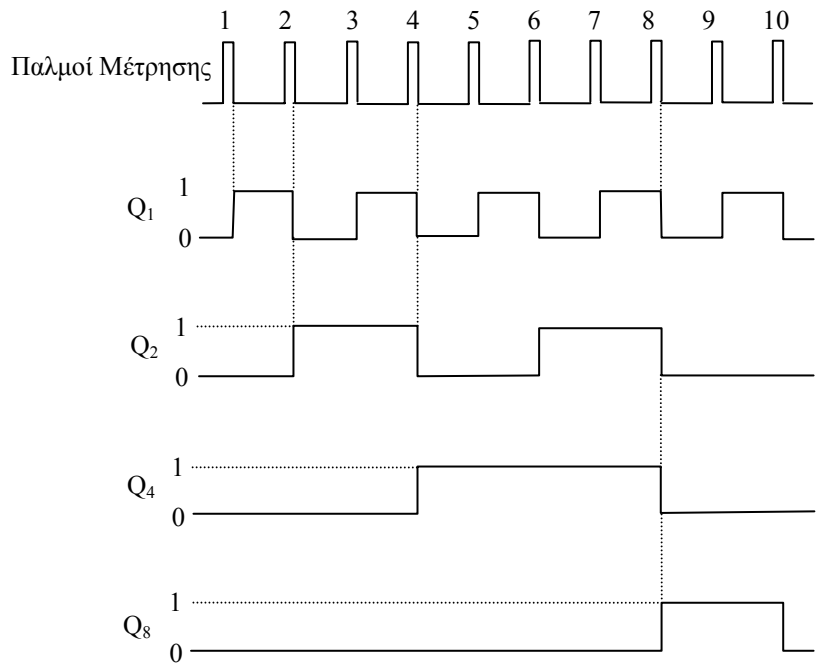
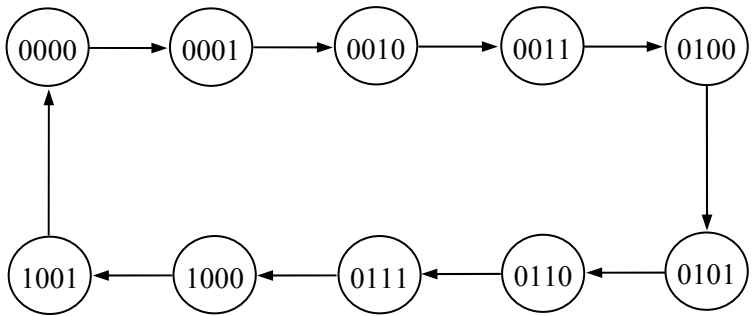
Ένα ακολουθιακό κύκλωμα που περνάει από μια προδιεγεγραμμένη ακολουθία καταστάσεων, όταν του εφαρμόζουμε παλμούς στην είσοδο, ονομάζεται “απαριθμητής” (counter). Οι παλμοί εισόδου, που τους ονομάζουμε “παλμούς μέτρησης” (count pulses), μπορεί να είναι παλμοί ρολογιού ή να προέρχονται από κάποια εξωτερική πηγή και μπορεί να έρχονται σε κανονικά ή σε ακανόνιστα διαστήματα. Σε έναν απαριθμητή η ακολουθία των καταστάσεων μπορεί να είναι η δυαδική σειρά μέτρησης ή μια οποιαδήποτε άλλη σειρά. Απαριθμητές βρίσκουμε σχεδόν μέσα σε όλα τα ψηφιακά συστήματα. Τους χρησιμοποιούμε για να μετράμε πόσες φορές συμβαίνει κάποιο γεγονός ή για τη δημιουργία ακολουθιών χρονισμού για τον έλεγχο των λειτουργιών ενός ψηφιακού συστήματος.

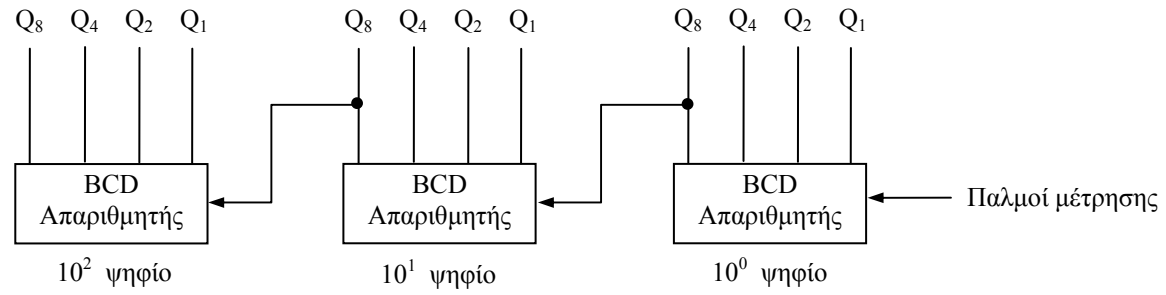
Οι απαριθμητές απαρτίζονται από flip-flops κατάλληλα συνδεδεμένα μεταξύ τους, και ανήκουν σε δύο κατηγορίες: τους *ασύγχρονους* απαριθμητές και τους *σύγχρονους* απαριθμητές.

### Ασύγχρονοι απαριθμητές



### Ασύγχρονος δεκαδικός απαριθμητής





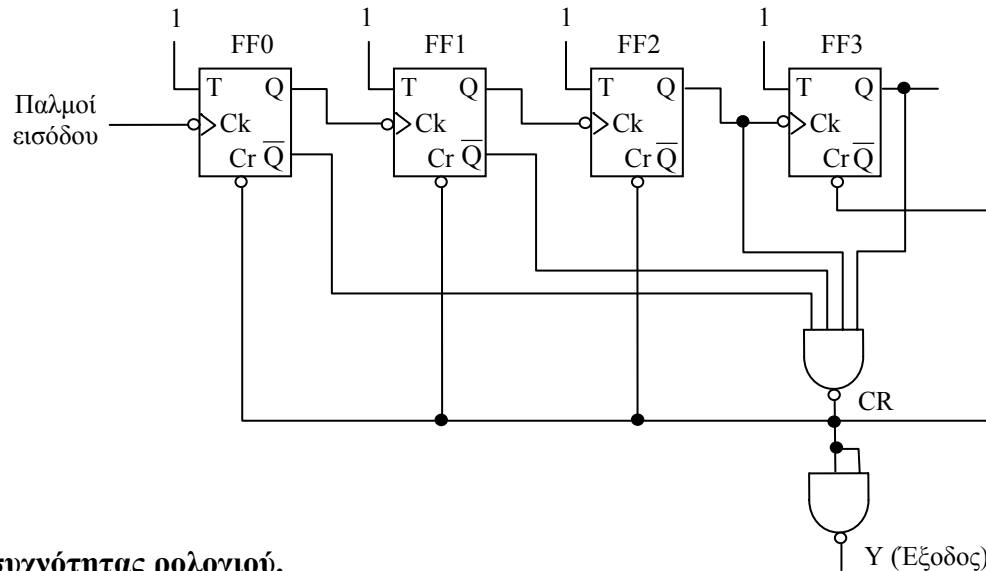
Σχηματικό διάγραμμα δεκαδικού απαριθμητή BCD 3 δεκάδων.



**Παράδειγμα:** Να σχεδιαστεί κύκλωμα που να διαιρεί τη συχνότητα του σήματος ρολογιού δια 12.

**Λύση:**

Το κύκλωμα που πραγματοποιεί τη ζητούμενη διαίρεση δίνεται στο επόμενο σχήμα.



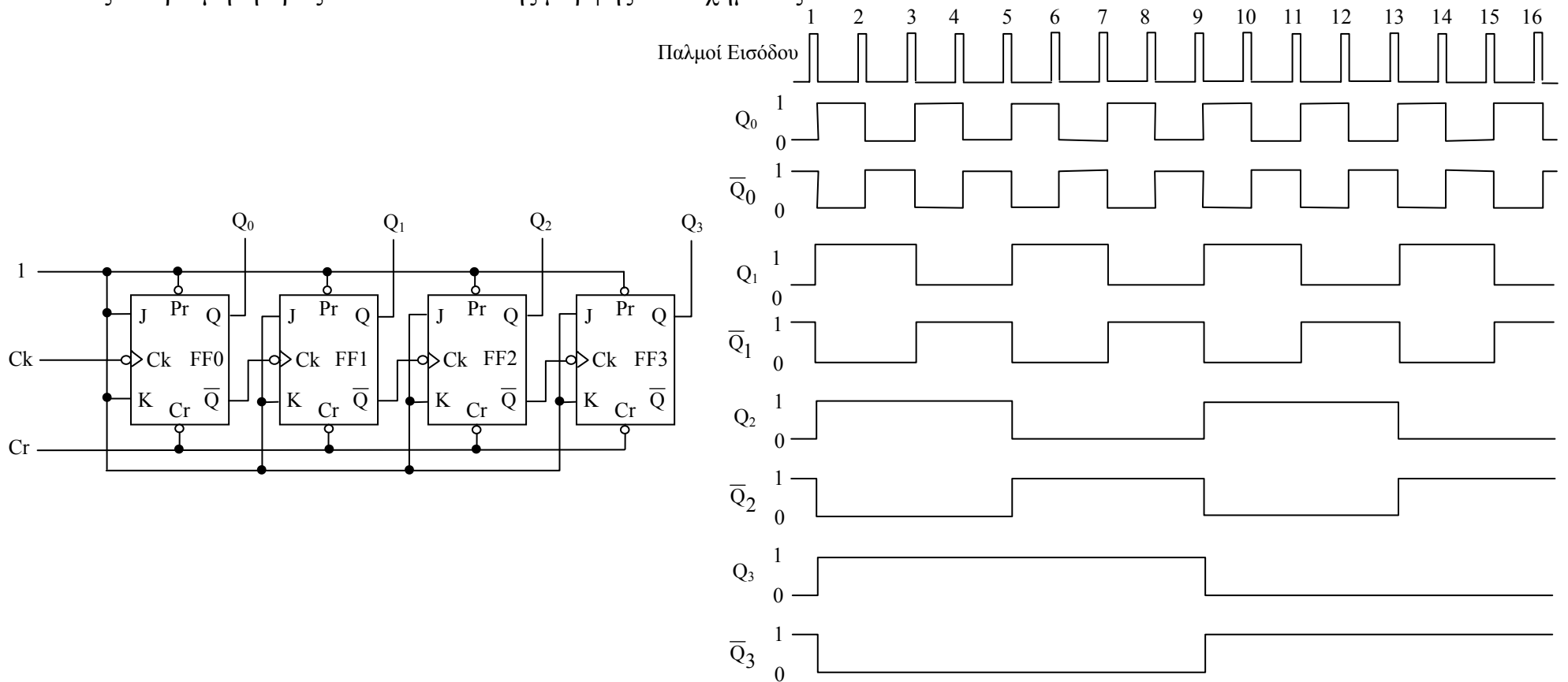
**Κύκλωμα διαίρεσης συχνότητας ρολογιού.**

Το κύκλωμα λειτουργεί ως εξής: Κατ' αρχήν χρησιμοποιούμε ένα ασύγχρονο απαριθμητή 4 bits ( $2^4 = 16 > 12$ ) ο οποίος μόλις μετρήσει τον 12ο παλμό θα έχει στις εξόδους του την ακολουθία 1100. Δηλαδή, οι εξοδοί των flip-flops μετά το 12ο παλμό θα είναι  $Q_3Q_2Q_1Q_0 = 1100$ . Έτσι, με τη συνδεσμολογία του Σχ. 6.37 μόνο μετά το 12ο παλμό η έξοδος CR της πρώτης πύλης NAND θα είναι 0, διότι  $CR = Q_3 Q_2 \overline{Q_1} \overline{Q_0} = 0$ , οπότε και θα γίνει εκκαθάριση όλων των flip-flops και θα αρχίσει ξανά η διαδικασία μέτρησης. Η έξοδος του κυκλώματος είναι η έξοδος Y της δεύτερης πύλης NAND, η οποία θα γίνει 1 μόλις η έξοδος CR γίνει 0, και στην οποία επομένως θα εμφανιστεί παλμός κάθε φορά που ο απαριθμητής θα έχει πρώτα μετρήσει 12 παλμούς ρολογιού. Κατ' αυτόν τον τρόπο επιτυγχάνουμε διαίρεση της συχνότητας ρολογιού διά 12.

**Παράδειγμα:** Να δοθεί το διάγραμμα χρονισμού ενός ασύγχρονου απαριθμητή 4 bits προς τα κάτω.

**Λύση**

Ένας απαριθμητής προς τα κάτω σχεδιάζεται όπως και ο απαριθμητής προς τα πάνω, μόνο που στην είσοδο ρολογιού κάθε flip-flop συνδέεται το συμπλήρωμα της εξόδου του flip-flop μικρότερης σημαντικότητας. Έτσι, το κυκλωματικό διάγραμμα ενός απαριθμητή προς τα κάτω είναι της μορφής του σχήματος.



Το διάγραμμα χρονισμού του ασύγχρονου απαριθμητή προς τα κάτω προκύπτει εύκολα από τους πίνακες αληθείας των flip-flops τύπου T και από όσα είπαμε για τους ασύγχρονους απαριθμητές.

**Παράδειγμα:** Να σχεδιαστεί δυαδικός απαριθμητής προς τα άνω με τις εξής προδιαγραφές:

- 1) Όταν δέχεται σήμα εισόδου  $R = 1$  να μηδενίζεται.
- 2) Όταν δέχεται σήμα εισόδου  $F = 1$  να “παγώνει” (δηλ. να σταματάει η λειτουργία του).
- 3) Όταν δέχεται σήματα εισόδου  $R = F = 1$  να μην συμβαίνει τίποτα (δηλ. να αποκλείεται η ταυτόχρονη επενέργεια των δύο σημάτων ελέγχου  $F$  και  $R$ ).
- 4) Όταν το δεκαδικό ισοδύναμο του δυαδικού απαριθμητή είναι  $17 < N \leq 49$  να υπάρχει σήμα εξόδου  $S = 1$ .
- 5) Όταν  $N = 49$  να μηδενίζεται ο απαριθμητής.

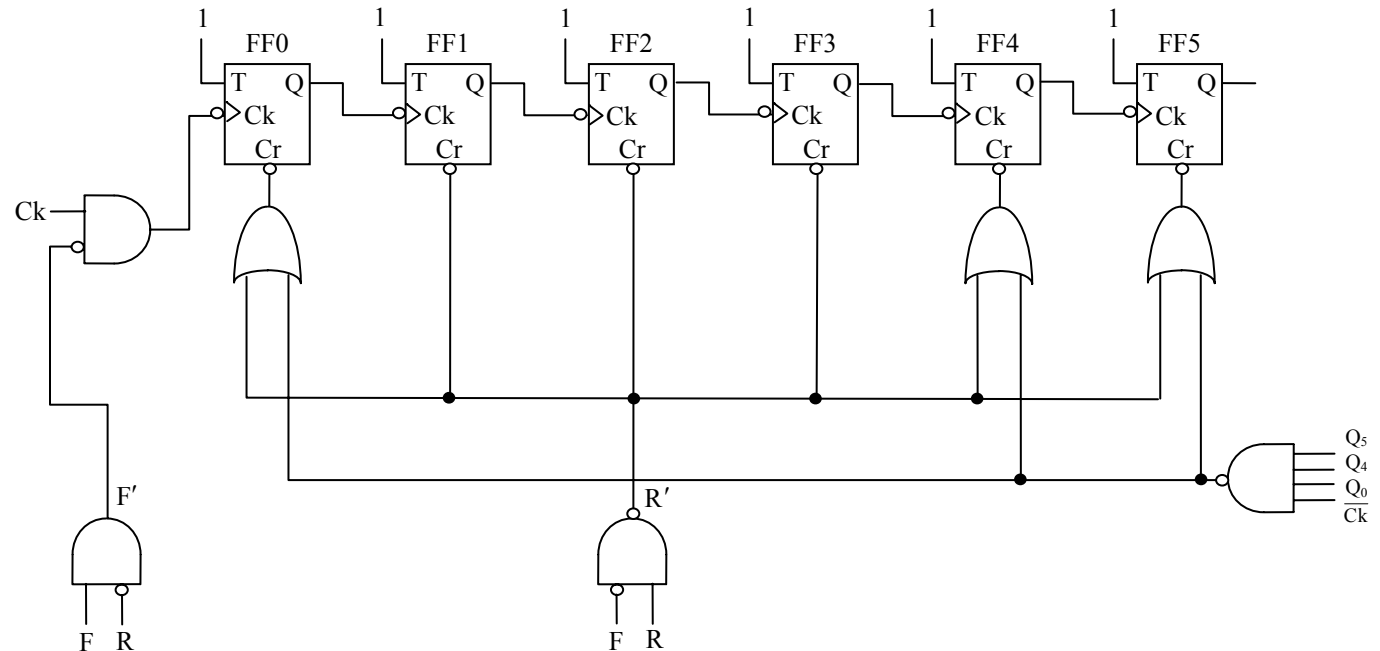
### Λύση

Από τα δεδομένα του παραδείγματος ο απαριθμητής, τον οποίο σχεδιάζουμε με flip-flops τύπου T, θα πρέπει να μπορεί να μετράει 49 παλμούς, να μηδενίζεται μόλις μετρηθεί και ο 49ος παλμός, και να αρχίζει ξανά τη διαδικασία μέτρησης. Άρα, ο ελάχιστος αριθμός flip-flops που πρέπει να διαθέτει είναι  $n = 6$  (διότι  $2^5 = 32 < 49$ ,  $2^6 = 64 > 49$ ). Εφόσον επίσης τα σήματα ελέγχου  $F$  και  $R$  πρέπει να είναι αμοιβαία αποκλειόμενα θα πρέπει να ισχύει:

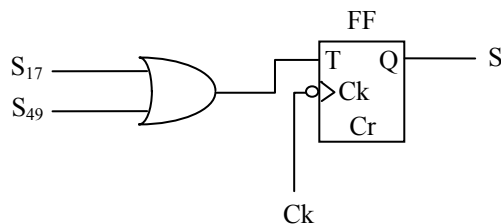
$$F' = (F \oplus R)F = (\overline{FR} + \overline{FR})F = \overline{FR}$$

$$R' = \overline{(F \oplus R)R} = \overline{(\overline{FR} + \overline{FR})R} = \overline{\overline{FR}} = F + \overline{R}$$

που υλοποιείται με τις πύλες OR που φαίνονται στο σχήμα. Έτσι, όταν  $F = 1$  και  $R = 0$ , έχουμε  $F' = 1$  και αποτρέπονται οι παλμοί εισόδου  $Ck$  λόγω της πύλης AND στο αριστερό άκρο του σήματος και επομένως ο απαριθμητής “παγώνει”.



Όταν  $R = 1$  και  $F = 0$ , έχουμε  $R' = 0$  και ενεργοποιούνται οι είσοδοι Clear (Cr) των flip-flops με αποτέλεσμα να μηδενίζεται ο απαριθμητής. Όταν  $F = R = 1$ , έχουμε  $\bar{F} = \bar{R} = 0$ , και επομένως δεν συμβαίνει τίποτα διότι ακυρώνεται η ταυτόχρονη επενέργεια των δύο σημάτων ελέγχου και ο απαριθμητής συνεχίζει να απαριθμεί κανονικά με τους παλμούς εισόδου. Η λύση του ερωτήματος (4) υλοποιείται πολύ εύκολα με το ακόλουθο σχήμα:



**Υλοποίηση του κυκλώματος του ερωτήματος (4).**

Για το σχεδιασμό του πιο πάνω κυκλώματος λαμβάνουμε υπόψη ότι η δυαδική απεικόνιση του 17 είναι  $(17)_{10} = (010001)_2$  και του 49 είναι  $(49)_{10} = (110001)_2$ . Έτσι, εάν οι είσοδοι σε ένα flip-flop τύπου T μέσω της πύλης OR είναι οι  $S_{17}$  και  $S_{49}$  που ορίζονται ως εξής:

$$S_{17} = \bar{Q}_5 Q_4 \bar{Q}_3 \bar{Q}_2 \bar{Q}_1 Q_0 \quad S_{49} = Q_5 Q_4 \bar{Q}_3 \bar{Q}_2 \bar{Q}_1 Q_0$$

τότε, μόλις ο απαριθμητής μετρήσει τον 17ο παλμό, και σύμφωνα με τον πίνακα αληθείας ενός flip-flop τύπου T, η είσοδος  $S_{17}$  θα γίνει 1 οπότε η έξοδος του flip-flop θα συμπληρωθεί και από  $S = 0$  (θεωρούμε εδώ ότι στο flip-flop έχουμε κάνει αρχικά εκκαθάριση Clear) θα γίνει  $S = 1$ , θα παραμείνει δε ίση με 1 διότι μέχρι τον 49ο παλμό και οι δύο είσοδοι  $S_{17}$  και  $S_{49}$  είναι 0 (οπότε το flip-flop παραμένει στην προηγούμενη κατάσταση 1). Μόλις μετρηθεί ο 49ος παλμός η είσοδος  $S_{49}$  γίνεται 1, οπότε η έξοδος του flip-flop συμπληρώνεται και από 1 γίνεται 0, όπως ακριβώς ήταν και το ζητούμενο της άσκησης. Σημειώνουμε ότι για λόγους ευκρίνειας του Σχ. 6.32 δεν περιλάβαμε το Σχ. 6.33 σε αυτό.

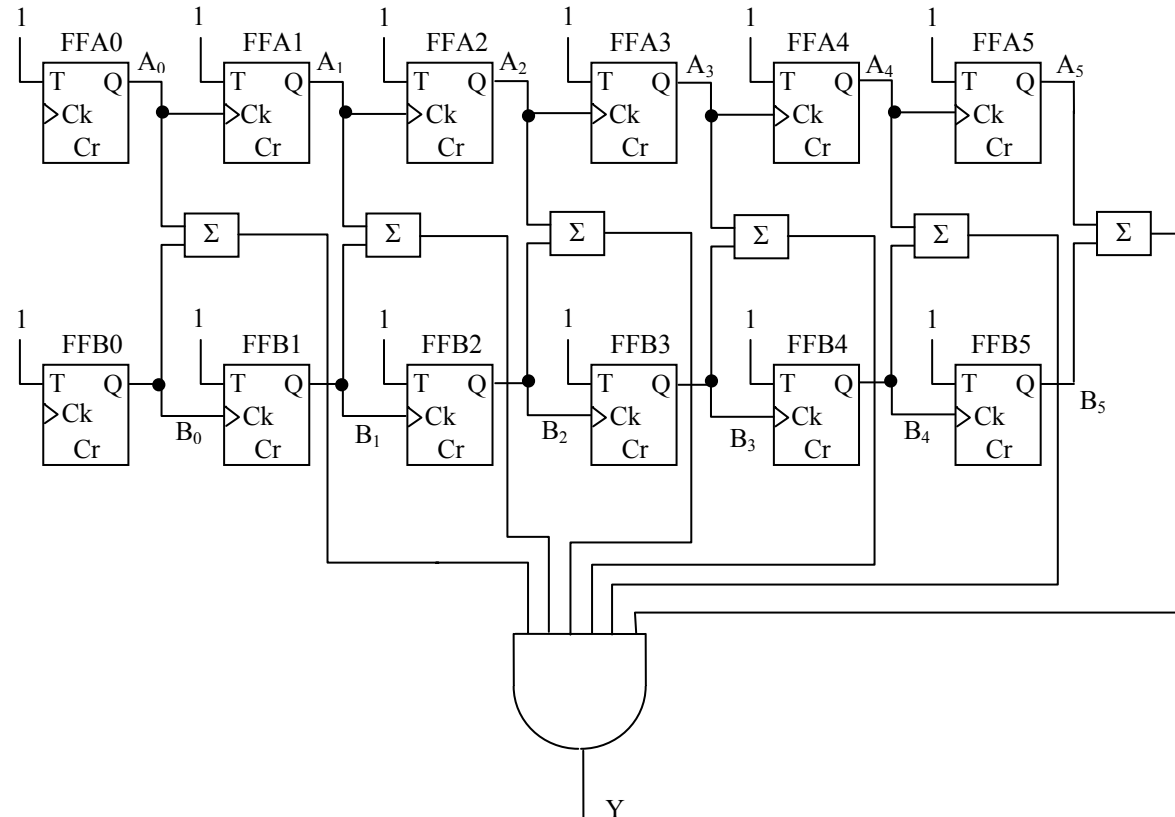
Τέλος, δίνουμε την υλοποίηση του κυκλώματος του ερωτήματος (5) που αποτελείται ουσιαστικά από την πύλη NAND στο κάτω δεξιό μέρος του σχήματος με εισόδους  $Q_5 Q_4 Q_0 \bar{Ck}$  (θυμηθείτε ότι τα flip-flop ενεργοποιούνται με την αρνητική ακμή ρολογιού, άρα όταν ο παλμός ρολογιού έχει πέσει στο 0). Μόλις μετρηθεί ο 49ος παλμός και οι τέσσερις είσοδοι της πύλης NAND είναι 1, οπότε η έξοδός της που γίνεται 0 οδηγείται στην είσοδο Clear των flip-flops FF0, FF4, FF5, τα οποία και μηδενίζει ώστε να αρχίσει ξανά τη διαδικασία μέτρησης ο απαριθμητής.

## Παράδειγμα

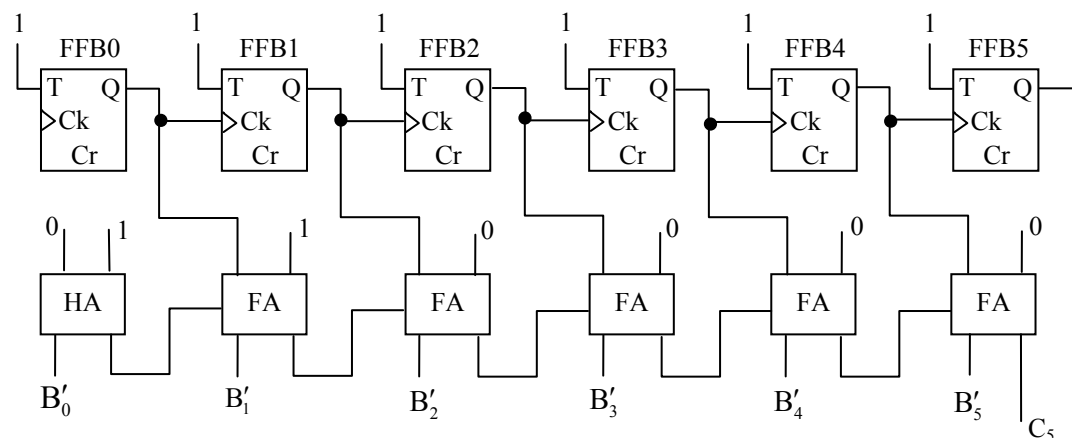
Δίνονται δύο εξαψήφιοι απαριθμητές A και B, με  $a$  και  $\beta$  το περιεχόμενό τους. Να σχεδιαστεί κύκλωμα το οποίο όταν  $a = \beta$  να παράγει σήμα  $Y = 1$ . Ομοίως για  $a = 2\beta + 3$ . (Διατίθεται μόνο ένας παράλληλος αθροιστής).

## Λύση

Η πρώτη συνθήκη η οποία πρέπει να ικανοποιείται, είναι το περιεχόμενο του απαριθμητή  $\beta$  να μην είναι πολύ μεγάλο, δηλαδή να μην είναι το μέγιστο σημαντικό του ψηφίο 1 (MSB = 1), διότι δεν θα μπορούμε να παραστήσουμε τον αριθμό  $2\beta$  με τον ίδιο απαριθμητή (γιατί;). Στο επόμενο σχήμα δίνουμε την κυκλωματική υλοποίηση της συνδεσμολογίας των δύο απαριθμητών προκειμένου να παράγεται το σήμα  $Y = 1$  όταν  $a = \beta$ .



Τα κυκλώματα που συμβολίζονται με το σύμβολο  $\Sigma$  είναι κυκλώματα συγκριτή ενός bit. Τέτοιο κύκλωμα είναι η πύλη XNOR η οποία σύμφωνα με τον πίνακα αληθείας της δίνει στην έξοδό της 1 όταν και οι δύο εισοδοί της είναι ίδιες (0 ή 1), ενώ δίνει στην έξοδό της 0 όταν οι εισοδοί της είναι διαφορετικές (δηλαδή η μία 1 και η άλλη 0). Έτσι, στο κύκλωμα του παραπάνω σχήματος, η έξοδος Y θα είναι ίση με 1 όταν και μόνο όταν και οι έξι εισοδοί της είναι 1. Αυτό θα συμβεί όταν οι έξοδοι και των έξι συγκριτών είναι 1, δηλαδή όταν το περιεχόμενο και των δύο απαριθμητών είναι ίδιο (η σύγκριση γίνεται ξεχωριστά σε κάθε bit). Για οποιοδήποτε άλλο συνδυασμό η έξοδος Y είναι 0.



### Κυκλωματική υλοποίηση της πράξης $2\beta+3$ .

Η ζητούμενη έξοδος Y θα προκύψει από μία πύλη AND της οποίας οι εισοδοί θα είναι τα σήματα:  $\overline{B_5}$ ,  $(A_0 \Sigma B'_0)$ ,  $(A_1 \Sigma B'_1)$ ,  $(A_2 \Sigma B'_2)$ ,  $(A_3 \Sigma B'_3)$ ,  $(A_4 \Sigma B'_4)$ ,  $(A_5 \Sigma B'_5)$ ,  $\overline{C_5}$ . Με το σύμβολο  $(A_i \Sigma B'_i)$  συμβολίζουμε το αποτέλεσμα της σύγκρισης του bit  $A_i$  του απαριθμητή A με το αντίστοιχο bit  $B'_i$  του Σχ. 6.43. Η πύλη AND πρέπει να έχει ως είσοδο το σήμα  $\overline{B_5}$  για να ελέγχουμε εάν το αρχικό περιεχόμενο του απαριθμητή B είχε  $MSB = 1$  όπως είπαμε στην αρχή. Επίσης, η πύλη AND πρέπει να έχει ως είσοδο το σήμα  $\overline{C_5}$  ώστε να ελέγχουμε εάν τυχόν προκύψει κρατούμενο από την πράξη  $2\beta+3$ . Έτσι, η έξοδος Y θα δίνεται από το γινόμενο, με χρήση μιας πύλης AND

$$Y = \overline{B_5} \cdot (A_0 \Sigma B'_0) \cdot (A_1 \Sigma B'_1) \cdot (A_2 \Sigma B'_2) \cdot (A_3 \Sigma B'_3) \cdot (A_4 \Sigma B'_4) \cdot (A_5 \Sigma B'_5) \cdot \overline{C_5}$$

και θα έχουμε

$$Y = 1 \text{ όταν } \alpha = 2\beta + 3$$

$$Y = 0 \text{ όταν } \alpha \neq 2\beta + 3$$

Έτσι, η κυκλωματική υλοποίηση θα είναι ίδια με αυτήν του Σχ. 6.42 όπου όμως οι έξοδοι  $B_0 - B_5$  θα αντικατασταθούν με τις εξόδους  $B'_0 - B'_5$ , και έχουν προστεθεί στην πύλη AND οι έξοδοι  $\overline{B_5}$  και  $\overline{C_5}$ . Για λόγους σχεδιαστικής πολυπλοκότητας παραλείπουμε την πλήρη κυκλωματική υλοποίηση.

## Παράδειγμα

Να σχεδιαστεί BCD απαριθμητής με δυνατότητα μέτρησης 99 παλμών. Ο απαριθμητής θα πρέπει:

- 1) Όταν δέχεται σήμα εισόδου  $R = 1$  να μηδενίζεται.
- 2) Να παράγει σήμα εξόδου  $Y = 1$  όταν το αποτέλεσμα της μέτρησης είναι διαιρετό με το  $(11)_{10}$ .

## Λύση

Από τα δεδομένα του παραδείγματος είναι φανερό ότι θα πρέπει να χρησιμοποιήσουμε δύο δεκαδικούς BCD απαριθμητές, έναν για κάθε δεκάδα, δεδομένου ότι πρέπει να μπορούμε να μετρήσουμε 99 παλμούς. Η είσοδος  $R$  συνδέεται στο Reset των δεκαδικών απαριθμητών και όταν ενεργοποιείται (δηλ.  $R = 1$ ) μηδενίζει τους απαριθμητές. Για να είναι διαιρετή η έξοδος του απαριθμητή με το 11 θα πρέπει τα δύο δεκαδικά ψηφία να είναι ίσα, δηλαδή να είναι 11, 22, 33, ... 99.

Έτσι, συνδέουμε ανά δύο τα ψηφία των εξόδων ίδιας σημαντικότητας των απαριθμητών BCD ως εισόδους σε συγκριτές ενός bit, και τα αποτελέσματα των συγκρίσεων διοχετεύονται σε μια πύλη AND. Η έξοδος της πύλης αυτής θα είναι 1 όταν και μόνο όταν τα όλα τα ψηφία ίδιας σημαντικότητας είναι ίσα μεταξύ τους, δηλαδή όταν οι δεκαδικοί αριθμοί στην έξοδο των απαριθμητών είναι ίσοι.

