3^η Θεματική Ενότητα : Σύγχρονα Ακολουθιακά Κυκλώματα

Επιμέλεια διαφανειών:
Χρ. Καβουσιανός

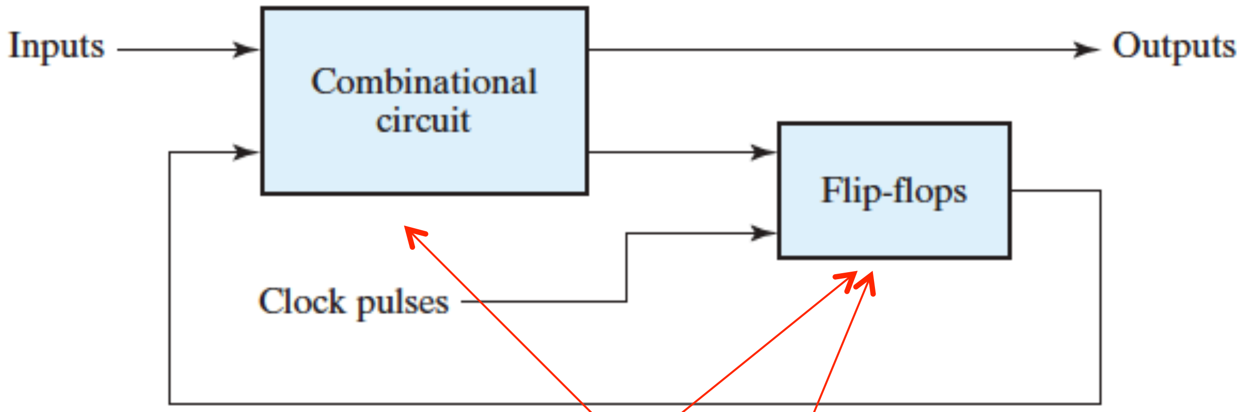
Εισαγωγή

- Κατάσταση Ακολουθιακού Κυκλώματος : περιεχόμενα στοιχείων μνήμης.
- Η έξοδος εξαρτάται από τις εισόδους και την κατάσταση του κυκλώματος.
- Η κατάσταση εξαρτάται από τις εισόδους και την προηγούμενη κατάσταση.

Ακολουθιακά Κυκλώματα

- Σύγχρονα: οι τιμές των σημάτων του αλλάζουν σε διακριτές χρονικές στιγμές (ρολόι).
- Ασύγχρονα: οι τιμές των σημάτων του αλλάζουν σε οποιαδήποτε χρονική στιγμή (συνδυαστικά κυκλώματα με ανάδραση).

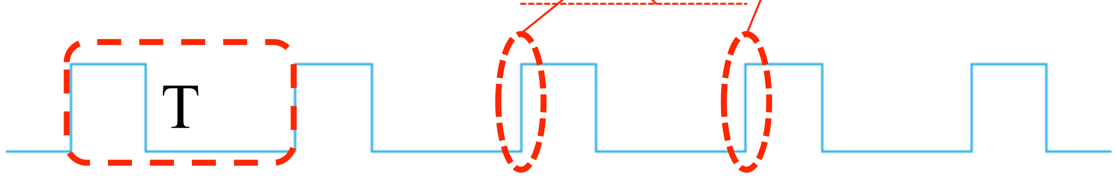
Ρολόι και Συγχρονισμός



(a) Block diagram

T = Περίοδος Ρολογιού

F = 1/T



(b) Timing diagram of clock pulses

Η αποθήκευση γίνεται σε συγκεκριμένες διακριτές χρονικές στιγμές: **θετικές** ή **αρνητικές** ακμές

Latches & Flip Flops

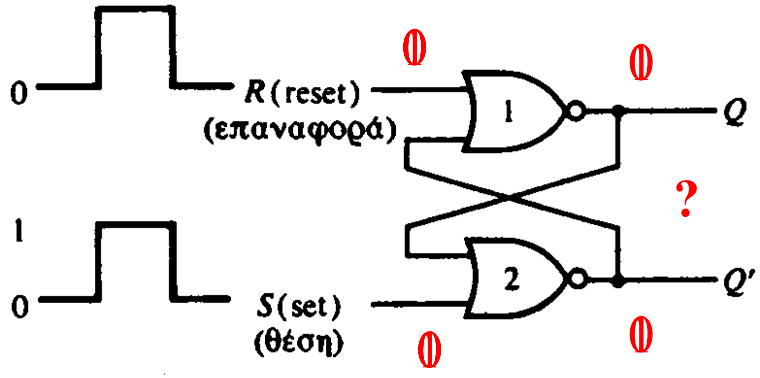
Αποθηκευτικά
Στοιχεία

Latches (Μανδαλωτές)	Flip Flops (ακμοπυροδότητα)
Αποθηκεύουν σε επίπεδο	Αποθηκεύουν σε ακμή
Χρησιμοποιούνται πολύ σε ασύγχρονα συστήματα	Χρησιμοποιούνται σε σύγχρονα συστήματα
Είναι δομικά στοιχεία των Flip Flop	Σχεδιάζονται από Latches

Βασικό Κύκλωμα RS Μανδαλωτής (ΟΥΤΕ)

Ένα κύκλωμα latch μπορεί να διατηρήσει την εσωτερική του κατάσταση επ'αόριστο έως ότου κάποιο σήμα εισόδου το κάνει να αλλάξει κατάσταση.

<u>Μανταλωτής SR (latch)</u>



(α) Λογικό διάγραμμα

S	R	Q	Q'	
1	0	1	0	
0	0	1	0	(μετά από $S=1$, $R=0$)
0	1	0	1	
0	0	0	1	(μετά από $S=0$, $R=1$)
1	1	0	0	

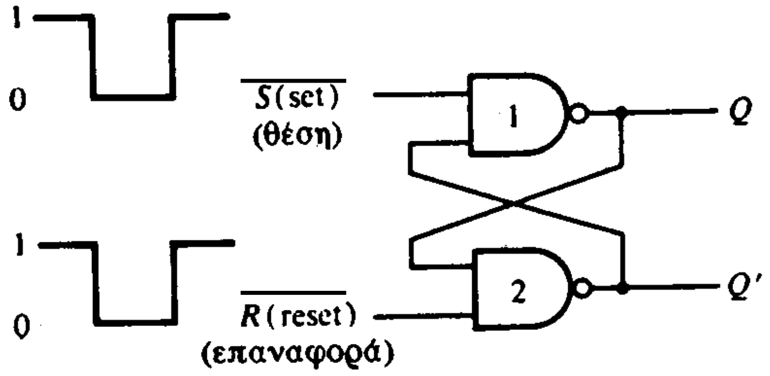
(β) Πίνακας αληθείας

- Q : κανονική έξοδος
- Q' : συμπληρωματική έξοδος
- R : είσοδος επαναφοράς
- S : είσοδος θέσης

Για ίδια είσοδο δίνει διαφορετική έξοδο.

Κρατάει την προηγούμενη κατάσταση (ακολουθιακό).

Βασικό Κύκλωμα RS Μανδαλωτή



(α) Λογικό διάγραμμα

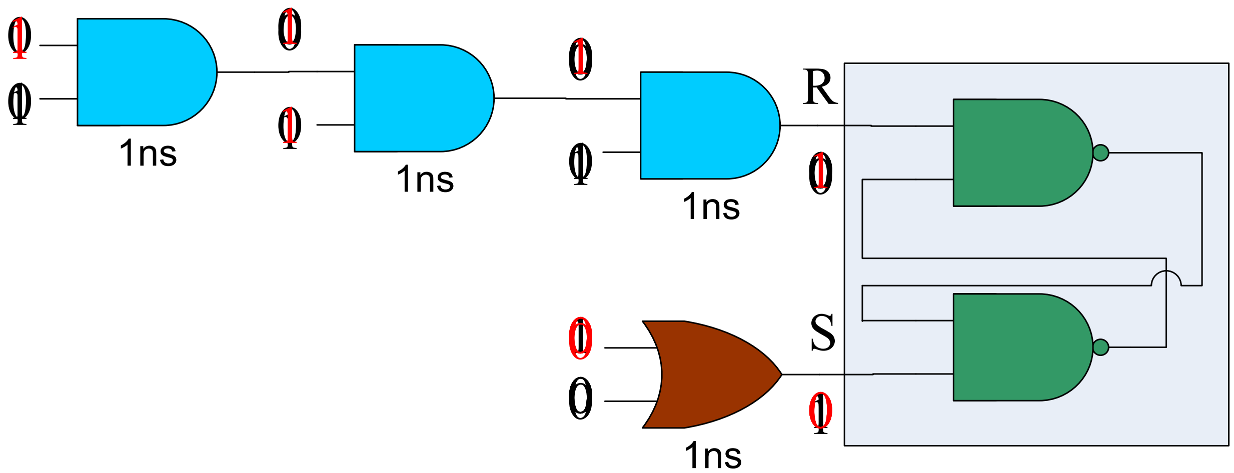
$\overline{S}$	$\overline{R}$	Q	Q'	
1	0	0	1	
1	1	0	1	(μετά από $S=1$, $R=0$)
0	1	1	0	
1	1	1	0	(μετά από $S=0$, $R=1$)
0	0	1	1	

(β) Πίνακας αληθείας

- Το κύκλωμα λειτουργεί με τις δύο του εισόδους κανονικά στο 1 εκτός αν θέλουμε να αλλάξουμε κατάσταση.
- Με την εφαρμογή ενός στιγμιαίου 0 στην είσοδο θέσης, η έξοδος Q γίνεται 1.
- Με την εφαρμογή ενός στιγμιαίου 0 στην είσοδο επαναφοράς, η έξοδος Q γίνεται 0.

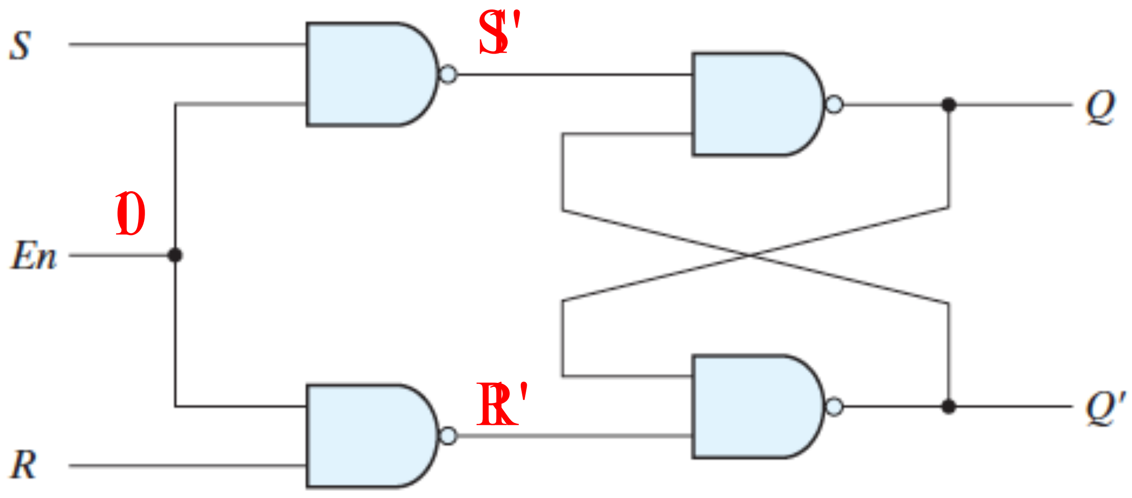
RS Μανδαλωτής

Οι ψηφιακές πύλες έχουν καθυστέρηση → Οι είσοδοι των flip flop μπορεί να αλλάζουν συνεχώς μέχρι να σταθεροποιηθούν



t = 4 ns → Οι είσοδοι των Latches πρέπει να απομονώνονται μέχρι το συνδυαστικό κύκλωμα να ηρεμήσει

RS Μανδαλωτής με επίτρεψη



(a) Logic diagram

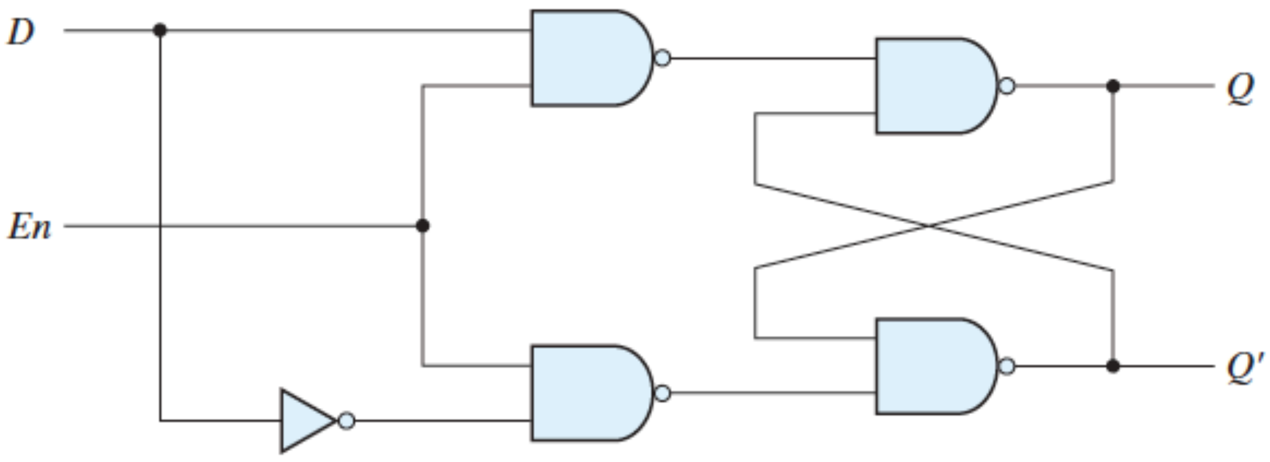
En	S	R	Next state of Q
0	X	X	No change
1	0	0	No change
1	0	1	$Q = 0$; reset state
1	1	0	$Q = 1$; set state
1	1	1	Indeterminate

(b) Function table

Η λειτουργία του flip flop τροποποιείται με την τοποθέτηση πρόσθετης εισόδου ελέγχου που καθορίζει πότε θα αλλαχθεί η κατάστασή του.

D Μανδαλωτής

Εξασφαλίζει ότι οι είσοδοι του flip flop δεν θα πάνε ποτέ στο 1 ταυτόχρονα. Ταυτόχρονα όμως δεν πάνε ποτέ ούτε στο 0 οπότε χάνουμε την διατήρηση της προηγούμενης κατάστασης



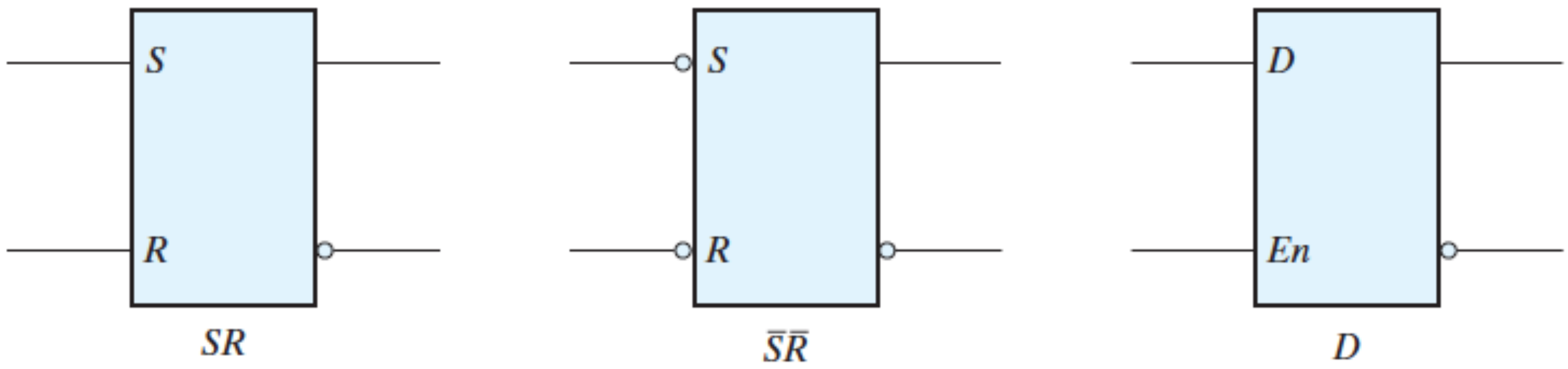
(a) Logic diagram

En	*D*	Next state of *Q*
0	X	No change
1	0	$Q = 0$; reset state
1	1	$Q = 1$; set state

(b) Function table

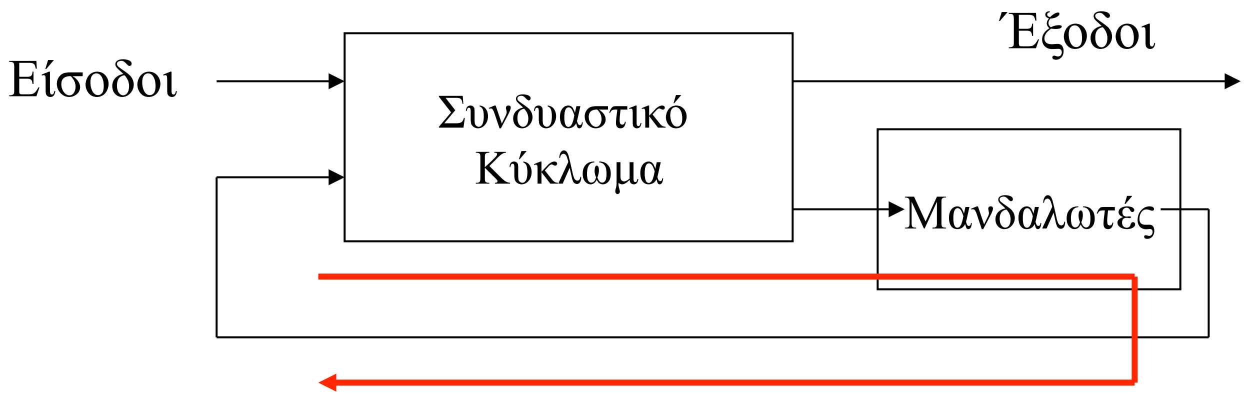
Φυλασσόμενος Μανταλωτής D (gated D-latch).

Γραφικά Σύμβολα Μανδαλωτών



Στη σχεδίαση των ακολουθιακών κυκλωμάτων
δεν μας ενδιαφέρει το εσωτερικό των
μανδαλωτών αλλά μόνο η διεπαφή τους

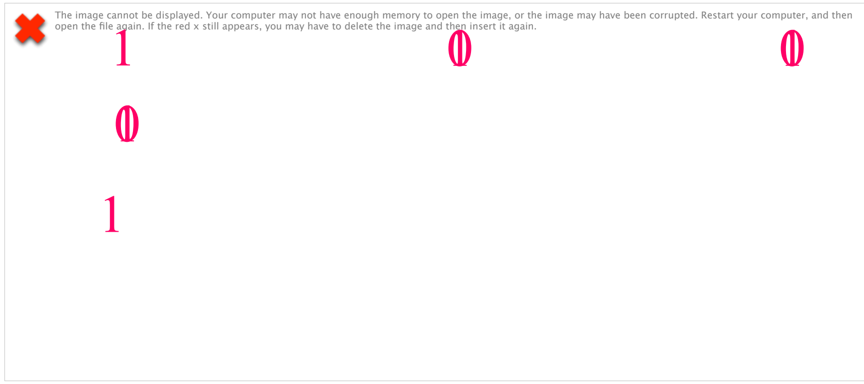
Πρόβλημα Μανδαλωτών

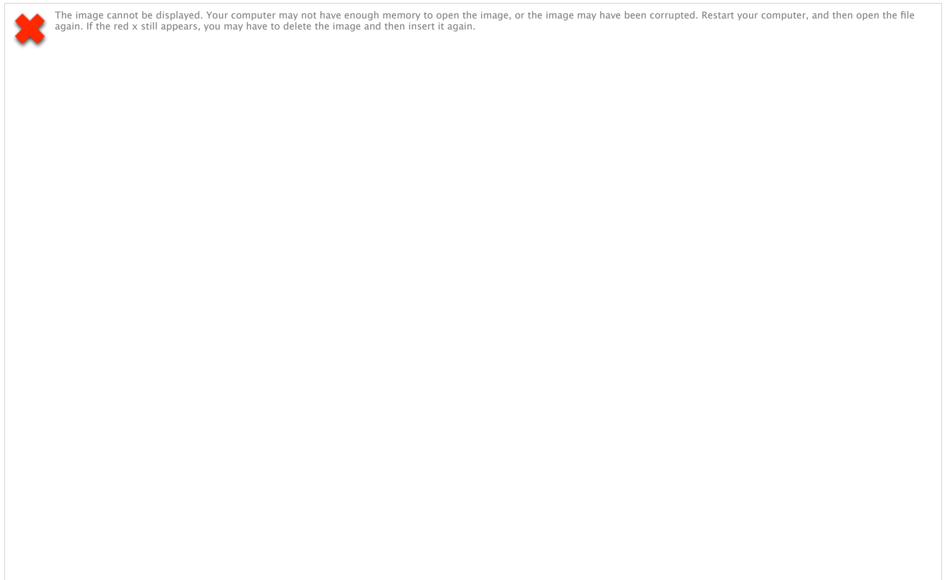


Οι μανδαλωτές δημιουργούν αστάθεια στα σύγχρονα κυκλώματα καθώς αλλαζουν τιμή για μεγάλο χρονικό διάστημα

Όσο οι μαδαλωτές αλλάζουν τιμή το συνδυαστικό κύκλωμα επαναϋπολογίζει την κατάσταση του και μπορεί να αλλάξει την τιμή στις εισόδους των μανδαλωτών

Πυροδότηση των Flip-Flops





Η πυροδότηση εξαλείφει το πρόβλημα της αστάθειας ακολουθώντας δύο διαφορετικές μεθοδολογίες:

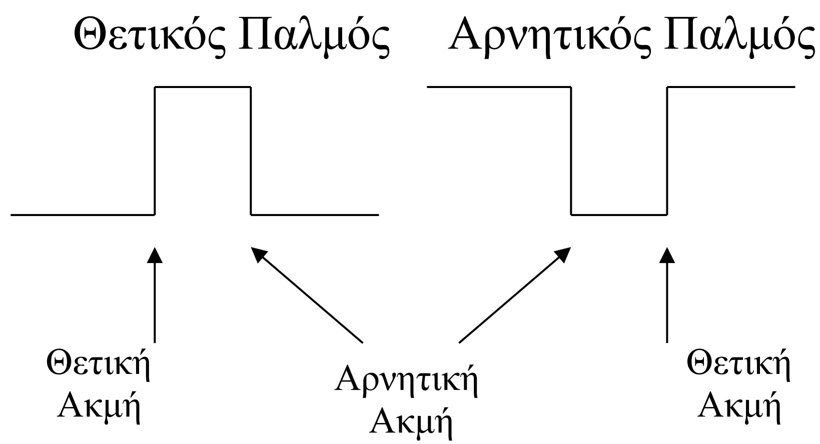
1. Τοποθετεί ένα δεύτερο αποθηκευτικό στοιχείο το οποίο λειτουργεί συμπληρωματικά με το 1ο και κόβει την ανάδραση που δημιουργεί το πρόβλημα.
2. Επιτρέπει την αποθήκευση να γίνεται στιγμιαία (κατά την άνοδο ή κάθοδο του ρολογιού) ώστε να μην υπάρχει χρόνος ενεργοποίησης της ανάδρασης και πρόκλησης έτσι αστάθειας.

Πυροδότηση των Flip-Flops

Πυροδότηση είναι η αλλαγή κάποιας εισόδου του flip-flop που προκαλεί αλλαγή στην κατάστασή του. Είδη: level sensitive - edge triggered

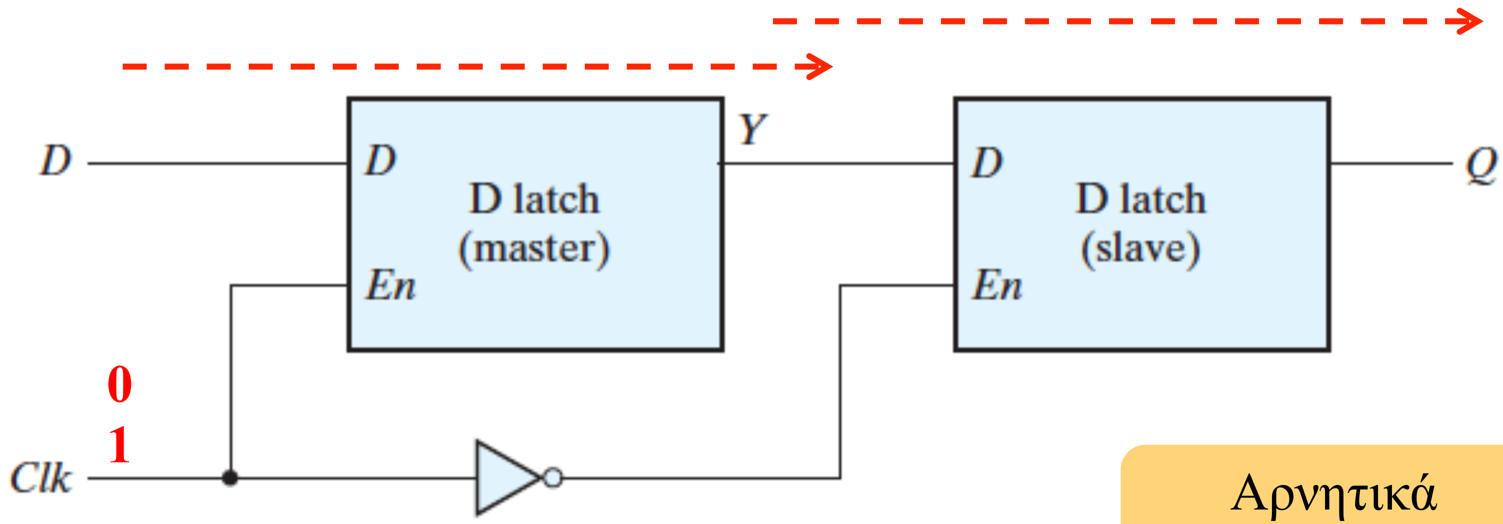
Σε δειγματοληψία με τον παλμό ρολογιού (level) το κύκλωμα μπορεί να οδηγηθεί σε αστάθεια.

Σε δειγματοληψία με την ακμή ρολογιού (edge) το κύκλωμα δεν θα έχει πρόβλημα.



Flip-Flop Αφέντη-Σκλάβου

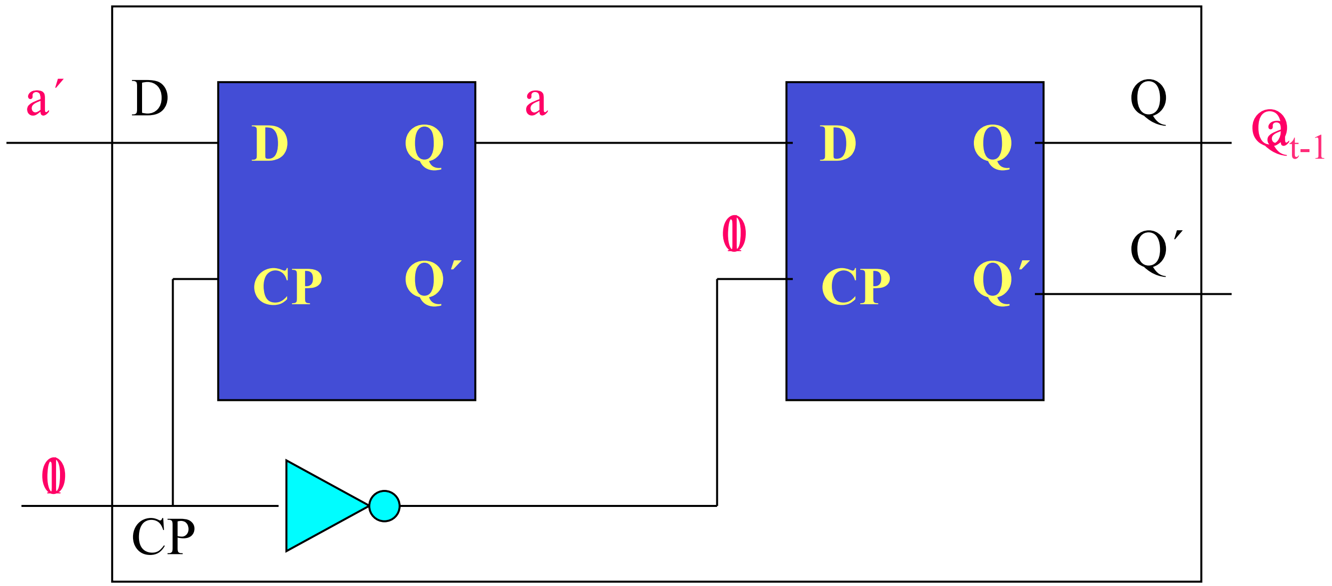
Το flip flop Αφέντη-Σκλάβου περιέχει δύο απλά flip flops. Το ένα εκτελεί χρέη αφέντη και το άλλο χρέη σκλάβου. Μπορούν να κατασκευαστούν όλοι οι τύποι flip-flops.



Αρνητικά ακμοπυροδότητο

- CLK = 0 → Αφέντης: απενεργοποιημένος, Σκλάβος: ενεργός
- CLK = 1 →. Αφέντης: ενεργός, Σκλάβος: απενεργοποιημένος

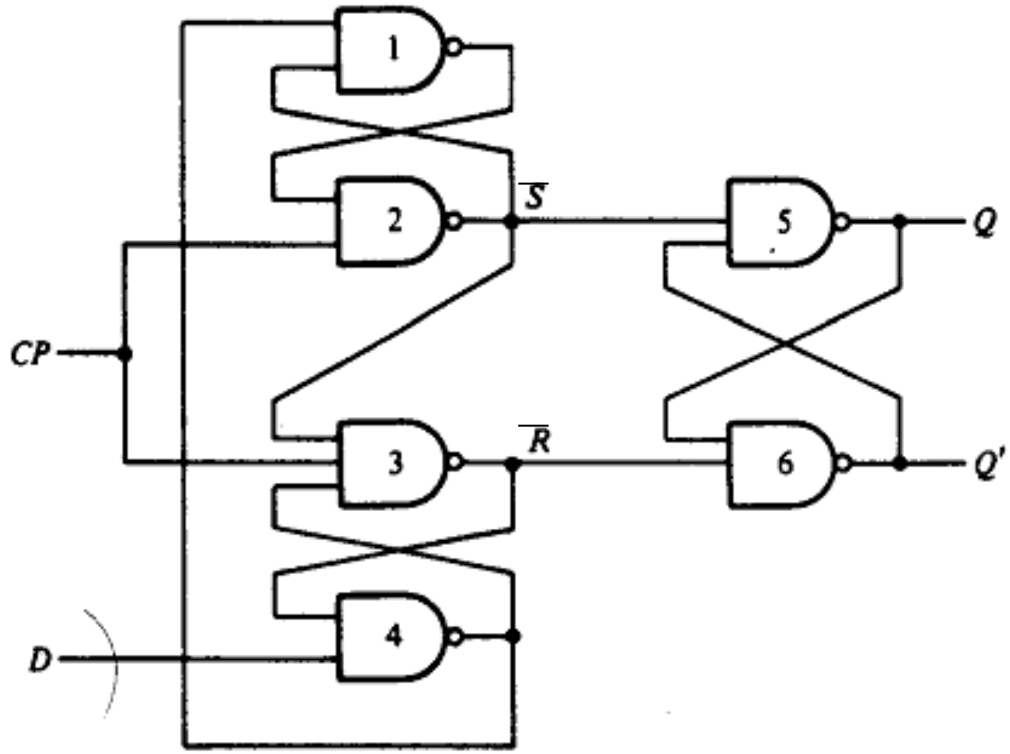
Flip-Flop Αφέντη-Σκλάβου



- CP = 0 → Αφέντης: απενεργοποιημένος, Σκλάβος: ενεργός
- CP = 1 →. Αφέντης: ενεργός, Σκλάβος: απενεργοποιημένος

Ακμοπυροδότητα Flip-Flops

Στο ακμοπυροδότητο flip-flop όλες οι αλλαγές στις εξόδους συμβαίνουν σε μία ακμή.

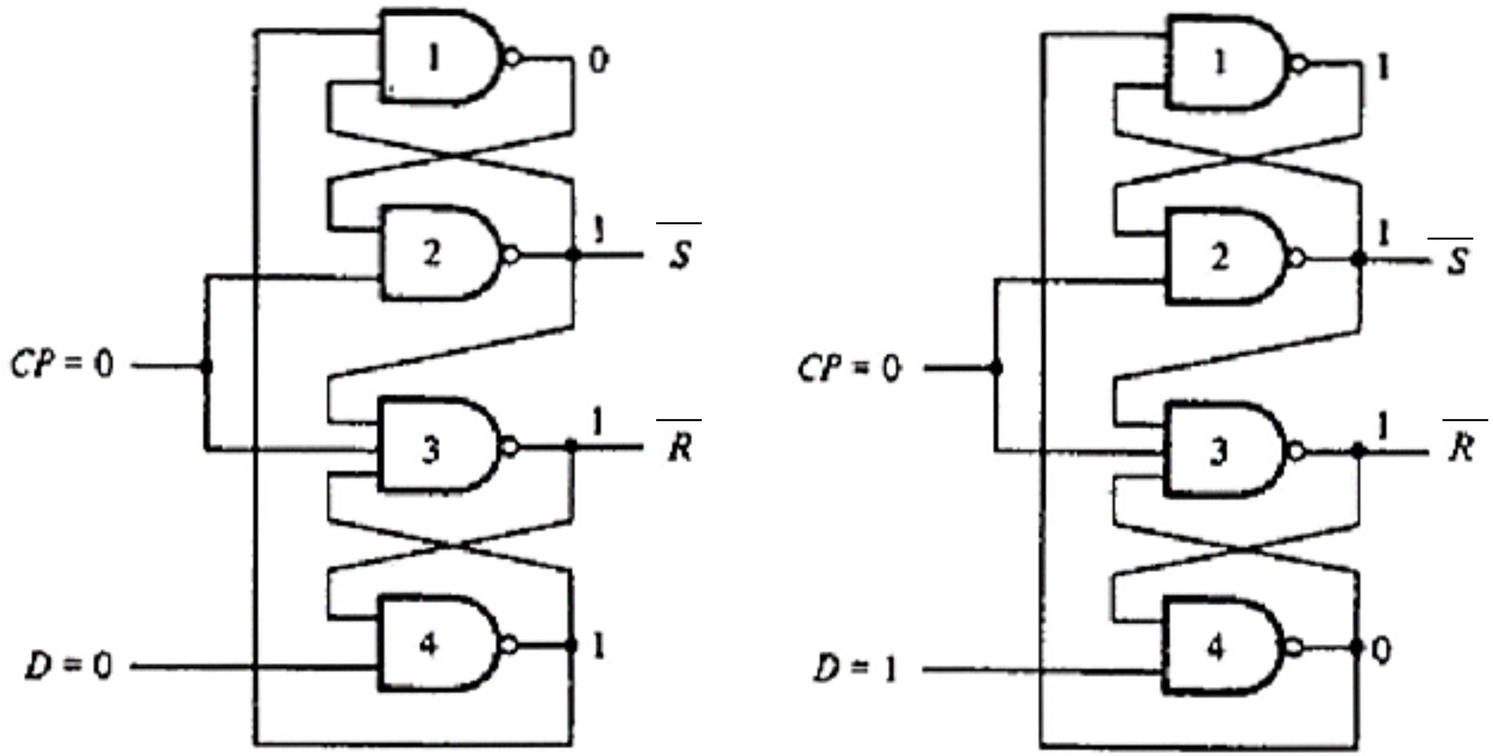


Αποτελείται ουσιαστικά από τρία βασικά flip-flops.

Μερικά ακμοπυροδότητα flip-flops αντιδρούν στην αρνητική ακμή του ρολογιού και άλλα στη θετική ακμή.

Ακμοπυροδότητα Flip-Flops

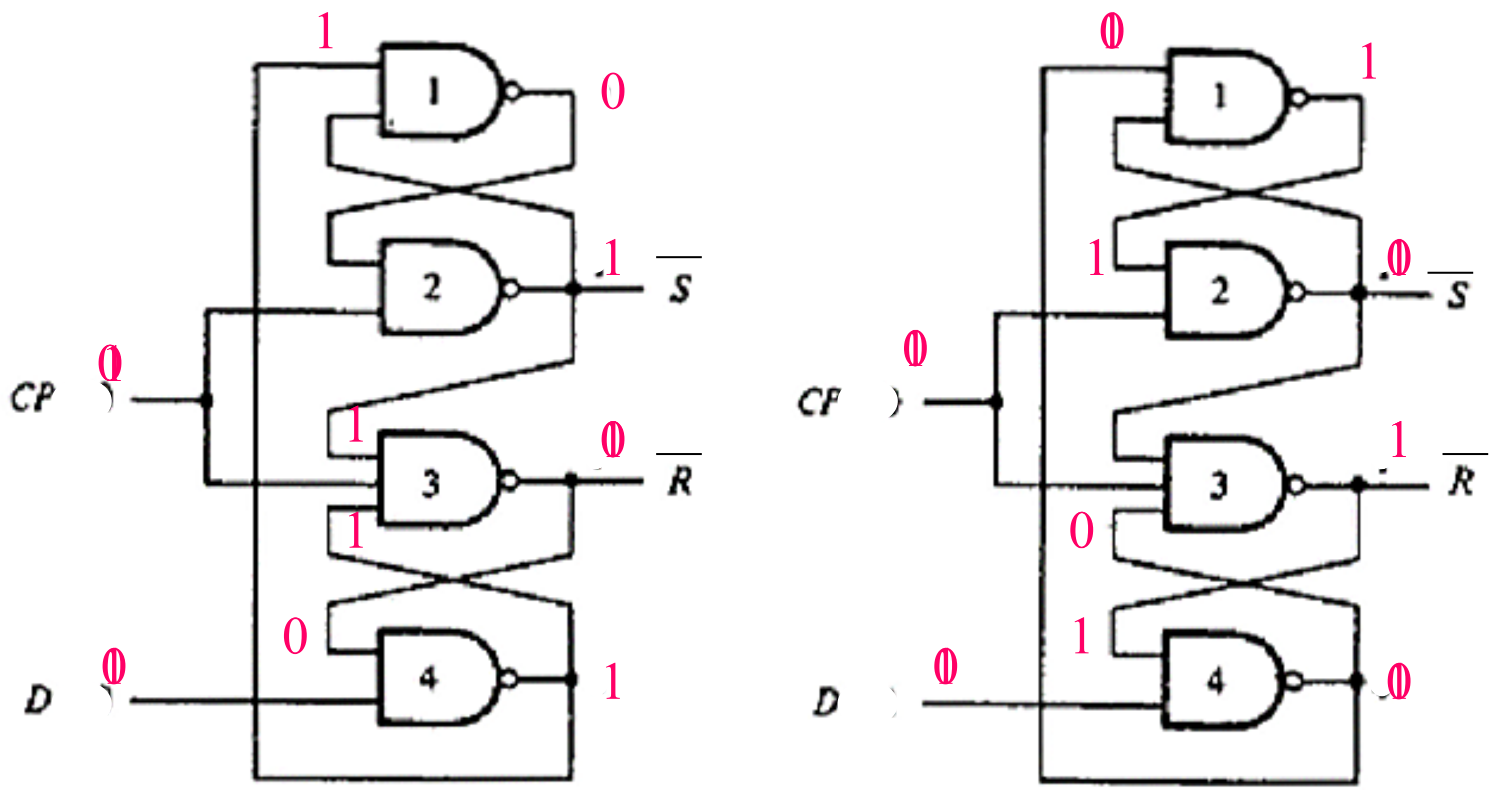
Επιβάλει την τήρηση ορισμένων χρονικών προδιαγραφών για να λειτουργήσει σωστά:
Χρόνος Προετοιμασίας και Χρόνος Κρατήματος



(α) Όταν CP = 0

Όταν CP=0, το 3ο FF παίρνει εισόδους (1,1) και διατηρεί την κατάσταση του.

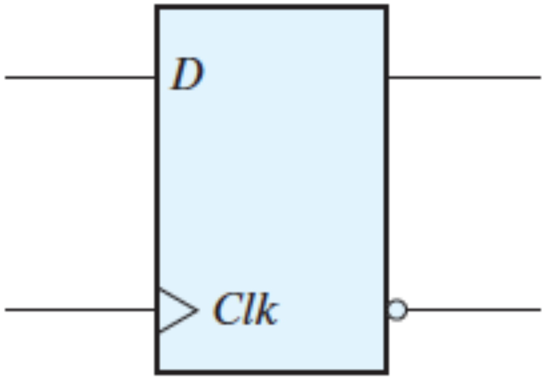
Ακμοπυροδότητα Flip-Flops



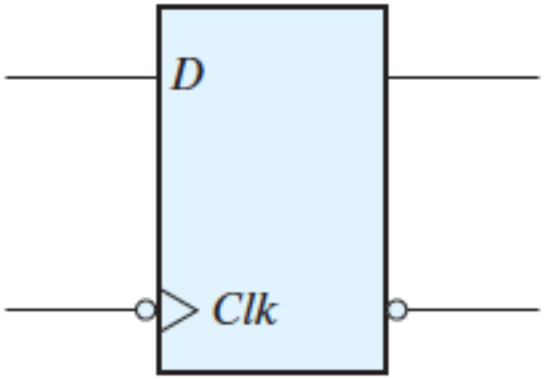
Ακμοπυροδότητα Flip-Flops

Setup Time: Χρόνος Προετοιμασίας πριν την ακμή στον οποίο η είσοδος D πρέπει να κρατηθεί σταθερή για να κλειδωθεί.

Hold Time: Χρόνος Κρατήματος μετά την ακμή στον οποίο η είσοδος D πρέπει να κρατηθεί σταθερή για να κλειδωθεί.



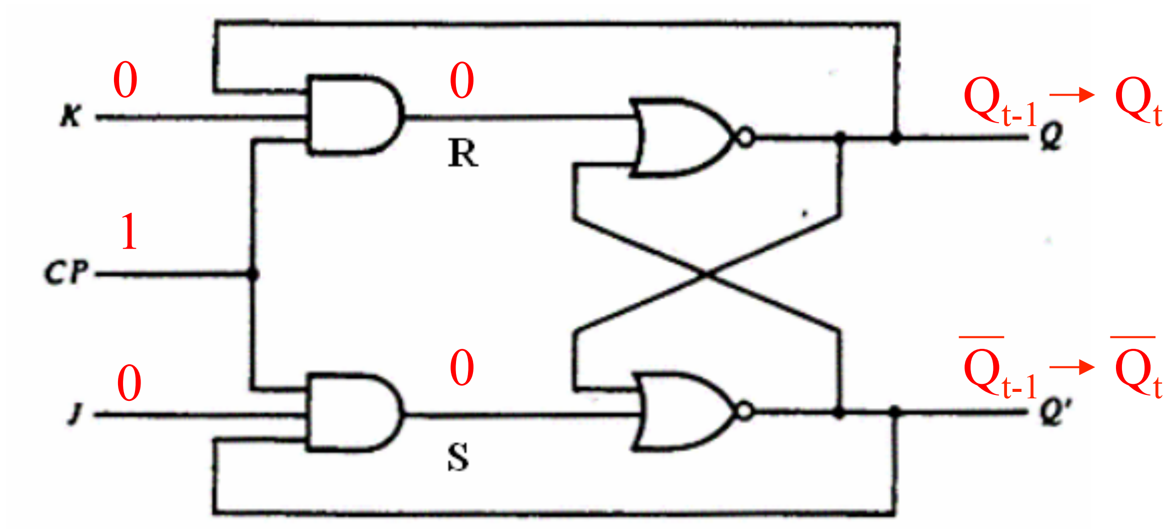
(a) Positive-edge



(a) Negative-edge

JK Flip-Flop

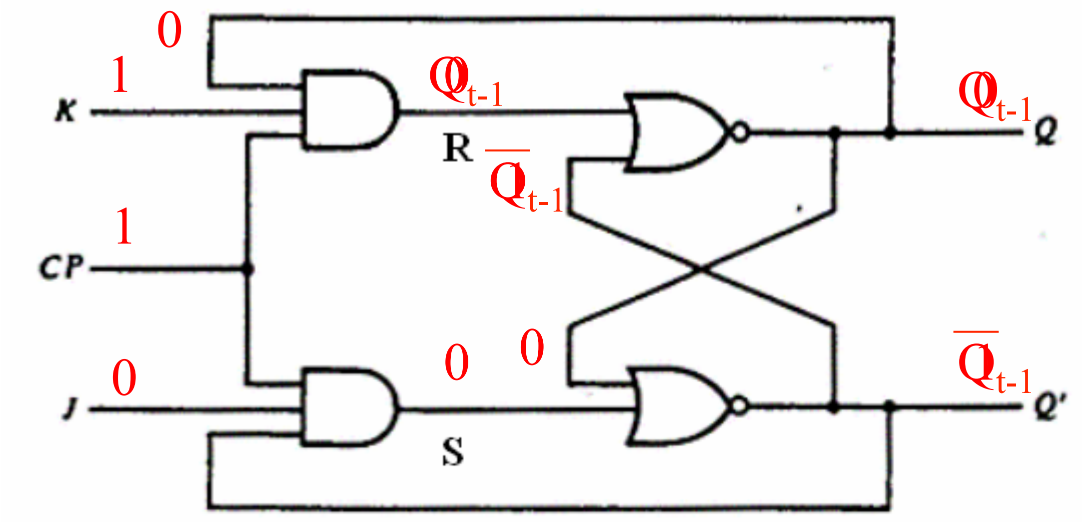
Η απροσδιόριστη κατάσταση του RS εδώ προσδιορίζεται και αξιοποιείται.



Με JK=00 διατηρεί την προηγούμενη κατάσταση

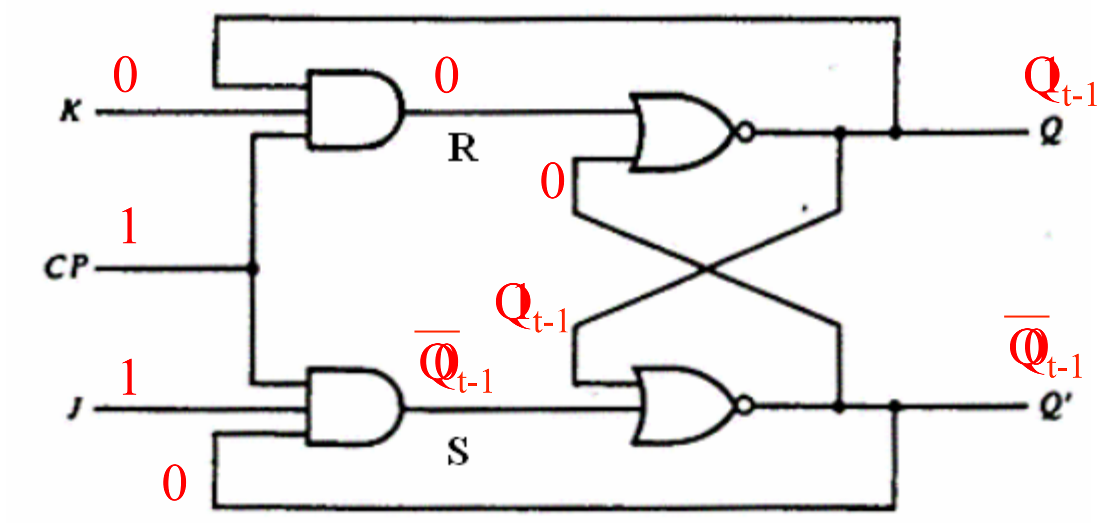
JK Flip-Flop

Η απροσδιόριστη κατάσταση του RS εδώ προσδιορίζεται και αξιοποιείται.



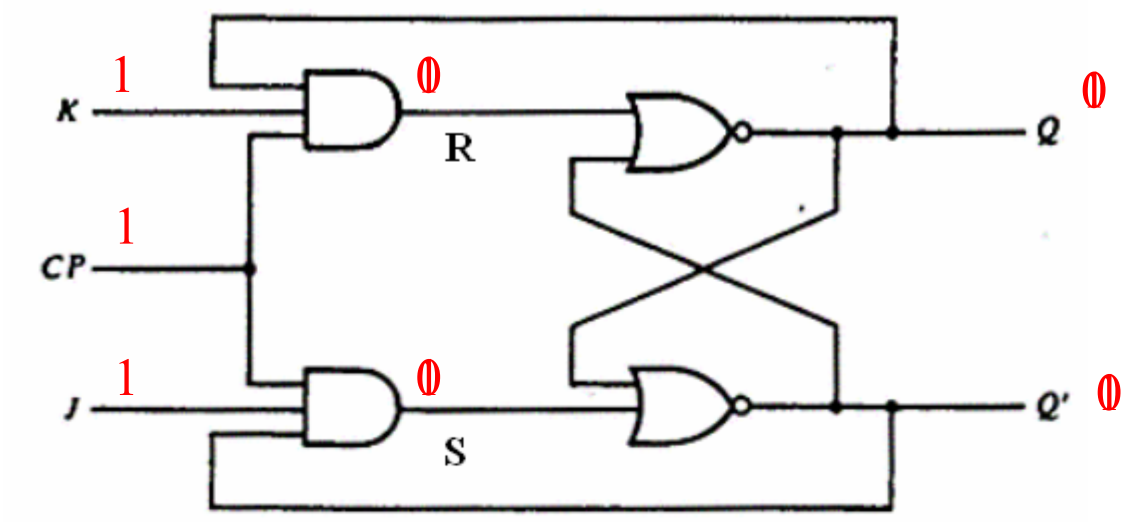
Με JK=01 μηδενίζεται

JK Flip-Flop



Με JK=10 τίθεται

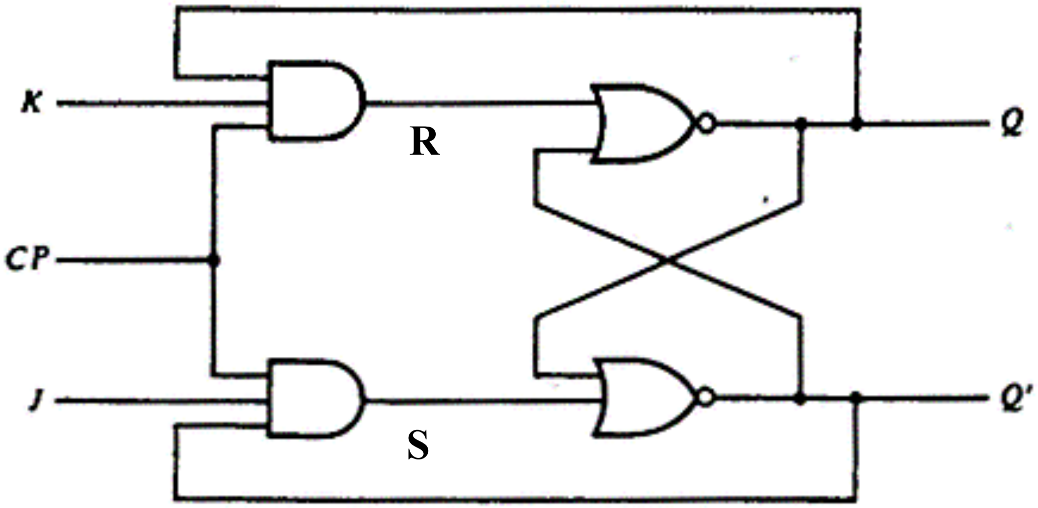
JK Flip-Flop



Με JK=11 αντιστρέφει την προηγούμενη κατάσταση

JK Flip-Flop

Η απροσδιόριστη κατάσταση του RS εδώ προσδιορίζεται και αξιοποιείται.



(α) Λογικό διάγραμμα

Αν CP=1 για αρκετό χρόνο και J=K=1 τότε οι έξοδοι θα αντιστρέφονται συνεχώς. Για τον λόγο αυτό χρησιμοποιούνται τα edge-triggered, master-slave ffs.

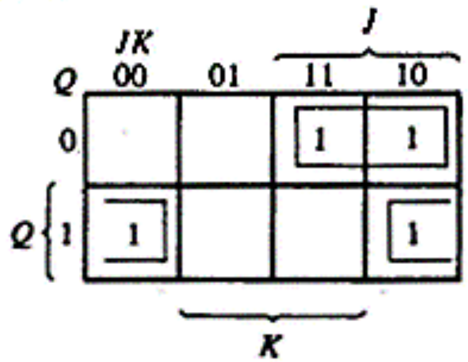
Q	J	K	Q(t + 1)
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	0

retain

set-reset

invert

(β) Χαρακτηριστικός πίνακας

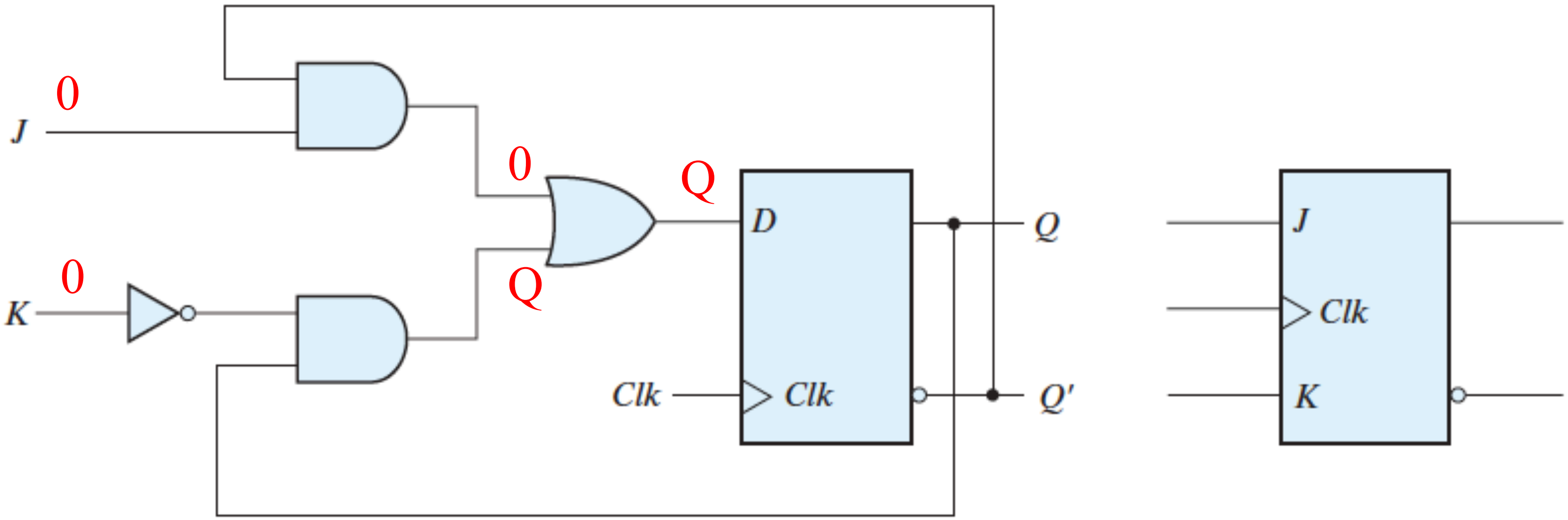


$$Q(t+1) = JQ' + K'Q$$

(γ) Χαρακτηριστική εξίσωση

JK Flip-Flop

Με JK = 00 διατηρεί την προηγούμενη κατάσταση

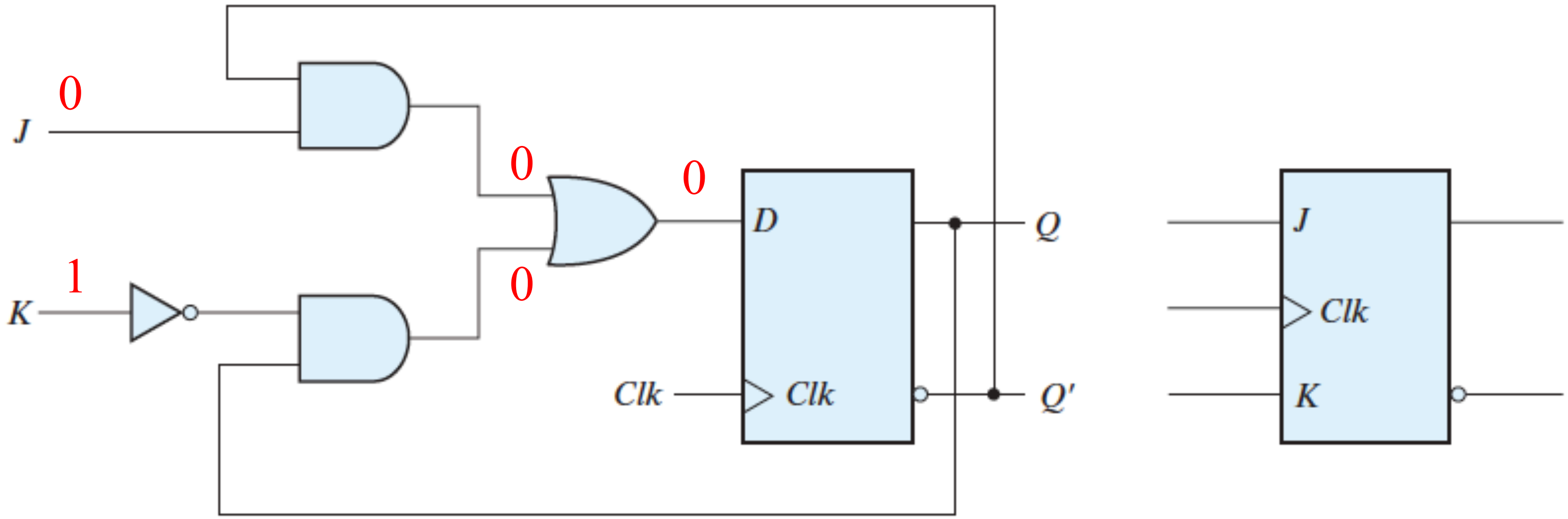


(a) Circuit diagram

(b) Graphic symbol

JK Flip-Flop

Με JK = 01 μηδενίζει το D Flip Flop

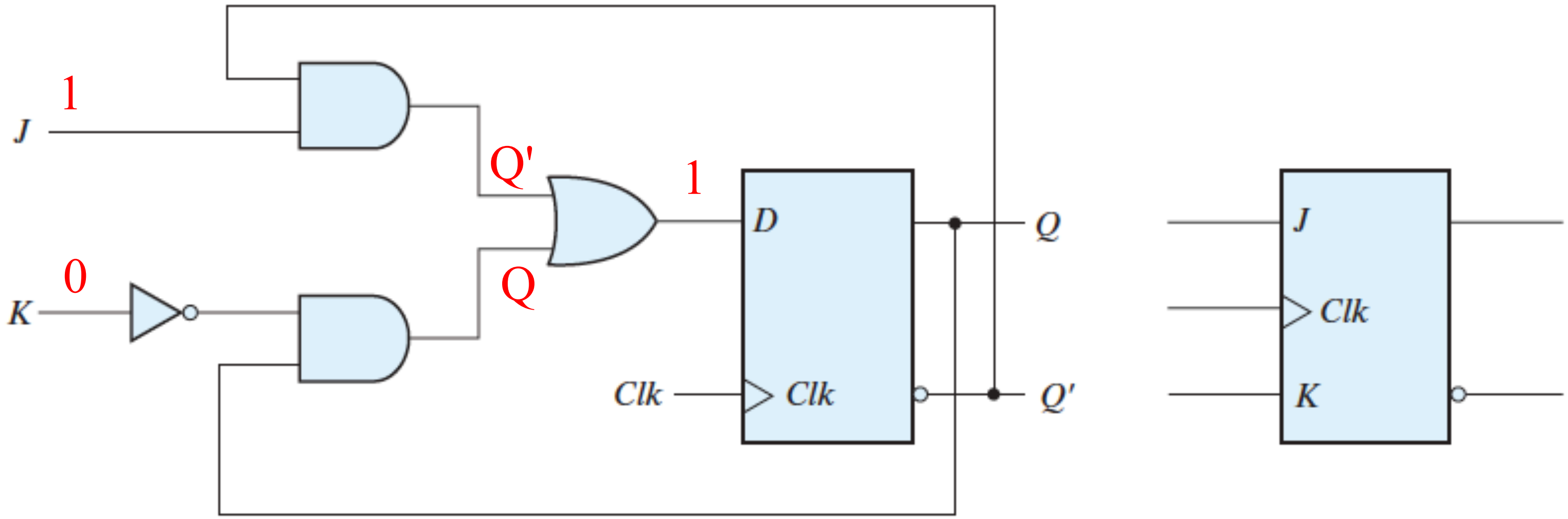


(a) Circuit diagram

(b) Graphic symbol

JK Flip-Flop

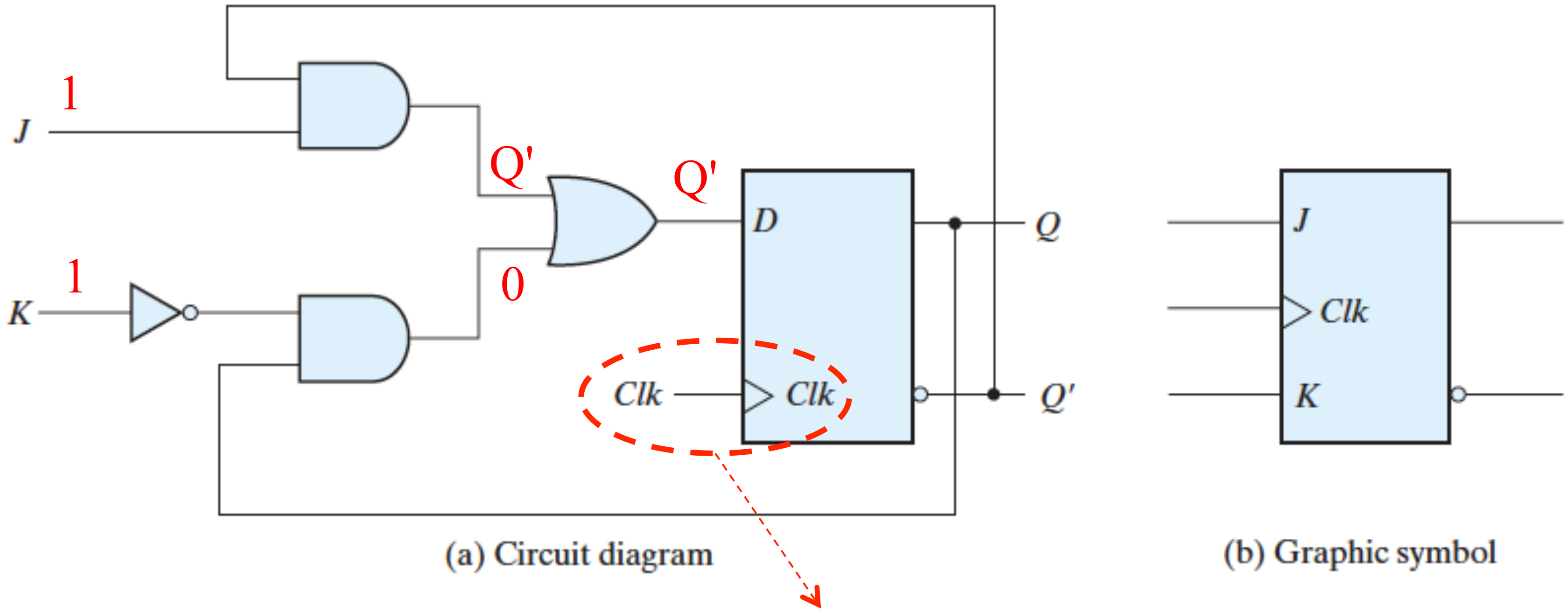
Με JK = 10 θέτει το FF στην μονάδα



(a) Circuit diagram (b) Graphic symbol

JK Flip-Flop

Με JK = 11 αντιστρέφει



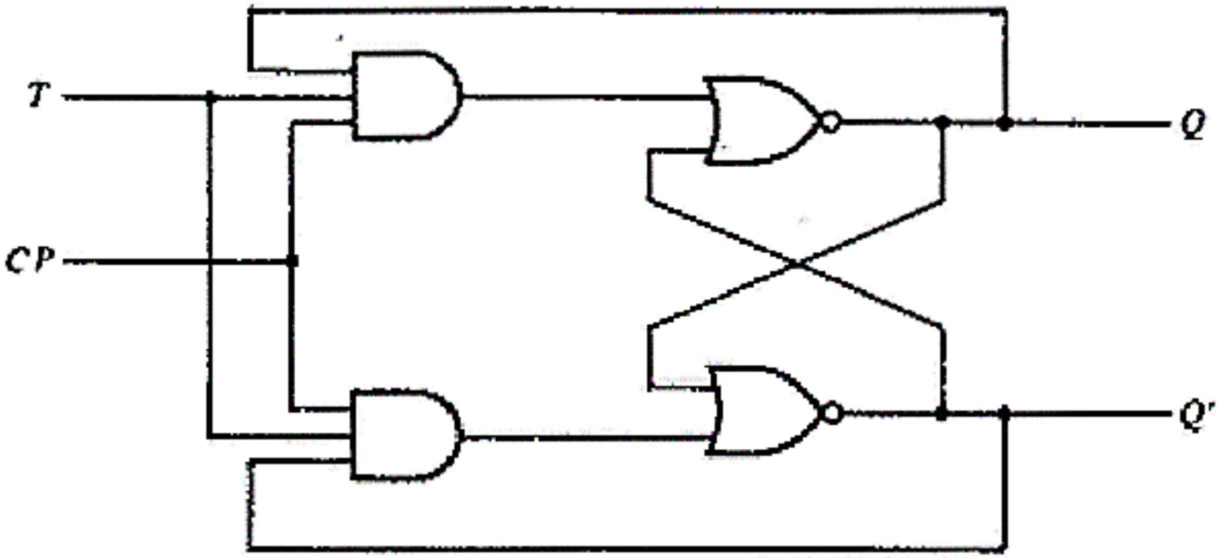
(a) Circuit diagram

(b) Graphic symbol

Η θετική ακμή εξασφαλίζει ότι θα γίνει μία φορά η αντιστροφή

T Flip-Flop

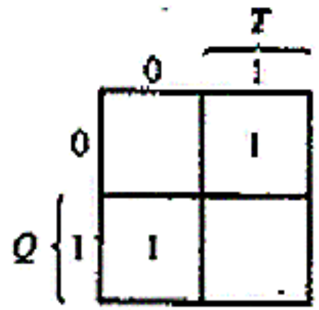
Είναι μία παραλλαγή του JK με μία μόνο είσοδο T και όταν T=1 αντιστρέφει την κατάστασή του.



(α) Λογικό διάγραμμα

Q	T	$Q(t+1)$
0	0	0
0	1	1
1	0	1
1	1	0

(β) Χαρακτηριστικός πίνακας

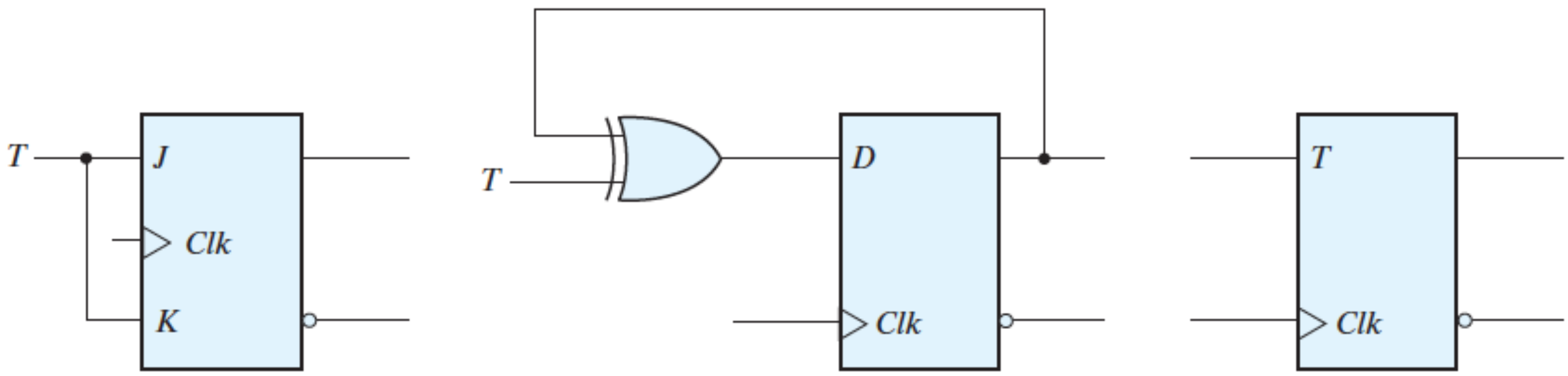


$$Q(t+1) = TQ' + T'Q$$

(γ) Χαρακτηριστική εξίσωση

T Flip-Flop

Υπάρχουν πολλοί τρόποι σχεδίασης ενός Flip Flop



(a) From *JK* flip-flop (b) From *D* flip-flop (c) Graphic symbol

Χαρακτηριστικοί Πίνακες

JK Flip-Flop

J	K	Q(t + 1)	
0	0	$Q(t)$	No change
0	1	0	Reset
1	0	1	Set
1	1	$Q'(t)$	Complement

$$Q(t+1) = JQ'+K'Q$$

D Flip-Flop

D	Q(t + 1)	
0	0	Reset
1	1	Set

$$Q(t+1) = D$$

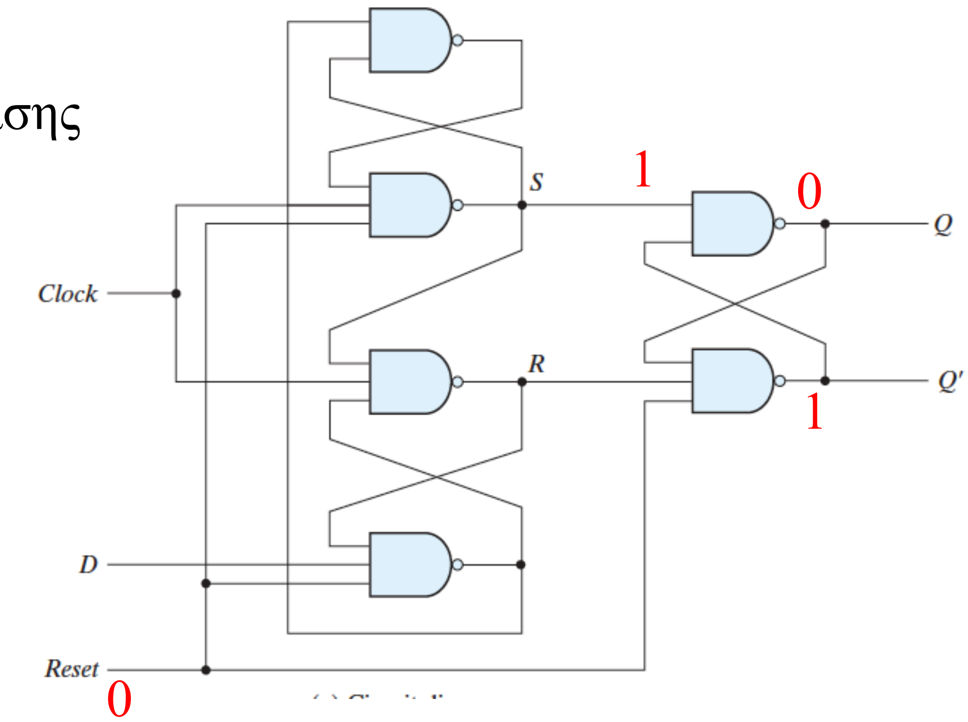
T Flip-Flop

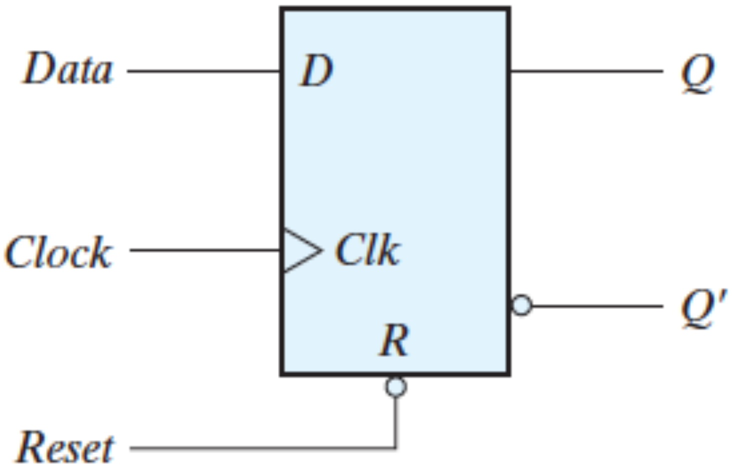
T	Q(t + 1)	
0	$Q(t)$	No change
1	$Q'(t)$	Complement

$$Q(t+1) = T \text{ xor } Q$$

Ασύγχρονες είσοδοι

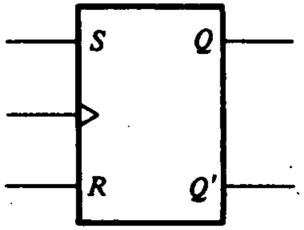
Παρέχουν δυνατότητα θέσης /μηδένισης ανεξάρτητης του ρολογιού



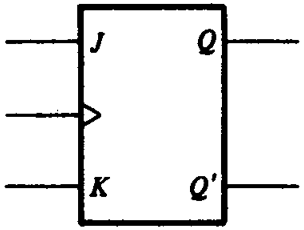


R	Clk	D	Q	Q'
0	X	X	0	1
0	↑	0	0	1
0	↑	1	1	0

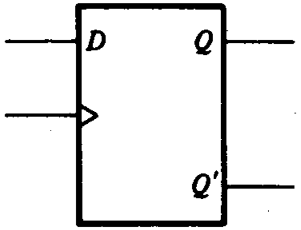
Γραφικά Σύμβολα Flip-Flops



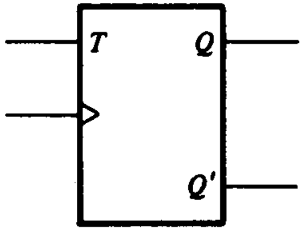
(α) *RS*



(β) *JK*

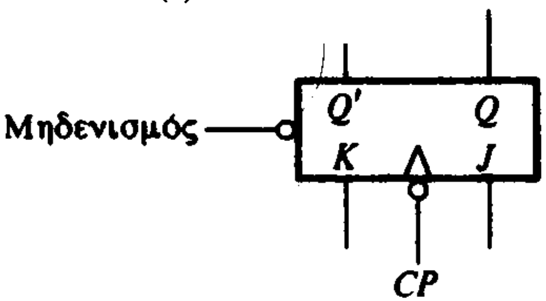


(γ) *D*



(δ) *T*

- Το τρίγωνο δείχνει λειτουργία στη θετική ακμή.
- Το τρίγωνο με ένα κύκλο δείχνει λειτουργία στην αρνητική ακμή.
- Παρέχονται και οι δύο συμπληρωματικές έξοδοι.
- Παρέχονται ασύγχρονες είσοδοι θέσης και μηδένισης.

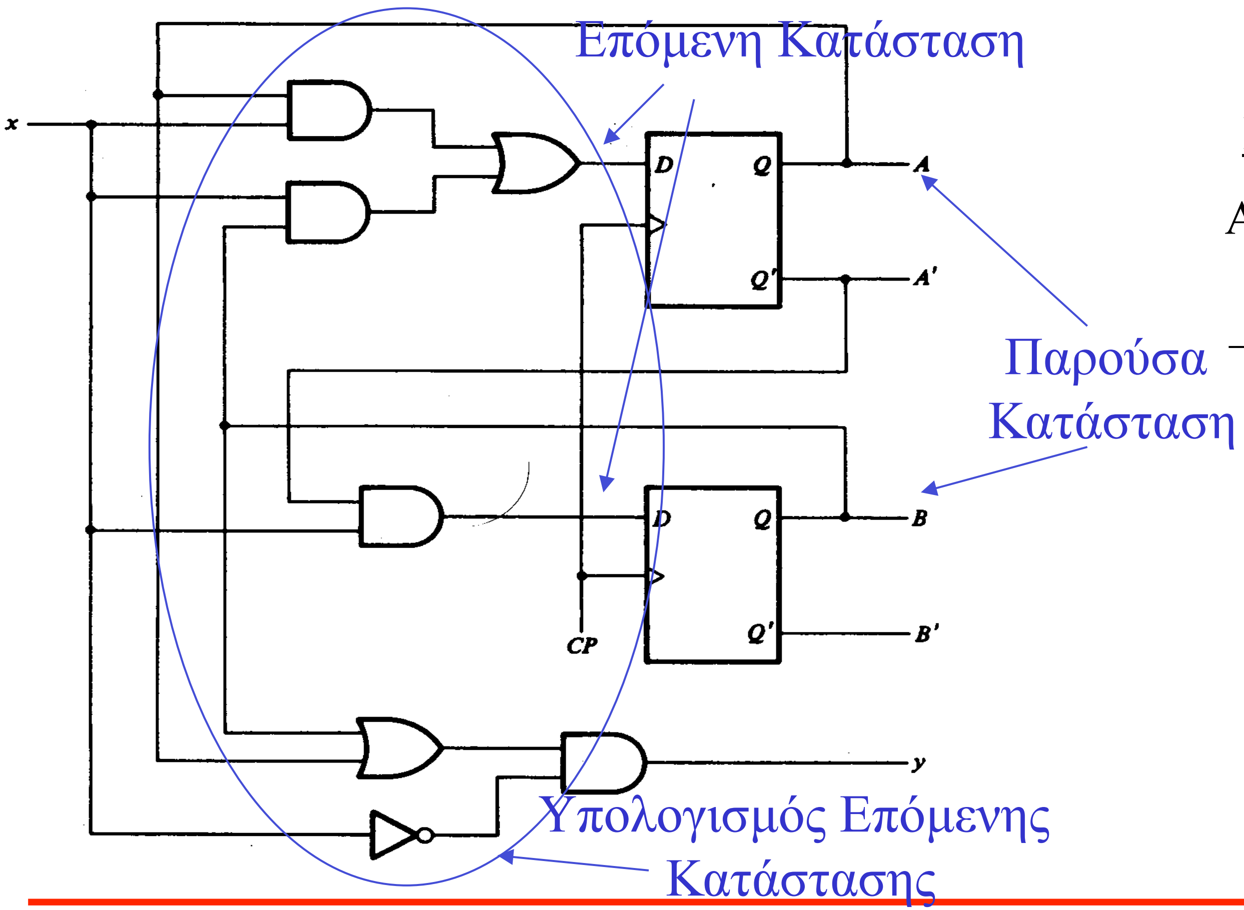


Πίνακας λειτουργίας

Είσοδοι				Έξοδοι	
Μηδενισμού	Ρολόι	*J*	*K*	*Q*	*Q'*
0	*X*	*X*	*X*	0	1
1	↓	0	0	Αμετάβλητες	
1	↓	0	1	0	1
1	↓	1	0	1	0
1	↓	1	1	Αντιστροφή	

Ανάλυση Ακολουθιακών Κυκλωμάτων

Η ανάλυση των ακολουθιακών κυκλωμάτων έγκειται στην εύρεση ενός πίνακα ή διαγράμματος για τη χρονική ακολουθία εισόδων, εξόδων και εσωτερικών καταστάσεων.



Εξισώσεις κατάστασης

$A(t+1)=A(t)x(t)+B(t)x(t)$

$B(t+1)=A'(t)x(t)$

$A(t+1)=Ax+Bx$

$B(t+1)=A'x$

$y(t)=(A+B)x'$

Πίνακας Καταστάσεων

$A(t+1)=Ax+Bx$

$B(t+1)=A'x$

$y(t)=(A+B)x'$

Present State		Input	Next State		Output
A	*B*	*x*	*A*	*B*	*y*
0	0	0	0	0	0
0	0	1	0	1	0
0	1	0	0	0	1
0	1	1	1	1	0
1	0	0	0	0	1
1	0	1	1	0	0
1	1	0	0	0	1
1	1	1	1	0	0

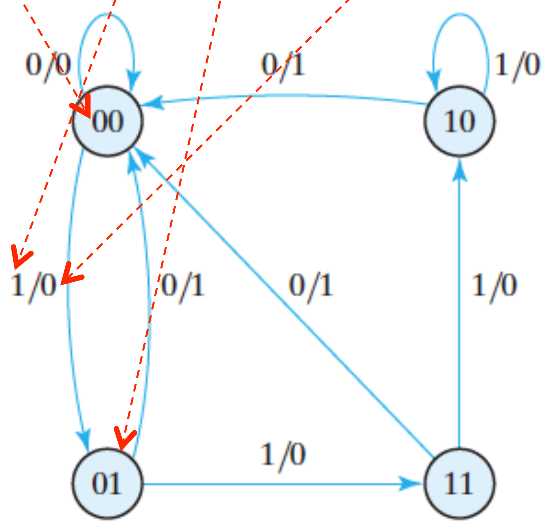
Η μετάβαση καταστάσεων προϋποθέτει την έλευση ακμών ρολογιού οι οποίες όμως δεν συμπεριλαμβάνονται στον πίνακα

Αντιμετωπίζεται όπως το αριστερό μέρος των πινάκων αληθείας

Πίνακας Καταστάσεων

Present State		Next State				Output	
		$x = 0$		$x = 1$		$x = 0$	$x = 1$
A	*B*	*A*	*B*	*A*	*B*	*y*	*y*
0	0	0	0	0	1	0	0
0	1	0	0	1	1	1	0
1	0	0	0	1	0	1	0
1	1	0	0	1	0	1	0

Εναλλακτική μορφή του Πίνακα Καταστάσεων

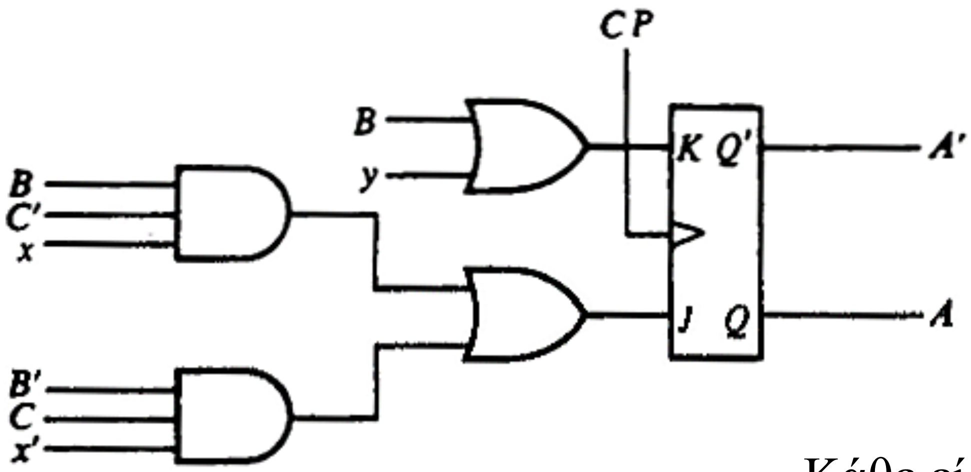


Οι πληροφορίες που περιέχονται στον πίνακα καταστάσεων μπορούν να παρασταθούν σχηματικά με το διάγραμμα καταστάσεων.

Χρήση σε HDLs

Συναρτήσεις εισόδου των Flip Flops

Ένα ακολουθιακό κύκλωμα μπορεί να περιγραφτεί από τις συναρτήσεις εξόδου καθώς και τις συναρτήσεις εισόδων των flip flops.



$J_A = BC'x + B'Cx'$

$K_A = B + y$

Κάθε είσοδος ενός FF ονομάζεται όπως και η έξοδος

Με τον ίδιο τρόπο προσπαθούμε να σχεδιάσουμε ένα ακολουθιακό κύκλωμα

Ανάλυση Ακολουθιακού Κυκλώματος

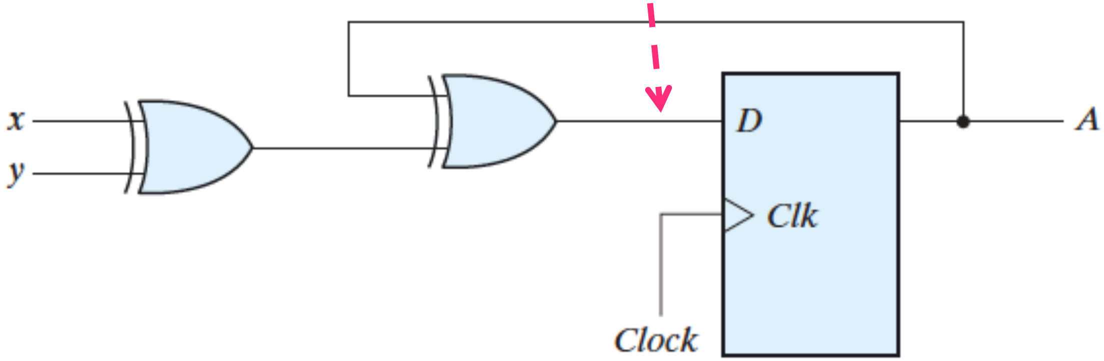
Ανάλυση ακολουθιακού κυκλώματος:

1. Υπολογισμός των δυαδικών τιμών κάθε συνάρτησης εισόδου flip flop με την παρούσα κατάσταση και τις μεταβλητές εισόδου.

2. Χρήση του χαρακτηριστικού πίνακα για καθορισμό της επόμενης κατάστασης.

Κύκλωμα με D flip flop

$D_A = A \text{ xor } x \text{ xor } y$



(a) Circuit diagram

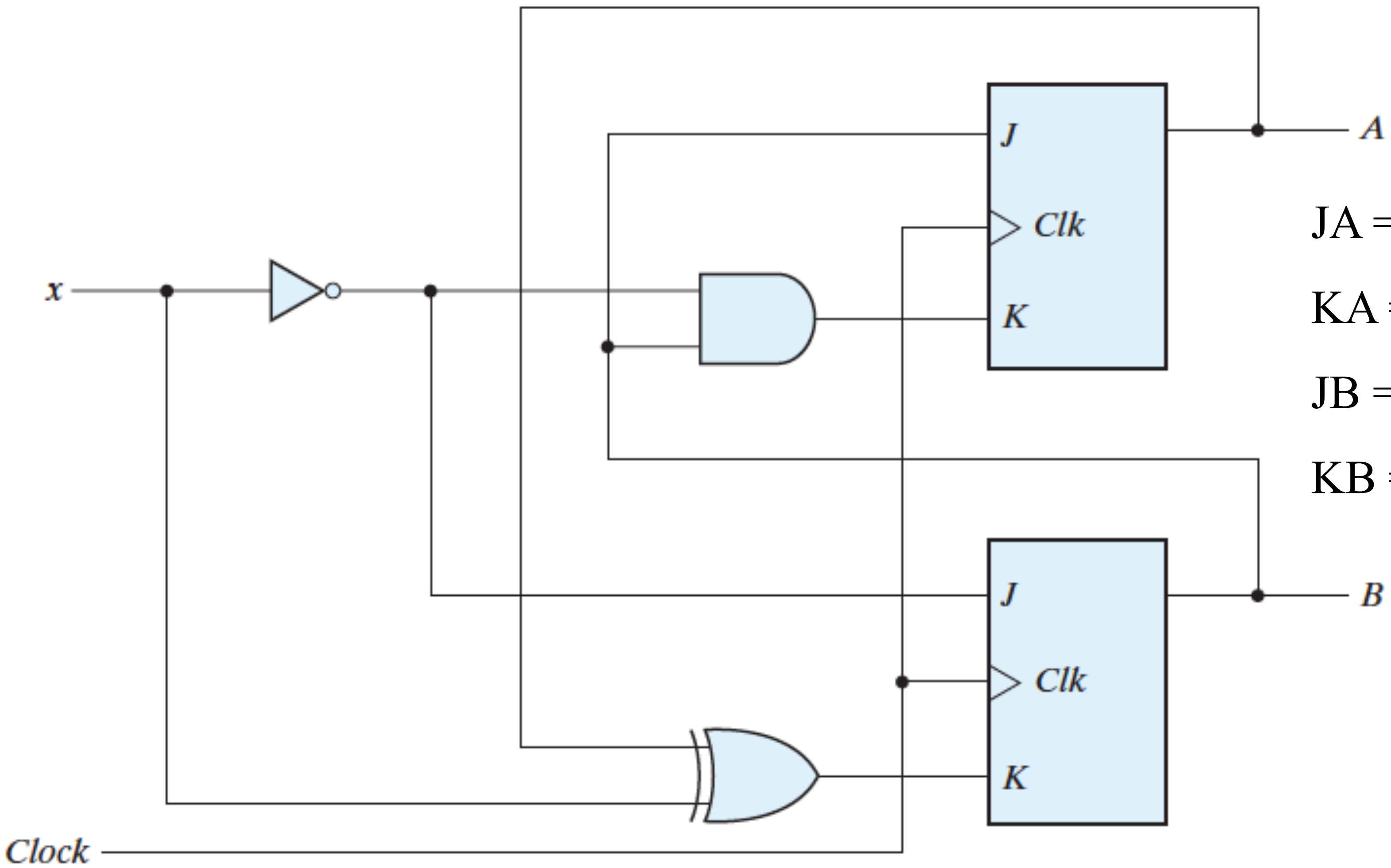
Present state	Inputs		Next state
A	x	y	A
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	1

(b) State table

Η επόμενη κατάσταση ταυτίζεται με την είσοδο D_A

(c) State diagram

Ακολουθιακό κύκλωμα με JK FFs



$JA = B$

$KA = Bx'$

$JB = x'$

$KB = A \oplus x$

Παράδειγμα: Ακολουθιακό κύκλωμα με JK FFs

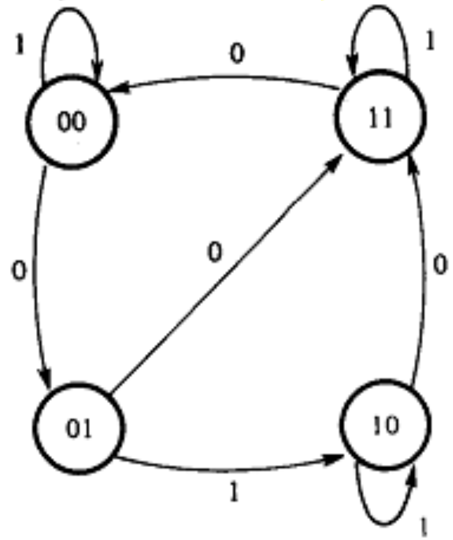
Παρούσα κατάσταση		Είσοδος	Επόμενη κατάσταση		Είσοδοι των flip-flops			
A	*B*	*x*	*A*	*B*	*JA*	*KA*	*JB*	*KB*
0	0	0	0	1	0	0	1	0
0	0	1	0	0	0	0	0	1
0	1	0	1	1	1	1	1	0
0	1	1	1	0	1	0	0	1
1	0	0	1	1	0	0	1	1
1	0	1	1	0	0	0	0	0
1	1	0	0	0	1	1	1	1
1	1	1	1	1	1	0	0	0

$JA = B$

$KA = Bx'$

$JB = x'$

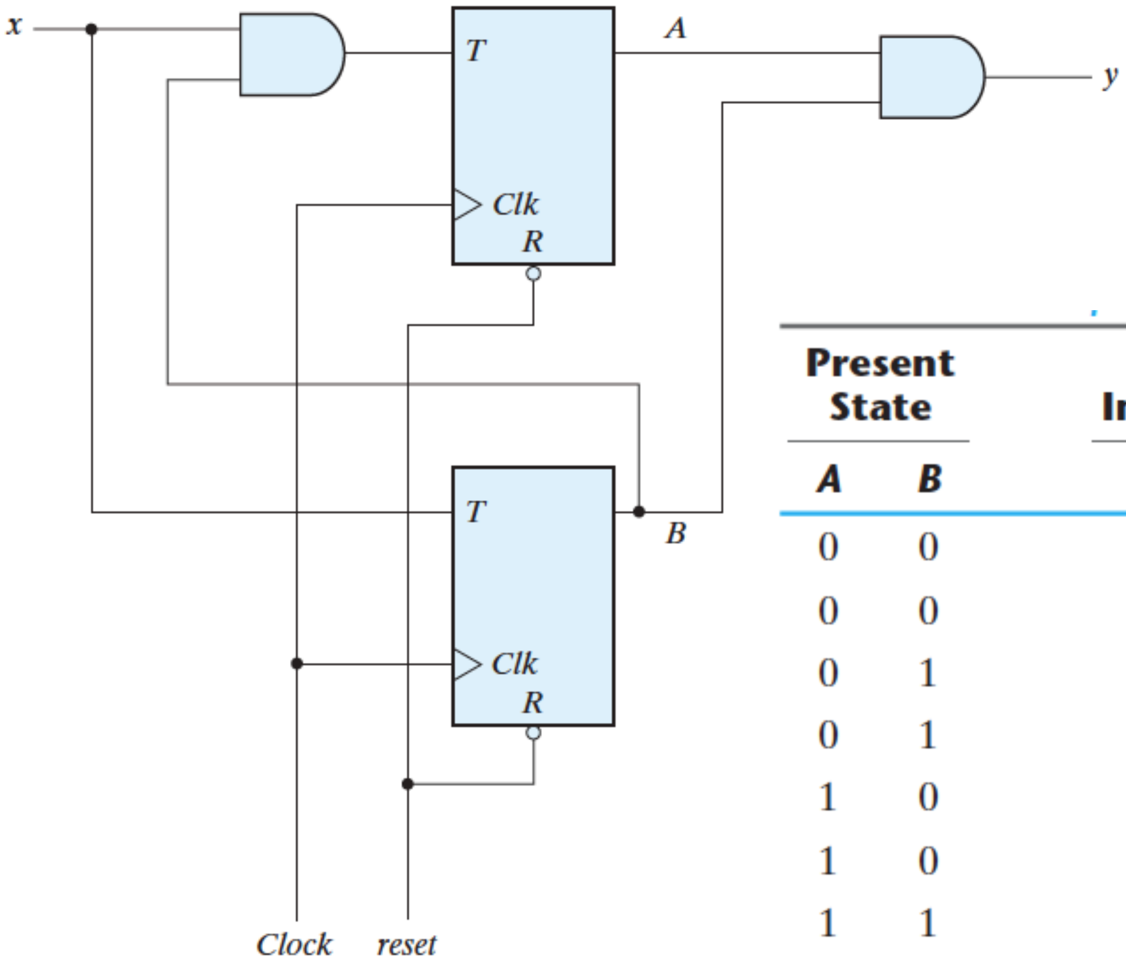
$KB = A \oplus x$



***JK* Flip-Flop**

J	*K*	$Q(t+1)$	
0	0	$Q(t)$	Αμετάβλητη
0	1	0	Επαναφορά
1	0	1	Θέση
1	1	$Q'(t)$	Συμπλήρωμα

Ακολουθιακό κύκλωμα με T FFs



(a) Circuit diagram

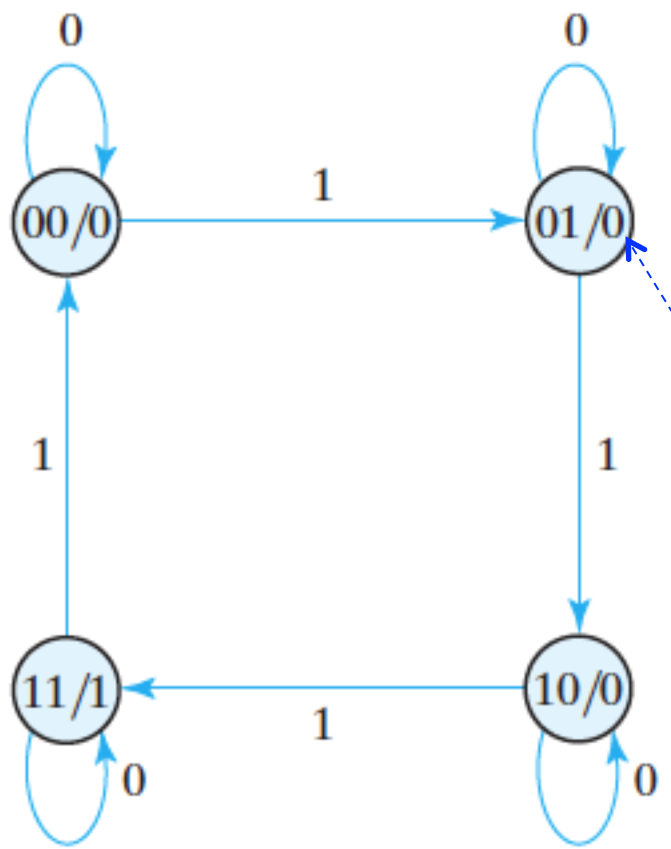
$T_A = Bx$

$T_B = x$

$y = AB$

Present State		Input	Next State		Output
A	***B***	***x***	***A***	***B***	***y***
0	0	0	0	0	0
0	0	1	0	1	0
0	1	0	0	1	0
0	1	1	1	0	0
1	0	0	1	0	0
1	0	1	1	1	0
1	1	0	1	1	1
1	1	1	0	0	1

Ακολουθιακό κύκλωμα με T FFs

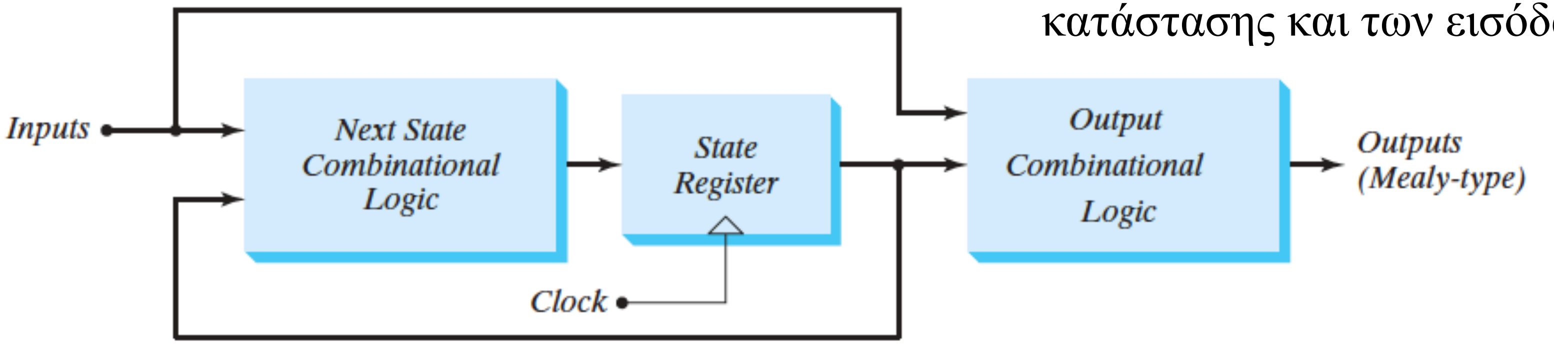


Προσοχή:

η έξοδος δηλώνεται και εσωτερικά στην κάθε κατάσταση

Μοντέλα Mealy και Moore

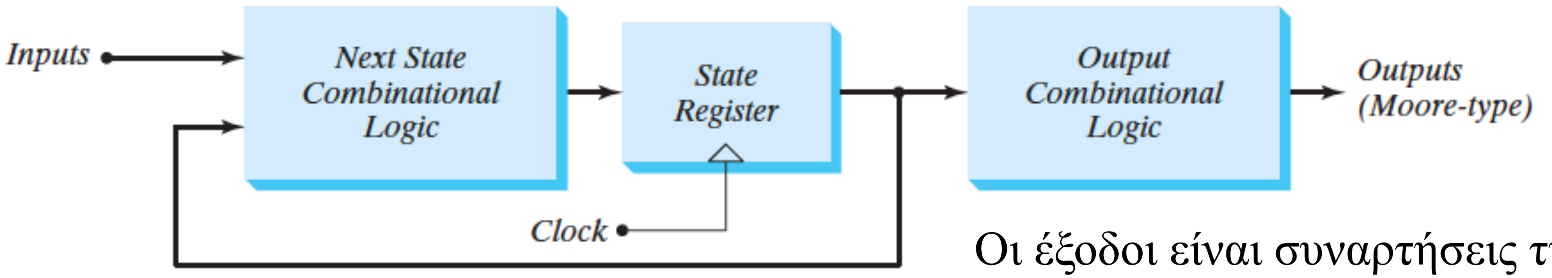
Mealy Machine



(a)

Οι έξοδοι είναι συναρτήσεις της παρούσας κατάστασης και των εισόδων.

Moore Machine

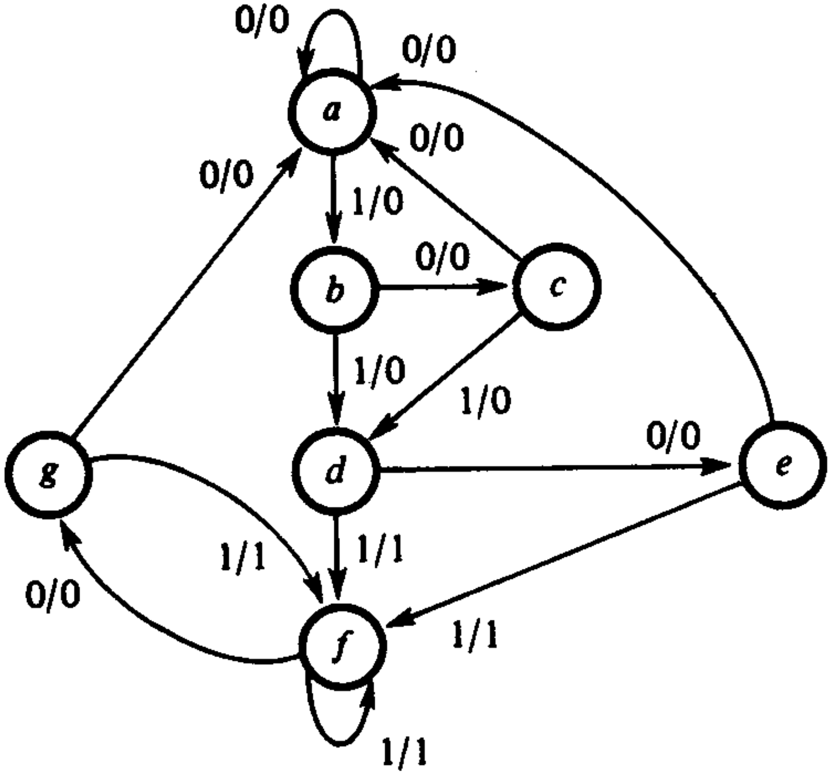


(b)

Οι έξοδοι είναι συναρτήσεις της παρούσας κατάστασης μόνο.

Ελαχιστοποίηση Καταστάσεων

Ελαχιστοποίηση του κυκλώματος: ελαχιστοποίηση πυλών και αριθμού flip flops (ή αλλιώς αριθμού καταστάσεων). Οι αλγόριθμοι ελαχιστοποιούν τις εσωτερικές καταστάσεις χωρίς να αλλάζουν τις προδιαγραφές εισόδου/εξόδου.



Κατάσταση	a a b c d e f f g f g a
Είσοδος	0 1 0 1 0 1 1 0 1 0 0
Έξοδος	0 0 0 0 0 1 1 0 1 0 0

Οι εσωτερικές καταστάσεις είναι αδιάφορες. Στόχος είναι να διατηρηθεί ίδια η ακολουθία εισόδων-εξόδων.

Ελαχιστοποίηση Καταστάσεων

Κανόνας: Δύο καταστάσεις είναι ισοδύναμες αν για κάθε είσοδο δίνουν ακριβώς την ίδια έξοδο και στέλνουν το κύκλωμα στην ίδια ή σε ισοδύναμη κατάσταση. Όταν δύο καταστάσεις είναι ισοδύναμες τότε η μία από τις δύο μπορεί να απαλειφθεί.

Παρούσα κατάσταση	Επόμενη κατάσταση		Έξοδος	
	$x=0$	$x=1$	$x=0$	$x=1$
a	a	b	0	0
b	c	d	0	0
c	a	d	0	0
d	e	f	0	1
e	a	f	0	1
f	g	f	0	1
g	a	f	0	1

Παρούσα κατάσταση	Επόμενη κατάσταση		Έξοδος	
	$x=0$	$x=1$	$x=0$	$x=1$
a	a	b	0	0
b	c	d	0	0
c	a	d	0	0
d	e	~~f~~ d	0	1
e	a	~~f~~ d	0	1
~~f~~	~~g~~ e	~~f~~	~~0~~	~~1~~
~~g~~	~~a~~	~~f~~	~~0~~	~~1~~

Αντικατάσταση της g με την ισοδύναμη της

Ελαχιστοποίηση Καταστάσεων

Παρούσα κατάσταση	Επόμενη κατάσταση $x=0$	Επόμενη κατάσταση $x=1$	Έξοδος $x=0$	Έξοδος $x=1$
a	a	b	0	0
b	c	d	0	0
c	a	d	0	0
d	e	~~f~~d	0	1
e	a	~~f~~d	0	1
~~f~~	~~g~~e	f	0	1
~~g~~	a	f	0	1

Ελαχιστοποίηση Καταστάσεων

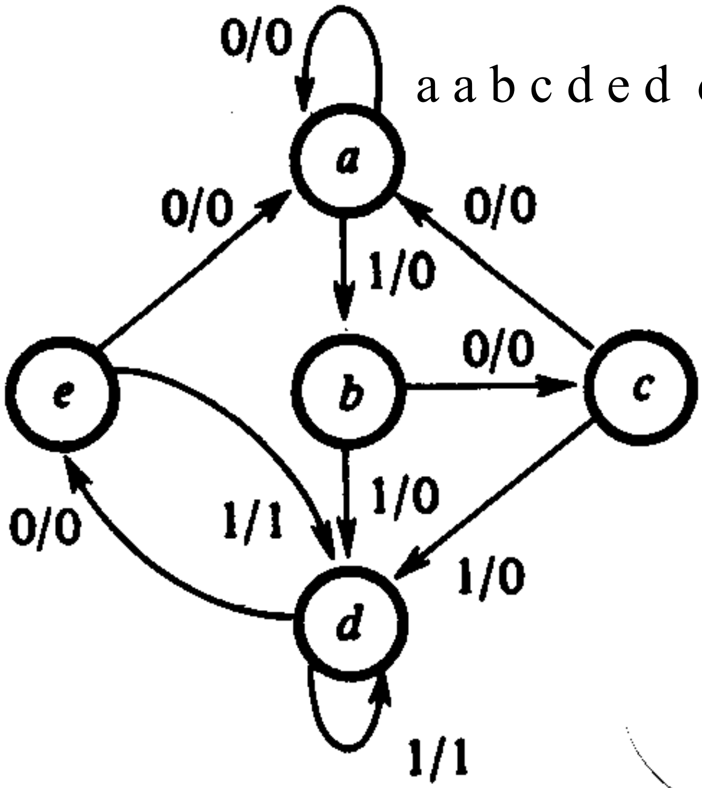
Παρούσα κατάσταση	Επόμενη κατάσταση $x=0$	Επόμενη κατάσταση $x=1$	Έξοδος $x=0$	Έξοδος $x=1$
a	a	b	0	0
b	c	d	0	0
c	a	d	0	0
d	e	d	0	1
e	a	d	0	1

Ελαχιστοποίηση Καταστάσεων

Κατάσταση	a a b c d e f f g f g a
Είσοδος	0 1 0 1 0 1 1 0 1 0 0
Έξοδος	0 0 0 0 0 1 1 0 1 0 0

Κατάσταση	a a b c d e d d e d e a
Είσοδος	0 1 0 1 0 1 1 0 1 0 0
Έξοδος	0 0 0 0 0 1 1 0 1 0 0

a a b c d e d d e d e a



Η μείωση των εσωτερικών καταστάσεων μπορεί να οδηγήσει σε μείωση του αριθμού των flip flops αλλά και απλοποίηση των συνδυαστικών κυκλωμάτων αφού οι αχρησιμοποίητες καταστάσεις ισοδυναμούν με αδιάφορους όρους.

Κωδικοποίηση Καταστάσεων

Κάθε κατάσταση πρέπει να έχει διακριτή τιμή

Σε κυκλώματα που δεν μας ενδιαφέρουν οι εσωτερικές καταστάσεις (οι έξοδοι δεν οδηγούνται κατευθείαν από αυτές) μπορούμε να τις κωδικοποιήσουμε με δυαδικά ψηφία όπως θέλουμε για να ελαχιστοποιήσουμε το κόστος του κυκλώματος.

Μεγάλη εφαρμογή σε περιγραφές υψηλού επιπέδου

Κωδικοποίηση Καταστάσεων

Present State	Next State $x = 0$	Next State $x = 1$	Output $x = 0$	Output $x = 1$
a	a	b	0	0
b	c	d	0	0
c	a	d	0	0
d	e	d	0	1
e	a	d	0	1

Πόσα δυαδικά ψηφία χρειαζόμαστε;

Επηρεάζει ο αριθμός των ψηφίων το κύκλωμα;

Διαφορετική ανάθεση συνδυασμών επηρεάζει το κύκλωμα;

Γνωστές Κωδικοποιήσεις

State	Assignment 1, Binary	Assignment 2, Gray Code	Assignment 3, One-Hot
a	000	000	00001
b	001	001	00010
c	010	011	00100
d	011	010	01000
e	100	110	10000

Present State	Next State		Output	
	$x = 0$	$x = 1$	$x = 0$	$x = 1$
000	000	001	0	0
001	010	011	0	0
010	000	011	0	0
011	100	011	0	1
100	000	011	0	1

Σχεδίαση Ακολουθιακών Κυκλωμάτων

Είναι η αντίστροφη διαδικασία της ανάλυσης.

Θέτουμε κάποιες προδιαγραφές για ένα κύκλωμα (πχ διάγραμμα καταστάσεων)

Ελαχιστοποιούμε το πλήθος των καταστάσεων

Αναθέτουμε δυαδικές τιμές στις καταστάσεις

Φτιάχνουμε τον κωδικοποιημένο πίνακα καταστάσεων

Απλοποιούμε τις συναρτήσεις κατάστασης και εξόδου

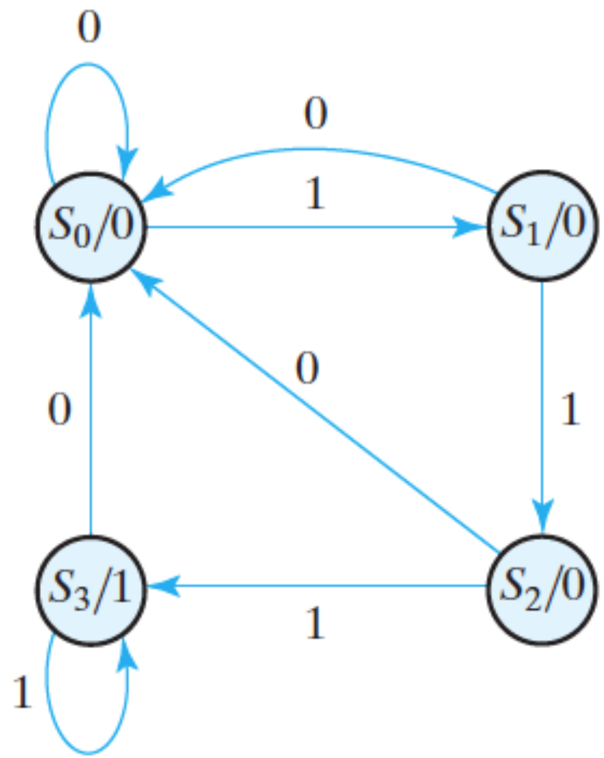
Σχεδιάζουμε το λογικό διάγραμμα

Αυτοματοποιημένα βήματα με αλγόριθμους σύνθεσης

Παράδειγμα με D-FF

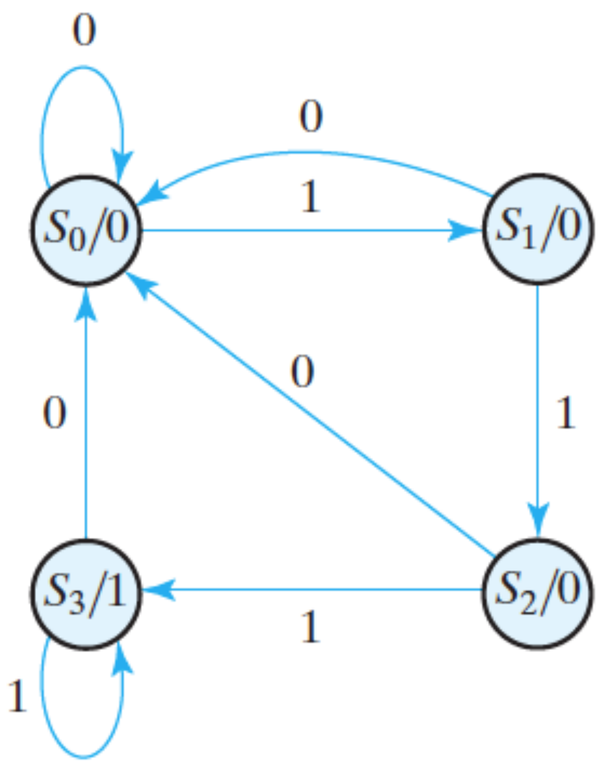
<u>Προδιαγραφές</u>:

Ζητείται κύκλωμα που θα ανιχνεύει μία ακολουθία τριών ή περισσοτέρων μονάδων σε μία σειριακή ακολουθία δυαδικών ψηφίων.



Παράδειγμα με D-FF

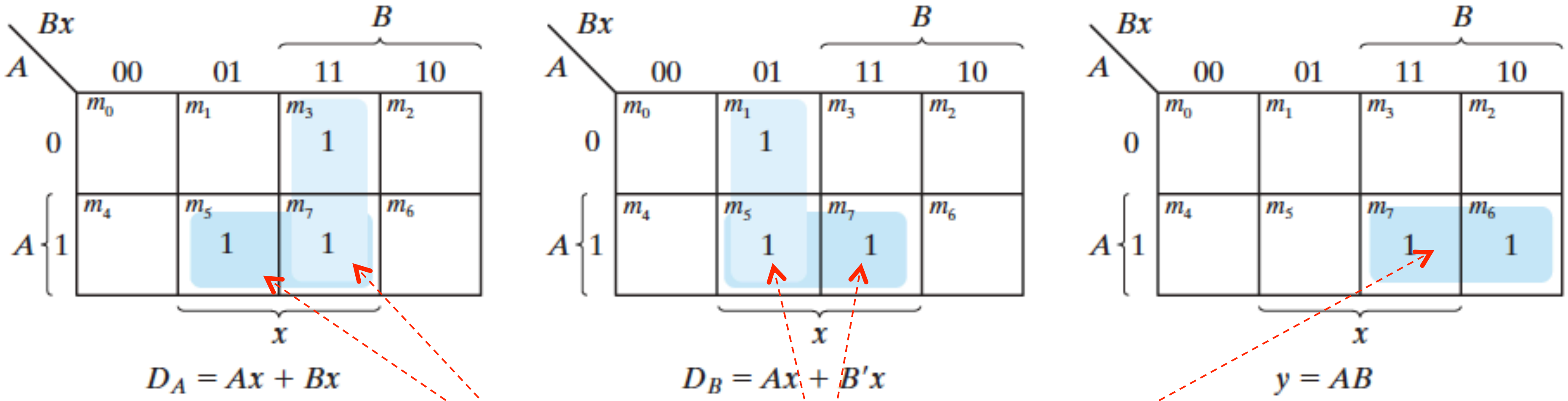
Έχουμε 4 καταστάσεις και τις κωδικοποιούμε με 4 δυαδικές λέξεις των 2 ψηφίων η κάθε μία: S_0:00, S_1:01, S_2:10, S_3:11



Present State		Input	Next State		Output
A	*B*	*x*	*A*	*B*	*y*
0	0	0	0	0	0
0	0	1	0	1	0
0	1	0	0	0	0
0	1	1	1	0	0
1	0	0	0	0	0
1	0	1	1	1	0
1	1	0	0	0	1
1	1	1	1	1	1

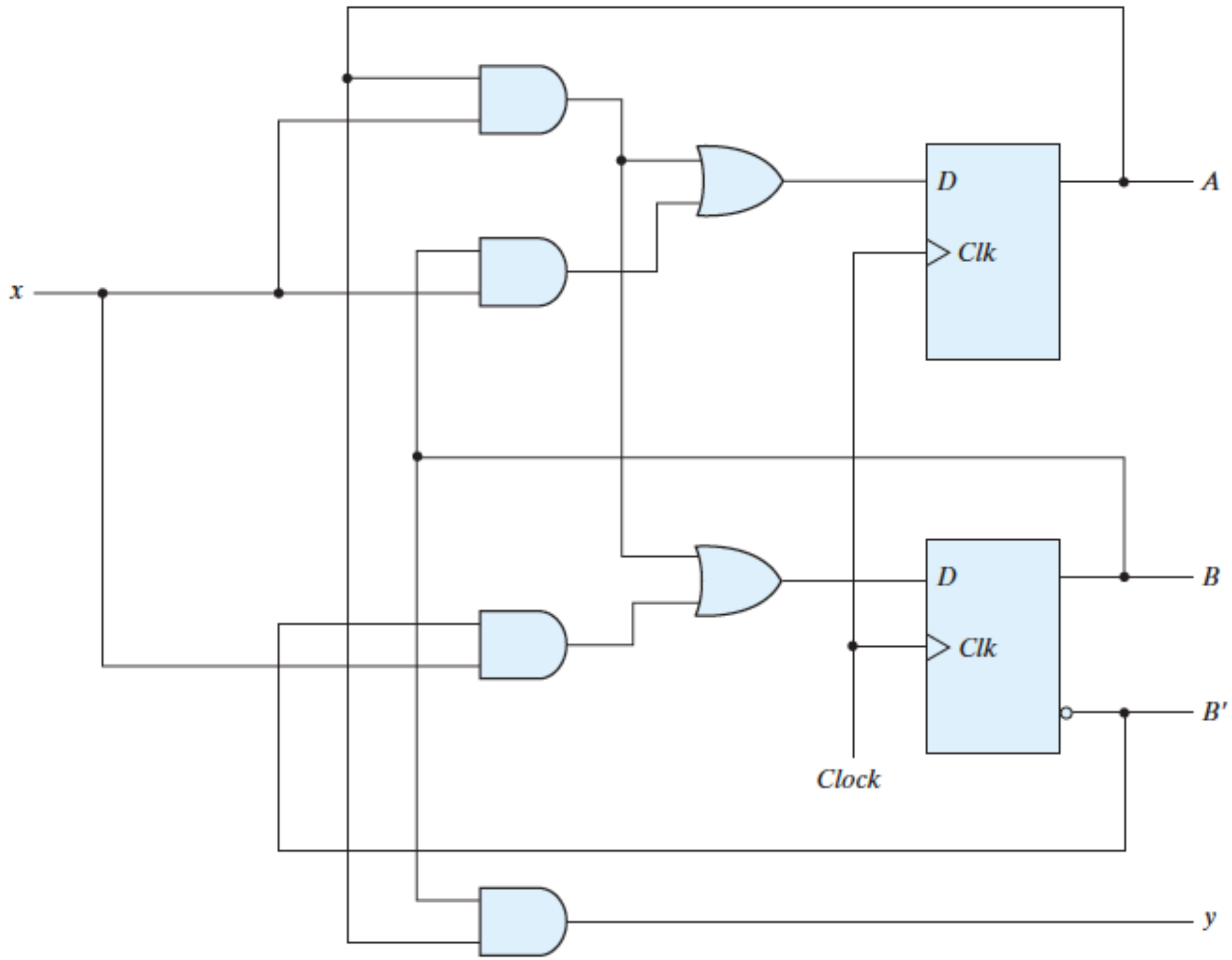
Παράδειγμα με D-FF

Στην περίπτωση των D-FF η επόμενη κατάσταση ταυτίζεται με την τιμή της εισόδου πριν την έλευση της ακμής του ρολογιού



Στους χάρτες τοποθετούμε τις τιμές των εισόδων των D-FF που ταυτίζονται με τις τιμές της επόμενης κατάστασης.

Παράδειγμα με D-FF



Πίνακες Διέγερσης των flip flops

Όταν χρησιμοποιούνται RS, T, JK Flip Flop η διαδικασία σχεδίασης είναι πιο περίπλοκη καθώς οι συναρτήσεις εισόδων πρέπει να υπολογιστούν από την μετάβαση μεταξύ καταστάσεων

JK Flip-Flop

J	K	$Q(t+1)$	
0	0	$Q(t)$	Αμετάβλητη
0	1	0	Επαναφορά
1	0	1	Θέση
1	1	$Q'(t)$	Συμπλήρωμα

<u>Πίνακας Διέγερσης (Excitation Table)</u>: Πίνακας που δίνει τις απαιτούμενες εισόδους για ορισμένη αλλαγή της κατάστασης.

$Q(t)$	$Q(t+1)$	J	K
0	0	0	X
0	1	1	X
1	0	X	1
1	1	X	0

Πίνακες Διέγερσης των flip flops

Πίνακας Διέγερσης (Excitation Table): Πίνακας που δίνει τις απαιτούμενες εισόδους για ορισμένη αλλαγή της κατάστασης.

$Q(t)$	$Q(t+1)$	S	R
0	0	0	X
0	1	1	0
1	0	0	1
1	1	X	0

(α) RS

$Q(t)$	$Q(t+1)$	J	K
0	0	0	X
0	1	1	X
1	0	X	1
1	1	X	0

(β) JK

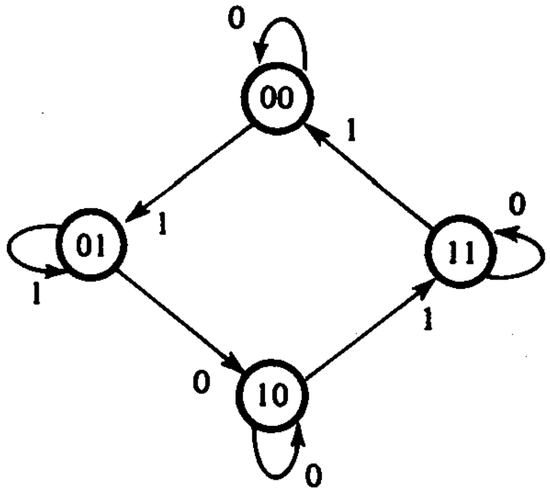
$Q(t)$	$Q(t+1)$	D
0	0	0
0	1	1
1	0	0
1	1	1

(γ) D

$Q(t)$	$Q(t+1)$	T
0	0	0
0	1	1
1	0	1
1	1	0

(δ) T

Παράδειγμα Σχεδίασης με JK FF

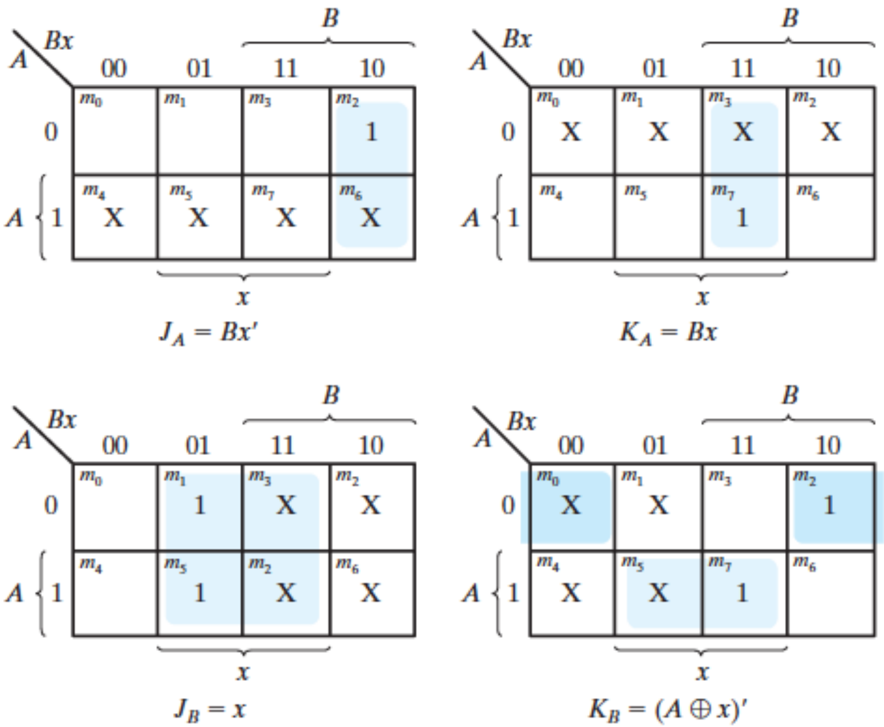


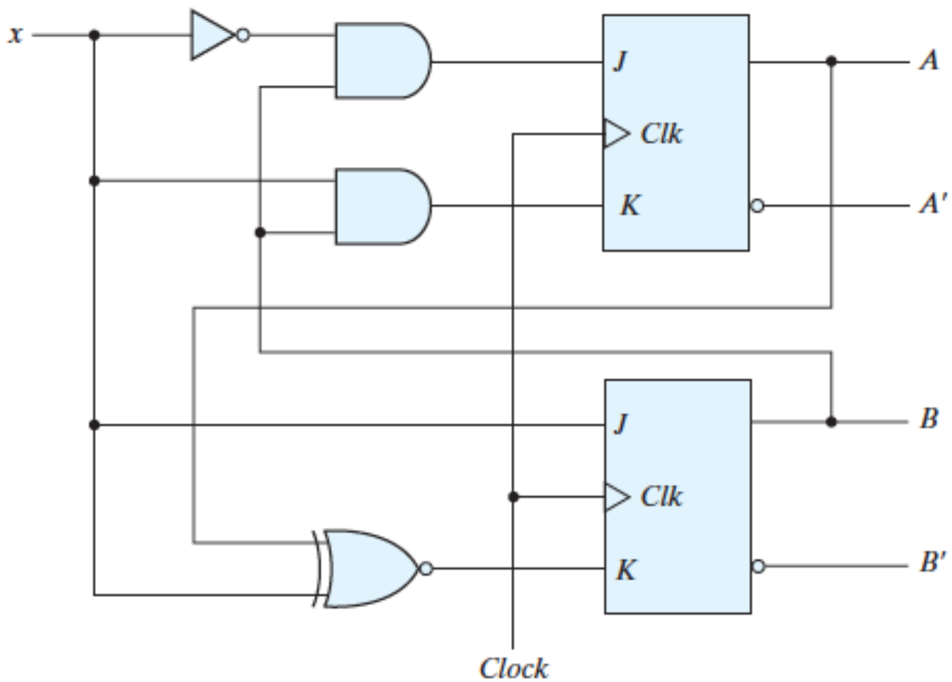
Παρούσα κατάσταση		Επόμενη κατάσταση			
		$x = 0$		$x = 1$	
A	*B*	*A*	*B*	*A*	*B*
0	0	0	0	0	1
0	1	1	0	0	1
1	0	1	0	1	1
1	1	1	1	0	0

Παράδειγμα Σχεδίασης με JK FF

Είσοδοι του συνδυαστικού κυκλώματος		
Παρούσα κατάσταση		Είσοδος
A	*B*	*x*
0	0	0
0	0	1
0	1	0
0	1	1
1	0	0
1	0	1
1	1	0
1	1	1

Παράδειγμα Σχεδίασης με JK FF

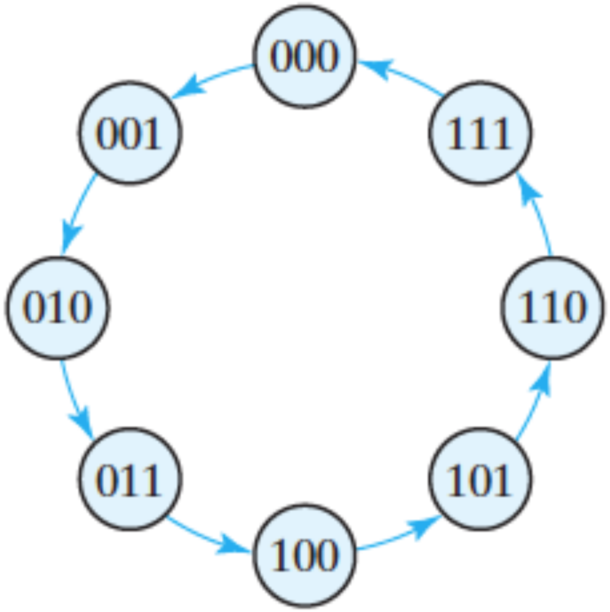




Σχεδίαση με T Flip Flops

Προδιαγραφές: Ζητείται μετρητής από το 000...111

Μετρητής: κύκλωμα που διέρχεται από όλες τις καταστάσεις 000, 001, 010, 011, 100, 101, 110, 111, 000, ...

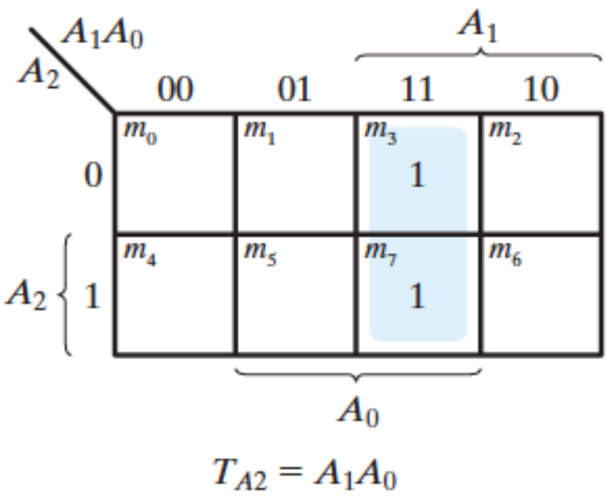


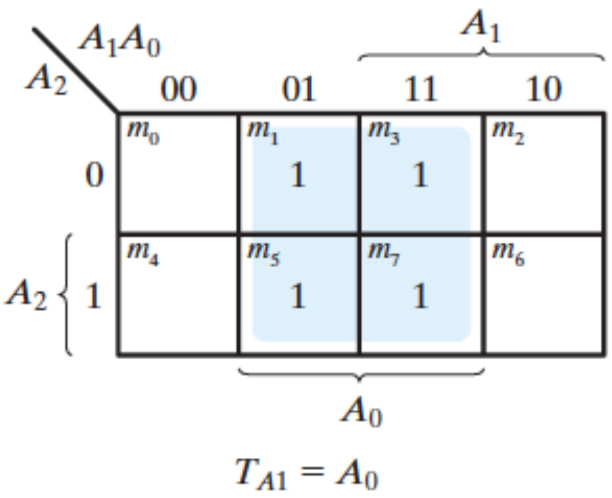
Σχεδίαση με T Flip Flops

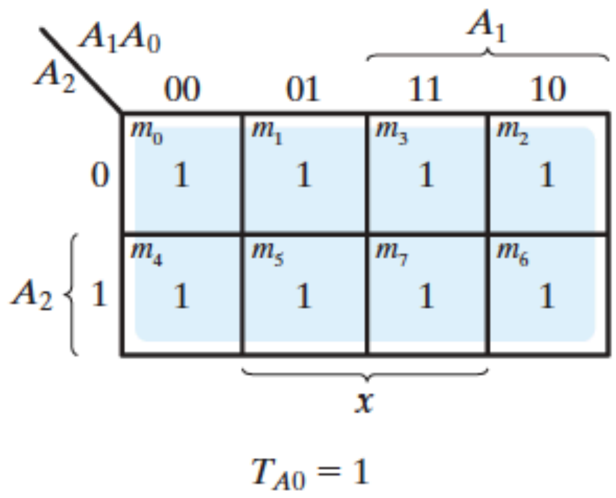
Present State			Next State			Flip-Flop Inputs		
A_2	A_1	A_0	A_2	A_1	A_0	T_{A2}	T_{A1}	T_{A0}
0	0	0	0	0	1	0	0	1
0	0	1	0	1	0	0	1	1
0	1	0	0	1	1	0	0	1
0	1	1	1	0	0	1	1	1
1	0	0	1	0	1	0	0	1
1	0	1	1	1	0	0	1	1
1	1	0	1	1	1	0	1	1
1	1	1	0	0	0	1	1	1

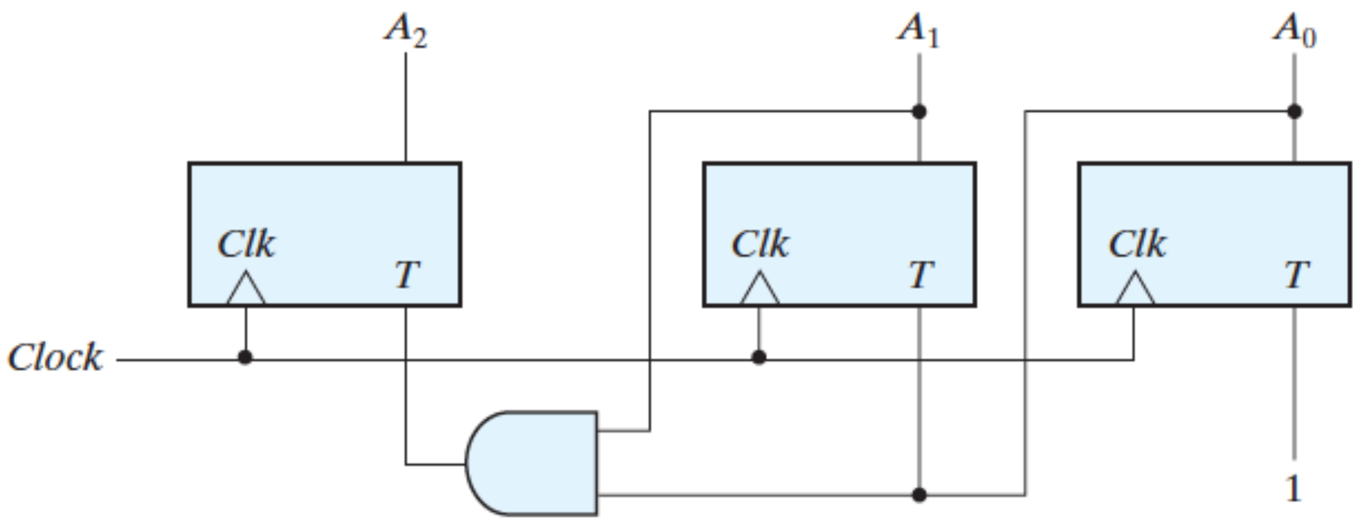
Οι αχρησιμοποίητες καταστάσεις αποτελούν αδιάφορους όρους.

Σχεδίαση με T Flip Flops









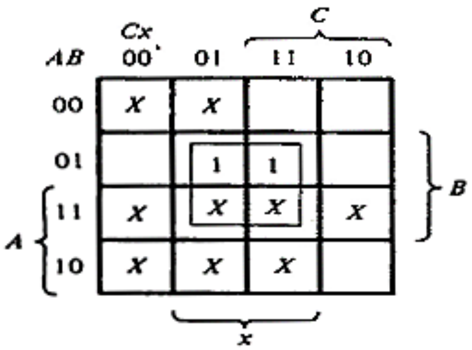
Σχεδίαση με Αχρησιμοποίητες Καταστάσεις

Οι αχρησιμοποίητες καταστάσεις αποτελούν αδιάφορους όρους.

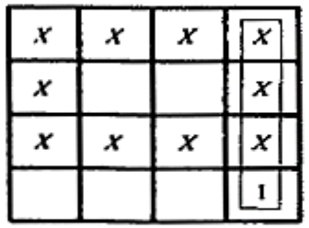
<u>Παράδειγμα</u>: Υλοποίηση ακολουθιακού κυκλώματος με RS FFs

Παρούσα κατάσταση			Είσοδος	Επόμενη κατάσταση			Είσοδοι των flip-flop						Έξοδος
A	*B*	*C*	*x*	*A*	*B*	*C*	*SA*	*RA*	*SB*	*RB*	*SC*	*RC*	
0	0	1	0	0	0	1	0	*X*	0	*X*	*X*	0	0
0	0	1	1	0	1	0	0	*X*	1	0	0	1	0
0	1	0	0	0	1	1	0	*X*	*X*	0	1	0	0
0	1	0	1	1	0	0	1	0	0	1	0	*X*	0
0	1	1	0	0	0	1	0	*X*	0	1	*X*	0	0
0	1	1	1	1	0	0	1	0	0	1	0	1	0
1	0	0	0	1	0	1	*X*	0	0	*X*	1	0	0
1	0	0	1	1	0	0	*X*	0	0	*X*	0	*X*	1
1	0	1	0	0	0	1	0	1	0	*X*	*X*	0	0
1	0	1	1	1	0	0	*X*	0	0	*X*	0	1	1

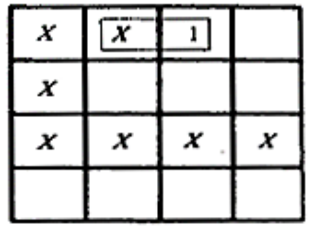
Παράδειγμα



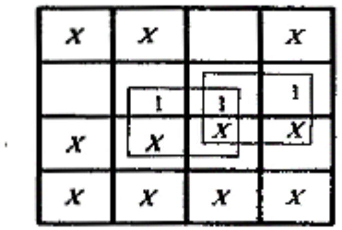
$SA = Bx$



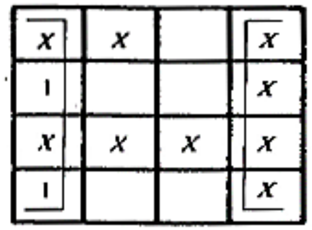
$RA = Cx'$



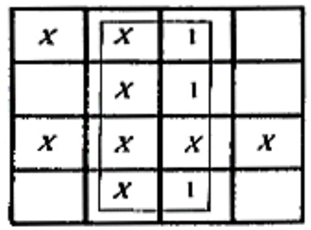
$SB = A'B'x$



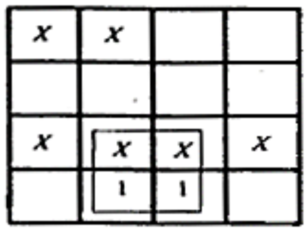
$RB = BC + Bx$



$SC = x'$

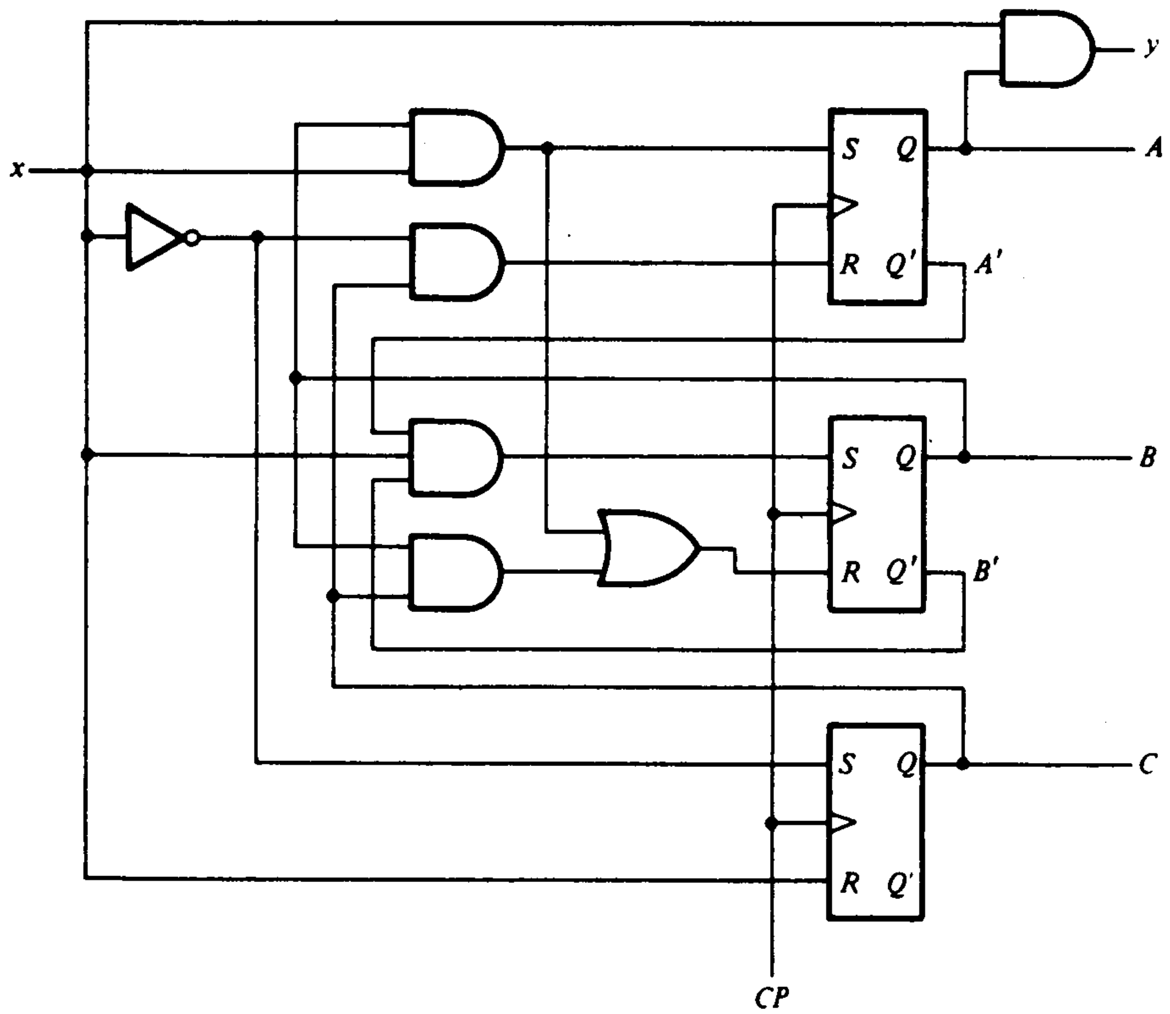


$RC = x$



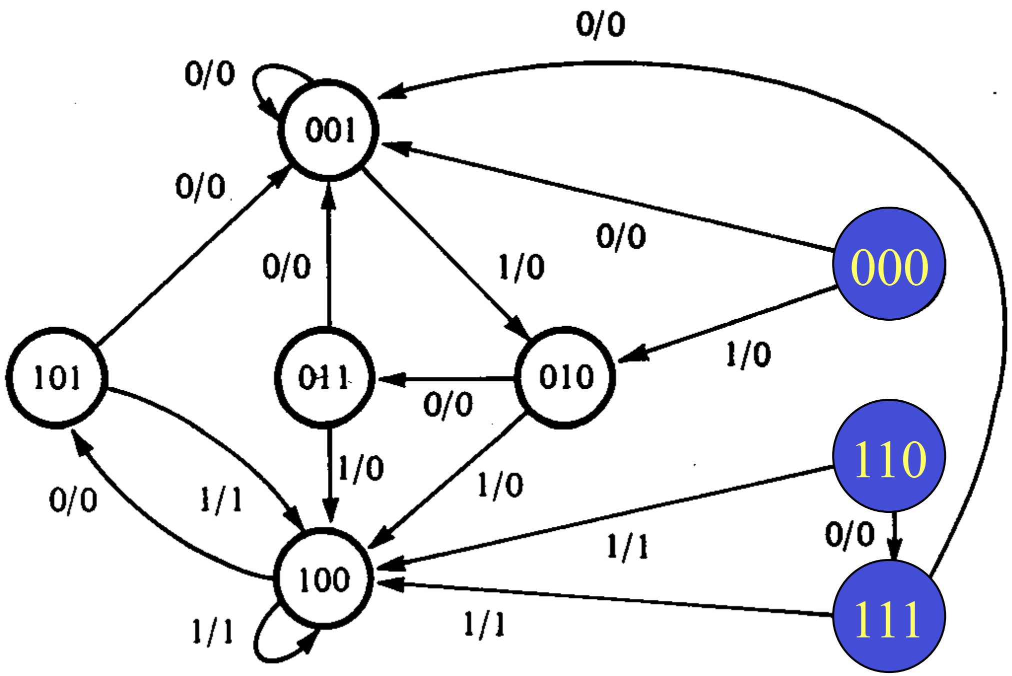
$y = Ax$

Παράδειγμα



Σχεδίαση με Αχρησιμοποίητες Καταστάσεις

Το κύκλωμα δεν πρέπει να βρεθεί σε μία από τις αχρησιμοποίητες καταστάσεις του γιατί τότε θα έχει απροσδιόριστη συμπεριφορά. Έτσι αρχικά τα Flip Flops αρχικοποιούνται σε προκαθορισμένη κατάσταση.



Πρέπει να εξασφαλίζουμε ότι δεν υπάρχει περίπτωση να ταλαντεύεται το κύκλωμα ανάμεσα σε δύο ή περισσότερες αχρησιμοποίητες καταστάσεις με κίνδυνο να μην μπορεί να εξέλθει από αυτές.

Π.χ αχρησ.: 000, 110, 111

Αν συμβεί αυτό τότε ξανασχεδιάζουμε το κύκλωμα έτσι ώστε να σπάσουμε τους κύκλους.

Σχεδίαση με Αχρησιμοποίητες Καταστάσεις

Αποφυγή άκυρων καταστάσεων με κύκλους

Όλες οι αδιάφορες καταστάσεις θα πηγαίνουν σε μοναδική επόμενη που θα χειρίζεται το λάθος.

State Machines με χειρισμό λάθους

Όλες οι αδιάφορες καταστάσεις θα πηγαίνουν σε οποιαδήποτε έγκυρη επόμενη.

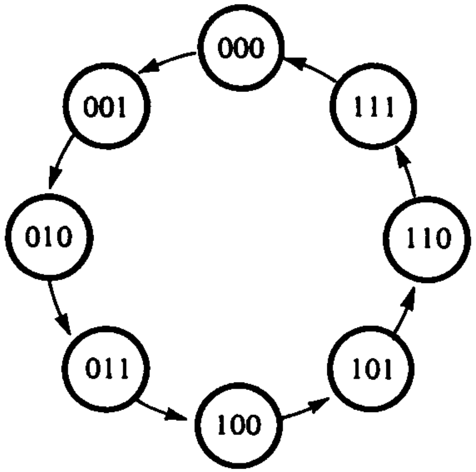
Το κύκλωμα μπορεί να είναι απλούστερο αλλά δεν υπάρχει χειρισμός λάθους

Σχεδίαση με αδιάφορες καταστάσεις και σε περίπτωση κύκλων επανασχεδίαση

Σχεδίαση Μετρητών

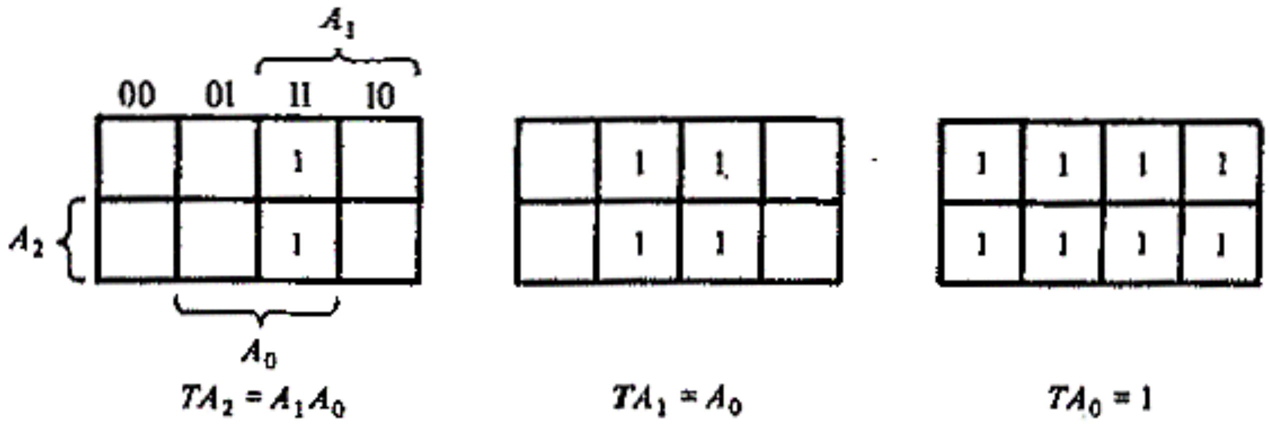
Ένα ακολουθιακό κύκλωμα που περνάει από μία προδιαγεγραμμένη ακολουθία καταστάσεων με απλούς παλμούς ονομάζεται μετρητής. Δεν έχει εισόδους

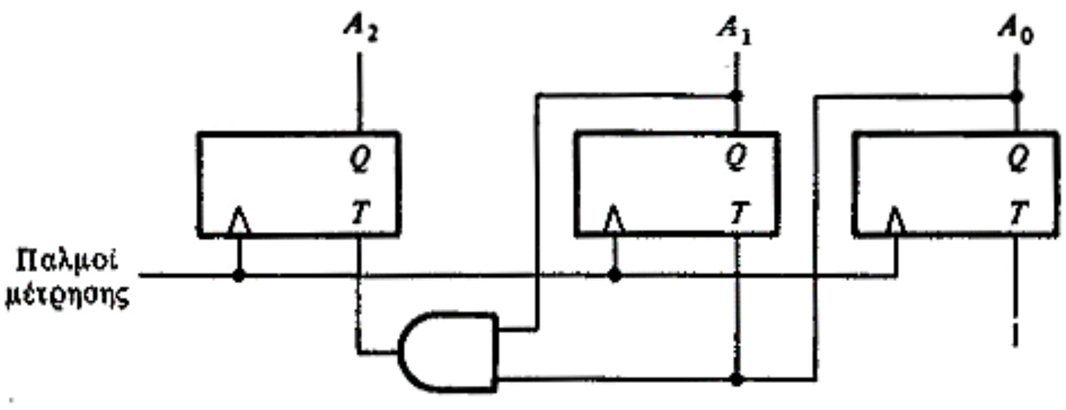
Παράδειγμα: Δυαδικός μετρητής 3 bit



Παρούσα κατάσταση			**Επόμενη κατάσταση**			**Είσοδος flip-flop**		
A_2	A_1	A_0	A_2	A_1	A_0	TA_2	TA_1	TA_0
0	0	0	0	0	1	0	0	1
0	0	1	0	1	0	0	1	1
0	1	0	0	1	1	0	0	1
0	1	1	1	0	0	1	1	1
1	0	0	1	0	1	0	0	1
1	0	1	1	1	0	0	1	1
1	1	0	1	1	1	0	0	1
1	1	1	0	0	0	1	1	1

Παράδειγμα


A1
00 01 11 10
A2
A0
TA2 = A1A0
TA1 = A0
TA0 = 1


A2
A1
A0
Q
T
Παλμοί μέτρησης
1

Παράδειγμα: Μετρητής με μη δυαδική ακολουθία

→ 000 → 001 → 010 → 100 → 101 → 110 →

Παρούσα κατάσταση			Επόμενη κατάσταση			Είσοδοι flip-flop					
A	*B*	*C*	*A*	*B*	*C*	*JA*	*KA*	*JB*	*KB*	*JC*	*KC*
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	1	0	0	1	X	X	1	0	X
1	0	0	1	0	1	X	0	0	X	1	X
1	0	1	1	1	0	X	0	1	X	X	1
1	1	0	0	0	0	X	1	X	0	0	X

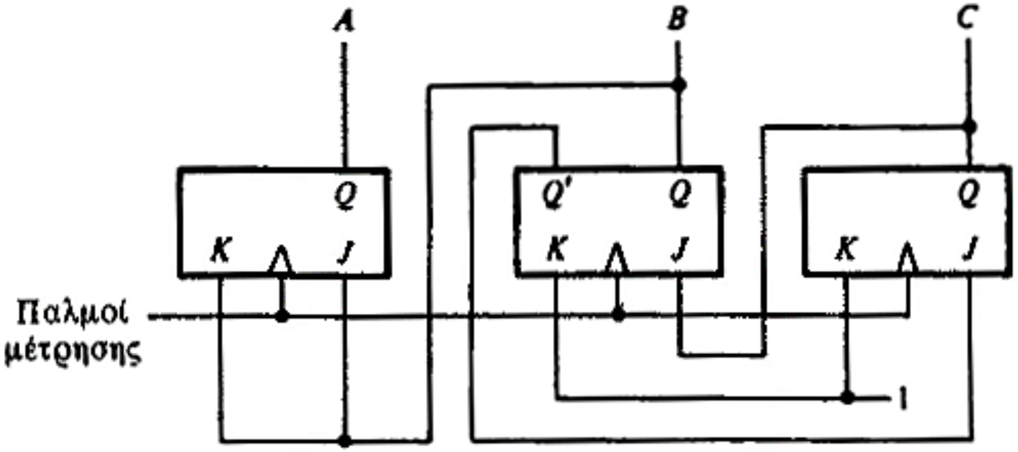
Παράδειγμα: Μετρητής με μη δυαδική ακολουθία

JA = B   KA = B

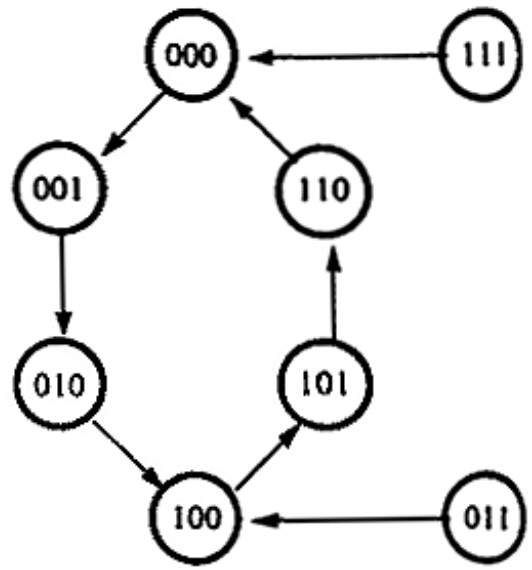
JB = C   KB = 1

JC = B΄   KC = 1

Με ανάλυση εξετάζουμε αχρησιμοποίητες καταστάσεις



(α) Λογικό διάγραμμα του μετρητή



(β) Διάγραμμα καταστάσεων του μετρητή