

Ψηφιακή Σχεδίαση

Διάλεξη 5 – Ακολουθιακά Κυκλώματα

Γεώργιος Κεραμίδας, Επίκουρος Καθηγητής
2^ο Εξάμηνο, Τμήμα Πληροφορικής





Συνδυαστικά vs Ακολουθιακά

- **Συνδυαστικά** : οι έξοδοι εξαρτώνται σε κάθε χρονική στιγμή αποκλειστικά και μόνο από τις εισόδους που εφαρμόζονται. Δεν εξαρτώνται ούτε από τη σειρά εφαρμογής, ούτε από τη κατάσταση του κυκλώματος πριν εφαρμοστούν.
- **Ακολουθιακά** : οι έξοδοι κάθε χρονική στιγμή εξαρτώνται όχι μόνο από τις τιμές των εισόδων εκείνη τη στιγμή αλλά και από τις τιμές των εισόδων όλες τις προηγούμενες χρονικές στιγμές. Οι τελευταίες έχουν επιβάλλει μια κατάσταση στο κύκλωμα.

Συνδυαστικά vs Ακολουθιακά



- Η κατάσταση αποθηκεύεται σε **στοιχεία με ικανότητα μνήμης**. Τα στοιχεία αυτά ονομάζονται **ακολουθιακά στοιχεία**. Ένα κύκλωμα που περιέχει ακολουθιακά στοιχεία είναι ένα ακολουθιακό κύκλωμα.
- Υπάρχουν 2 μεγάλες κατηγορίες ακολουθιακών κυκλωμάτων : **σύγχρονα και ασύγχρονα κυκλώματα**. Στα σύγχρονα οι αλλαγές της κατάστασης γίνονται σε διακριτές στιγμές χρόνου.

Σύγχρονα vs Ασύγχρονα



- **Σύγχρονα**

- Οι τιμές σε συγκεκριμένες (διακριτές) τιμές χρόνου το περιγράφουν πλήρως
- Συγχρονισμός επιτυγχάνεται μέσω ενός σήματος χρονισμού, γνωστό ως ρολόι που έχει μια περιοδική παλμοσειρά.
- Δε παρουσιάζουν αστάθεια.

- **Ασύγχρονα**

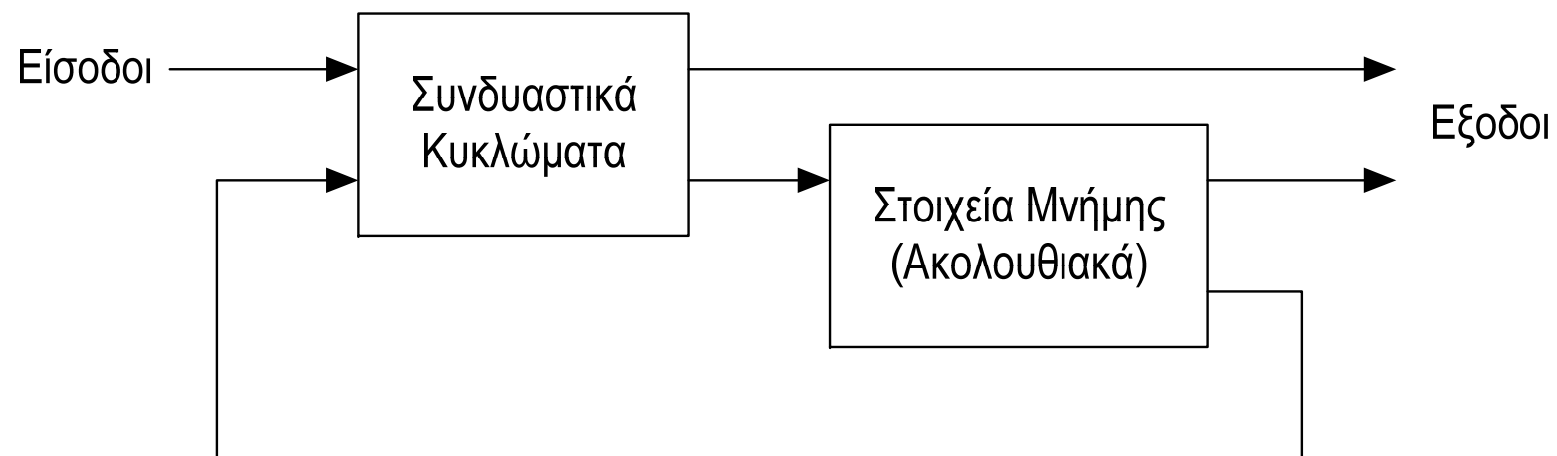
- Οι τιμές εξαρτώνται από τη σειρά εναλλαγής των σημάτων εισόδου. Νέα τιμή μπορεί να προκύψει ανά πάσα χρονική στιγμή.
- Ένα συνδυαστικό κύκλωμα με ανατροφοδότηση (feedback) είναι ένα ασύγχρονο ακολουθιακό κύκλωμα.
- Αστάθεια.

Σύγχρονα vs Ασύγχρονα

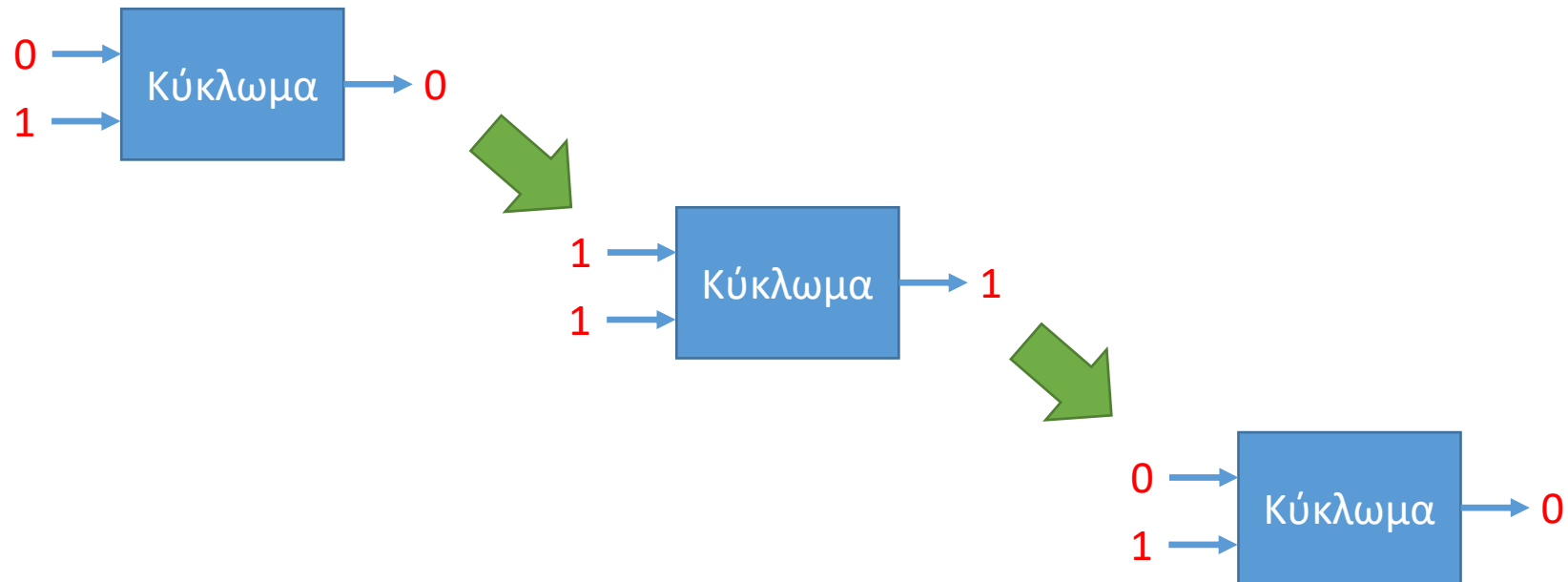


- Θα μελετήσουμε κυρίως τα σύγχρονα κυκλώματα, μιας και τα ασύγχρονα τείνουν να εξαλειφθούν εντελώς από την βιομηχανία και τα εμπορικά προϊόντα

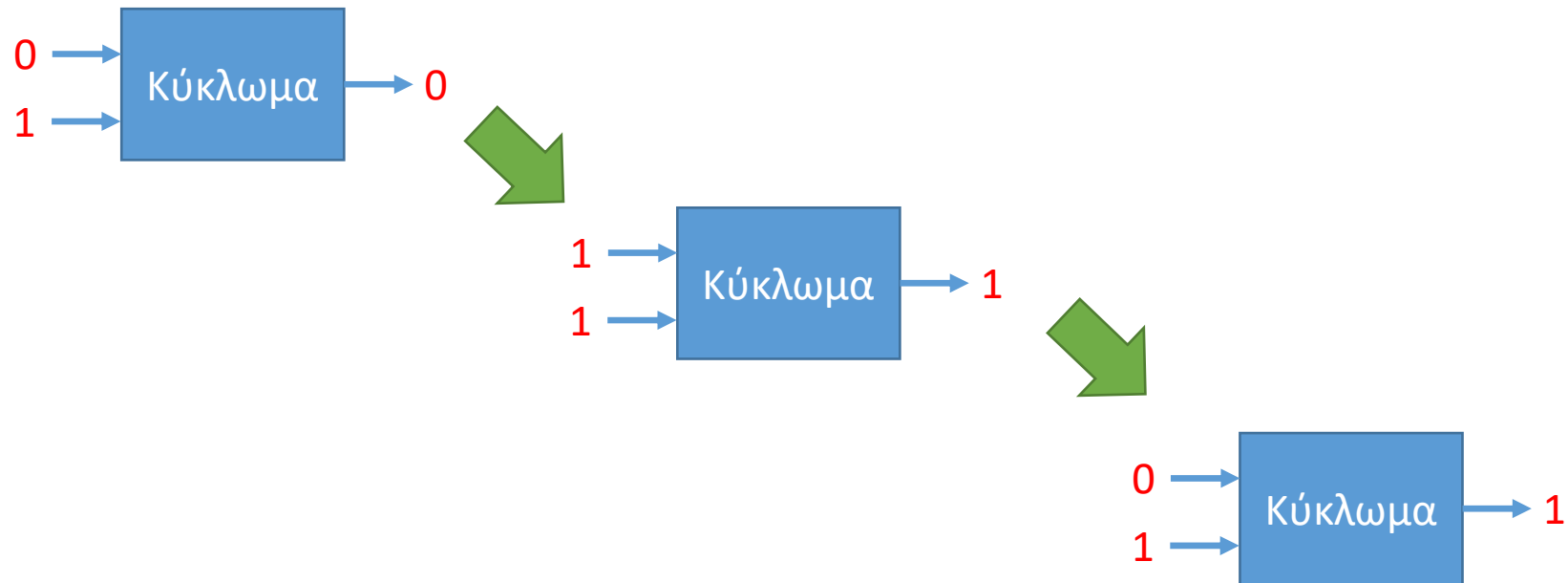
Γενικό μοντέλο ενός πλήρους κυκλώματος



Ένα κύκλωμα χωρίς μνήμη

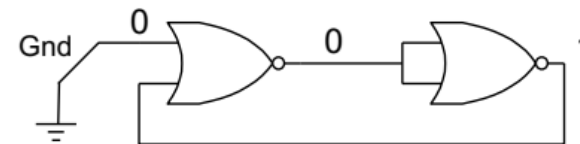
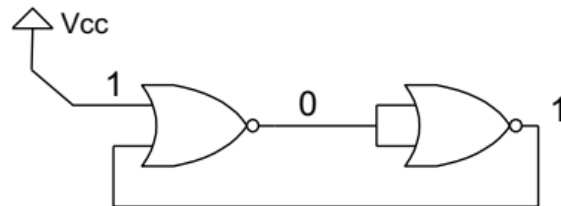
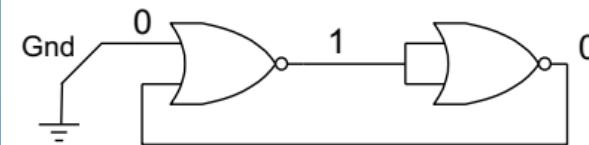
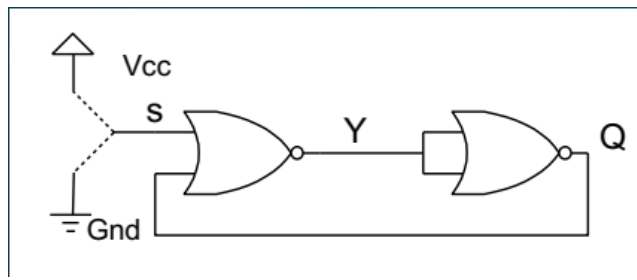


Ένα κύκλωμα με μνήμη

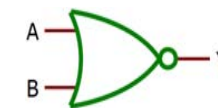


- Το κύκλωμά μας απέκτησε μνήμη !

Το πρώτο κύκλωμα με μνήμη



- Το κύκλωμά μας απέκτησε μνήμη !
 - "Θυμάται" ότι κάποτε η είσοδος πήγε στο 1.
 - Το κύκλωμα δεν επανέρχεται !!!

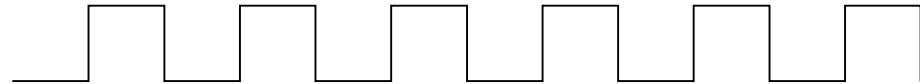


A	B	Y
0	0	1
0	1	0
1	0	0
1	1	0

Ακολουθιακά στοιχεία / στοιχεία μνήμης - γενικά

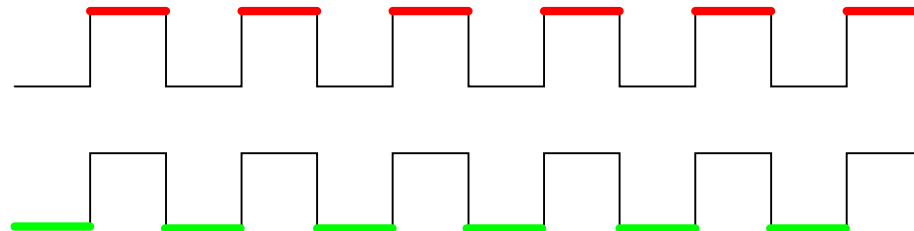


• Έστω το σήμα χρονισμού :



• Υπάρχουν δύο (2) κατηγορίες :

• Αυτά που είναι ενεργά βάσει μιας στάθμης δυναμικού :

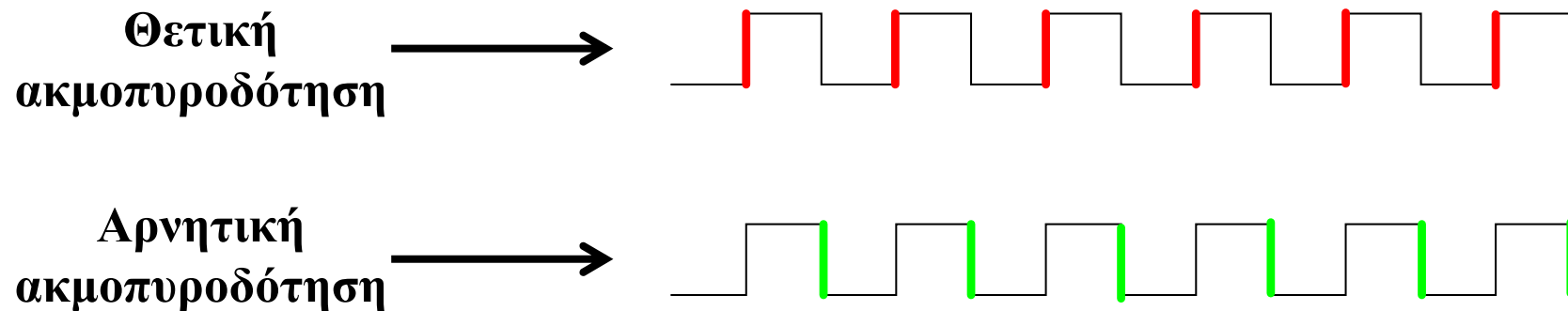


- Το στοιχείο αυτό μπορεί να αλλάζει κατάσταση όσο το σήμα χρονισμού βρίσκεται στην ενεργή στάθμη
- Τα στοιχεία αυτής της κατηγορίας ονομάζονται μανταλωτές (**latches**)

Ακολουθιακά στοιχεία / στοιχεία μνήμης – γενικά



- Αυτά που είναι ενεργά μόνο κατά την μία ακμή αλλαγής δυναμικού



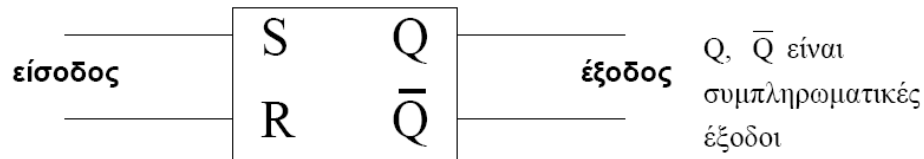
- Τα στοιχεία αυτής της κατηγορίας ονομάζονται **flip-flops**



Ακολουθιακά στοιχεία / στοιχεία μνήμης – γενικά

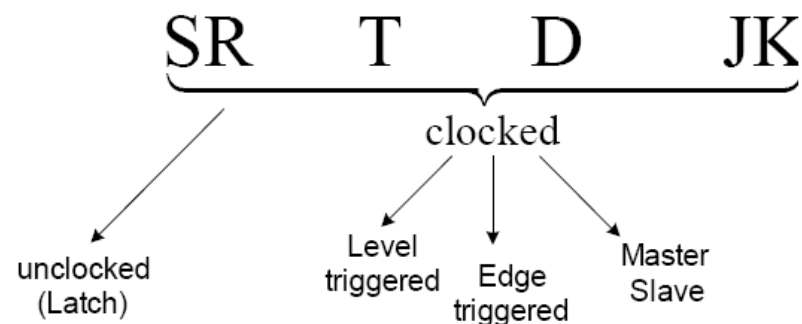
- Τα flip-flops αποτελούν τα βασικά δομικά στοιχεία για το σχεδιασμό των ακολουθιακών κυκλωμάτων

Το SR (Set-Reset) flip-flop (ff)



Πίνακας αληθείας του SR-ff

S	R	Q_{n+1}	
0	0	Q_n	επόμενη κατάσταση
0	1	0	παρούσα κατάσταση
1	0	1	
1	1	-	ακαθόριστη έξοδος (μη επιτρεπτή περίπτωση)





To SR Flip-Flop (FF)

- Για την υλοποίηση του SR-ff δημιουργούνται ο εκτεταμένος πίνακας αληθείας και οι πίνακες Karnaugh, όπου το Q_n (παρούσα κατάσταση εξόδου) χρησιμοποιείται ως μεταβλητή εισόδου:

R	S	Q_n	Q_{n+1}	
0	0	0	0	Μνήμη
0	0	1	1	Μνήμη
0	1	0	1	Set
0	1	1	1	Set
1	0	0	0	Reset
1	0	1	0	Reset
1	1	0	x	
1	1	1	x	

Πίνακας-K

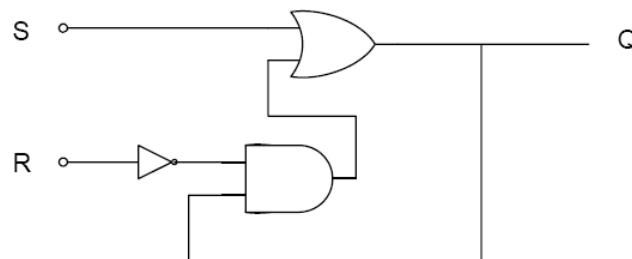
		Q_n			
		00	01	11	10
Q_{n+1}	0	0	1	1	1
	1	0	0	X	X

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

Χαρακτηριστική
εξίσωση

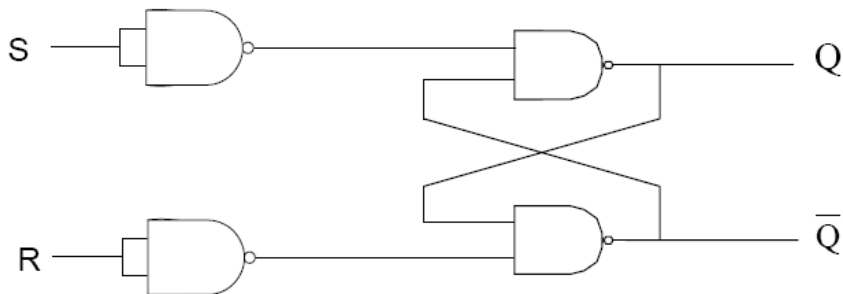
To SR Flip-Flop (FF)

- Το κύκλωμα που υλοποιεί την παραπάνω σχέση είναι:



- Χρησιμοποιώντας το θεώρημα De Morgan, η σχέση για σχεδιασμό με πύλες NAND έχει ως εξής:

$$Q_{n+1} = \overline{\overline{S} \cdot RQ_n}$$



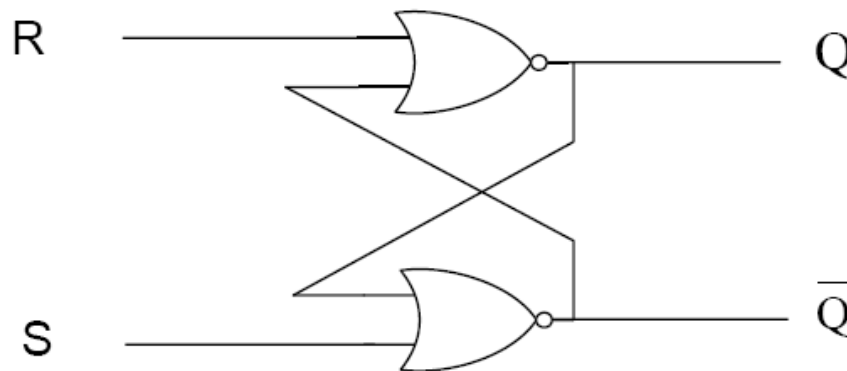
To SR Flip-Flop (FF)



Χρησιμοποιώντας έναν εκτεταμένο πίνακα αληθείας και τους πίνακες-K που συσχετίζουν τα \overline{Q}_{n+1} , \overline{Q}_n , R και S προκύπτει μια άλλη χαρακτηριστική εξίσωση η οποία δίνεται από τη σχέση:

$$\overline{Q}_{n+1} = R + \overline{S}\overline{Q}_n$$

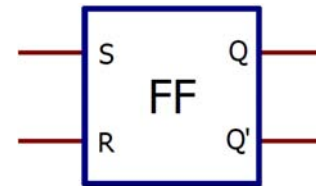
από την οποία εξάγεται η σχέση για την υλοποίηση του ff με πύλες NOR



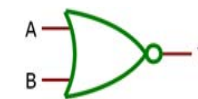
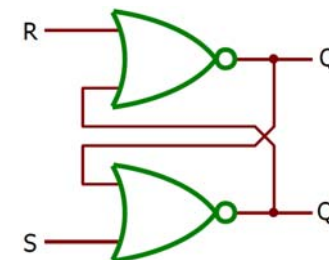
$$Q_{n+1} = \overline{R + S + Q_n}$$

To SR Flip-Flop (FF)

- Για $S=1, R=0$, τότε η έξοδος Q' είναι '0', ανεξάρτητα από την τιμή της άλλης εισόδου
- Η άλλη πύλη έχει και τις δύο εισόδους της στο '0' και άρα $Q=1$
 - **Ενεργοποιήθηκε (SET) το FF**
- Αν βάλουμε $S=0$, αυτή η αλλαγή δεν επηρεάζει την κατάσταση του FF. Η άλλη είσοδος της κάτω πύλης εξακολουθεί να είναι στο '1'.
 - **το FF «θυμάται» την προηγούμενη κατάσταση.**
- Για $R=1, S=0$, η έξοδος της πάνω γίνεται '0' και η έξοδος της δεύτερης πύλης γίνεται '1'
 - **Άρα, επιτυγχάνεται η εκκαθάριση (RESET) του FF**
- Αν βάλουμε $R=0$, η κατάσταση του FF δεν αλλάζει
 - **το FF «θυμάται» την προηγούμενη κατάσταση.**



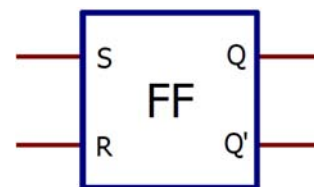
R	S	Q_n	Q_{n+1}	
0	0	0	0	Μνήμη
0	0	1	1	Μνήμη
0	1	0	1	Set
0	1	1	1	Set
1	0	0	0	Reset
1	0	1	0	Reset
1	1	0	x	}
1	1	1	x	



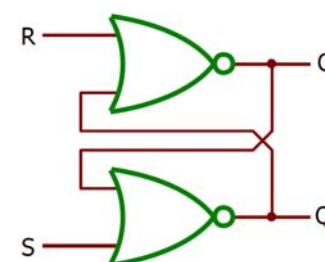
A	B	Y
0	0	1
0	1	0
1	0	0
1	1	0

To SR Flip-Flop (FF)

- Μπορούμε να αποθηκεύσουμε 1 ή 0 αν $S=1$ και $R=0$ ή $S=0$ και $R=1$.
- Όσο $S=0$ και $R=0$, το FF θυμάται την προηγούμενη κατάσταση
- Αλλάζουμε εισόδους μόνο όταν θέλουμε να αποθηκεύσουμε ένα νέο δεδομένο
- Μετά την είσοδο του νέου δεδομένου, οι είσοδοι S και R επαναφέρονται στο '0'



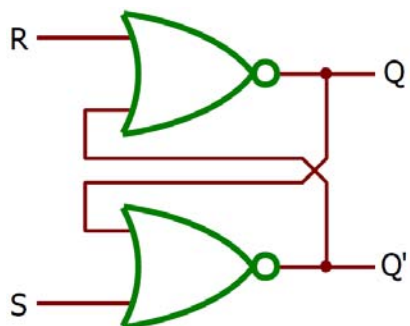
R	S	Q_n	Q_{n+1}	
0	0	0	0	Μνήμη
0	0	1	1	Μνήμη
0	1	0	1	Set
0	1	1	1	Set
1	0	0	0	Reset
1	0	1	0	Reset
1	1	0	x	}
1	1	1	x	



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

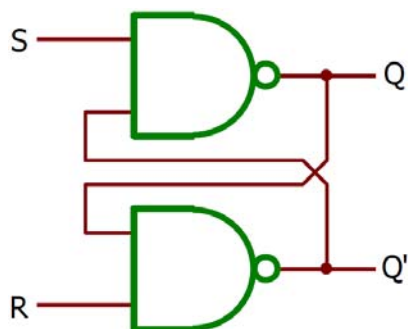
To SR Flip-Flop (FF)

• Υλοποίηση με πύλες NOR



S	R	Q	Q'	
0	0	Q	Q'	ηρεμία (μνήμη)
0	1	0	1	Reset ($Q=0$)
1	0	1	0	Set ($Q=1$)
1	1	0	0	απροσδιοριστία

• Υλοποίηση με πύλες NAND

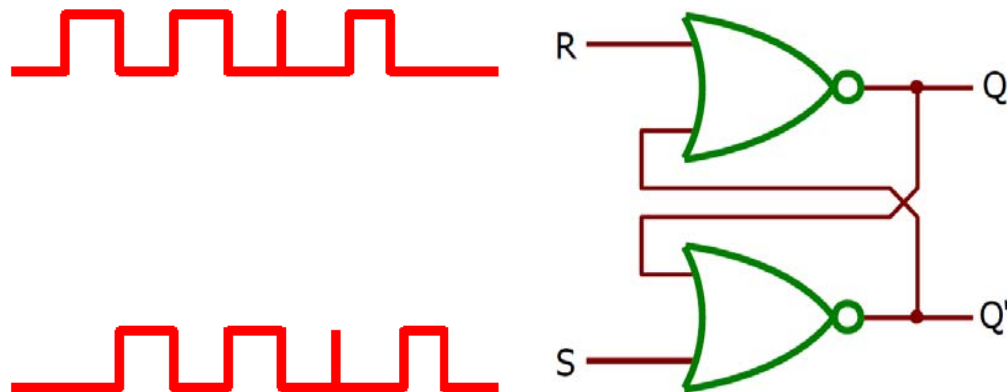


S	R	Q	Q'	
0	0	1	1	απροσδιοριστία
0	1	1	0	Set ($Q=1$)
1	0	0	1	Reset ($Q=0$)
1	1	Q	Q'	ηρεμία (μνήμη)

To SR Flip-Flop (FF)



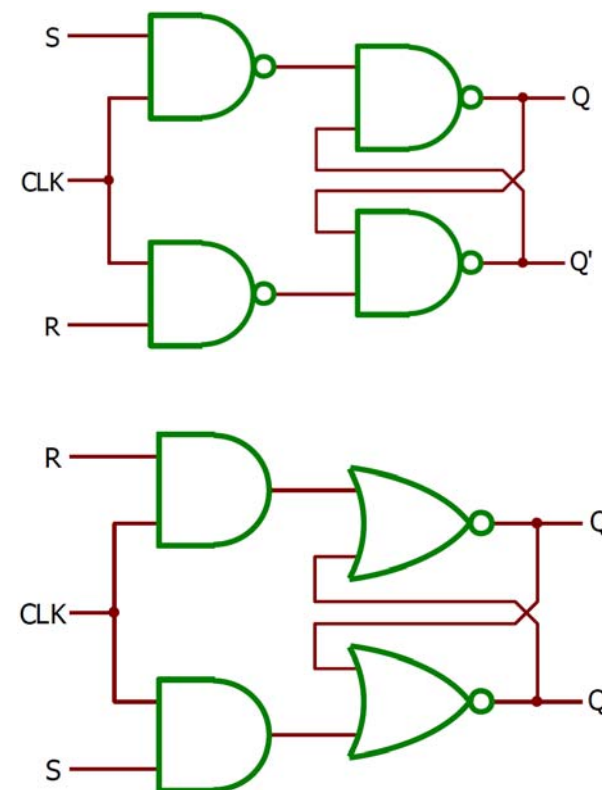
- Πρόβλημα: Αρκεί να μην μπει το κύκλωμα στις καταστάσεις απροσδιοριστίας → **Το κύκλωμα δεν επανέρχεται !!!**
- Παράδειγμα αν εμφανιστούν ανεπιθύμητοι παλμοί πολύ μικρού εύρους ή αιχμές (glitches)



To clocked SR Flip-Flop (FF)

- Για να αποφύγουμε τέτοια προβλήματα, συγχρονίζουμε τις εισόδους του FF με ένα επιπλέον επίπεδο πυλών
- Το σήμα ρολογιού δρα ως ένα σήμα ελέγχου που ενεργοποιεί το SR-FF και επομένως οι έξοδοι του μπορεί να αλλάξουν μόνο όταν ο παλμός του ρολογιού είναι 1

CLK	S	R	Q_{n+1}	Q'_{n+1}	
0	X	X	Q_n	Q'_n	
1	0	0	Q_n	Q'_n	ηρεμία (μνήμη)
1	0	1	0	1	Reset ($Q=0$)
1	1	0	1	0	Set ($Q=1$)
1	1	1	1	1	απροσδιοριστία

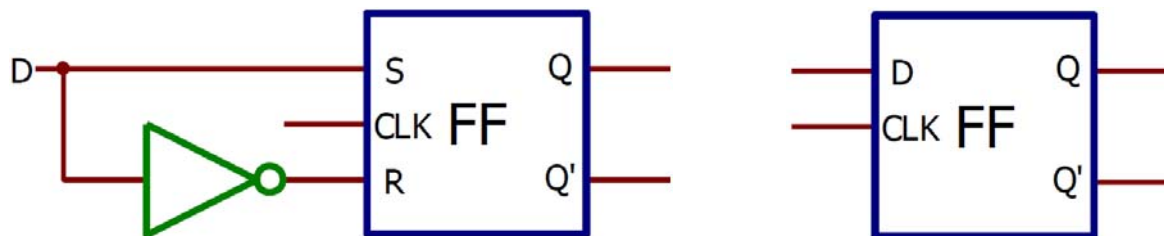


Flip-Flop τύπου D (D-FF)

- Βάζουμε στο RS FF έναν αντιστροφέα μεταξύ των εισόδων του
- Αυτό ονομάζεται FF τύπου D
 - Ουσιαστικά μόνο οι περιπτώσεις της 2ης και 3ης γραμμής του RS FF
- Λειτουργία: $D=0 \rightarrow Q=0$, $D=1 \rightarrow Q=1$ (μόνο όταν $CLK=1$)
- Όταν $CLK=1$, το D FF κρατάει στην έξοδο την προηγούμενη τιμή
- Ονομάζεται και μανταλωτής D (D latch)

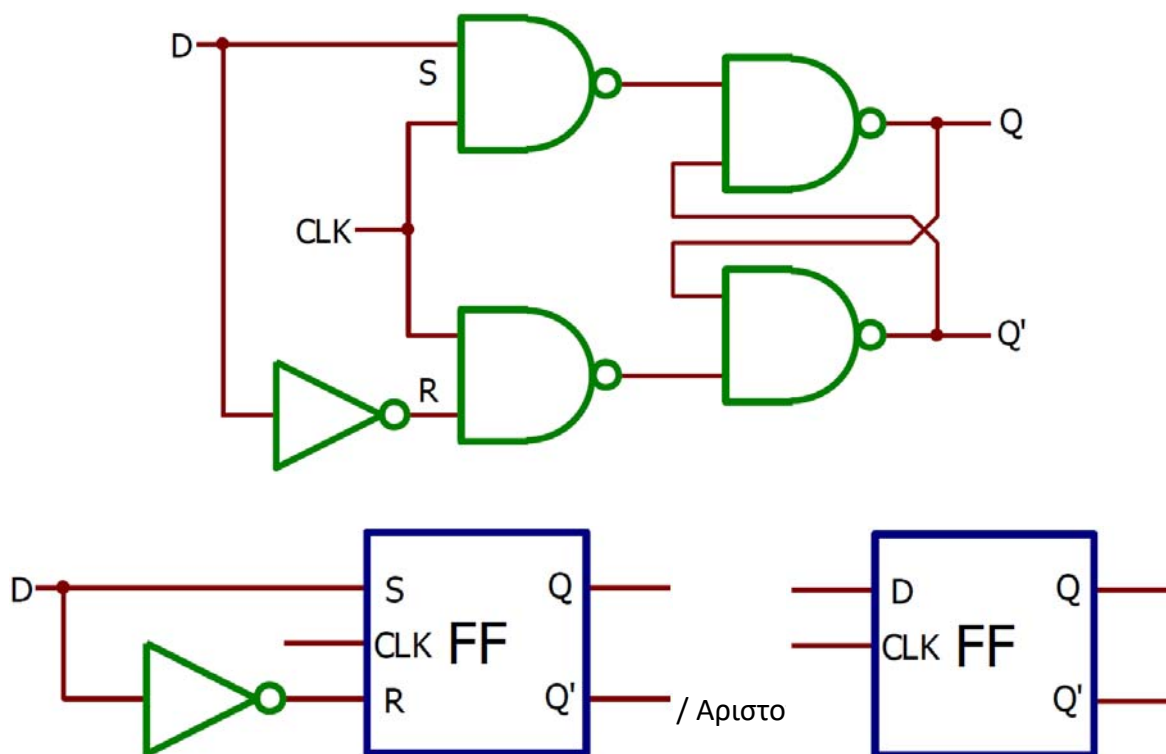
CLK	S	R	Q_{n+1}	Q'_{n+1}	
0	X	X	Q_n	Q'_n	
1	0	0	Q_n	Q'_n	ηρεμία (μνήμη)
1	0	1	0	1	Reset ($Q=0$)
1	1	0	1	0	Set ($Q=1$)
1	1	1	1	1	απροσδιοριστία

CLK	D	Q_{n+1}	Q'_{n+1}
0	X	Q_n	Q'_n
1	0	0	1
1	1	1	0



Flip-Flop τύπου D (D-FF)

• Υλοποίηση



CLK	S	R	Q_{n+1}	Q'_{n+1}	
0	X	X	Q_n	Q'_n	
1	0	0	Q_n	Q'_n	ηρεμία (μνήμη)
1	0	1	0	1	Reset ($Q=0$)
1	1	0	1	0	Set ($Q=1$)
1	1	1	1	1	απροσδιοριστία

CLK	D	Q_{n+1}	Q'_{n+1}
0	X	Q_n	Q'_n
1	0	0	1
1	1	1	0

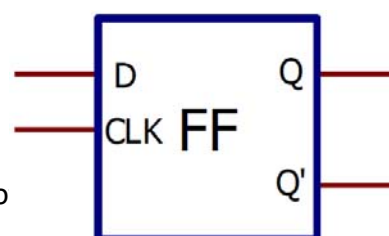
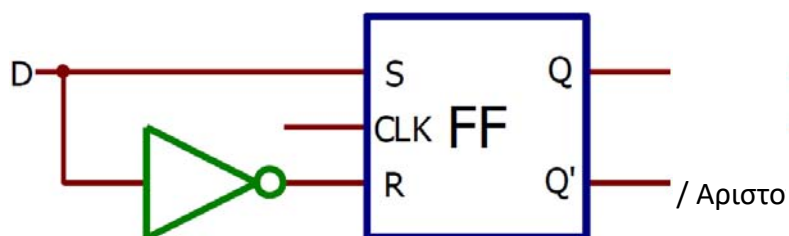
Flip-Flop τύπου D (D-FF)

• D latch : το πρώτο μας ακολουθιακό στοιχείο

- Εξαλείφουμε τη κατάσταση απροσδιοριστίας με το να μην επιτρέπουμε S, R να πάρουν ταυτόχρονα τη τιμή 1
- Μπορούμε εναλλακτικά να πούμε ότι η είσοδος D, **δειγματοληπτείται** (αντιγράφεται στην έξοδο) **ΔΙΑΡΚΩΣ** όσο το C είναι στο 1

CLK	S	R	Q_{n+1}	Q'_{n+1}	
0	X	X	Q_n	Q'_n	
1	0	0	Q_n	Q'_n	ηρεμία (μνήμη)
1	0	1	0	1	Reset ($Q=0$)
1	1	0	1	0	Set ($Q=1$)
1	1	1	1	1	απροσδιοριστία

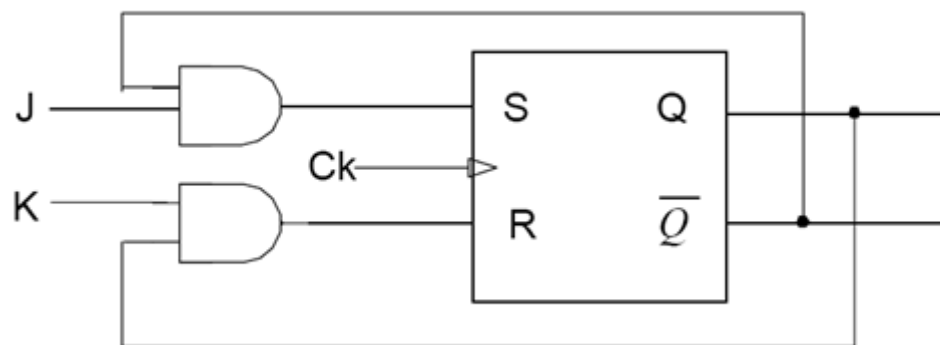
CLK	D	Q_{n+1}	Q'_{n+1}
0	X	Q_n	Q'_n
1	0	0	1
1	1	1	0



νίκης, Τμήμα Πληροφορικής

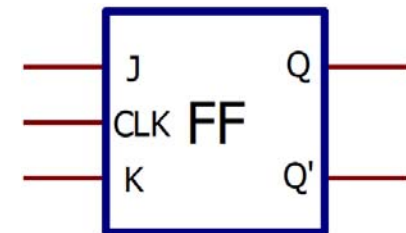
Flip-Flop τύπου JK

- Για να λύσουμε την απροσδιόριστη κατάσταση του RS FF ανατροφοδοτούμε τις εξόδους στις εισόδους
- Ονομάζεται JK flip-flop ($S \rightarrow J, R \rightarrow K$).
- Δεν έχει απροσδιόριστη κατάσταση όταν και οι είσοδοί είναι '1'
- Όταν $J=K=1$ και ($CLK=1$), τότε η κατάσταση του FF αντιστρέφεται ('1' \rightarrow '0' και '0' \rightarrow '1')



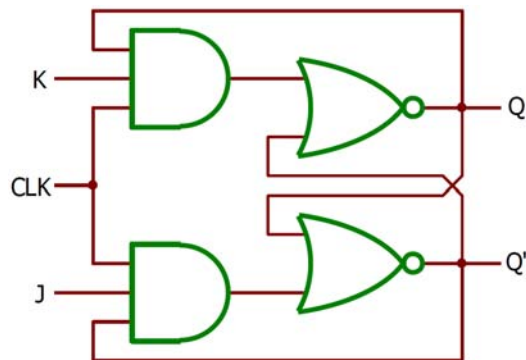
$$S = J\bar{Q}$$

$$R = KQ$$

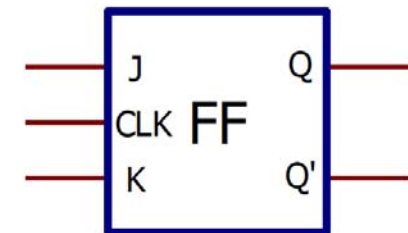


Flip-Flop τύπου JK

- Η απροσδιόριστη κατάσταση του RS FF ανατροφοδοτούμε τις εξόδους στις εισόδους.
- Ονομάζεται JK flip-flop ($S \rightarrow J, R \rightarrow K$).
- Δεν έχει απροσδιόριστη κατάσταση όταν και οι είσοδοί είναι '1'
- Όταν $J=K=1$ και ($CLK=1$), τότε η κατάσταση του FF αντιστρέφεται ('1' \rightarrow '0' και '0' \rightarrow '1')

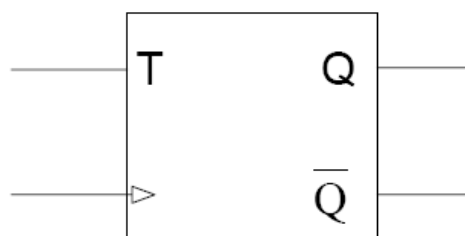


CLK	J	K	Q_{n+1}	Q'_{n+1}	
0	X	X	Q_n	Q'_n	
1	0	0	Q_n	Q'_n	ηρεμία (μνήμη)
1	0	1	0	1	Reset ($Q=0$)
1	1	0	1	0	Set ($Q=1$)
1	1	1	Q'_n	Q_n	toggle



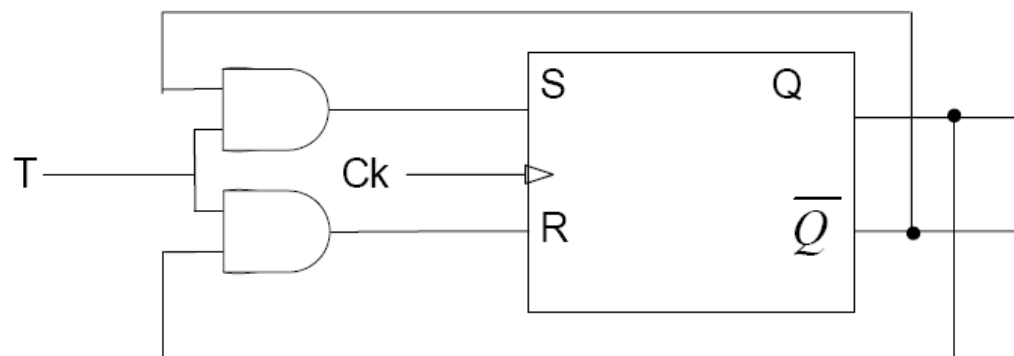
Flip-Flop τύπου T (Toggle)

Flip – Flop τύπου T (Toggle ff)



Ck	T	Q_{n+1}
0	0	Q_n
0	1	Q_n
1	0	Q_n
1	1	$\overline{Q_n}$

Υλοποίηση:



$$S = T\overline{Q}$$

$$R = TQ$$



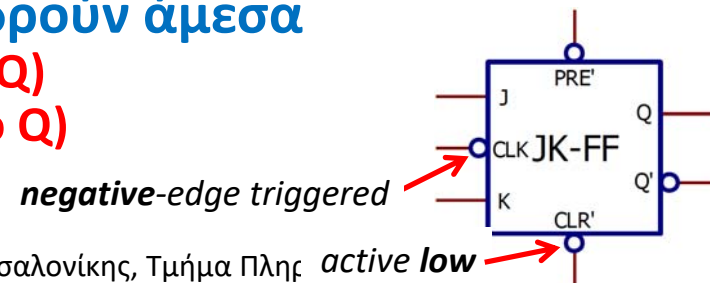
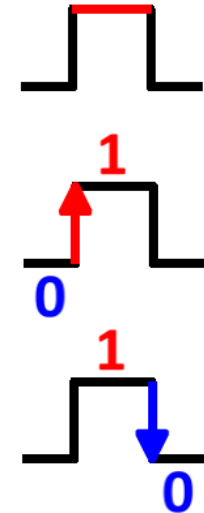
Πυροδότηση των Flip-flops

- Το JK FF παρουσιάζει μια αδυναμία. Όταν $CLK=1$ και $J=K=1$ (ή $T=1$), τότε αυτό αλλάζει κατάσταση
- Αποτέλεσμα \rightarrow Το FF θα αλλάζει καταστάσεις όσο $CLK=1$
- Το πρόβλημα αυτό μπορεί να αντιμετωπιστεί με δύο διαφορετικούς τρόπους:
 - (α) με JK FFs τύπου αφέντης-σκλάβος (master-slave) και
 - (β) με FFs τα οποία ενεργοποιούνται κατά την άνοδο ή κάθοδο του παλμού του ρολογιού (ακμοπυροδοτούμενα, edge-triggered), και όχι κατά τη διάρκεια του παλμού



Πυροδότηση των Flip-flops

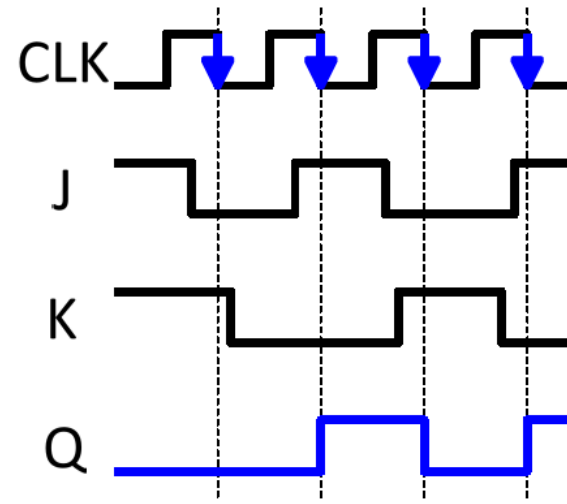
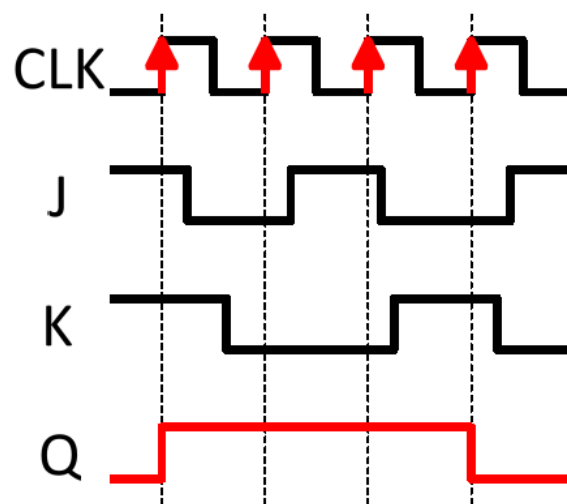
- Επιπεδο-πυροδοτούμενα (level-triggered) ή (pulse-triggered)
- Ακμοπυροδοτούμενα (edge-triggered)
 - **Θετικής Ακμής (positive-edge)**
 - Σκανδαλίζονται (λαμβάνει τιμή η έξοδος Q) κατά την άνοδο του παλμού CLK (μετάβαση 0 → 1)
 - **Αρνητικής Ακμής (negative-edge)**
 - Σκανδαλίζονται (λαμβάνει τιμή η έξοδος Q) κατά την κάθοδο του παλμού CLK (μετάβαση 1 → 0)
- **Preset, Clear: ασύγχρονες εισοδοι που επιδρούν άμεσα**
 - **Preset:** λειτουργία Set (ενεργοποιεί την έξοδο Q)
 - **Clear:** λειτουργία Reset (εκκαθαρίζει την έξοδο Q)



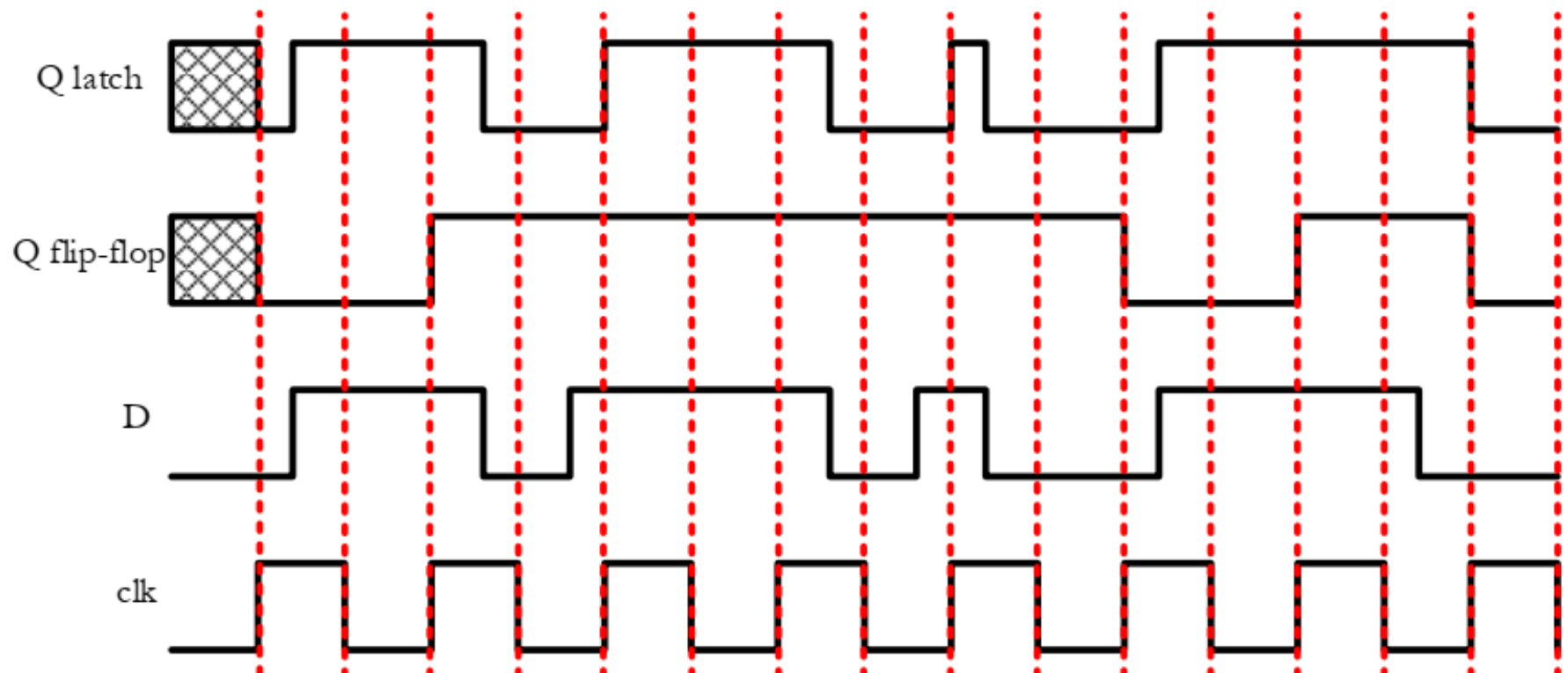
Πυροδότηση των Flip-flops

- Παραδείγματα

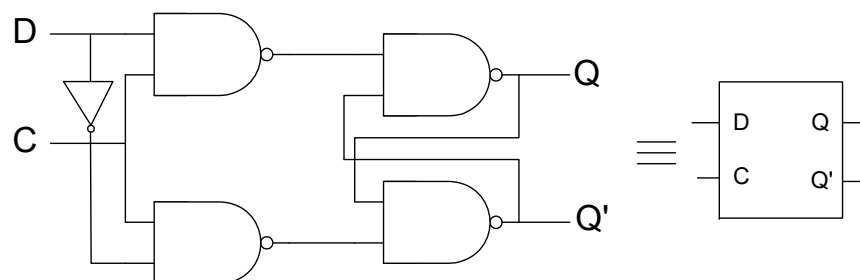
J	K	Q_{n+1}
0	0	Q_n
0	1	0
1	0	1
1	1	Q_n'



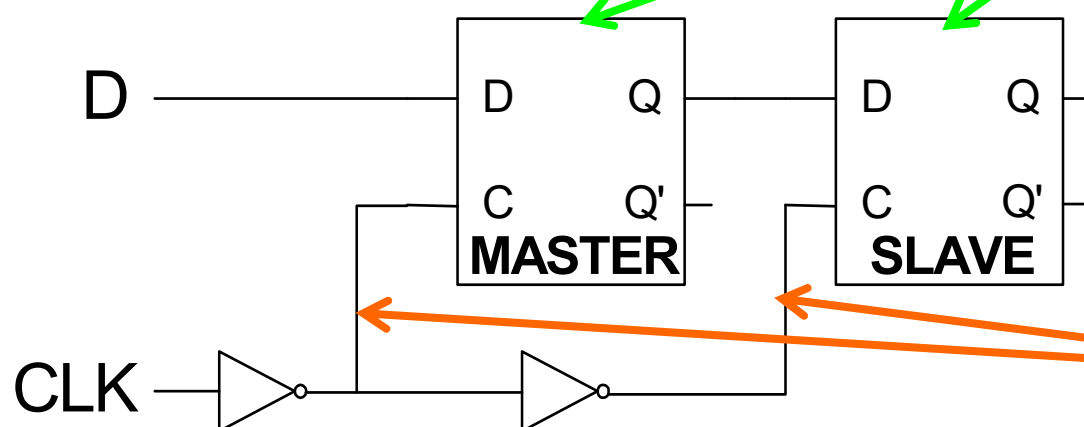
Latches vs Flip-flops



Πως φτιάχνεται ένα θετικά ακμοπυροδότητο D flip-flop ?



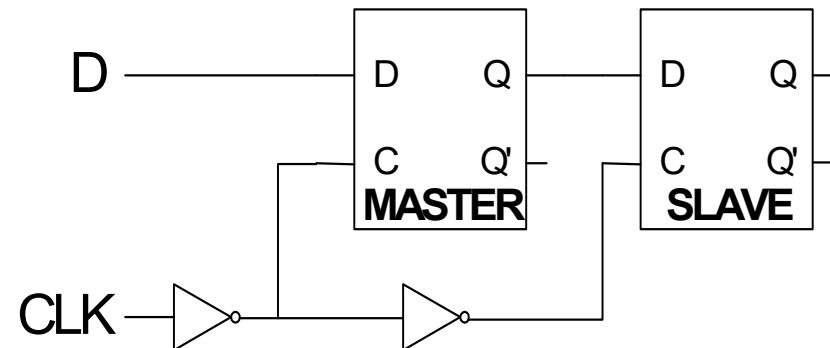
Δύο D latches
(master and slave)



Με συμπληρωματικά
ρολόγια

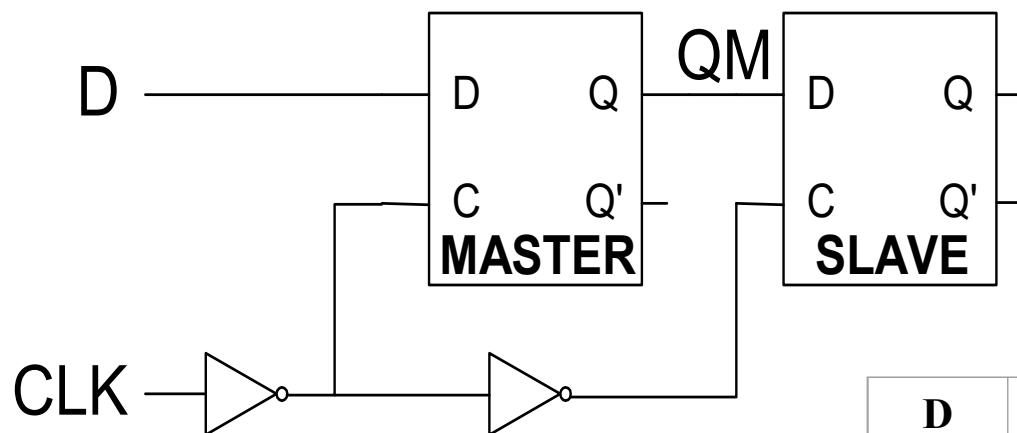
Λειτουργία ενός D flip-flop

- **CLK = LOW** => Master ενεργό, Slave ανενεργό. Η είσοδος αντιγράφεται στην έξοδο του Master. Η έξοδος του Slave διατηρεί τη προηγούμενη κατάσταση



- **CLK = HIGH** => Master ανενεργό και συνεπώς διατηρεί τη προηγούμενη κατάσταση. Το Slave ενεργό και αντιγράφει την έξοδο του Master στην έξοδο του κυκλώματος.
- **Αφού η έξοδος του Master δεν αλλάζει όσο το ρολόι είναι στο 1, και η έξοδος του Slave δεν αλλάζει όσο το ρολόι είναι στο 0 ότι δειγματοληπτήθηκε στην ανοδική ακμή του CLK θα είναι η έξοδος του κυκλώματος μέχρι την επόμενη ανοδική ακμή**

Πίνακας αληθείας



D	CLK	Q	Q'
0	↑	0	1
1	↑	1	0
X	0	Q(t-1)	Q'(t-1)
X	1	Q(t-1)	Q'(t-1)

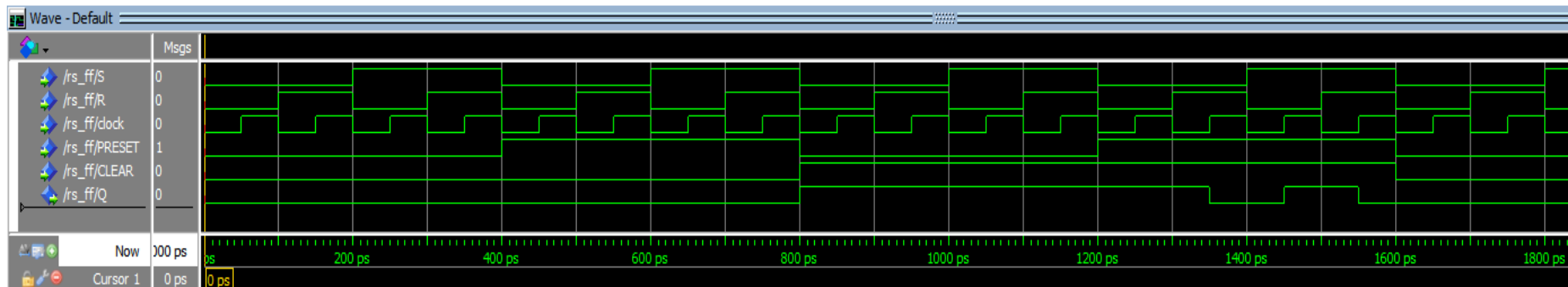
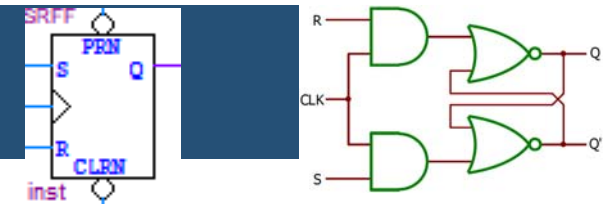
Άμεσες είσοδοι σε ένα FF



- Μερικές φορές θέλουμε να θέσουμε ή να καθαρίσουμε ένα FF χωρίς να χρειάζεται να οδηγήσουμε τόσο την είσοδο D όσο και την είσοδο CLK
- Το πλέον προφανές παράδειγμα είναι η αρχικοποίηση των ακολουθιακών στοιχείων του κυκλώματός μας σε κάποια αρχική κατάσταση
- Αυτό επιτυγχάνεται με τη προσθήκη άμεσων εισόδων :
 - **Άμεση είσοδος θέσης (Preset / Direct Set).** Η ενεργοποίησή της οδηγεί το Q στο 1 (Q' στο 0)
 - **Άμεση είσοδος μηδενισμού / καθαρισμού (Reset / Clear).** Η ενεργοποίησή της οδηγεί το Q στο 0 (Q' στο 1)
- Οι άμεσες είσοδοι μπορούν να δρουν
 - **Ασύγχρονα**
 - **Σύγχρονα**

Επαλήθευση Λειτουργίας RS-FF

* πρόκειται για το RS-FF με 2 AND και 2 NOR και όχι με 4 NAND



S	X	X	0	0	1	1
R	X	X	0	1	0	1
clock	X	X	↑	↑	↑	↑
PRESET	X	0 active low	1			
CLEAR	0 active low	1	1			
Q	0	1	1	0	1	0

positive-edge triggered

clear

preset

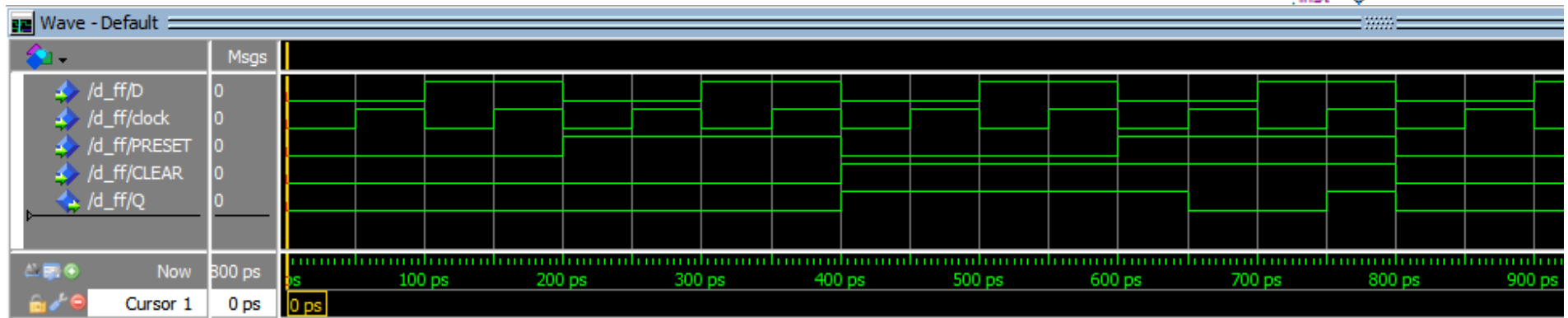
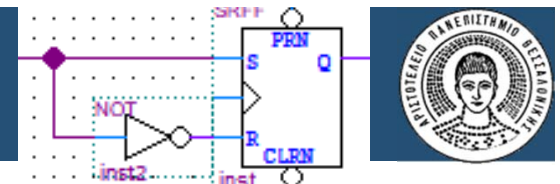
Q_n μνήμη

reset

set

απροσδιοριστία*

Επαλήθευση Λειτουργίας D-FF



D	X	X	0	1
Clock	X	X	↑	↑
PRESET	X	0	1	1
CLEAR	0	1	1	1
Q	0	1	0	1

21 April 2021

clear

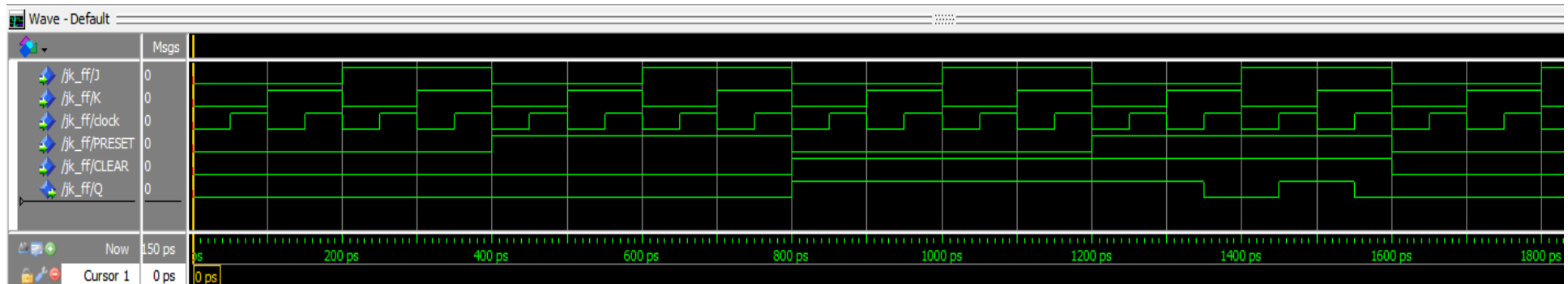
preset

D

D

36

Επαλήθευση Λειτουργίας JK-FF



J	X	X	0	0	1	1
K	X	X	0	1	0	1
Clock	X	X	↑	↑	↑	↑
PRESET	X	0	1			
CLEAR	0	1	1			
Q	0	1	1	0	1	0

21 April 2021

clear

preset

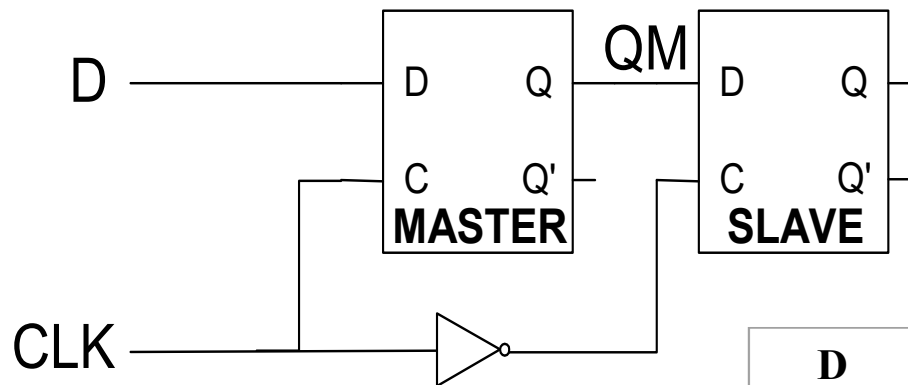
Q_n μνήμη

reset

set

Q'_n toggle

Αρνητικά ακμοπυροδότητο D flip-flop

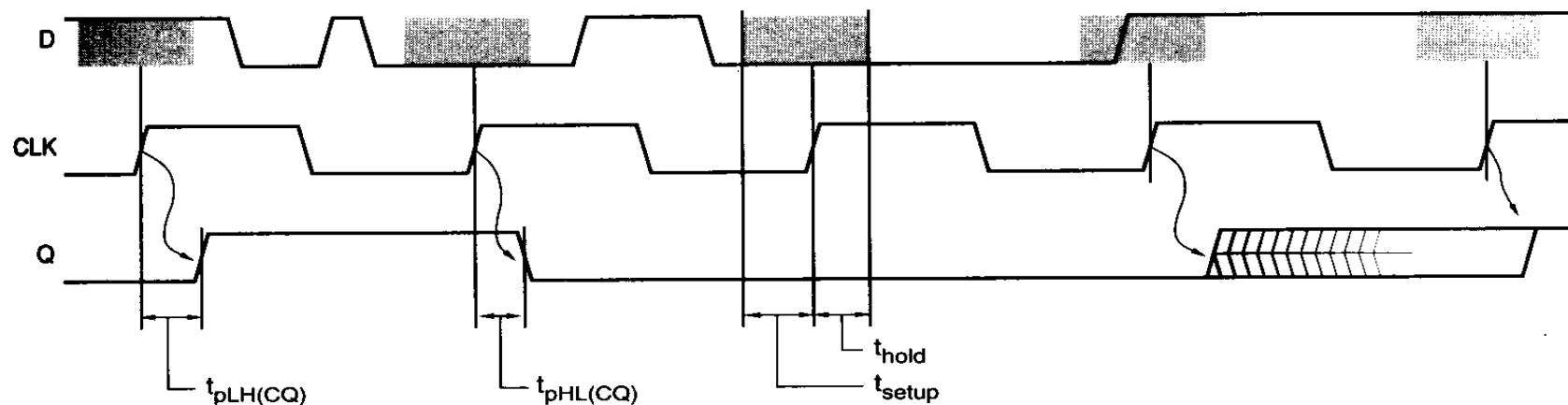
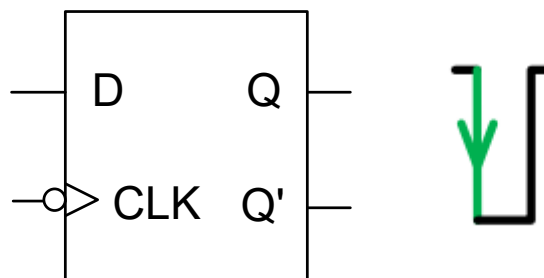
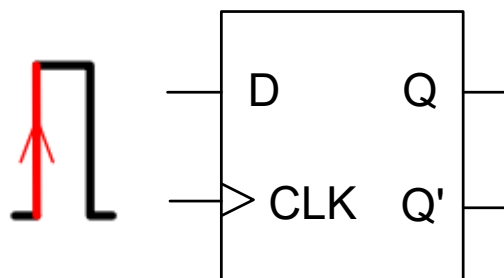


D	CLK	Q	Q'
0	↓	0	1
1	↓	1	0
X	0	Q(t-1)	Q'(t-1)
X	1	Q(t-1)	Q'(t-1)

Συμβολισμοί και χρόνοι ενδιαφέροντος

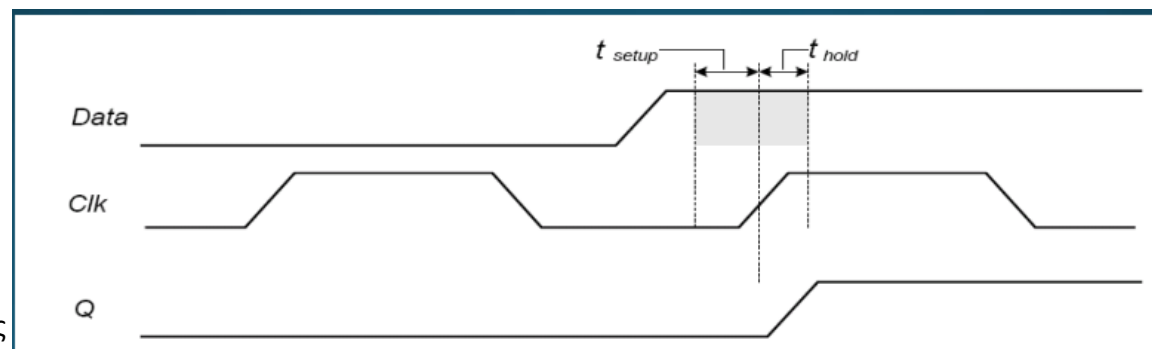
Θετικά
ακμοπυροδοτούμενο
D flip-flop

Αρνητικά
ακμοπυροδοτούμενο
D flip-flop



Προβλήματα Χρονισμού (Μεταστάθεια)

- Ορίζουμε δύο χρονικά διαστήματα που θα πρέπει να τηρούνται **ευλαβικά** από κάθε σχεδιαστή ώστε να μην εμφανιστεί ποτέ το πρόβλημα της μεταστάθειας
 - **Setup time (χρονικό διάστημα προετοιμασίας)** Χρόνος που το D πρέπει να έχει σταθερή τιμή πριν την αλλαγή του Clk στην κατάσταση επίτρεψης
 - **Hold time (χρονικό διάστημα αποκατάστασης)** Χρόνος που το D πρέπει να διατηρήσει την ίδια τιμή μετά την αλλαγή του Clk στην κατάσταση επίτρεψης

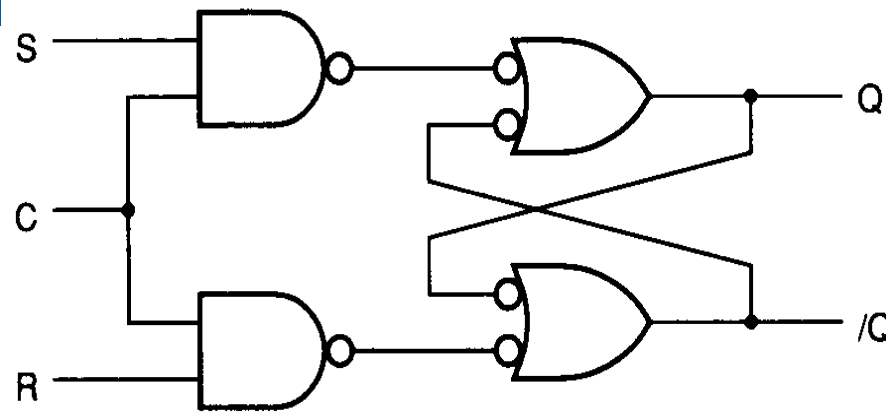




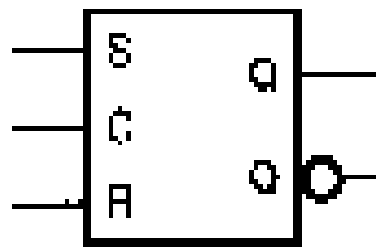
Αλλα είδη flip - flop

- Το **D flip flop** είναι το πιο οικονομικό από άποψης απαιτούμενων πυλών
- Ωστόσο δεν είναι το μόνο που μπορούμε να φτιάξουμε. Τα πιο διαδεδομένα άλλα είδη flip flop είναι :
 - To Master / Slave S-R Flip – Flop
 - To Master / Slave J-K Flip – Flop
 - Το ακμοπυροδότητο J-K Flip – Flop
 - To T Flip – Flop
 - To Scan Flip – Flop
- Σε καθένα από αυτά, μπορούμε να προσθέτουμε άμεσες εισόδους είτε σύγχρονες είτε ασύγχρονες
- Η συλλογή όλων αυτών των ακολουθιακών στοιχείων αποτελεί μια βιβλιοθήκη στοιχείων με την οποία φτιάχνουμε ακολουθιακά κυκλώματά

Master / Slave S-R Flip Flop



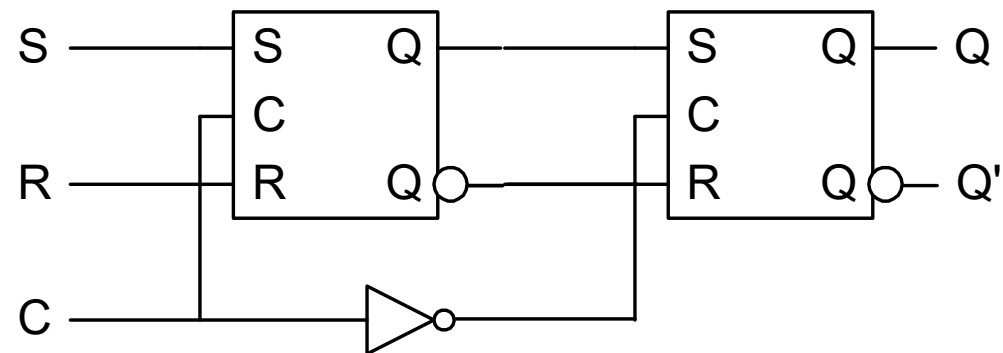
C	S	R	Q (next state)
0	X	X	Q (t-1)
1	0	0	Q (t-1)
1	0	1	0 (μηδενισμός)
1	1	0	1 (θέση)
1	1	1	Απροσδιοριστία



Οπως φτιάξαμε ένα Master / Slave D FF με τη χρήση δύο D latches, μπορούμε με τη χρήση δύο S-R latches με είσοδο επίτρεψης να φτιάξουμε ένα Master / Slave S-R FF

Master / Slave S-R Flip Flop - 2

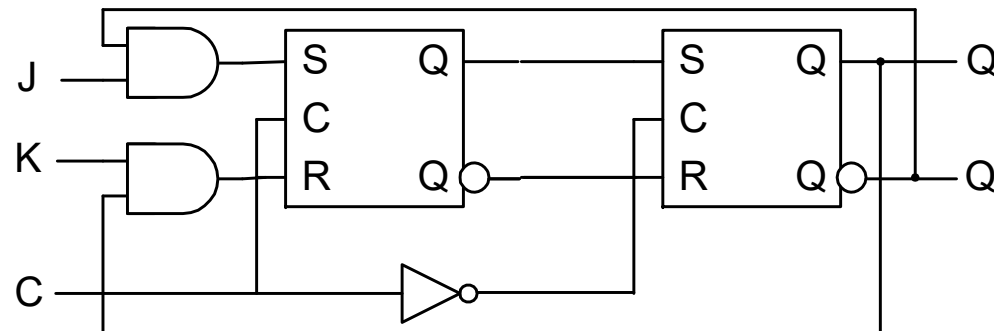
- Όταν το ρολόι είναι στο 1, το Slave δεν αλλάζει κατάσταση.
- Όταν το ρολόι μεταβαίνει στο 0 αλλάζει η έξοδος του Slave αλλά το Master είναι ανενεργό και συνεπώς δίνει συνεχώς την ίδια είσοδο στο Slave.
- Αρα αλλαγή της εξόδου μόνο στην \downarrow του C.



C	S	R	Q (next state)
0	X	X	Q (t-1)
1	0	0	Q (t-1)
1	0	1	0 (μηδενισμός)
1	1	0	1 (θέση)
1	1	1	Απροσδιοριστία

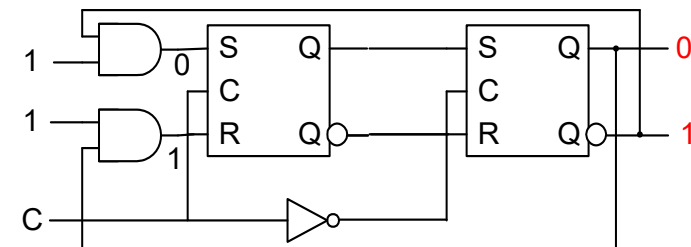
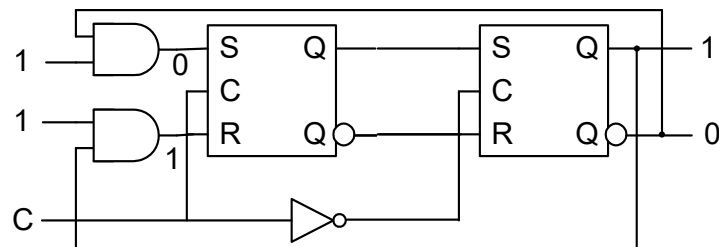
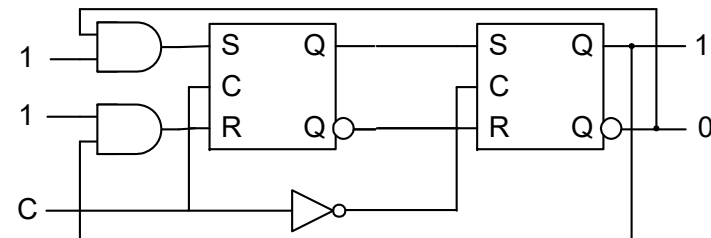
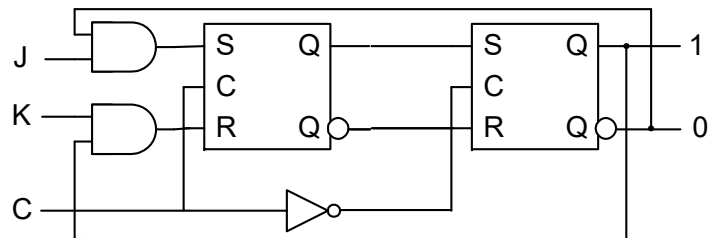
Master / Slave J-K Flip Flop

- Προτάθηκε για να λύσει το πρόβλημα της απροσδιοριστίας.
- Φτιάχνεται κι αυτό με 2 S-R latches με είσοδο επίτρεψης και επιπλέον λογική που αποτρέπει την είσοδο $S=R=1$ στο Master latch.



- Όταν η J πάρει τη τιμή 1, η S γίνεται 1, μόνο αν το Q' είναι 1, δηλαδή αν το Q είναι 0 !
- Όταν η K πάρει τη τιμή 1, η R γίνεται 1, μόνο αν το Q είναι 1, δηλαδή αν το Q' είναι 0 !

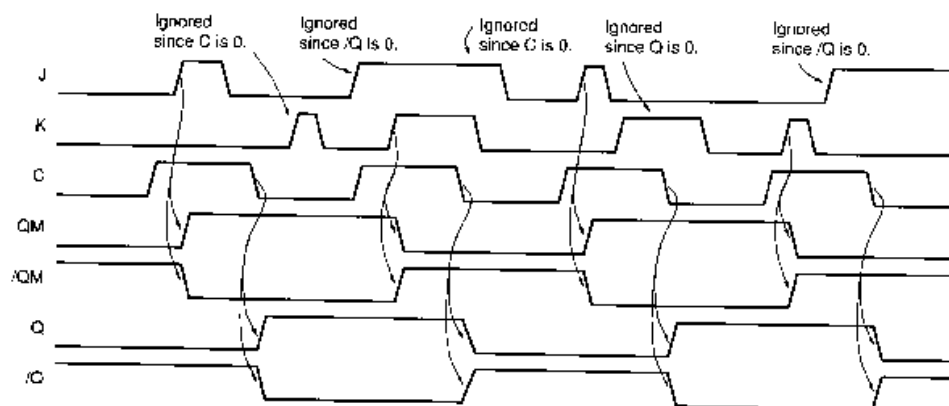
Master / Slave J-K Flip Flop - 2



- Η εφαρμογή της εισόδου $J = K = 1$ είχε ως αποτέλεσμα την αντιστροφή της προηγούμενης κατάστασης.
- Ακριβώς το ίδιο θα συνέβαινε αν η αρχική μας κατάσταση ήταν η $Q=0$ και $\sim Q=1$.

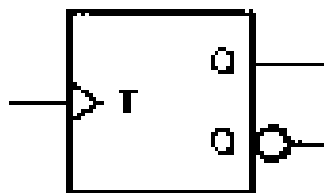
Master / Slave J-K Flip Flop – 3

C	J	K	Q (next state)
0	X	X	Q (t-1)
1	0	0	Q (t-1)
1	0	1	0 (μηδενισμός)
1	1	0	1 (θέση)
1	1	1	$\sim Q(t-1)$



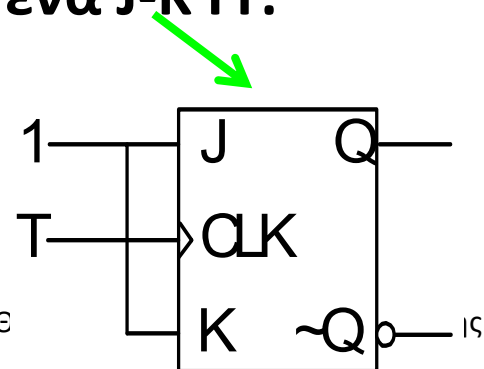
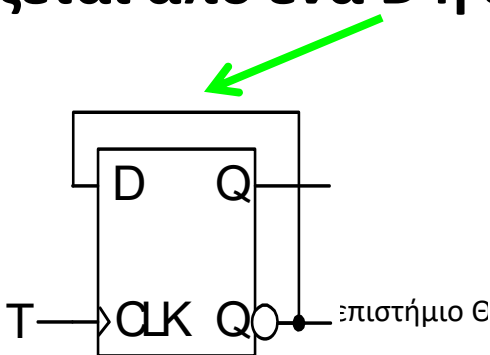
T(oggle) Flip Flop

- Το T FF αλλάζει κατάσταση σε κάθε πυροδότηση !!!



- ΠΑΡΑΤΗΡΗΣΗ : Η έξοδος ενός T FF έχει τη μισή συχνότητα της εισόδου του !!!

- Ένα T FF κατασκευάζεται από ένα D ή ένα J-K FF.

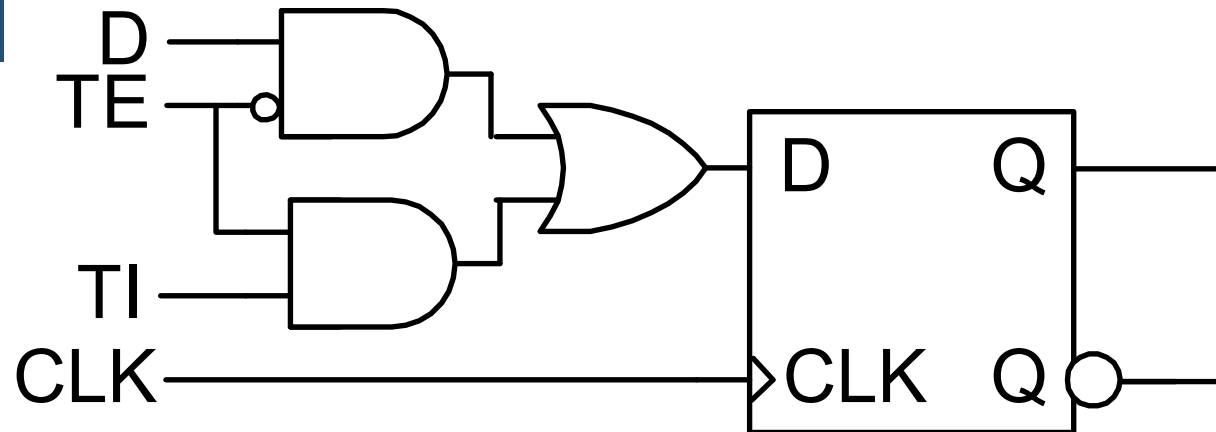


Scan Flip Flop (σάρωσης)



- Στα σημερινά chip υπάρχουν χιλιάδες έως εκατοντάδες χιλιάδες FF
- Όταν θέλουμε να δοκιμάσουμε αν ένα τέτοιο κύκλωμα δουλεύει σωστά θα πρέπει να μπορούμε να βάζουμε τα FF του σε συγκεκριμένες καταστάσεις και να διαβάζουμε τις εξόδους
- Αυτό δε μπορεί να γίνει (ή είναι εξαιρετικά χρονοβόρο) χρησιμοποιώντας τις εισόδους / εξόδους λειτουργίας του FF
- Σκοπός του scan FF είναι να παρέχει εναλλακτική είσοδο / έξοδο δεδομένων που τη χρησιμοποιούμε κατά τη διάρκεια των δοκιμών μόνο

Scan D FF



- Ισοδύναμο με ένα DFF και ένα πολυπλέκτη στην είσοδο με σήμα επιλογής το $\sim TE$ (Test Enable).
- Για κανονική λειτουργία $TE = 0$ και παίρνουμε ένα D FF.
- Όταν $TE = 1$, η είσοδος D, οδηγείται από το TI (Test Input).
- Οι επιπλέον είσοδοι χρησιμοποιούνται για τη σύνδεση όλων των FF του κυκλώματος σε μια αλυσίδα σάρωσης (scan chain).

Τι θα πρέπει να θυμάστε

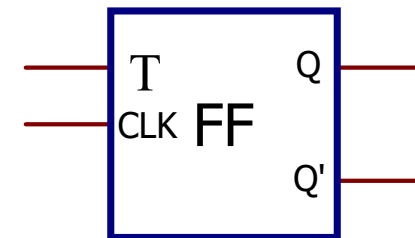
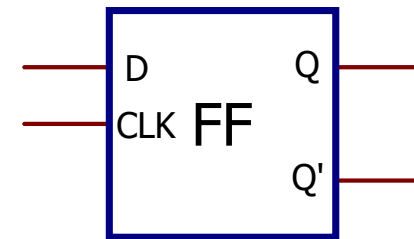


- Ακμοπυροδότητα και πυροδότητα με τιμή (level vs edge triggered) στοιχεία
- Άμεσες είσοδοι για Reset και Preset
- Το D FF είναι το πιο οικονομικό και γι' αυτό χρησιμοποιείται κατά κόρο
- Άλλα χρησιμοποιούμενα FFs είναι τα J-K και T

Χαρακτηριστικοί πίνακες FF

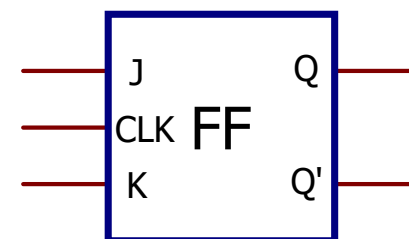
CLK	D	Q_{n+1}	
↑	1	1	Load "1" (Set)
↑	0	0	Load "0" (Reset)

CLK	T	Q_{n+1}	
↑	0	Q_n	Μνήμη/Παραμονή
↑	1	Q'_n	Εναλλαγή (Toggle)



Χαρακτηριστικοί πίνακες FF

CLK	J	K	Q_{n+1}	
↑	0	0	Q_n	ηρεμία (μνήμη)
↑	1	0	1	Load "1" (Set)
↑	0	1	0	Load "0" (Reset)
↑	1	1	Q'_n	Toggle





Πίνακες Διέγερσης

- Άλλη μορφή του πίνακα λειτουργίας ενός flip flop
 - Για κάθε πιθανή τρέχουσα κατάσταση και τιμή εισόδου την επόμενη κατάσταση του flip flop
 - Μας δίνει τις τιμές εισόδου για να μεταβούμε από μια τρέχουσα κατάσταση, σε μια επιθυμητή επόμενη κατάσταση

$Q(t)$	$Q(t+1)$	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

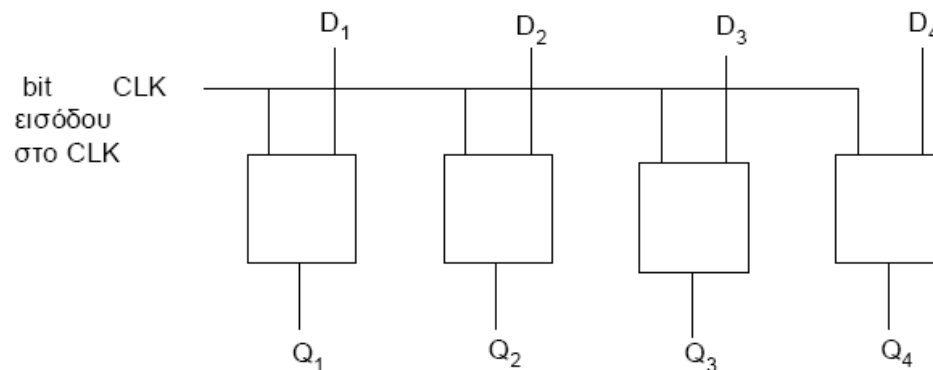
(δ) T

Εφαρμογές ff

- Τυπικές εφαρμογές των ff's είναι:
 - Απλοί καταχωρητές
 - Κυκλώματα καταχωρητών ολίσθησης
 - Μετρητές

Απλοί καταχωρητές

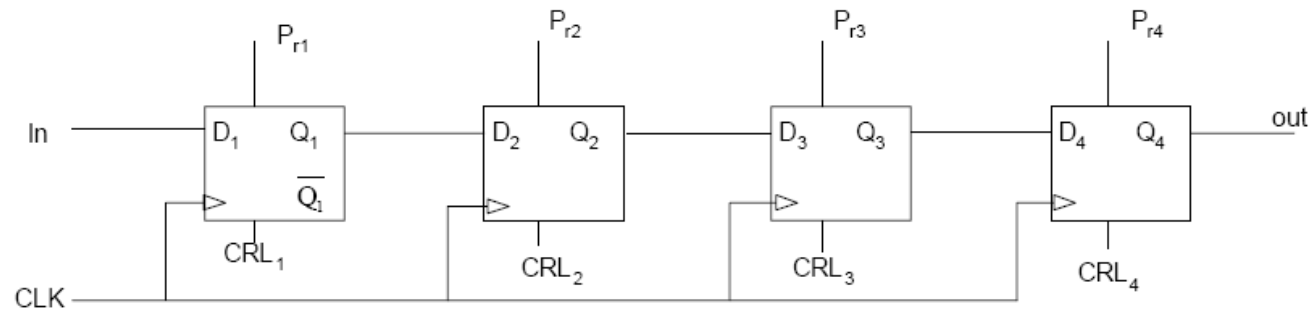
Π.χ. Καταχωρητής 4 bit



Μια ομάδα από m ff's αποθηκεύει m bits πληροφορίας. Τέτοιες ομάδες συχνά χρησιμοποιούνται για αποθήκευση πληροφορίας σε συστήματα υπολογιστών. (Προσωρινή αποθήκευση δεδομένων)

Καταχωρητές ολίσθησης

Για παράδειγμα καταχωρητής ολίσθησης 4 bit χρησιμοποιώντας D-ff



DATA IN	Q ₁	Q ₂	Q ₃	Q ₄
1	0	0	0	0
1	1	0	0	0
0	1	1	0	0
0	0	1	1	0
1	0	0	1	1
1	1	0	0	1

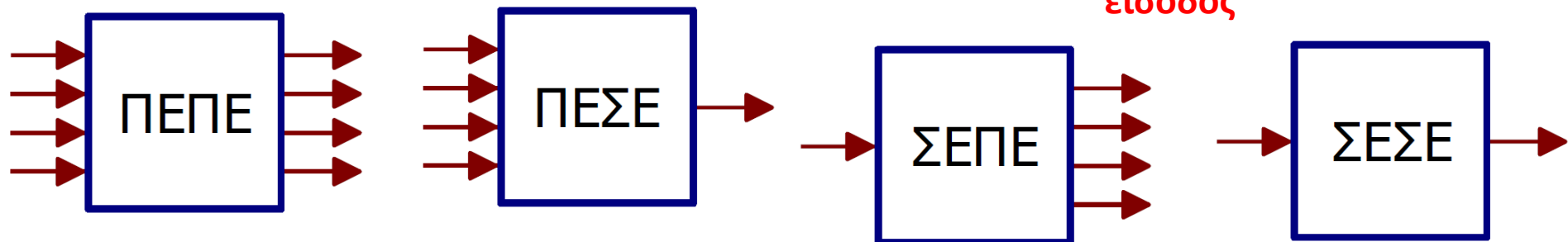
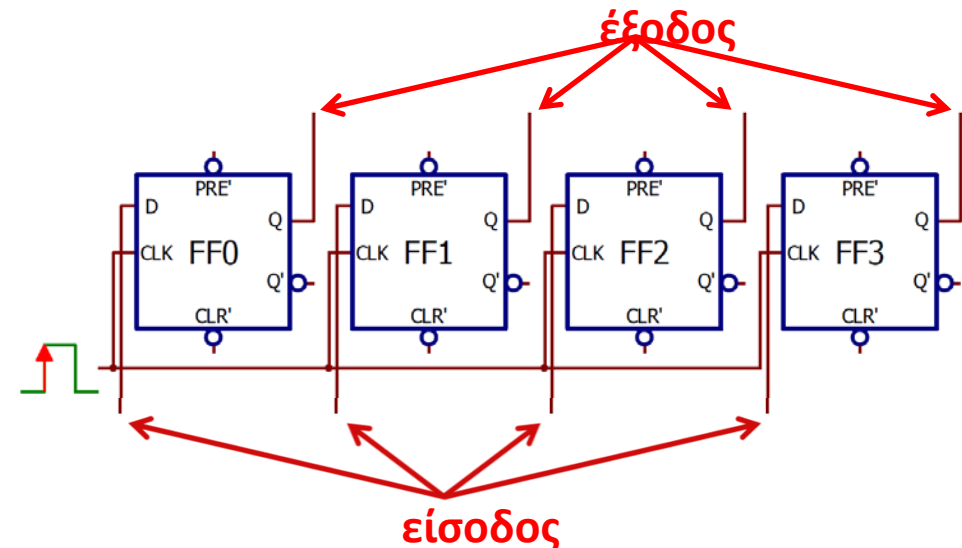
SISO καταχωρητής



- Ο παραπάνω καταχωρητής είναι γνωστός και ως καταχωρητής SISO (Serial In Serial Out).
- **Σημείωση:** Κάθε είσοδος 0 στον καταχωρητή μετατόπισης έχει ως αποτέλεσμα
 - Τη διαίρεση με το 2 εάν είναι ο καταχωρητής μετατόπισης προς τα δεξιά
 - τον πολλαπλασιασμό με το 2 εάν είναι καταχωρητής μετατόπισης προς τα αριστερά

Serial-Parallel Input & Serial-Parallel Output

- Καταχωρητές
 - Σειριακής Εισόδου Σειριακής Εξόδου (ΣΕΣΕ)
 - Σειριακής Εισόδου Παράλληλης Εξόδου (ΣΕΠΕ)
 - Παράλληλης Εισόδου Παράλληλης Εξόδου (ΠΕΠΕ)
 - Παράλληλης Εισόδου Σειριακής Εξόδου (ΠΕΣΕ)



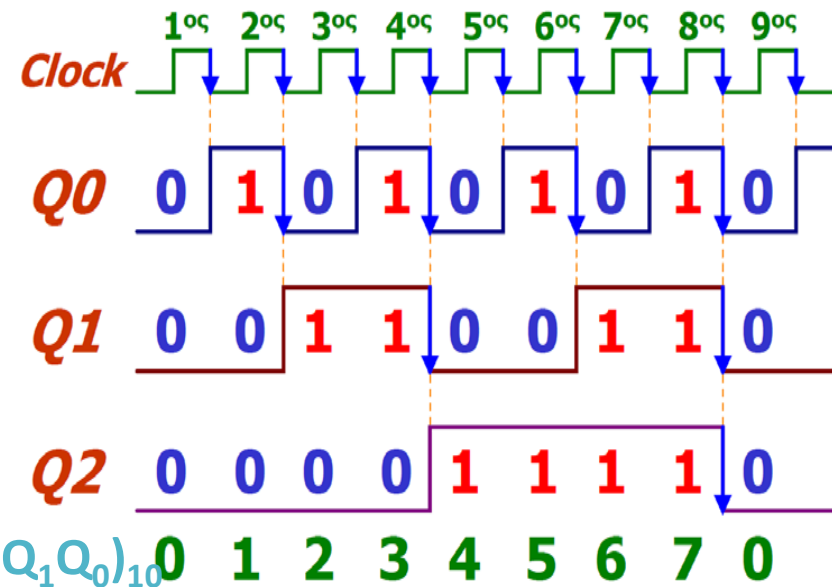
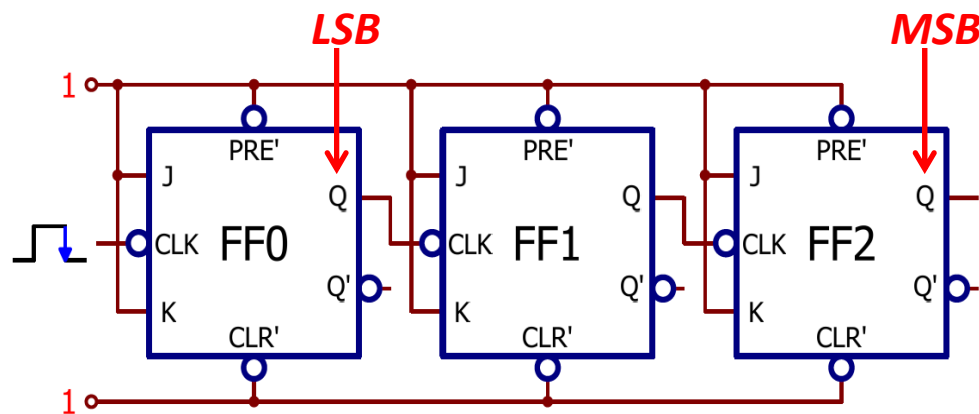


Μετρητές (Counters)

- **n flip-flops απαριθμούν μέχρι 2^n καταστάσεις ($0 \rightarrow 2^n - 1$)**
 - Binary counters: 2^n καταστάσεις (π.χ. mod-8)
 - Non-Binary counters (π.χ. mod-10)
- **Ασύγχρονοι Μετρητές**
 - Το clock διεγείρει μόνο το πρώτο flip-flop. Τα επόμενα flip-flops παίρνουν είσοδο από τα προηγούμενα
- **Σύγχρονοι Μετρητές**
 - Το clock διεγείρει ταυτόχρονα όλα τα flip-flops

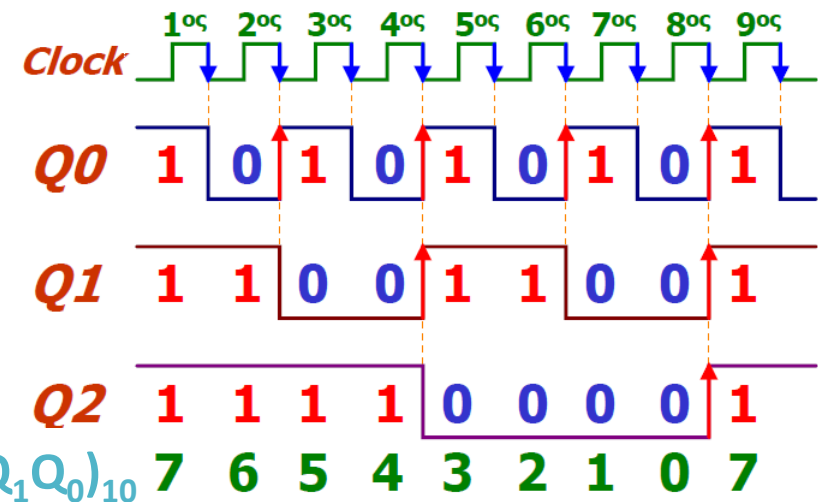
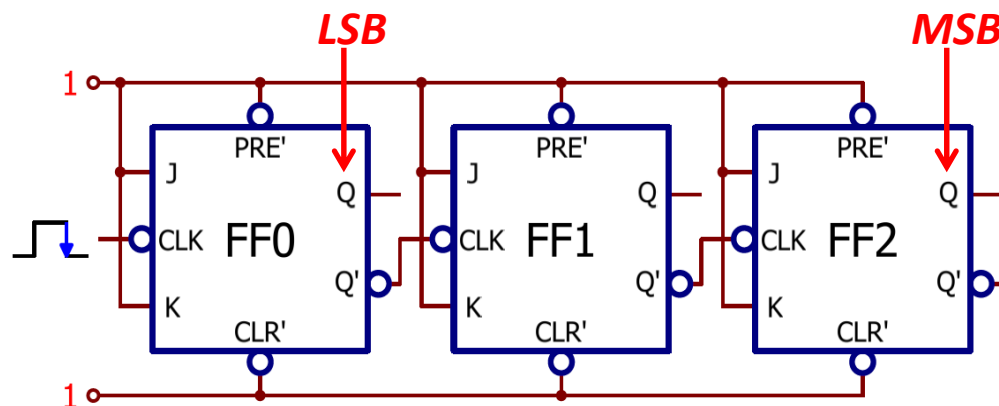
Ασύγχρονοι Μετρητές

- 3-bit count-up binary counter (mod-8)
 - MSB \rightarrow Q2, LSB \rightarrow Q0
 - Διπλασιασμός περιόδου \rightarrow διαίρεση συχνότητας



Ασύγχρονοι Μετρητές

- 3-bit count-down binary counter (mod-8)
 - MSB \rightarrow Q2, LSB \rightarrow Q0
 - Διπλασιασμός περιόδου \rightarrow διαίρεση συχνότητας

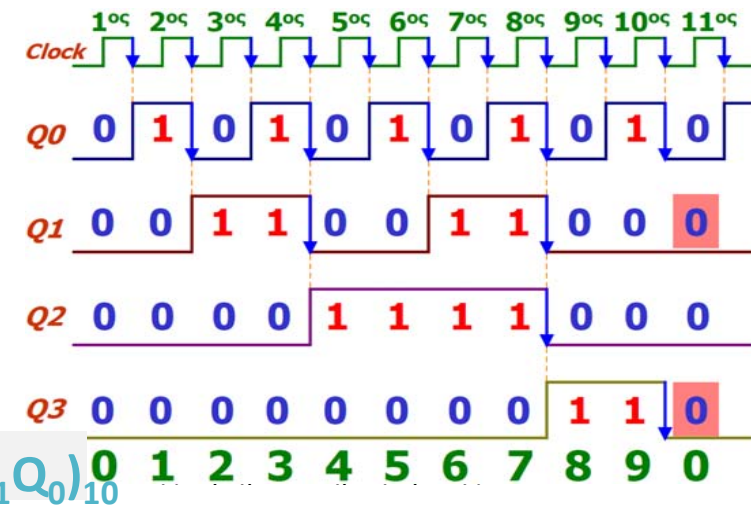
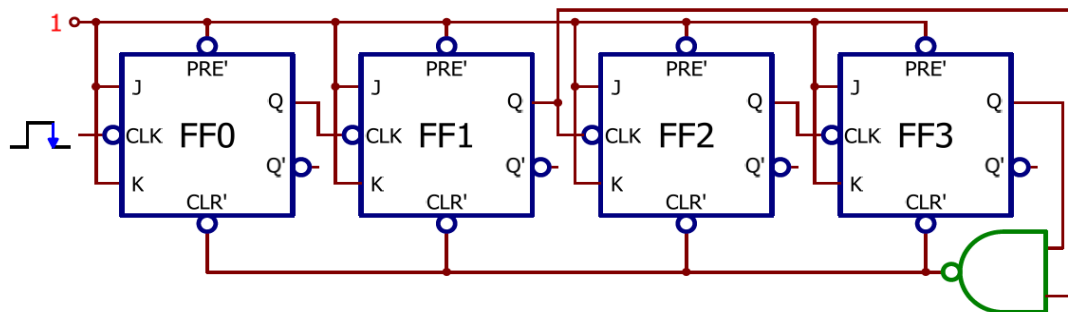


21 April 2021

Γεώργιος Κεραμίδας / Αριστοτέλειο Πα

Ασύγχρονοι Μετρητές

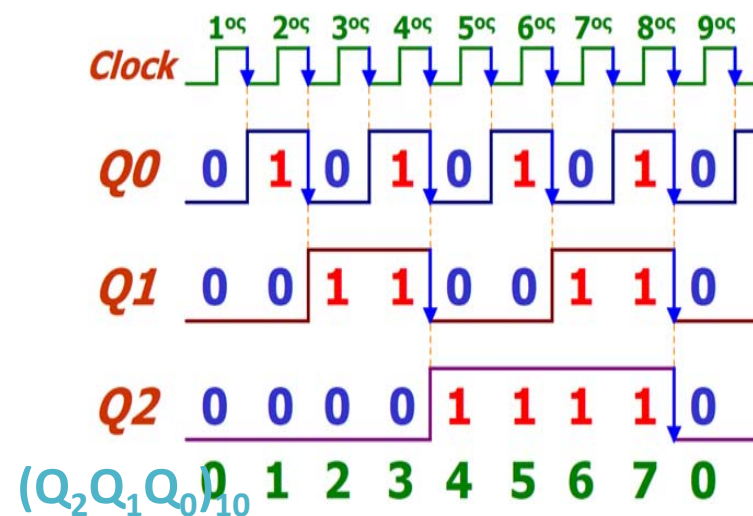
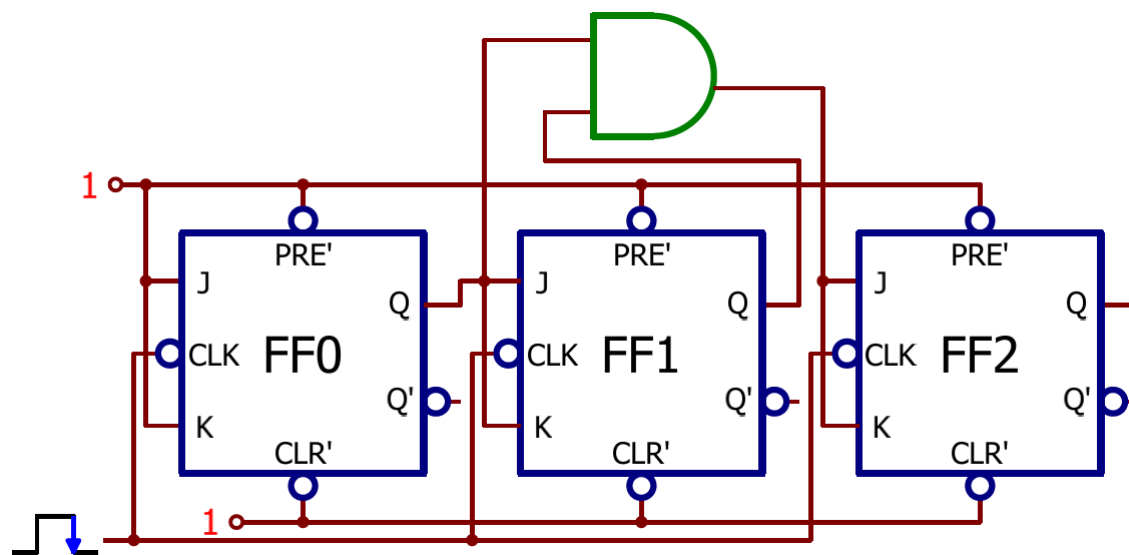
- 4-bit count-up non-binary counter (mod-10)
 - CLEAR FFs στον 10ο παλμό (1001 \rightarrow 0000)
 - 1010 \rightarrow



Σύγχρονοι Μετρητές

• Count-up binary 3-bit (mod-8)

- Q_0 : clock $1 \rightarrow 0$ & $J_0 = K_0 = 1$
- Q_1 : clock $1 \rightarrow 0$ & $J_1 = 1$
- Q_2 : clock $1 \rightarrow 0$ & $J_2 = Q_0 Q_1 = 1$

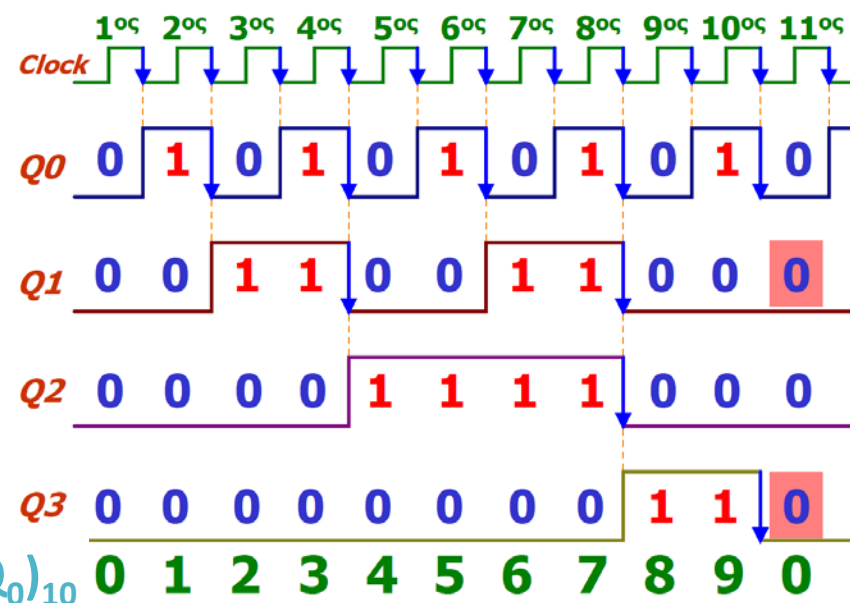




Σύγχρονοι Μετρητές

- Count-up non-binary 4-bit (mod-10)
- Θα χρησιμοποιήσουμε χάρτες Karnaugh

J	K	Q_{n+1}	Q	J	K
0	0	Q_n	$0 \rightarrow 0$	0	X
0	1	0	$0 \rightarrow 1$	1	X
1	0	1	$1 \rightarrow 0$	X	1
1	1	Q_n'	$1 \rightarrow 1$	X	0



Σύγχρονοι Μετρητές

- Count-up non-binary 4-bit (mod-10)
- Θα χρησιμοποιήσουμε χάρτες Karnaugh

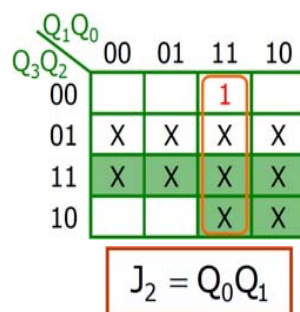
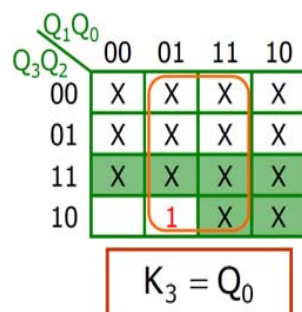
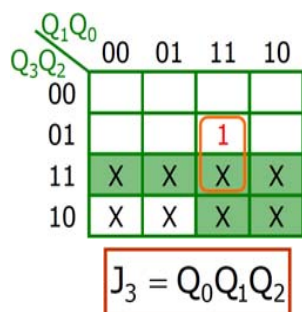
	Προηγούμενη Κατάσταση				Επόμενη Κατάσταση											
	Q ₃	Q ₂	Q ₁	Q ₀	Q ₃	Q ₂	Q ₁	Q ₀	J ₃	K ₃	J ₂	K ₂	J ₁	K ₁	J ₀	K ₀
(Q ₃ Q ₂ Q ₁ Q ₀) ₁₀																
0 → 1	0	0	0	0	0	0	0	1	0	X	0	X	0	X	1	X
1 → 2	0	0	0	1	0	0	1	0	0	X	0	X	1	X	X	1
2 → 3	0	0	1	0	0	0	1	1	0	X	0	X	X	0	1	X
3 → 4	0	0	1	1	0	1	0	0	0	X	1	X	X	1	X	1
4 → 5	0	1	0	0	0	1	0	1	0	X	X	0	0	X	1	X
5 → 6	0	1	0	1	0	1	1	0	0	X	X	0	1	X	X	1
6 → 7	0	1	1	0	0	1	1	1	0	X	X	0	X	0	1	X
7 → 8	0	1	1	1	1	0	0	0	1	X	X	1	X	1	X	1
8 → 9	1	0	0	0	1	0	0	1	X	0	0	X	0	X	1	X
9 → 0	1	0	0	1	0	0	0	0	X	1	0	X	0	X	X	1

Q	J	K
0 → 0	0	X
0 → 1	1	X
1 → 0	X	1
1 → 1	X	0

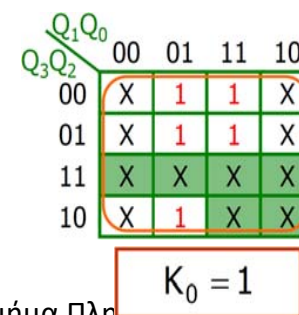
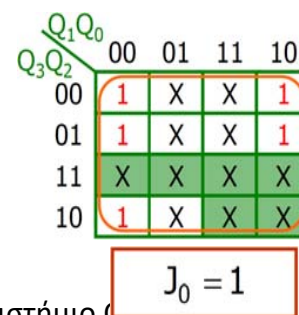
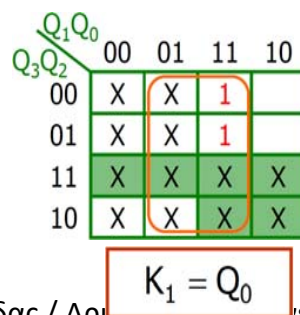
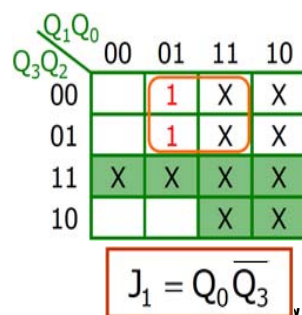
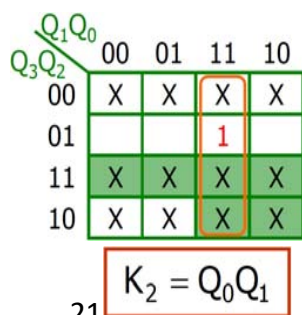
μα Πληροφορικής

Σύγχρονοι Μετρητές

• Count-up non-binary 4-bit (mod-10)

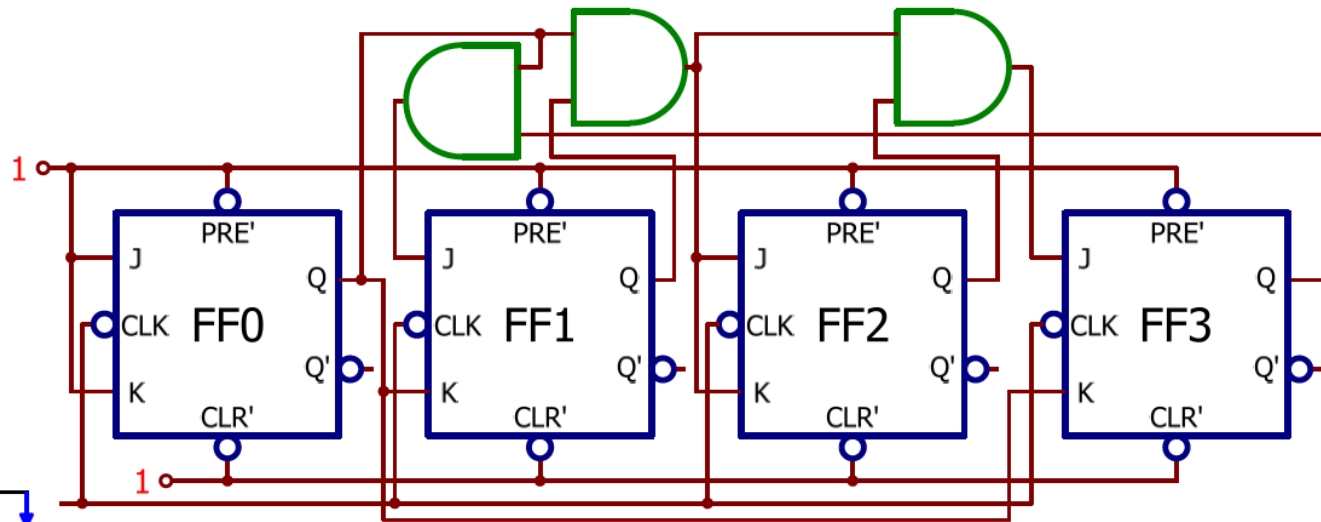


Προηγούμενη Κατάσταση				Επόμενη Κατάσταση											
Q ₃	Q ₂	Q ₁	Q ₀	Q ₃	Q ₂	Q ₁	Q ₀	J ₃	K ₃	J ₂	K ₂	J ₁	K ₁	J ₀	K ₀
0	0	0	0	0	0	0	1	0	X	0	X	0	X	1	X
0	0	0	1	0	0	1	0	0	X	0	X	1	X	X	1
0	0	1	0	0	0	1	1	0	X	0	X	X	0	1	X
0	0	1	1	0	1	0	0	0	X	1	X	X	1	X	1
0	1	0	0	0	1	0	1	0	X	X	0	0	X	1	X
0	1	0	1	0	1	1	0	0	X	X	0	1	X	X	1
0	1	1	0	0	1	1	1	0	X	X	0	X	0	1	X
0	1	1	1	1	0	0	0	1	X	X	1	X	1	X	1
1	0	0	0	1	0	0	1	X	0	0	X	0	X	1	X
1	0	0	1	0	0	0	0	X	1	0	X	0	X	X	1



- 21 April 2021

$$J_3 = Q_0 Q_1 Q_2 = J_2 Q_2$$



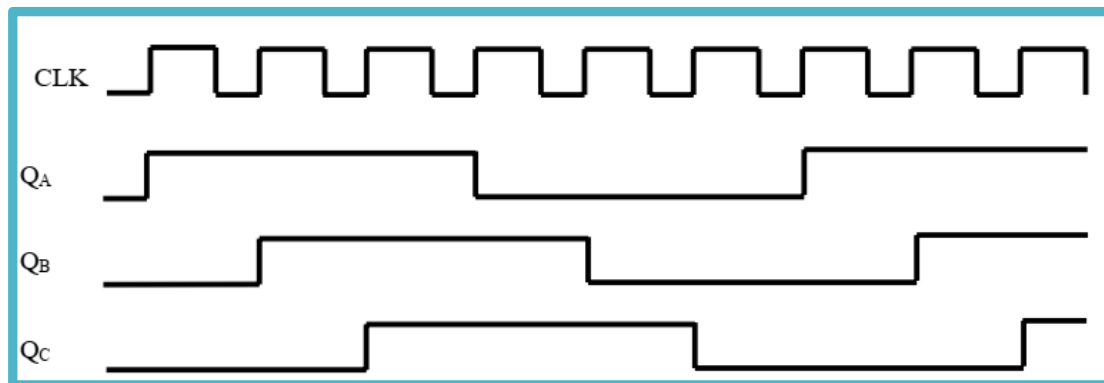
Άσκηση 1 – Εκφώνηση



- Σύγχρονο Ακολουθιακό Κύκλωμα με 3 D-FF Q_A , Q_B , Q_C .
- Οι είσοδοι των FF είναι: $D_A = Q'_C$, $D_B = Q_A$, $D_C = (Q_A + Q_C)Q_B$
- Να σχεδιαστούν οι κυματομορφές των εξόδων του κυκλώματος.

Άσκηση 1 – Λύση

- Πίνακας μετάβασης καταστάσεων. Θεωρώ ότι η αρχική κατάσταση είναι 000
- Από τον πίνακα ότι: $Q_A Q_B Q_C = 000 \rightarrow 100 \rightarrow 110 \rightarrow 111 \rightarrow 011 \rightarrow 001 \rightarrow 000 \dots$



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

- Το κύκλωμα αυτό είναι ο γνωστός απαριθμητής Johnson. Χρησιμοποιείται κυρίως για τη δημιουργία σημάτων χρονισμού με διαφορές φάσης

Άσκηση 2 – Εκφώνηση



- Με FF JK θετικής ακμής πυροδότησης να σχεδιαστεί σύγχρονο ακολουθιακό κύκλωμα το οποίο να διατρέχει τις καταστάσεις 0, 4, 6, 3, 1, 0, 4, 6, ...
- Να σχεδιαστούν οι κυματομορφές των εξόδων του κυκλώματος

Άσκηση 2 – Λύση



- Ζητούμενη ακολουθία: **0, 4, 6, 3, 1, 0, 4, 6, ...**
- Πρώτο βήμα: Πίνακας 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

Άσκηση 2 – Λύση



- Δεύτερο βήμα: Χάρτες Karnaugh

Q _A \ 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 \ 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 \ Q _B Q _C	00	01	11	10
0			X	X
1	1		X	X

$$J_B = Q_A Q_C' = (Q_A' + Q_C)'$$

Q _A \ Q _B Q _C	00	01	11	10
0	X	X	1	1
1	X	X	1	

$$K_B = Q_A' + Q_C$$

Άσκηση 2 – Λύση



- Δεύτερο βήμα: Χάρτες Karnaugh

Q_A	$Q_B Q_C$		00	01	11	10
	0	1		X	X	
1	0	1		X	X	1

$J_C = Q_A Q_B$

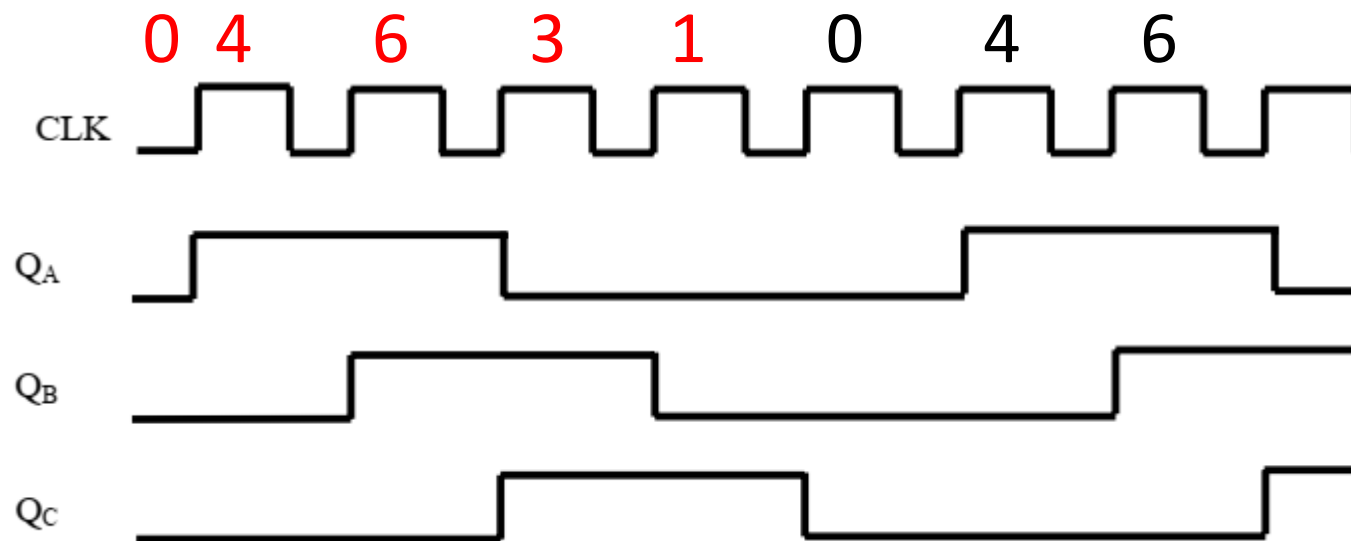
Q_A	$Q_B Q_C$		00	01	11	10
	0	1	X	1		X
1	0	1	X	1	1	X

$K_C = Q_A + Q_B'$

Άσκηση 2 – Λύση



- Τρίτο βήμα: Κυματομορφές



Άσκηση 3 – Εκφώνηση



- Με D-FF αρνητικής ακμής πυροδότησης, να σχεδιαστεί σύγχρονο ακολουθιακό κύκλωμα το οποίο να διατρέχει τις καταστάσεις **0, 1, 2, 3, 4, 5, 6, 7, 0, 1, 2,...**

Άσκηση 3 – Λύση



- Πρώτο βήμα: Πίνακας 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

Άσκηση 3 – Λύση



- Δεύτερο βήμα: Χάρτες Karnaugh

Q _A \ Q _B Q _C	00	01	11	10
0			1	
1	1	1		1

$$\begin{aligned} D_A &= Q_A' Q_B Q_C + Q_A (Q_B' + Q_C') = \\ &= Q_A' (Q_B Q_C) + Q_A (Q_B Q_C)' = \\ &= \text{XOR} (Q_A, Q_B Q_C) \end{aligned}$$

Q _A \ Q _B Q _C	00	01	11	10
0		1		1
1		1		1

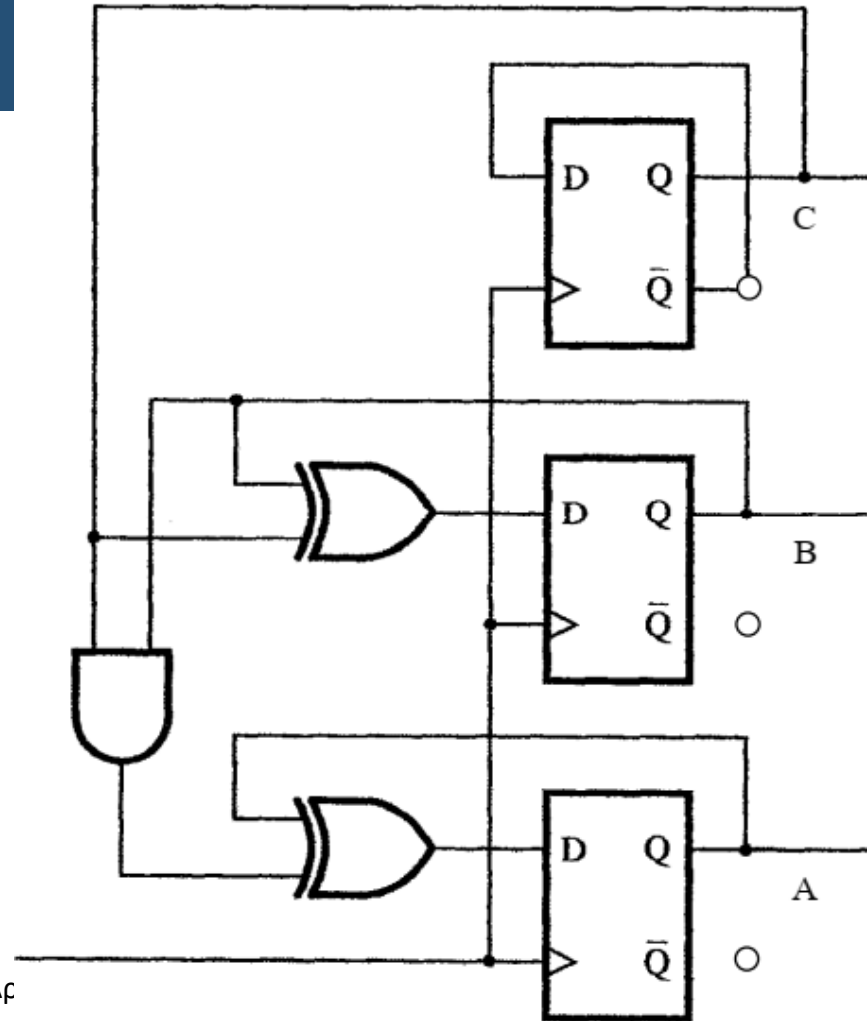
$$D_B = \text{XOR} (Q_B, Q_C)$$

Q _A \ Q _B Q _C	00	01	11	10
0	1			1
1	1			1

$$D_C = Q_C'$$

Άσκηση 3 – Λύση

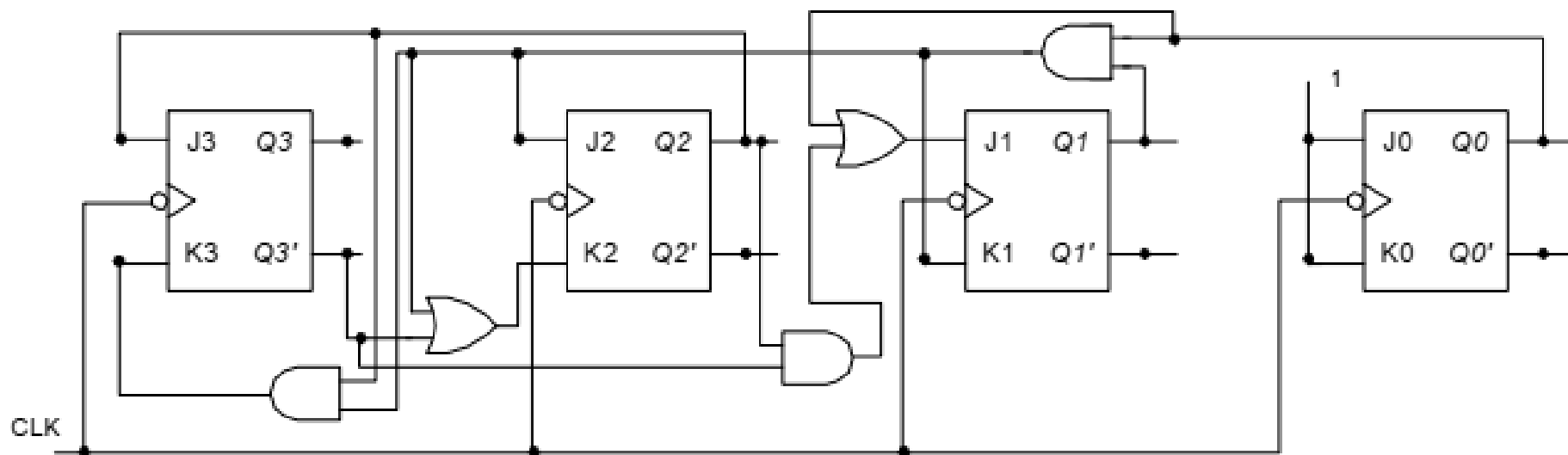
- Κύκλωμα



Άσκηση 4 – Εκφώνηση



- Να αναλύσετε το ακόλουθο κύκλωμα (JK αρνητικής ακμής πυροδότησης)



Άσκηση 4 – Λύση



- Πρώτο βήμα: Συναρτήσεις εισόδου.

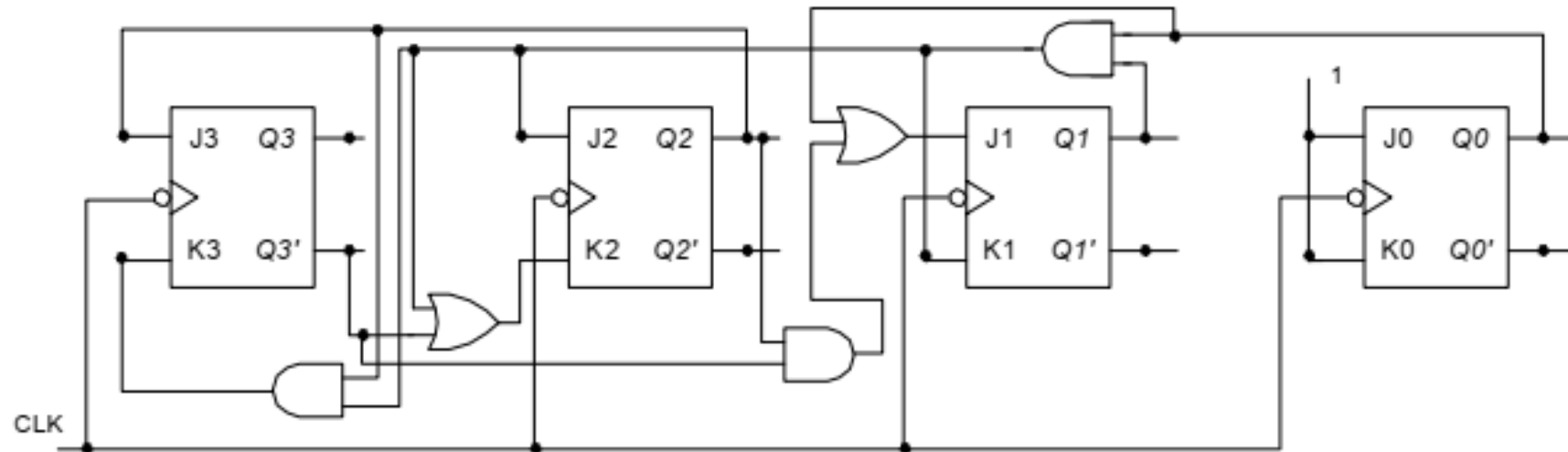
$$J_0 = 1 \quad J_1 = (Q_2 Q_3') + Q_0 \quad J_2 = Q_0 Q_1$$

$$J_3 = Q_2$$

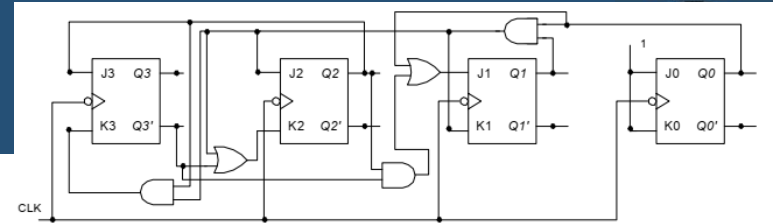
$$K_0 = 1 \quad K_1 = Q_0 Q_1$$

$$K_2 = (Q_0 Q_1) + Q_3'$$

$$K_3 = (Q_0 Q_1) Q_2$$



Άσκηση 4 – Λύση



- Δεύτερο βήμα: Πίνακα Καταστάσεων. 4 FFs → 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

Άσκηση 4 – Λύση

- Τρίτο βήμα: Διάγραμμα Καταστάσεων. Θεωρούμε αρχική κατάσταση την 0000
- $Q_3Q_2Q_1Q_0$: $0 \rightarrow 1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 11 \rightarrow 12 \rightarrow 13 \rightarrow 14 \rightarrow 15$

ΠΑΡΟΥΣΑ ΚΑΤΑΣΤΑΣΗ				ΕΠΟΜΕΝΗ ΚΑΤΑΣΤΑΣΗ			
Q_3	Q_2	Q_1	Q_0	Q_3	Q_2	Q_1	Q_0
0	0	0	0	0	0	0	1
0	0	0	1	0	0	1	0
0	0	1	0	0	0	1	1
0	0	1	1	0	1	0	0
0	1	0	0	1	0	1	1
0	1	0	1	1	0	1	0
0	1	1	0	1	0	1	1
0	1	1	1	1	0	0	0
1	0	0	0	1	0	0	1
1	0	0	1	1	0	1	0
1	0	1	1	1	1	0	0
1	1	0	0	1	1	0	1
1	1	0	1	1	1	1	0
1	1	1	0	1	1	1	1
1	1	1	1	0	0	0	0

πιστήμ

