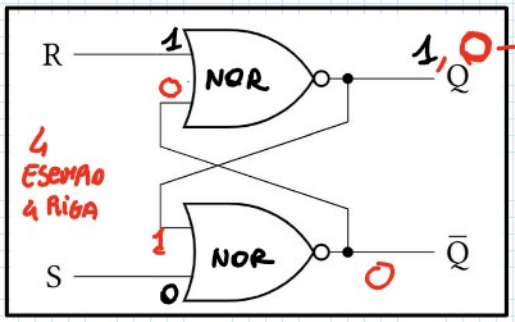
RIASSUNTO
mercoledì 2 novembre 2022 21:27

LATCH SR

N.C = NON CAMBIA
a'è il successivo di Q



NOR

x_1	x_2	y
0	0	1
0	1	0
1	0	0
1	1	0

S	R	Q	$\bar{Q}$
0	0	N.C	N.C
0	1	0	1
1	0	1	0
1	1	x	x

S	R	Q	Q'	SAVE
0	0	0	0	
0	0	1	1	
0	1	0	0	
0	1	1	0	4
1	0	0	1	
1	0	1	1	
1	1	0	indeterminato	
1	1	1	indeterminato	

S=0, R=1 → Q=0, $\bar{Q}$=1 (RESET=0) [IL PRECEDENTE È UGUALE AL PRESENTE]
SE UNO DEGLI INGRESSI È 1, L'USCITA DELLA NOR È ZERO

S=1, R=0 → Q=1, $\bar{Q}$=0 (SET=1) [IL PRECEDENTE È UGUALE AL PRESENTE]

S=0, R=0 → DIPENDE DALL'INGRESSO RETROAZIONATO

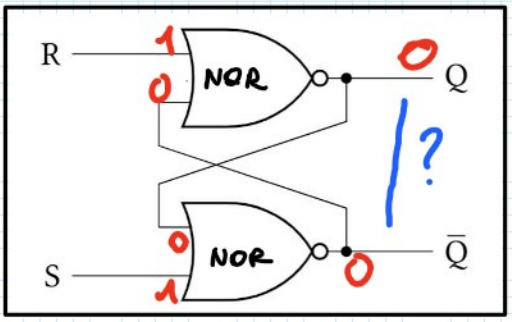
QUINDI SI FANNO DELLE IPOTESI RELATIVE ALLO STATO PRECEDENTE DELL'USCITA Q
L'USCITA PRECEDENTE PUÒ ESSERE 0 o 1.

1. Q=1 → $\bar{Q}$=0, Q=1
   (PARTENDO DA UNO STATO ALTO PRECEDENTE, SI OTTIENE ANCORA UNO STATO ALTO).

2. Q=0 → $\bar{Q}$=1, Q=0
   (PARTENDO DA UNO STATO PRECEDENTE BASSO, SI OTTIENE ANCORA UNO STATO PRESENTE BASSO).

IN ENTRAMBE LE IPOTESI, L'USCITA mantiene lo STATO PRECEDENTE.
IL SISTEMA RICORDA.

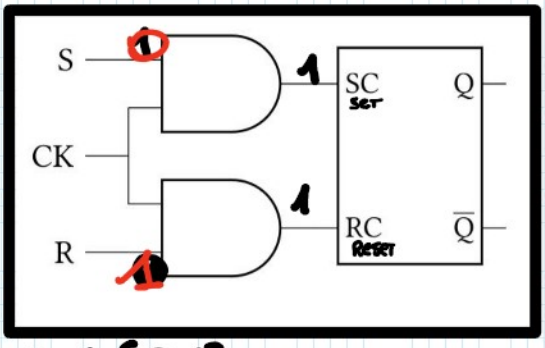
S=1, R=1 NON È PRESA IN CONSIDERAZIONE PERCHÈ È CONTRADDITTORIA.



FLIP FLOP SR

PRIMA DI QUESTO CIRCUITO HO ALTRI CIRCUITI CHE PRIMA O POI SI STABILIZZERÀ, CI VUOLE TEMPO PRIMA CHE UN CIRCUITO COMPLESSO SI STABILIZZI, QUANDO SONO SICURO CHE I VALORI IN INPUT SI STABILIZZANO ALLORA CERCO DI CAMPIONARE

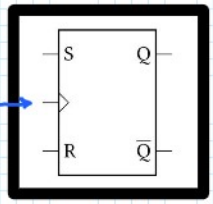
Ci vuole tempo prima che un circuito complesso si stabilizzi, quando sono sicuro che i valori in input si stabilizzano allora cerco di campionare il valore. Quando vado ad effettuare questo campionamento, il circuito si chiama **FLIP FLOP**, un circuito digitale.



CK	S	R	Q	Q'
0	*	*	0	0
0	*	*	1	1
1	0	0	0	0
1	0	0	1	1
1	0	1	0	0
1	0	1	1	0
1	1	0	0	1
1	1	0	1	1
1	1	1	0	×
1	1	1	1	×

CK $\in \{0,1\}$

= segnale di clock →

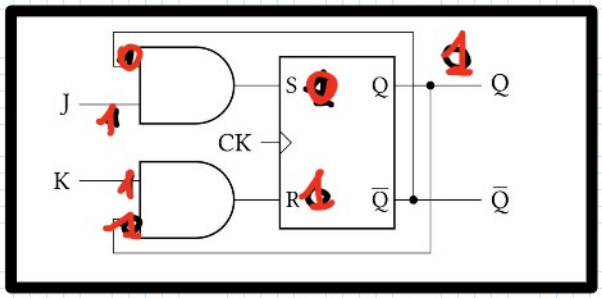


FLIP FLOP JK

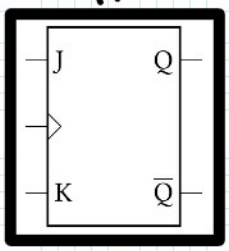
È un flip flop che previene le situazioni oscillanti in un latch. Abbiamo risolto il problema della stabilizzazione dei valori, ma non le oscillazioni.

Dentro al Flip Flop JK abbiamo un flip flop SR.

SR LATCH



=

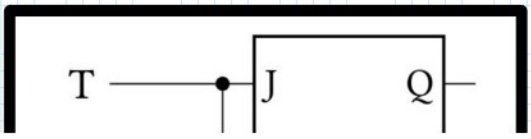


J	K	CK	S	R	Q	Q'	
*	*	0	*	*	0	0	
*	*	0	*	*	1	1	
0	0	1	0	0	0	0	
0	0	1	0	0	1	1	
0	1	1	0	0	0	0	
0	1	1	0	1	1	0	= RESET
1	0	1	1	0	0	1	= SET
1	0	1	0	0	1	1	= SET
1	1	1	1	0	0	1	
1	1	1	0	1	1	0	

CLOCK : 0→1, commutazione fronte di salita

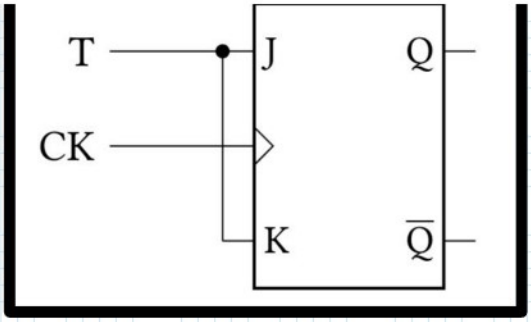
FLIP FLOP T J=K!

A volte è interessante avere un flip flop che si comporta come uno switch. Ogni volta che scrivo un informazione, commuta da 0 a 1.



dipende dall'ingresso RETROAZIONATO

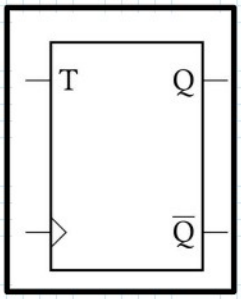
T	J	K	CK	S	R	Q	Q'
0	0	0	1	0	0	0	0



T	J	K	CK	S	R	Q	$\overline{Q}$
0	0	0	1	0	0	0	0
0	0	0	1	0	0	1	1
1	1	1	1	1	0	0	1
1	1	1	1	0	1	1	0

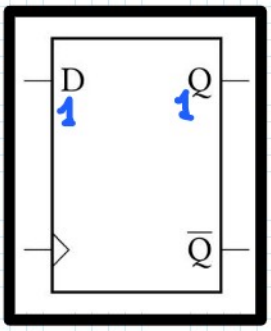
Questo flip flop presenta solo due modi di funzionamento:

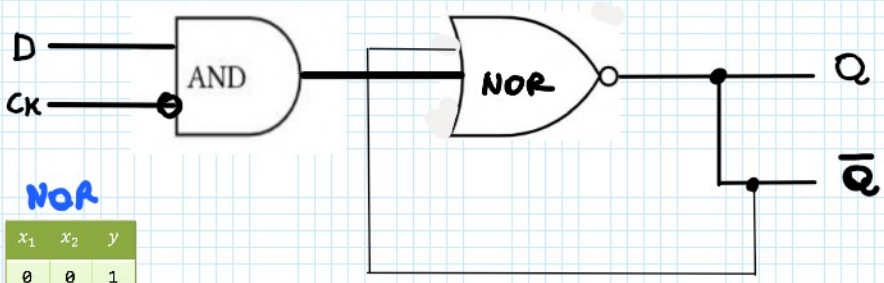
1. se T=0 il flip flop memorizza lo stato precedente;
2. se T=1 lo stato del flip flip commuta cambiando valore (*toggle*).



FLIP FLOP D

QUANDO IL MIO SEGNALE di CLOCK abilita la STRUTTURA, OSSIA quando il SEGNALE vale 1, CIOÉ che i CIRCUITI SI SONO STABILIZZATI, allora memorizziamo l'INFORMAZIONE, e questo oggetto CAMPIONERÀ IL VALORE SU D, SE È 1, allora aggiornerà lo STATO Q a 1, Se è ZERO, lo STATO Q commuterà a ZERO;
QUINDI È UN CIRCUITO BASATO sul latch.





NOR

x_1	x_2	y
0	0	1
0	1	0
1	0	0
1	1	0

D	CK	Q	Q'
1	1	1	1
0	1	0	0

SONO TUTTI CIRCUITI SEQUENZIALI!