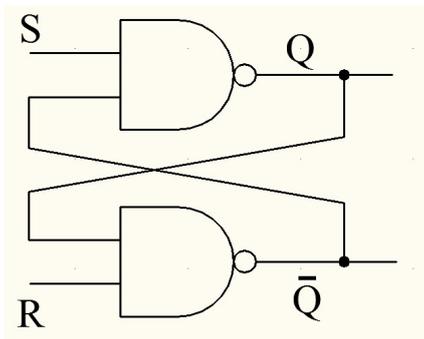


Circuiti combinatori e sequenziali

Un **circuito combinatorio**, ovvero una **rete combinatoria**, è un sistema di porte logiche, connesse opportunamente tra loro, organizzato con un insieme di ingressi e un insieme di uscite, nel quale i valori logici delle uscite sono determinati direttamente e univocamente dai valori logici presenti negli ingressi. Il circuito combinatorio si può rappresentare, complessivamente, come una scatola composta da ingressi e da uscite, con una tabella di verità che stabilisce i valori delle uscite in base ai valori degli ingressi.

Un **circuito digitale** si dice **sequenziale** se l'uscita dipende dagli ingressi applicati e dallo stato precedente della stessa uscita. Un circuito sequenziale, pertanto, deve ricordare il suo *stato precedente* e quindi deve possedere uno o più elementi di *memoria*. In un circuito sequenziale la stessa combinazione di dati in ingresso può dare uscite diverse in base agli stati precedenti.

I circuiti sequenziali elementari per memorizzare un bit sono il Latch SR (SET –RESET) e il Flip Flop JK. (flip the switch si traduce in gira l'interruttore. Una volta gli interruttori per la luce avevano una parte da girare per aprire e rigirare per chiudere da cui flip e flop. Nei circuiti digitali la commutazione è tra 0 e 1). Ci sono diversi tipi di latch e FF, qui vedremo alcuni esempi.



Latch SR (SET-RESET)

Il circuito sequenziale più semplice

Le uscite sono due e per definizione devono essere una complementare all'altra Q (quit) e la sua negazione \bar{Q}

	S	R	Q_n
è la combinazione da evitare sia per incongruenza logica sia perché porterebbe entrambe le uscite al valore 1 e quindi non sarebbero una il complemento dell'altra	0	0	impossibile
si realizza la funzione di set per cui l'uscita Q si porta a 1	0	1	1
si realizza la funzione di reset per cui l'uscita Q si porta a 0	1	0	0
si realizza la funzione di memoria per cui l'uscita conserva il precedente valore memorizzato	1	1	Q_{n-1}

Q_{n-1} e Q_n sono i valori di Q prima e dopo ogni cambiamento degli ingressi S e R. Guardando il grafico accanto vediamo in sequenza come può variare Q in funzione di S e R. Tra gli istanti 0 e 1 abbiamo S = 0, R = 1 quindi, come da tabella, Q = 1. Tra gli istanti 1 e 2 abbiamo S = 1, R = 1 quindi Q_n (valore di Q dopo l'istante 1) è uguale a Q_{n-1} (valore di Q prima dell'istante 1), diciamo che Q non commuta, cioè non cambia e rimane a 1, il latch ha mantenuto in memoria il valore precedente all'istante 1. Tra gli istanti 2 e 3 abbiamo S = 1, R = 0 quindi, come da tabella, Q = 0, in questo caso Q deve commutare.

Tra gli istanti 3 e 4 abbiamo S = 1, R = 1 quindi Q_n (valore di Q dopo l'istante 3) è uguale a Q_{n-1} (valore di Q prima dell'istante 3), diciamo che Q non commuta, cioè non cambia e rimane a 0, il latch ha mantenuto in memoria il valore precedente all'istante 3.

Tra gli istanti 4 e 5 abbiamo S = 0, R = 1 quindi, come da tabella, Q = 1 in questo caso Q deve commutare e passa da 0 a 1.

\bar{Q} negato è sempre il contrario di Q.

Quindi con questo circuito possiamo comandare Q a zero 0 o a 1 oppure memorizzare il bit delle situazioni precedenti. Ci può essere una commutazione (variazione) di Q ad ogni cambiamento di S o R.

Da notare che i cambiamenti di Q avvengono subito dopo il cambiamento di S o R.

Ora vediamo un altro circuito più evoluto: il Flip Flop JK

