

Campionamento e memoria



Engineering Department in Ferrara

Introduzione

Latch di tipo D

Flip-flop

Introduzione

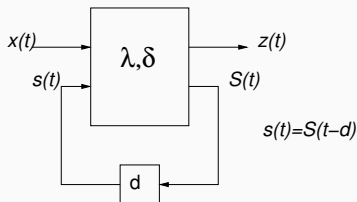
Latch di tipo D

Flip-flop

- Le macchine a stati finiti necessitano di elementi di memoria controllati da un segnale di clock per realizzare un ritardo T
- Questi elementi di memoria sono anch'essi delle macchine a stati finiti di tipo asincrono
- La teoria di questi sistemi é relativamente complessa e non risulta trattabile nell'ambito di questo corso
- Ci si limiterá a considerare i meccanismi base di memorizzazione in un sistema digitale e ad analizzare il comportamento di alcuni elementi di memoria studiandone l'evoluzione dei segnali nel dominio dei tempi

Modello di Huffman per un sistema asincrono

- L'aggiornamento delle variabili di stato non avviene in presenza degli istanti di sincronizzazione, ma in conseguenza di eventi sugli ingressi del sistema o (su altre variabili di stato)
- La memoria é in pratica data dal ritardo associato alle porte logiche e alle interconnessioni del sistema
- Nella figura tale ritardo é concentrato sulle linee di retroazione



Macchine asincrone

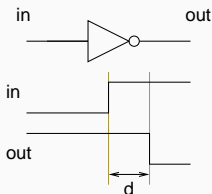
- Il funzionamento di una macchina a stati asincrona dipende dall'evoluzione nel tempo di tutti i segnali (ingressi e variabili di stato futuro)
- Questo ne rende il comportamento dipendente dai transitori della rete combinatoria complicandone (rispetto al caso sincrono) la progettazione
- Considereremo alcuni esempi di circuiti elementari di tipo asincrono che realizzano funzionalità di memoria
- Tali circuiti verranno analizzati al livello strutturale considerando l'evoluzione dei loro segnali interni nel tempo
- Per fare questo dobbiamo considerare il modo in cui i segnali digitali evolvono nel tempo

Fenomeni transitori

- La visione delle reti combinatorie che é stata fornita fino a questo momento é indipendente dal tempo
- In pratica, si é ipotizzato che tutte le porte logiche e quindi la rete abbiano un ritardo nullo
- Il comportamento fisico é invece piuttosto differente, in pratica, ciascun gate é caratterizzato da un ritardo di propagazione

Approfondimento

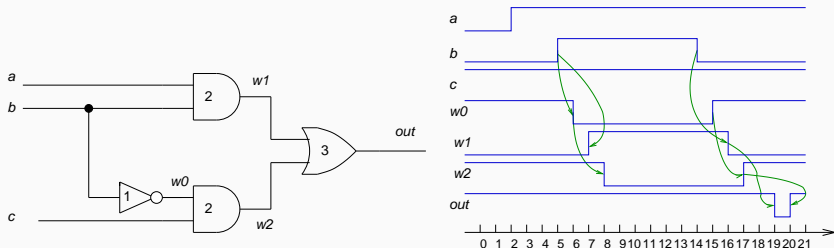
Il ritardo di propagazione di un gate é proporzionale alla sua complessità e al suo fan-out



- La risposta di una rete combinatoria al cambiamento degli ingressi non é istantanea
- Le forme d'onda dei segnali di uscita evolvono nel tempo fino ad assestarsi una volta esauriti i transitori
- Se una configurazione viene mantenuta in ingresso per un tempo sufficientemente grande, allora le uscite si assestano ai valori definiti dalla funzione della rete combinatoria
- Cosa succede prima di questo ?

Esempio di Fenomeni transitori

- Si considera un semplice MPX
- Si applica inizialmente un ingresso $abc = 101$, poi viene applicata la sequenza $abc = 110$ seguita di nuovo da $abc = 101$
- La funzione $out = f(a, b, c)$ é 1 con tutte e 3 le configurazioni di ingresso
- Si osserva però un alea (hazard) in uscita



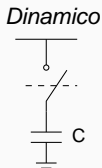
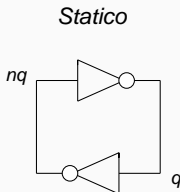
Strumenti per l'analisi dei fenomeni transitori

- Per analizzare i fenomeni transitori, lo strumento principale é la simulazione logica
- Un simulatore é un programma in grado di ricevere in ingresso una rete logica (eventualmente caratterizzata con i ritardi di propagazione dei gate), una sequenza di stimoli e di predire l'evoluzione nel tempo dei segnali del circuito
- Il circuito può essere descritto graficamente o mediante un apposito linguaggio per la descrizione dell'hardware
- Esistono diversi simulatori di dominio pubblico (www.tkgate.org)

Memoria

Nei sistemi digitali, si hanno due possibili modi per memorizzare un informazione

- **statico**: la rete contiene un ciclo non invertente (retroazione) che da luogo a un circuito bistabile che contiene due punti di equilibrio stabile corrispondenti alla memorizzazione di uno 0 e di un 1 (si noti che i punti di equilibrio risultano stabili a causa del guadagno dei gate)
- **dinamico**: l'informazione viene memorizzata (s)caricando un condensatore



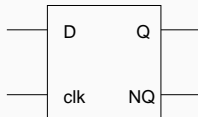
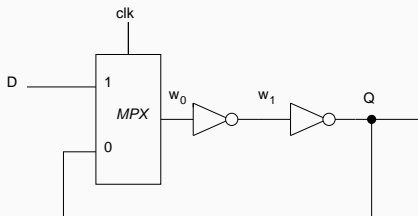
Introduzione

Latch di tipo D

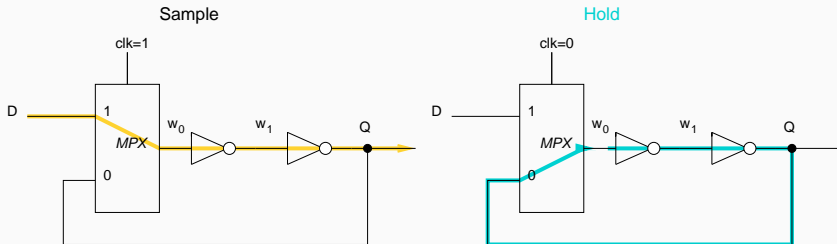
Flip-flop

Latch (tipo D trasparente)

- Nelle memorie statiche nasce il problema di come alterare l'informazione memorizzata (scrittura)
- Un possibile approccio consiste nell'utilizzare un MPX controllato da un segnale (*clk*) che quando é a 1 abilita la fase di scrittura (**sample**) in cui viene campionato il dato *D* in ingresso e quando é a 0 chiude l'anello di retroazione consentendo di mantenere memorizzata (**hold**) l'informazione campionata

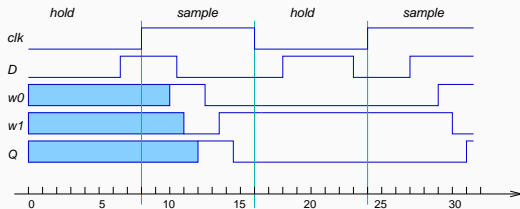
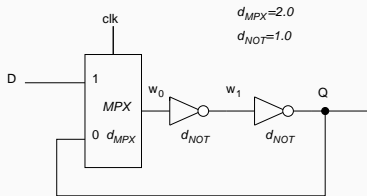


Cammini selezionati nelle due fasi di funzionamento del latch D



Esempio di comportamento

Analisi nel dominio dei tempi del comportamento di un D latch trasparente. Durante la fase di hold, l'uscita mantiene il valore campionato indipendentemente dai cambiamenti dell'ingresso

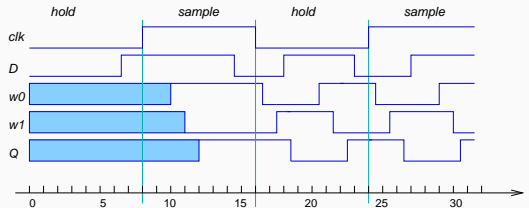
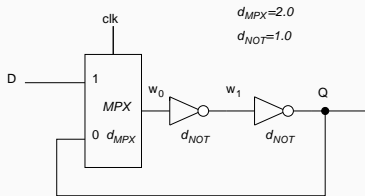


Parametri temporali del funzionamento di un latch

- Il tempo che intercorre fra l'istante di ingresso nella fase di campionamento e l'eventuale nuovo valore di Q , viene definito tempo di risposta (τ_{CQ})
- Il comportamento di un latch é quello corretto a condizione che alcuni condizioni siano verificate, altrimenti si possono presentare dei malfunzionamenti

Malfunzionamento

Se il segnale di ingresso ha avuto dei cambiamenti troppo "vicini" al fronte di discesa del segnale di clock, é possibile che l'anello di retroazione venga chiuso mentre una transizione si sta ancora propagando \Rightarrow oscillazioni



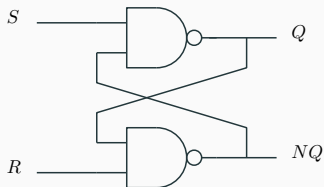
Tempo di setup

- Il fenomeno precedente viene definito come **violazione del tempo di setup**
- Il segnale di ingresso D deve rimanere stabile prima dell'ingresso del componente nella fase di hold per un tempo almeno pari a quello necessario per propagare il valore di D fino a Q
- Tale tempo viene detto τ_{DC} e nell'esempio é uguale al ritardo lungo al cammino: D, w_0, w_1, Q

Latch set - reset (SR)

- Il latch di tipo D é il piú largamente utilizzato
- Esistono altri tipi di latch fra cui quello **set - reset (SR)**
- Anche in questo caso si tratta di una rete asincrona che verrà descritta in maniera intuitiva
- Si lascia per esercizio l'analisi con le forme d'onda

Schema logico e funzionalità



- realizzazione a **NAND** (ne esiste la duale a NOR)
- comandi attivi bassi
 - $S = 0$ (**set**) porta l'uscita a 1
 - $R = 0$ (**reset**) porta l'uscita a 0
 - $S = R = 1$ (**latch**)
memorizzazione

Introduzione

Latch di tipo D

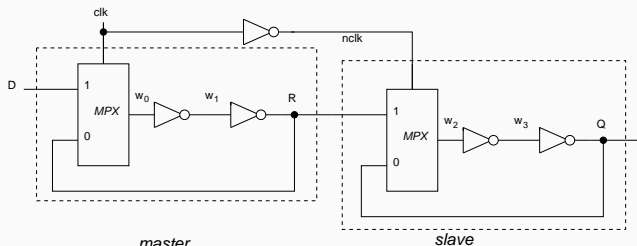
Flip-flop

Istante vs. intervallo di campionamento

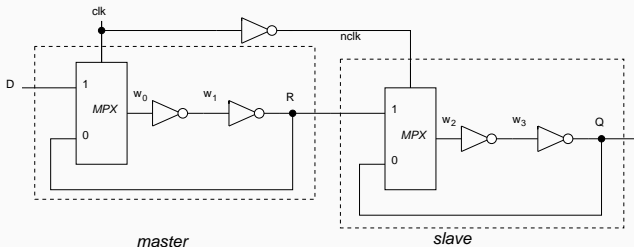
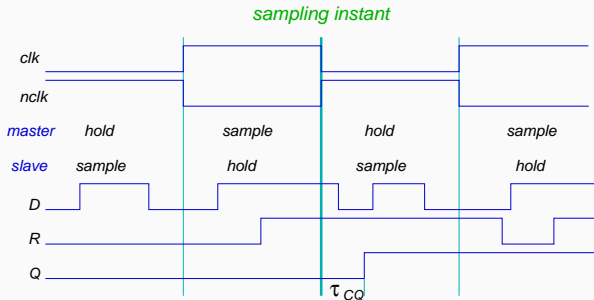
- Nel caso delle reti sincrone si considera un istante di campionamento
- Il latch D trasparente mette invece a disposizione un intervallo di campionamento, e quindi il segnale rimane stabile non per T , ma per il periodo in cui il latch é nella fase di hold
- Per risolvere il problema esistono due soluzioni:
 1. utilizzare una forma d'onda del clock con un periodo di campionamento ridotto
 2. utilizzare elementi di memoria (flip-flop) che campionano in presenza di un evento sul segnale di clock anziché su un livello

Flip-flop D master-slave

- L'idea é quella di utilizzare due latch D connessi in cascata controllati da due segnali di clock opposti
- Mentre il primo (master) campiona il dato, il secondo (slave) lo mantiene stabile in uscita riuscendo così ad avere l'uscita stabile per un periodo di clock
- Si può quindi affermare che tale FF campiona mentre il master passa dallo stato di sample a quello di hold



Analisi del comportamento di un FF D master-slave



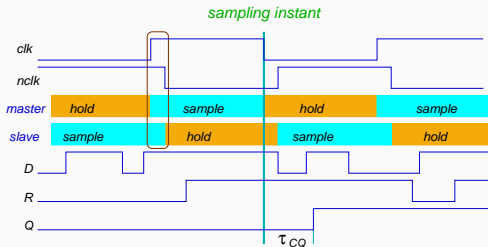
- Tale FF può essere impropriamente definito edge-triggered in quanto il campionamento avviene apparentemente in presenza di un evento (il clock va da 1 a 0), ma in realtà questo effetto è dovuto all'azione combinata di due campionamenti su livelli
- Esistono FF detti true edge-triggered in cui effettivamente è un evento a campionare il dato in ingresso

Aspetti relativi alle temporizzazioni del FF D

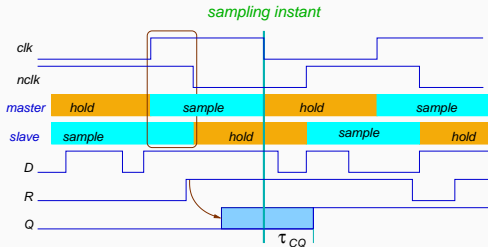
- In maniera simile al caso del latch, D deve rimanere stabile per un tempo pari a τ_{CD} (tempo di setup) prima della transizione in discesa del clock
- Si ha anche un tempo di hold (τ_{CD}) che si misura a partire dall'evento di campionamento e durante il quale D deve stare stabile
- Se queste condizioni sono verificate l'uscita Q assume il valore di D quando il clock passa da 1 a 0 con un ritardo detto tempo di risposta τ_{CQ}
- **Bisogna fare attenzione ai ritardi della rete che potrebbero dare luogo a malfunzionamenti**

Esempio di malfunzionamenti (feedthrough)

- **Caso senza problemi:**
il ritardo del NOT é $<$
di quello del master



- **Caso con problemi:** il ritardo del NOT é $>$ di quello del master
- Le due fasi di *sample* di master e slave si sovrappongono
- Il dato passa in uscita

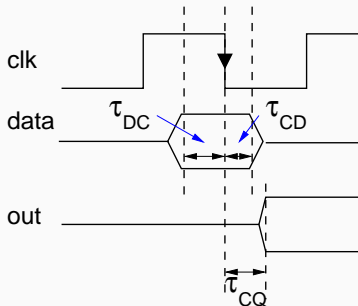
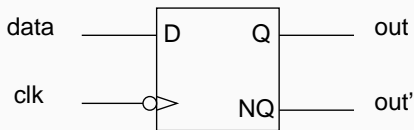


Equazione caratteristica del FF di tipo D

Il FF di tipo *D* può essere astratto come una macchina sincrona con le seguenti equazioni

- stato futuro, δ : $Q_{k+1} = D_k$ (detta equazione caratteristica)
- uscita, uscita λ : $out = Q_k$

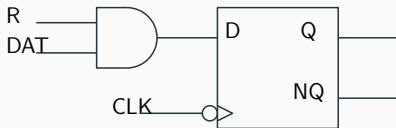
Simbolo e vincoli sulla dinamica di *D* e *clk*



FF D: comandi di reset e enable

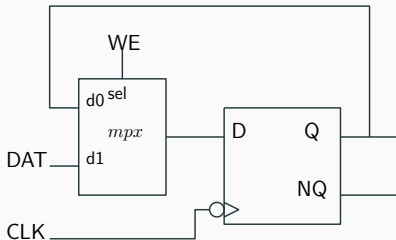
FF di tipo D con comando di reset (R attivo basso) sincrono:

$Q_{k+1} = R_k DAT_k$ (serve per portare in uno stato noto tutti i FF)



FF di tipo D con comando di write enable (WE):

$Q_{k+1} = Q_k \cdot WE'_k + DAT_k WE_k$ (serve per evitare di campionare il dato quando non interessa)



- Si sono visti latch e flip-flop che servono per realizzare una rete sequenziale sincrona
- Questi componenti vengono forniti nelle librerie di progetto insieme ai gate e al livello logico non importa descriverli al livello gate
- I flip-flop possono essere organizzati in array di n FF che ricevono n ingressi in parallelo e producono n uscite in parallelo