



Ritardi e alee delle reti combinatorie

▼ Creatore originale: @LucaCaffa

Ritardi nelle Porte Logiche

Ritardo di Propagazione

Ogni porta logica introduce un certo ritardo tra la variazione dell'ingresso e la corrispondente variazione dell'uscita.

Ritardi di Trasporto (Transport Delays)

I ritardi di trasporto sono i seguenti:

- t_{PLH} : tempo di transizione da livello basso a alto (Low-to-High);
- t_{PHL} : tempo di transizione da livello alto a basso (High-to-Low).

Ritardi Inerziali (Inertial Delays)

I ritardi inerziali si hanno quando si parla di transizioni non istantanee, con un andamento lineare nel tempo. In pratica, i segnali impiegano un tempo finito per stabilizzarsi.

Si considereranno **solo i ritardi di propagazione** per semplicità, ignorando i ritardi inerziali.

Alee (Hazards) nelle reti combinatorie

Le alee sono transizioni impulsive nell'uscita, causate dai ritardi di propagazione asimmetrici tra i percorsi logici. La causa è il disallineamento temporale tra segnali

che dovrebbero cambiare simultaneamente. Si hanno, quindi, delle brevi instabilità nell'uscita, che però possono essere mascherati.

Alee Statiche

Le alee statiche si verificano quando l'uscita dovrebbe rimanere stabile, ma mostra un impulso indesiderato a causa di ritardi di trasporto.

Esempio - Circuito con Alea statica

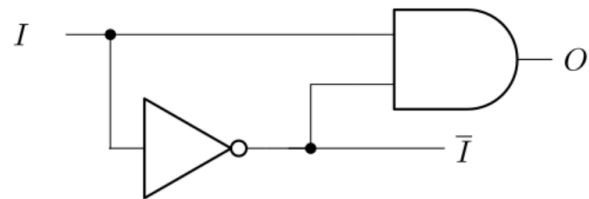
1. Prendiamo in considerazione il circuito a lato, che comprende una porta AND e una porta NOT.

In teoria, l'uscita O non dovrebbe mai essere 1, perché in ingresso alla porta AND arrivano i segnali I e \bar{I} .

In realtà, è presente un breve istante in cui l'uscita O è 1.

2. Se guardiamo il segnale I , ad un certo istante va a 1.

La porta NOT, però, non manda immediatamente il segnale \bar{I} (quindi 0) alla porta AND, e, quindi, per un breve istante si vede che l'uscita O diventa 1, portando ad avere un'alea statica.



Circuito con una porta AND e una porta NOT

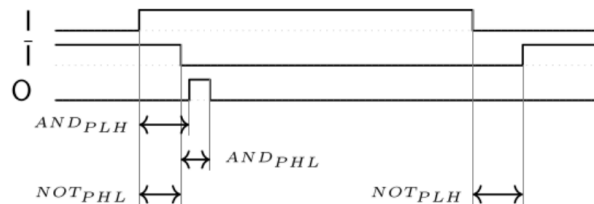


Diagramma temporale del circuito sopra.

Analizziamo i ritardi uno per volta:

- AND_{PLH} è il tempo che la porta AND ci mette a passare da 0 a 1, ed inizia quando I va a 1, terminando quando la porta AND imposta 1 in uscita;
- NOT_{PLH} è il tempo che la porta NOT ci mette a far passare il segnale \bar{I} da 0 a 1, iniziando nell'istante in cui I cambia;

- AND_{PHL} è il tempo che la porta AND ci mette a passare da 1 a 0, iniziando quando \bar{I} diventa 0;
- NOT_{PLH} è il tempo che la porta NOT ci mette a passare da 0 a 1, iniziando quando I passa a 0.

Eliminazione delle Alee Statiche

Teorema Fondamentale

Il seguente **teorema fondamentale** definisce la **condizione di libertà da Alee**: una rete combinatoria a due livelli (SOP - Sum of Products) è libera da alee statiche se, per ogni coppia di 1 adiacenti nella mappa di Karnaugh, esiste un implicante primo che li copre entrambi.

Metodo Sistematico

Come metodo sistematico, si definiscono due punti:

1. identificare le **transizioni critiche**, cercando coppie di 1 adiacenti nella mappa non coperti dallo stesso implicante;
2. aggiungere gli implicanti ridondanti, introducendo termini di prodotto aggiuntivi per mascherare le alee, anche se ciò aumenta la complessità del circuito.

Esempio - Correzione

La K-map a lato mostra una possibile alee, poiché sappiamo, dal **teorema fondamentale**, che due 1 adiacenti non coperti dallo stesso "rettangolo" possono creare una alee.

		AB		A	
		00	01	11	10
C	0	0	0	1	1
C	1	0	1	1	0
		B			

K-map con un possibile problema di alee

Possiamo risolvere il problema eliminando la alea, per cui dobbiamo prendere un implicante aggiuntivo ridondante, in modo da coprire ogni 1 adiacente con uno stesso implicante.

Dalla [figura a lato](#) si vede l'implicante aggiuntivo.

		A			
		AB		11	10
C	0	0	0	1	1
	1	0	1	1	0

B

K-map con problema di alea risolto

Trade-off

Per garantire la stabilità temporale, e quindi nessuna presenza di alea statiche, bisogna aumentare il numero di porte logiche. Il costo aumenta e il circuito non è minimo, ma si ha la sicurezza di non avere alee.