**Esercizio Propedeutico**

*Settembre 2023*

Si supponga che un progettista abbia disponibile come elemento di memoria un circuito asincrono il cui funzionamento è descrivibile dal seguente codice VerilogHDL:

**module** circuito(a,b,clk,res,z);

input a,b,clk,res;

inout z;

wire y1,y2,y1f,y2f;

assign y1= (y1f&(a|b|~clk|~y2f)|(~a&b&clk&~y2f));

assign y2= ((y1&~clk)|(~y1&clk));

assign #4 y1f=y1&~res;

assign #4 y2f=y2&~res;

assign z = y1f;

**endmodule**

IL candidato, impiegando detto elemento realizzi uno shift register a 3bit.

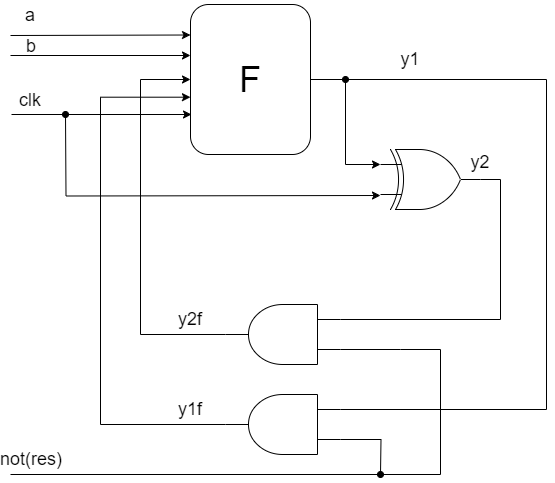
**Traccia**:

1. Si analizzi **DETTAGLIATAMENTE** il funzionamento del circuito asincrono proposto, sia analiticamente che **attraverso opportune simulazioni** atte a validare i risultai ottenuti.
2. Si individui quindi la mappa delle eccitazioni che caratterizza tale elemento di memoria.
3. Si passi a descrivere la macchina sequenziale che realizza lo “shift register” desiderato.
4. Sfruttando le mappe delle eccitazioni trovata al punto 2 si definiscano le equazioni di eccitazione.
5. Si Completi la realizzazione del circuito.
6. Si Simuli il funzionamento del circuito finale istanziando l’elemento di memoria descritto nel testo quante volte serve.

## Svolgimento

**Considerazioni iniziali**: già da un’analisi superficiale del codice si evince che la cella di memoria fornita è dotata di due segnali di controllo (a e b) un segnale di sincronismo (clk) ed un segnale di reset (res) il quale, quando attivato a livello alto, porterà ad azzerare i segnali y1f ed y2f che altrimenti vengono a coincidere con i segnali y1 ed y2. Essi appaiono essere quindi i segnali di “feedback” ovvero quei i segnali interni che servono a “ricordare” al circuito lo stato precedente. Nel momento in cui, per una particolare combinazione di ingressi (ed in assenza del reset), i segnali generati y1 ed y2 dovessero coincidere con i segnali y1f ed y2f, il sistema si viene a trovare in una situazione di stabilità.

Il segnale d’uscita, non riportato nello schema, coincide con la variabile di stato y1f.



Si va quindi ad analizzare come y1 ed y2 vengono a dipendere da a,b,clk nonché da y1f ed y2f (che poi è quanto espresso all’interno del codice VerilogHDL)

Le tabelle di verità delle due funzioni sono pertanto:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| y1 |  | clk=0 |  |  |  | clk=1 |  |  |
| y1f,y2f\ab | 00 | 01 | 11 | 10 | 00 | 01 | 11 | 10 |
| 00 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| 01 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 10 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 |
| 11 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 |
|  |  |  |  |  |  |  |  |  |
| y2 |  | clk=0 |  |  |  | clk=1 |  |  |
| y1f,y2f\ab | 00 | 01 | 11 | 10 | 00 | 01 | 11 | 10 |
| 00 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 |
| 01 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |
| 10 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 |
| 11 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 |
|  |  |  |  |  |  |  |  |  |

Che riunite insieme forniscono la seguente tavola di flusso

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| y1,y2 |  | clk=0 |  |  |  | clk=1 |  |  |
| y1f,y2f\ab | 00 | 01 | 11 | 10 | 00 | 01 | 11 | 10 |
| 00 | 00 | 00 | 00 | 00 | 01 | 10 | 01 | 01 |
| 01 | 00 | 00 | 00 | 00 | 01 | 01 | 01 | 01 |
| 10 | 11 | 11 | 11 | 11 | 01 | 10 | 10 | 10 |
| 11 | 11 | 11 | 11 | 11 | 01 | 10 | 10 | 10 |

Dove in rosso sono evidenziate le condizioni di stabilità

Fornendo un nome mnemonico per ciascuno stato si può ottenere la seguente tavola di Huffman

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| y1,y2 |  | clk=0 |  |  |  | clk=1 |  |  |  |
| y1f,y2f\ab | 00 | 01 | 11 | 10 | 00 | 01 | 11 | 10 | Out |
| A | A | A | A | A | B | D | B | B | 0 |
| B | A | A | A | A | B | B | B | B | 0 |
| C | C | C | C | C | B | D | D | D | 1 |
| D | C | C | C | C | D | D | D | D | 1 |

**Considerazioni:** da una attenta analisi di tale tavola si evince che:

1. Fintanto che il segnale “clk” rimane stabile (a 0 o a 1) la macchina NON cambia di stato
2. Durante il fronte di discesa del clock la macchina passa sempre ad uno stato che ha la medesima uscita (da B finisce sempre in A e da D finisce sempre in B)
3. Ciò implica che semmai vi è una modifica dell’uscita questa avverrà SOLO durante il fronte positivo del clock
4. Perché l’uscita passi da 0 ad 1 vi è una sola possibilità ovvero che i segnali di controllo a,b siano al valore 0,1 ed arrivi un fronte di salita del clock
5. Perché l’uscita si disattivi ovvero passi da 1 a 0 vi è una sola possibilità ovvero che i segnali di controllo a,b siano al valore 0,0 ed arrivi un fronte di salita di clock
6. In tutte le altre condizioni la cella mantiene l’uscita inalterata.

Quindi concludendo la cella di memoria **appare** come un **FF di tipo D** sincronizzato sul **fronte di salita del clock**, dotata di un segnale di **“enable” attivo basso** (b) ed un segnale di **reset asincrono**.

Attraverso una **corretta** simulazione si può verificare quanto ipotizzato:



Si noti che una “**corretta simulazione**” dovrebbe: fornire segnali congrui col funzionamento atteso del circuito; ad esempio un segnale di clock che fornisca una corretta cadenza, ovvero sia regolare e presenti molteplici istanti di campionamento, dei segnali di controllo che prendano in esame più situazioni possibili, inoltre essendo l’analisi del circuito asincrono si cercherà (per quanto possibile) di evitare la commutazione simultanea di più segnali di ingresso, in quanto questa situazione sarebbe di difficile interpretazione oltre che portatrice di alee !

La tavola di flusso del dispositivo, inteso ora come **circuito sincrono,** una volta assodato che il segnale di clock fornisce i sincronismi sul fronte positivo, potrebbe essere rappresentata come segue:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| st\a,b | 00 | 01 | 11 | 10 | Out |
| 0 | 0 | 1 | 0 | 0 | 0 |
| 1 | 0 | 1 | 1 | 1 | 1 |

Ovvero la sua mappa di transizione può essere rappresentata attraverso la seguente tabella

|  |  |  |  |
| --- | --- | --- | --- |
|  | Transizione | a | b |
| 0 | 0-->0 | -/1 | 0/- |
| 1' | 0-->1 | 0 | 1 |
| 0' | 1-->0 | 0 | 0 |
| 1 | 1-->1 | -/1 | 1/- |

Espresso a parole:

* Per mantenere l’uscita a 0 si può
  + Mettere qualsiasi valore ad “a” purchè “b” sia a 0
  + Mettere qualsiasi valore a “b” purchè a sia a 1
* Per attivare la transizione da 0 a 1 “a” deve essere 0 e “b” deve essere 1
* Per attivare la transizione da 1 a 0 “a” deve essere 0 e “b” deve essere 0
* Per mantenere l’uscita a 1 si può
  + Mettere “a” ad un valore qualsiasi purchè “b” sia a 1
  + Mettere “b” ad un valore qualsiasi purchè “a” sia ad 1

A questo punto si dispone di un particolare elemento di memoria (Flip Flop) e hanno tutte le informazioni su come pilotarlo per creare le transizioni desiderate, esso può quindi essere impiegato per la realizzazione di QUALSIASI circuito sincrono.

Si prenda quindi in esame il circuito desiderato: esso è uno “shift resister” a 3 bit.

**Trattazione a carattere generale**

Si descriva la macchina sincrona che si va a realizzare attraverso la sua tavola di Huffman oppure direttamente attraverso la tavola di flusso: Essa sarà dotata di 3 variabili di stato e di un ingresso, mentre l’uscita coinciderà con le variabili di stato stesse.

Il funzionamento della macchina è riassunto nelle seguenti tavole di flusso dove si sono evidenziate in rosso le transizioni nella variabili di stato da 1 a 0 o da 0 a 1

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| y1,y2,y3\x | 0 | 1 |  | y1,y2,y3\x | 0 | 1 |
| 000 | 000 | 001 |  | 000 | 000 | 001 |
| 001 | 010 | 011 |  | 001 | 010 | 011 |
| 011 | 110 | 111 |  | 011 | 110 | 111 |
| 010 | 100 | 101 |  | 010 | 100 | 101 |
| 100 | 000 | 001 |  | 100 | 000 | 001 |
| 101 | 010 | 011 |  | 101 | 010 | 011 |
| 111 | 110 | 111 |  | 111 | 110 | 111 |
| 110 | 100 | 101 |  | 110 | 100 | 101 |

Sfruttando la tavola delle transizioni dedotta prima si può arrivare quindi a stabilire quali eccitazioni fornire ai tre FF che memorizzano i valori di y1,y2, ed y3 in base alla tavola appena vista

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **Y1'** |  | **a1** |  | **b1** |  |
| y1,y2,y3\x | 0 | 1 | 0 | 1 | 0 | 1 |
| 000 | 0 | 0 | -/1 | -/1 | 0/- | 0/- |
| 001 | 0 | 0 | -/1 | -/1 | 0/- | 0/- |
| 011 | 1 | 1 | 0 | 0 | 1 | 1 |
| 010 | 1 | 1 | 0 | 0 | 1 | 1 |
| 100 | 0 | 0 | 0 | 0 | 0 | 0 |
| 101 | 0 | 0 | 0 | 0 | 0 | 0 |
| 111 | 1 | 1 | -/1 | -/1 | 1/- | 1/- |
| 110 | 1 | 1 | -/1 | -/1 | 1/- | 1/- |
|  |  |  |  |  |  |  |
|  | **Y2'** |  | **a2** |  | **b2** |  |
| y1,y2,y3\x | 0 | 1 | 0 | 1 | 0 | 1 |
| 000 | 0 | 0 | -/1 | -/1 | 0/- | 0/- |
| 001 | 1 | 1 | 0 | 0 | 1 | 1 |
| 011 | 1 | 1 | -/1 | -/1 | 1/- | 1/- |
| 010 | 0 | 0 | 0 | 0 | 0 | 0 |
| 100 | 0 | 0 | -/1 | -/1 | 0/- | 0/- |
| 101 | 1 | 1 | 0 | 0 | 1 | 1 |
| 111 | 1 | 1 | -/1 | -/1 | 1/- | 1/- |
| 110 | 0 | 0 | 0 | 0 | 0 | 0 |
|  |  |  |  |  |  |  |
|  | **Y3'** |  | **a3** |  | **b3** |  |
| y1,y2,y3\x | 0 | 1 | 0 | 1 | 0 | 1 |
| 000 | 0 | 1 | -/1 | 0 | 0/- | 1 |
| 001 | 0 | 1 | 0 | -/1 | 0 | 1/- |
| 011 | 0 | 1 | 0 | -/1 | 0 | 1/- |
| 010 | 0 | 1 | -/1 | 0 | 0/- | 1 |
| 100 | 0 | 1 | -/1 | 0 | 0/- | 1 |
| 101 | 0 | 1 | 0 | -/1 | 0 | 1/- |
| 111 | 0 | 1 | 0 | -/1 | 0 | 1/- |
| 110 | 0 | 1 | -/1 | 0 | 0/- | 1 |

Ora **Cella per cella** si può “scegliere” se sia più conveniente l’opzione riportata a destra o quella riportata a sinistra, purchè la medesima scelta sia fatta tanto per il segnale di eccitazione “an” che per “bn”

A scopo **Puramente didattico** si potrebbe ad esempio scegliere, per quanto riguarda la variabile y1, l’opzione di destra per le prime due righe e l’opzione di sinistra per le ultime due.   
Si ribadisce che tale scelta è puramente didattica e non porta alcun beneficio nella realizzazione del circuito.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | ***Y1'*** |  | ***a1*** |  | ***b1*** |  | ***a1*** |  | ***b1*** |  |
| *y1,y2,y3\x* | *0* | *1* | *0* | *1* | *0* | *1* | *0* | *1* | *0* | *1* |
| *000* | *0* | *0* | *-/1* | *-/1* | *0/-* | *0/-* | *1* | *1* | *-* | *-* |
| *001* | *0* | *0* | *-/1* | *-/1* | *0/-* | *0/-* | *1* | *1* | *-* | *-* |
| *011* | *1* | *1* | *0* | *0* | *1* | *1* | *0* | *0* | *1* | *1* |
| *010* | *1* | *1* | *0* | *0* | *1* | *1* | *0* | *0* | *1* | *1* |
| *100* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* |
| *101* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* |
| *111* | *1* | *1* | *-/1* | *-/1* | *1/-* | *1/-* | *-* | *-* | *1* | *1* |
| *110* | *1* | *1* | *-/1* | *-/1* | *1/-* | *1/-* | *-* | *-* | *1* | *1* |

Il che porterebbe ad identificare come eccitazioni per il primo dei tre FF (responsabile della memorizzazione di y1) ad esempio

E magari altre scelte per quanto riguarda l’eccitazione degli altri 2 FlipFlop

Ma si possono fare anche scelte più “economiche” in termini di logica impiegata.

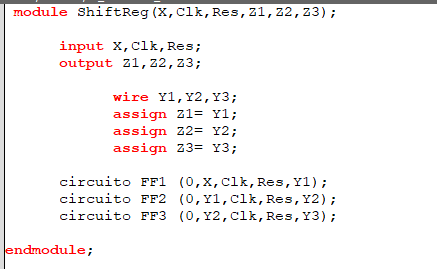
Una scelta assolutamente ragionevole potrebbe essere peraltro quella di scegliere per ciascuna cella la casella di sinistra, il che, sfruttando opportunamente le condizioni “don’t care” ci porta alla soluzione

**Trattazione a carattere euristico**

A tale soluzione vi si poteva giungere anche in modo assolutamente “euristico”: Poiché uno shift register può essere realizzato semplicemente impiegando la cascata di Tre Flip Flop di tipo D e poiché la cella di memoria fornita, presenta per l’appunto, il funzionamento di un FF di tipo D dotato in più di un segnale di abilitazione attivo basso (a), si poteva semplicemente scegliere di tenere i tre Flip Flop SEMPRE abilitati mettendo il loro segnale di eccitazione “a” livello basso fisso e sfruttare il secondo segnale (b) per creare la “catena” di FF, come ricavato qui sopra.

Per completare l’esercizio rimane solo da descrivere il circuito così realizzato e verificare se il suo funzionamento è congruo!

Il circuito può essere descritto dal seguente codice



Che come si può notare  
- istanzia 3 volte l’elemento “circuito” fornito nel testo dell’esercizio  
- in ciascuna istanza pone a 0 il primo dei due segnali di controllo (a)  
- realizza la “catena” di tre FF ove l’uscita dell’uno (Yi) è usata come segnale di eccitazione del FF che segue (bi+1)

Opportunamente simulato con stimoli “congrui” al dispositivo in esame porta alla generazione delle seguenti f.d.o



Da cui si evince il corretto funzionamento del dispositivo finale.