**Esercizi Propedeutici**

*Gennaio 2024*

1. Il Candidato progetti e realizzi in forma **asincrona** un dispositivo di memoria dotato di due ingressi di controllo **A** e **B**, di un segnale di sincronismo **clk** e di una uscita **Z.**Il dispositivo, in base ai segnali di controllo presenti le seguenti transizioni sincronizzate sul fronte di **discesa** del clock

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *Stato\AB* | *00* | *01* | *11* | *10* |
| *0* | 0 | 1 | 0 | 1 |
| *1* | 0 | 1 | 1 | 0 |

Il suddetto dispositivo venga opportunamente progettato e descritto attraverso un codice VerilogHDL. Se ne simuli quindi il funzionamento onde verificare la correttezza della soluzione.

1. Successivamente impiegando il suddetto dispositivo si realizzi questa volta in forma **sincrona** un dispositivo atto a creare un gioco di luci dotato di un segnale di controllo e quattro uscite (oltre naturalmente al clock) che esegua la sequenza i due diverse modalità:

- Se il segnale di controllo è a livello **alto** la sequenza si ripeta in modo ciclico

1001-0011-0110-1100 - …  
  
- Se il segnale di controllo è a livello **basso** invece la sequenza sia

0000-0001-0011-0111-1111-1110-1100-1000- …

Il dispositivo venga descritto in VerilogHDL **istanziando tante volte quante necessario** il dispositivo realizzato al passo precedente.

NOTA: Il candidato analizzi più soluzioni:

1. cercando di sfruttare la versatilità del dispositivo ed evitando di bloccare uno dei segnali di controllo A oppure B di ciascun dispositivo in una posizione fissa in maniera aprioristica.

2. oppure, viceversa usando in fase di controllo solamente uno dei due segnali

Si paragonino le diverse soluzioni trovate.

NOTA: Le soluzioni del presente elaborato siano complete di svolgimento, codice VerilogHDL e diagrammi di simulazione.

Il testo sottomesso deve inoltre contenere la seguente dichiarazione firmata:  
“Dichiaro che l’elaborato qui presente è stato svolto da me medesimo in piena autonomia”.

**Prima Parte:**

Si definisca la tavola di Huffman di una macchina che soddisfi alle specifiche. Tale macchina deve aver la possibilità di mantenere memoria del livello di clock (per poter rilevare uno specifico fronte) ed inoltre deve poter posizionare il segnale d’uscita in due modalità diverse (alto e basso) pertanto essa si dovrà comporre “come minimo” di quattro stati:

P: stato d’attesa del fronte **positivo** di clock con uscita pari a 0

Q: stato d’attesa del fronte **negativo** di clock con uscita pari a 0

R: stato d’attesa del fronte **positivo** di clock con uscita pari a 1

S: stato d’attesa del fronte **negativo** di clock con uscita pari a 1

Ogni qualvolta arriva un fronte di clock la macchina passa di stato e sceglie lo stato opportuno in base ai valori dei segnali di controllo così come riportati nel testo dell’esercizio. Il suo funzionamento potrebbe essere quello sotto riportato.



Si Adotti una opportuna codifica che eviti corse critiche:



Il che già consente di definire l’equazione dell’uscita ovvero:

Si possono quindi evidenziare singolarmente le evoluzioni relative a ciascuna variabile di stato:



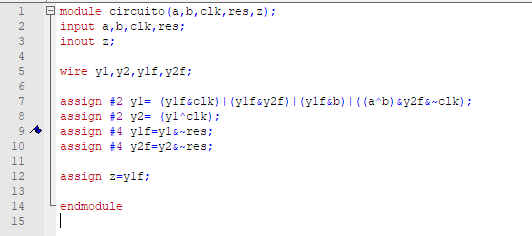
Interessante notare come la variabile Y2’ presenti gli stessi valori di Y1’ quando il segnale di clock è basso mentre presenti i valori invertiti quando invece esso è alto. Si può pertanto ottenere Y2’ noti che siano Y1’ e clock attraverso la seguente equazione

Inoltre la variabile di Stato Y1’, attraverso un’analisi degli implicanti

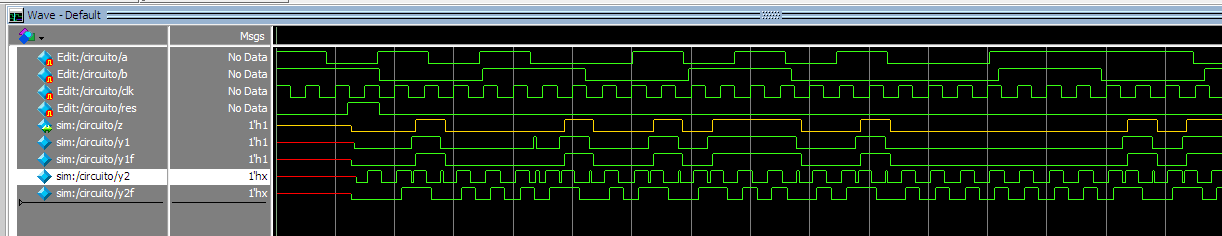


può essere espressa come:

Si può quindi passare a descrivere il sistema in Verilog HDL



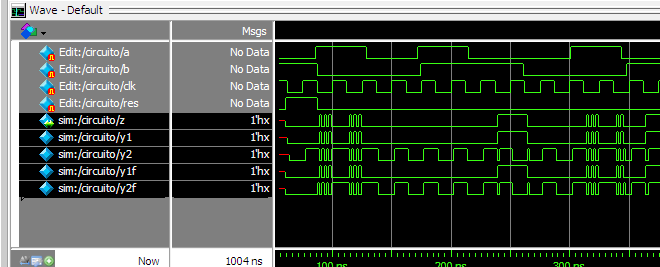
Il circuito così definito può essere ora simulato **fornendo segnali idonei ad indagare completamente** sul suo funzionamento.



Analizzando nel dettaglio le varie transizioni dell’uscita in base agli stimoli presenti in ingresso si desume che il circuito funziona correttamente.

**Approfondimenti:**

Si noti però in particolare che:

* Sebbene NON richiesto è stato inserito un segnale di reset asincrono attivo alto utile a “forzare” il circuito in uno stato noto (righe 9 e 10)
* I ritardi utilizzati fanno sì che le “variabili di stato” y1f e y2f si stabilizzino “**dopo**” tutte le altre. SE così non dovesse avvenire cosa accadrebbe? A quale fenomeno si andrebbe incontro? Si indaghi a fondo su cosa accadrebbe se ad esempio i ritardi alle righe 8 e 9 NON fossero presenti.
* Un Comportamento come quello evidenziato qui di seguito:  
  

A cosa è dovuto ?

* L’uscita è prelevata sulla variabile y1f ! Perché ? E se fosse stata prelevata sulla variabile y1 cosa accadrebbe ? Che differenza c’è tra le due variabili ?

**Seconda Parte:**

Questa parte si presta a molteplici soluzioni sia per quanto riguarda la definizione della macchina sincrona ed i suoi stati, sia per quanto concerne i segnali di eccitazione che consentono di volta in volta di far evolvere la macchina.

**Soluzione 1:**

Al di là dei valori che sono presenti sulle uscite la macchina fondamentalmente opera su due sequenze cicliche: la prima composta di 4 stati e la seconda da 8, per cui una sua descrizione piuttosto semplice potrebbe essere quella qui di seguito riportata:



Si vede pertanto che la macchina potrebbe essere realizzata facilmente (sfruttando le condizioni d.c della seconda colonna) attraverso un semplice contatore ciclico a 3bit che procede sequenzialmente su 8 stati diversi (0,1,2,3 …7,0) ,e da una logica opportuna in base allo stato ed alla variabile di controllo definisce le quattro uscite da fornire.



Trascurando momentaneamente le uscite la macchina è la seguente e la sua evoluzione NON dipende dalla variabile di controllo X:



La codifica da impiegare per il contatore può essere di qualsiasi tipo: Binario o Gray se si volesse minimizzare il numero di FF, ma potrebbe essere anche una codifica che usi più variabili di stato se questo potesse portare a dei benefici di altro tipo.

Supponendo di codificare gli stati in binario ci vogliono almeno 3 variabili di stato per codificare gli otto stati:



Nella tabella sono riportate le evoluzioni relative a ciascuna variabile di stato. Dovendo ora impiegare il dispositivo sviluppato nella prima parte dell’esercizio, bisogna evidenziare come le variabili di controllo possano controllare le transizioni. Dall’analisi della tabella fornita nel testo si desume che la mappa tabella delle eccitazioni per siffatto elemento sia la seguente:



Espressa in forma “verbosa” può essere esplicitata nelle seguenti affermazioni

* Perché lo stato si mantenga a 0 e segnali di eccitazione A e B devono essere uguali (o entrambi a 0 o entrambi a 1)
* Per far evolvere lo stato da 0 a 1 i segnali di eccitazione devono essere diversi (A=1 e B=0 o viceversa)
* Per portare lo stato a 0 il segnale B deve essere pari a 0 indipendentemente da A
* Per mantenere lo stato a 1 il segnale B deve essere pari a 1 indipendentemente da A.

Da quanto sopra esposto si nota che ci sono molteplici modalità per creare la trasmissione desiderata ad esempio si potrebbe tenere il segnale A fisso al valore 0 ed agire solo su B



O ancora si potrebbe mantenere A=1 e agire solo su B



Ma onde evitare di perdere il vantaggio solo per “pigrizia” manteniamo aperte tutte le possibilità. Nella tabella di evoluzione delle variabili di stato si vada a sostituire le eccitazioni che quelle evoluzioni controllano.

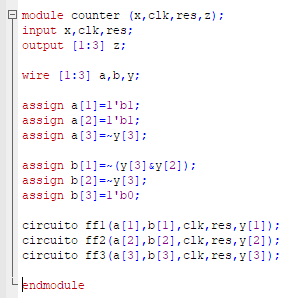


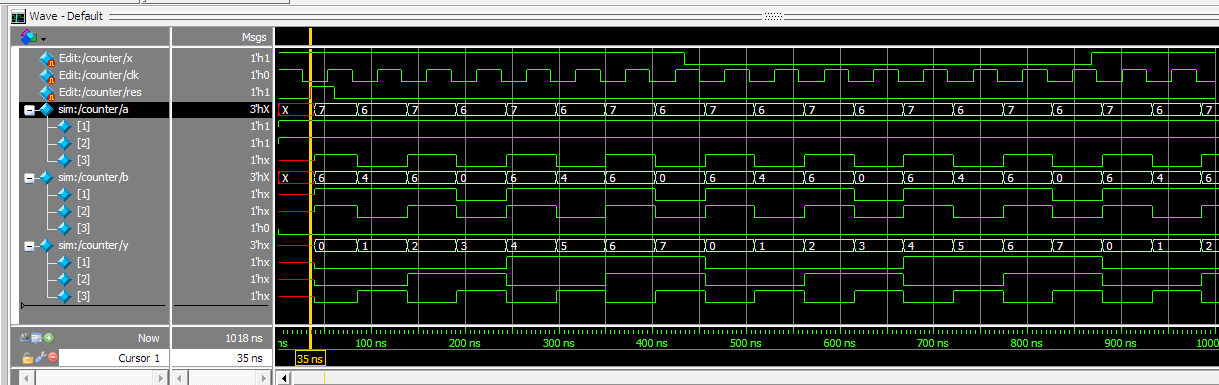
Adesso le scelte che si possono effettuare sono molteplici (per la precisione sono 2^24) e appare ovvio che non si possano analizzare tutte, si sceglierà quindi una soluzione che porti qualche beneficio dal punto di vista circuitale. Un esempio potrebbe essere il seguente:



Che porta alle seguenti equazioni di eccitazione:

Si può quindi descrivere il circuito in Verilog HDL e verificare il corretto funzionamento attraverso una opportuna simulazione:





Da cui si evince il corretto funzionamento del “contatore binario”

Rimane ancora da associare la corretta uscita ad ogni stato, ma questo può essere realizzato attraverso un semplice circuito combinatorio multiterminale basato sulle tabelle di verità già viste in precedenza:



Ovvero:



Che sviluppata secondo una variabile alla volta porta alle seguenti espressioni:



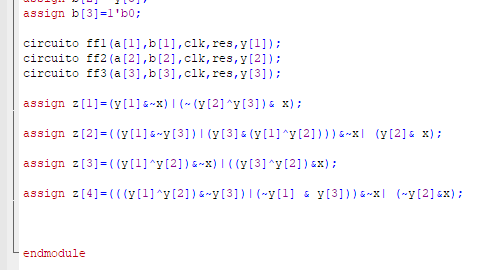
)x

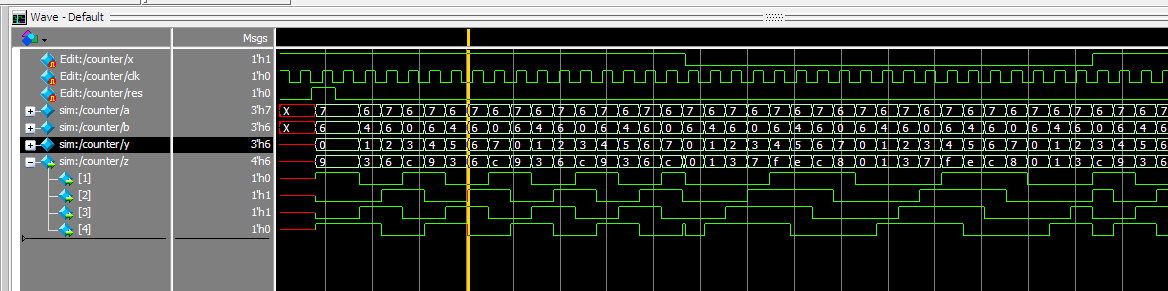






Si possono ora riportare le espressioni nel codice VerilogHDL e simulare





Evidenziando il corretto funzionamento.

**Soluzione 2:**

La Soluzione appena proposta sebbene limita al minimo il numero di elementi di memoria impiegati (infatti fa uso solo di 3 FF) richiede come contropartita l’uso di un circuito combinatorio abbastanza complesso per attivare le uscite come richiesto nei vari stati.

Si può peraltro ipotizzare che, al costo di impiegare un FF per ciascuna delle uscite (ovvero 4 invece di 3) si potrebbe ottenere una considerevole semplificazione dal punto di vista della logica che definisce le uscite. Ciò è realizzabile solo in quanto ogni combinazione delle uscite risulta unica e pertanto potrebbe essere impiegata anche come codifica dello stato .

Adottando quindi una codifica a 4 variabili che faccia coincidere le variabili di stato con le variabili di uscita si ottiene la seguente tavola di flusso:



Sfruttando opportunamente le condizioni d.c. e prestando particolare attenzione che durante le transizioni del segnale X la macchina non si ponga in qualche stato imprevisto si può proseguire evidenziando come controllare le variabili di eccitazione per garantire il corretto passaggio di stato della macchina.

Per ovviare a questa situazione vi sono molte soluzioni possibili, ma una soluzione abbastanza elegante potrebbe essere quella di seguito suggerita: se a seguito di una modifica del segnale di controllo la macchina dovesse trovarsi in una situazione che si pone “al di fuori della sequenza prevista” si può pensare di lasciar evolvere la macchina secondo la sequenza originale fintanto che essa non cade in uno dei due stati che risultano comuni ad entrambe ) le sequenze (0011 oppure 1100 e da qui inizia ad evolvere secondo la “nuova” sequenza. Si può pertanto riscrivere la macchina nel seguente modo.



Evidenziando le variabili di stato una alla volta:



Si noti in particolare che y1,y2 ed y3 risultano indipendenti da X, mentre l’unica variabile che dipende da esso è y4, Per cui le tabelle di cui sopra possono venir semplificate in: 

Dove sono state altresì evidenziate le transizioni di ciascuna variabile di stato.

Impiegando ora le mappe di eccitazione dei FF si possono ottenere le equazioni di eccitazione di ciascun FF:



Operando scelte opportune si può arrivare a questa soluzione



Che evidenzia le seguenti equazioni di eccitazione:

Inoltre per quanto riguarda la variabile b4, meglio esplicitare la sua tabella di verità attraverso le mappe di Karnaugh:



IL dispositivo così descritto può essere quindi descritto in Verilog HDL:



Ed infine simulato

