

Progetto 20 – CONTATORE A CODICE GRAY

Introduzione:

Lo scopo di questo progetto è di creare un contatore modulo 8 a codice Gray, senza ingressi e con tre uscite. Il contatore essendo modulo 8 conta da 0 a 7 e quindi riparte da 0. Esso parte dal codice 000 e generare in uscita il prossimo codice Gray a ogni fronte di salita del clock.

Per completare il progetto è possibile realizzare la macchina a stati finiti (FSM) relativa al contatore specificandone la codifica degli stati, sintetizzare, attraverso le mappe di Karnaugh, il sistema e successivamente tradurlo in un circuito Logisim e testarlo.

Descrizione:

Come richiesto dalla consegna del progetto il contatore Gray non avrà nessun tipo di ingresso ma solamente tre uscite (g_2 , g_1 , g_0) che rappresentano il contatore codificato in codice Gray. Esse verranno aggiornate in base allo stato del sistema ad ogni ciclo di clock essendo il contatore un circuito sincrono che si basa sul fronte di salita del clock.

Il sistema prevede 8 stati (da 0 a 7) che possono essere codificati utilizzando 3 bit (S2S1S0). L'automa che meglio può rappresentare questo sistema è quello di Moore. Ad ogni stato corrispondono i valori di uscita in quell'istante. La freccia che parte da uno stato è unica poiché ad ogni fronte di salita lo stato cambia e passa a quello futuro.

Disegnato il diagramma degli stati si può elaborare la tabella di transizione degli stati che mi permette di mostrare lo stato futuro ad ogni ciclo di clock in base allo stato corrente in cui sono.

In seguito si può effettuare la sintesi minima SP degli stati futuri mediante le mappe di Karnaugh. Gli stati futuri dipenderanno solamente dallo stato attuale poiché il sistema non ha ingressi.

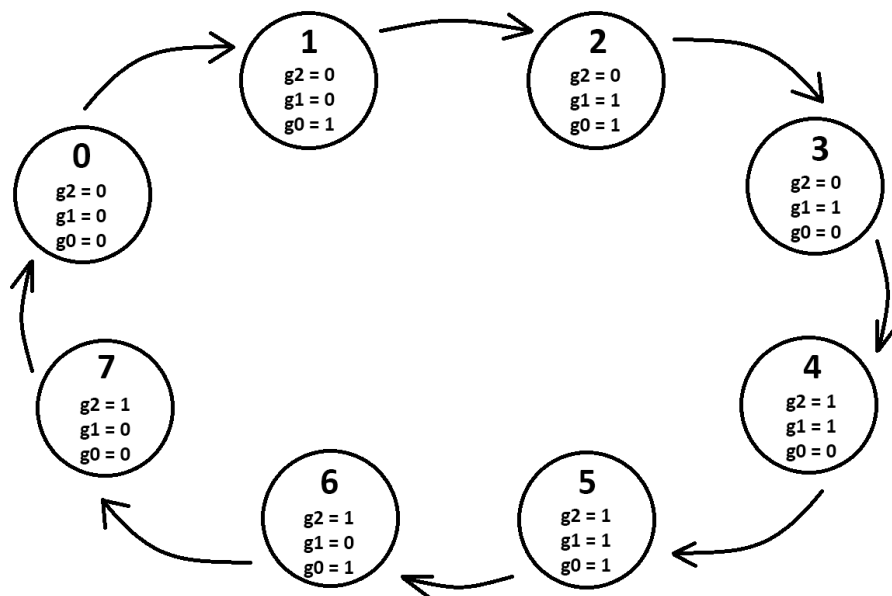
Le sintesi minime che si ottengono dalle mappe ci permettono di rappresentare il sistema anche tramite Logisim realizzando un contatore che, partendo dalla codifica 000, ci permette ad ogni ciclo di clock di ottenere la successiva codifica binaria.

Per trasformare la codifica binaria in codice Gray è necessario prima realizzare la sintesi minima delle uscite attraverso una semplice tabella e poi utilizzare tale sintesi per implementare la conversione in Logisim.

Procedimento:

DIAGRAMMA DEGLI STATI

STATO	0	1	2	3	4	5	6	7
CODIFICA	000	001	010	011	100	101	110	111
USCITE	g2 = 0	g2 = 0	g2 = 0	g2 = 0	g2 = 1	g2 = 1	g2 = 1	g2 = 1
	g1 = 0	g1 = 0	g1 = 1	g1 = 1	g1 = 1	g1 = 1	g1 = 0	g1 = 0
	g0 = 0	g0 = 1	g0 = 1	g0 = 0	g0 = 0	g0 = 1	g0 = 1	g0 = 0



Visto che tutte le uscite dipendono solo dallo stato in cui si trova il sistema e non dagli ingressi effettuiamo la sintesi dell'automa di Moore corrispondente.

Essendo il sistema a modulo 8 esso avrà 8 stati e quindi per la codifica sono necessari 3 bit ($2^3 = 8$). Ad ogni stato è associata una codifica in codice binario e 3 uscite che rappresentano i bit della codifica in codice Gray.

Il passaggio da uno stato all'altro avviene al fronte di salita del clock.

TABELLA DI TRANSIZIONE DEGLI STATI

Stato presente				Stato futuro			
s	s2	s1	s0	S	S2	S1	S0
0	0	0	0	1	0	0	1
1	0	0	1	2	0	1	0
2	0	1	0	3	0	1	1
3	0	1	1	4	1	0	0
4	1	0	0	5	1	0	1
5	1	0	1	6	1	1	0
6	1	1	0	7	1	1	1
7	1	1	1	0	0	0	0

Come convenzione all'interno della tabella di transizione degli stati si usa s minuscolo per lo stato presente e S maiuscolo per quello futuro.

Ogni stato è rappresentato con la corrispettiva codifica in binario a tre bit.

A dispetto delle usuali tabelle di transizione qui non figurano ingressi se non i tre bit dello stato presente poiché l'unico vero elemento in entrata è il segnale di temporizzazione. Per quanto riguarda invece le uscite vengono considerati i bit uscenti dai tre flip-flop D che rappresentano lo stato futuro.

MAPPE DI KARNAUGH

Sintesi minima degli stati futuri in forma SP

s2\s1s0	00	01	11	10
0	0	0	1	0
1	1	1	0	1

$$S2 = s2's1s0 + s2s1' + s2s0'$$

$s_2 \backslash s_1 s_0$	0 0	0 1	1 1	1 0
0	0	1	0	1
1	0	1	0	1

$$S1 = s1's0 + s1s0'$$

$s_2 \backslash s_1 s_0$	0 0	0 1	1 1	1 0
0	1	0	0	1
1	1	0	0	1

$$S0 = s0'$$

TABELLA DELLE USCITE

Senza la necessità di utilizzare le mappe di Karnaugh si riesce a ricavare la sintesi minima di ogni uscita.

s	s_2	s_1	s_0
0	0	0	0
1	0	0	1
2	0	1	0
3	0	1	1
4	1	0	0
5	1	0	1
6	1	1	0
7	1	1	1

g_2	g_1	g_0
0	0	0
0	0	1
0	1	1
0	1	0
1	1	0
1	1	1
1	0	1
1	0	0

$$g0 = s1 \oplus s0$$

$$g1 = s2 \oplus s1$$

$$g2 = s2$$

Implementazione:

Il circuito implementato in Logisim si compone di due circuiti: il “main” e il “Contatore”. Il main è il circuito principale mentre il circuito “contatore” è un sotto-circuito sviluppato per rendere più chiaro lo sviluppo del progetto.

MAIN



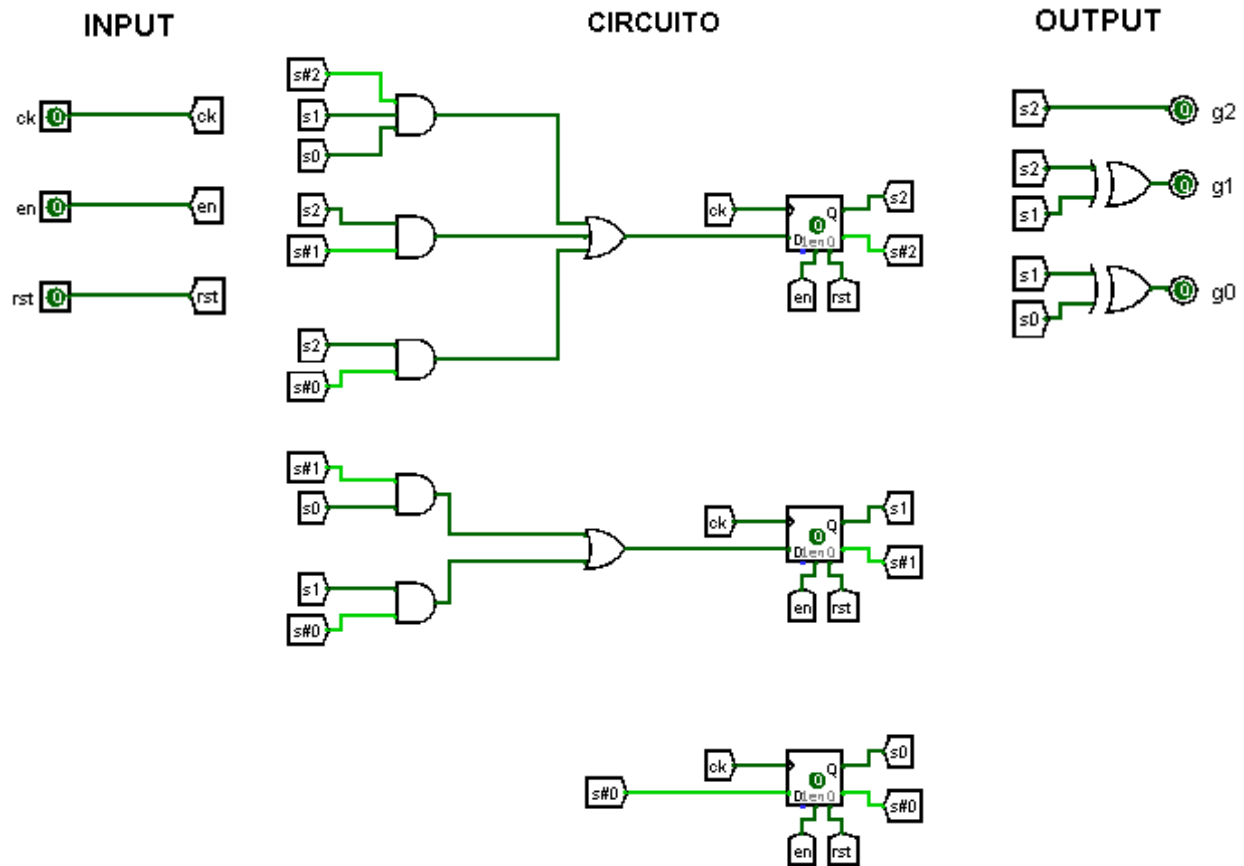
Nel circuito principale chiamato appositamente “main” abbiamo i segnali per controllare tutto il sistema e il contatore che dà in output i tre bit della codifica in codice Gray.

Come segnali vengono utilizzati:

- Clock (ck): segnale di temporizzazione per tutto il circuito. È in base ad esso che il contatore avanza da uno stato all'altro.
- Enable (en): segnale che dà l'abilitazione per aggiornare il circuito. Se esso non è attivo non può avvenire nessuna modifica.
- Reset (rst): segnale di azzeramento del circuito. Quando questo segnale si attiva tutto il circuito, in modo asincrono, viene ri-settato con le impostazioni iniziali.

Le uscite g2, g1, g0 rappresentano la codifica dei vari stati del contatore in codice Gray. Esse provengono dal sotto-circuito sviluppato separatamente. Esse se l'enable è attivo cambiano ad ogni fronte di salita del clock.

CONTATORE



Questo sotto-circuito in base ai segnali di controllo dati permette di contare da 0 a 7 e, una volta arrivato alla fine, ripartire da 0.

Il circuito è stato realizzato sfruttando le tre sintesi minime in forma SP degli stati futuri ricavate utilizzando le mappe di Karnaugh. Ogni bit ricavato dalle tre sintesi viene messo in ingresso ad un flip-flop D che campiona le entrate al fronte di salita del clock commutando il suo stato in base allo stato attuale e all'ingresso in quell'istante.

Le uscite dei flip-flop D sono poste sugli ingressi creando così un circuito retroattivo che permette di avere un sistema che basa lo stato futuro sullo stato attuale in cui si trova. Esse inoltre, attraverso le espressioni trovate grazie alla tabella delle uscite, modificano le uscite `g2`, `g1`, `g0` del sistema dando così in output il contatore non in binario ma codificato in codice Gray.