UNIVERSITÀ DEGLI STUDI DI PARMA

DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE

Fondamenti di Informatica B

Lezione n.6

Fondamenti di Informatica B

Lezione n.6

- **Circuiti Sincroni e Asincroni**
- **Modello Generale**
- **Tecniche di Progettazione**

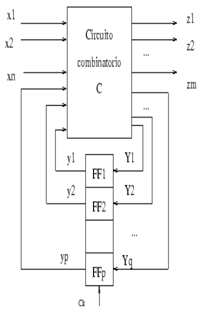
In questa lezione verranno considerati i modelli generali dei circuiti sequenziali e analizzate alcune semplici tecniche di progetto

Circuiti Sincroni e Asincroni

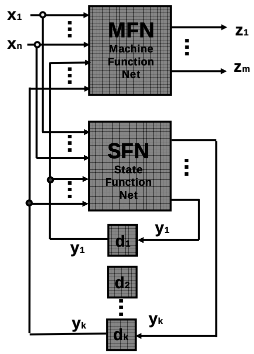
- Circuiti Sincroni:
  - Elementi di memoria cadenzati o sincroni
  - Tutte le variazioni di stato interno (contenuto delle celle di memoria) avvengono contemporaneamente, pilotate dal valore del clock
  - Il periodo del segnale di clock viene calcolato per dare tempo ai transitori di esaurirsi
- Circuiti Asincroni:
  - Le variazioni di stato interno (degli elementi di memoria) si verificano appena possibile, in funzione:
    - dei ritardi del circuito
    - delle variazioni dei segnali all'ingresso

Modello di Huffman

- Il modello di Huffman è uno schema generale per i circuiti sequenziali
  - separa la parte combinatoria dagli elementi di memoria
  - Lo stato totale del circuito è dato dal valore degli ingressi <x_i> e dallo stato interno <y_i> del circuito
- Il circuito C genera:
  - Uscite principali <z_i>
  - Comandi alla memoria <Y_i>



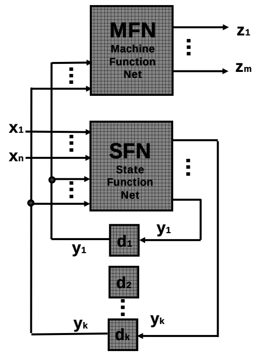
Modelli di Mealy



- Modello di Mealy:

MFN: $(X \times S) \rightarrow Z$ funzione di uscita
SFN: $(X \times S) \rightarrow S$ funzione di stato
dove:
  - X: spazio delle configurazioni di ingresso
  - S: spazio degli stati interni
  - Z: spazio delle configurazioni di uscita
  - $d_1, \ldots, d_k$: sottoreti elementari di marcatura. Possono essere:
    - FF sincroni (circuiti sincroni)
    - ritardi, latch SR o retroazioni dirette (circuiti asincroni)
- Se MFN non dipende da X viene detto modello di Moore

Modelli di Moore



- Modello di Moore:

MFN: $(X \times S) \rightarrow Z$ funzione di uscita
SFN: $(X \times S) \rightarrow S$ funzione di stato
dove:
  - X: spazio delle configurazioni di ingresso
  - S: spazio degli stati interni
  - Z: spazio delle configurazioni di uscita
  - $d_1, \ldots, d_k$: sottoreti elementari di marcatura. Possono essere:
    - FF sincroni (circuiti sincroni)
    - ritardi, latch SR o retroazioni dirette (circuiti asincroni)
- MFN non dipende da X

Esempio: il sommatore

- Progettare un sommatore seriale di numeri binari a lunghezza arbitraria

- Il circuito è sequenziale: $z(t) = x_1(t) + x_2(t) + c(t-1)$
  - S_0: c(t-1)= 0. La somma precedente non ha generato riporto
  - S_1: c(t-1)= 1. La somma precedente ha generato riporto

Esempio

- Una volta definiti ed enumerati gli stati (S0 ed S1 nell'esempio) di deve definire la modalità di evoluzione dello stato interno in funzione dei segnali di ingresso
- Lo strumento utilizzato è la tabella delle transizioni
  - Generalmente ci si aiuta con un grafico: il diagramma degli stati

Esempio

Diagramma degli stati:

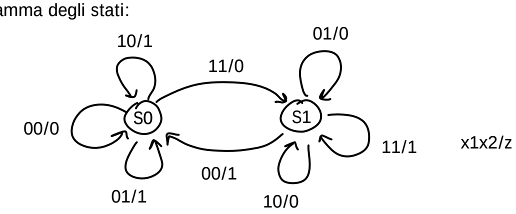


Tabella delle transizioni:

Stato \ X1,X2	00	01	11	10
S0	S0,0	S0,1	S1,0	S0,1
S1	S0,1	S1,0	S1,1	S1,0

Stato futuro, Uscita

Esempio

Tabella delle transizioni:

Stato \ X1,X2	00	01	11	10
S0	S0,0	S0,1	S1,0	S0,1
S1	S0,1	S1,0	S1,1	S1,0

Stato futuro, Uscita

Codifica degli stati: S0 $\Rightarrow$ y=0 S1 $\Rightarrow$ y=1

Tabella delle transizioni:

y \ X1,X2	00	01	11	10
0	0,0	0,1	1,0	0,1
1	0,1	1,0	1,1	1,0

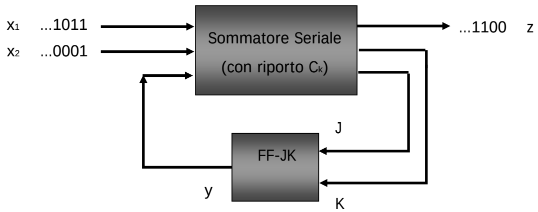
Stato interno, Uscita

Esempio

Scelta del tipo di Flip-Flop:

ad esempio in questo caso: FF-JK

J	K	Ck	Q(n+1)
X	X	0	Q(n)
1	0	1	1
0	1	1	0
0	0	1	Q(n)
1	1	1	$\overline{Q(n)}$



Esempio

- Ricondotto alla progettazione di 3 circuiti combinatori separati:
  - z = z (x1, x2, y)
  - J = j (x1, x2, y)
  - K = k (x1, x2, y)
- Le relazioni z, j, k sono definite nella tabella delle transizioni

Esempio

Tabella delle transizioni:

y \ x1,x2	00	01	11	10
0	0,0	0,1	1,0	0,1
1	0,1	1,0	1,1	1,0

Tabella FF-JK:

y(t)	y(t+1)	J(t)	K(t)
0	0	0	d
0	1	1	d
1	0	d	1
1	1	d	0

x1(t)	x2(t)	y(t)	y(t+1)	J(t)	K(t)	z(t)
0	0	0	0	0	d	0
0	1	0	0	0	d	1
1	1	0	1	1	d	0
1	0	0	0	0	d	1
0	0	1	0	d	1	1
0	1	1	1	d	0	0
1	1	1	1	d	0	1
1	0	1	1	d	0	0

Esempio

- Copertura di mappe

Tabella di J

y \ x1,x2	00	01	11	10
0	0	0	1	0
1	d	d	d	d

$J = x1 \cdot x2$

Tabella di K

y \ x1,x2	00	01	11	10
0	d	d	d	d
1	1	0	0	0

$K = \overline{x1} \cdot \overline{x2}$

Tabella di z

y \ x1,x2	00	01	11	10
0	0	1	0	1
1	1	0	1	0

$z = y \cdot \overline{x1} \cdot \overline{x2} + \overline{y} \cdot \overline{x1} \cdot x2 + y \cdot x1 \cdot x2 + \overline{y} \cdot x1 \cdot \overline{x2}$

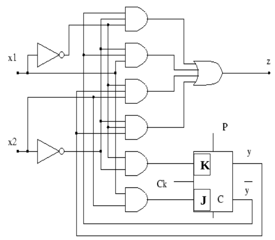
Esempio

- Realizzazione circuito

$J = x1 \cdot x2$

$K = \overline{x1} \cdot \overline{x2}$

$z = y \cdot \overline{x1} \cdot \overline{x2} + \overline{y} \cdot \overline{x1} \cdot x2 + y \cdot x1 \cdot x2 + \overline{y} \cdot x1 \cdot \overline{x2}$



Esercizi successivi

- Risolvere lo stesso esercizio utilizzando Flip-Flop di tipo SR, D, oppure T
- Ci sono vantaggi?

Le fasi del progetto

1. Modello del circuito (ad es. Huffman)
2. Diagramma degli stati
3. Tabella delle transizioni + codifica degli stati
4. Si scelgono gli elementi di memoria
5. Si trasforma il progetto nella sintesi di circuiti combinatori
6. Progetto circuiti combinatori separati