

Premessa

I contatori digitali sono circuiti sequenziali in grado di contare gli impulsi di clock applicati al proprio ingresso. In linea di principio, un contatore presenta un solo ingresso (clock) e n bit di uscita. Applicando in ingresso un impulso di clock, la parola binaria di uscita si incrementa di una unità.

Come è noto, con una parola binaria di n bit sono possibili 2^n configurazioni diverse che corrispondono ai numeri decimali compresi tra 0 e $2^n - 1$. Un contatore può assumere tutte le possibili combinazioni oppure una sola parte.

Si definisce *modulo* il numero di configurazioni distinte che il contatore può assumere. Il contatore si dice *binario* se il suo modulo vale proprio 2^n .

Il dispositivo si realizza collegando dei flip-flop opportunamente tra loro. Se il contatore è a modulo m il numero n di flip-flop necessari si determina nella seguente disuguaglianza:

$$2^{n-1} < m \leq 2^n \quad (1)$$

Un contatore a modulo m non binario (è il caso da te proposto) conta da 0 a $m-1$; al successivo impulso di clock, il dispositivo torna a riassumere la configurazione corrispondente al numero decimale 0 e così via.

I contatori integrati non binari più diffusi sono quelli decadici noti anche come *contatori BCD*. Essi sono a modulo 10 e, pertanto, assumono tutte le configurazioni binarie comprese tra 0 e 9.

Un contatore può essere progettato anche per contare indietro (*down counter*); altri invece, sono in grado di contare sia in avanti che indietro (*up/down counter*).

A seconda del modo con cui sono collegati gli ingressi di clock dei vari flip-flop, i contatori si dividono in due categorie: *asincroni* e *sincroni*. Nei primi un flip-flop cambia stato dopo la commutazione del flip-flop che lo precede; in questo caso il comando di commutazione si propaga da un flip-flop all'altro. Nei secondi l'ingresso di clock del contatore è applicato simultaneamente all'ingresso di clock di tutti i flip-flop.

Alcuni contatori integrati presentano delle linee supplementari come, ad esempio, l'ingresso di *clear* che serve per l'azzeramento del dispositivo.