

1 Lezione del 10-12-24

1.1 Conversione digitale/analogico e analogico/digitale

Finora abbiamo assunto che le interfacce lavorino solo e soltanto con segnali digitali. In verità nel mondo esterno al computer le grandezze variano su una scala continua. Occorrono appositi convertitori, detti convertitori digitale/analogico e analogico/digitale.

La grandezza analogica che consideriamo è un voltaggio appartenente alla scala FSR (Full Scale Range) $[5, 30]$ V. Questa verrà convertita in un intero x rappresentato su N bit con $N \in \{8, 16\}$. A seconda dell'interpolazione scelta, potremo distinguere fra:

- **Conversione ubipolare:** $v \in [0, FSR], x \in [0, 2^N - 1]$;
- **Conversione bipolare:** $v \in \left[-\frac{FSR}{2}, \frac{FSR}{2}\right], x \in [-2^{N-1}, +2^{N-1} - 1]$

1.1.1 Errori di conversione

Dato $K = \frac{FSR}{2^N}$, nel caso ideale vorremmo $v = Kx$. In realtà, avremo che $|v - Kx| \leq \varepsilon$, con un ε dovuto a errori di conversione:

- **Errore di non linearità;**
- **Errore di quantizzazione.**

1.1.2 Tempi di risposta

I convertitori D/A sono praticamente "combinatori", e quindi estremamente veloci (pochi nanosecondi). I convertitori A/D, di contro, hanno tempi di risposta variabili in base alle architetture. Noi vedremo i convertitori ad **approssimazioni successive (SAR)**, che hanno tempi di risposta di qualche centinaio di nanosecondi.

1.1.3 Convertitori bipolari

I convertitori bipolari lavorano con rappresentazioni degli interi in traslazione (detta appunto anche *binaria bipolare*). Il numero intero x viene quindi rappresentato dal naturale $X = x + 2^{N-1}$. In ogni caso, per riportare in complemento a 2 basterà complementare il MSB.

1.2 Convertitore D/A

Un convertitore D/A può essere realizzato come segue:

circuito
descrizione

Anche se non si considerano resistori e amplificatori operazionali come componenti combinatori, il circuito è effettivamente "combinatorio" nel senso che ha tempi di risposta estremamente veloci. Il problema è però quello delle transizioni multiple dello stato di uscita: questo si risolve attraverso un filtro *passa-basso* in uscita.

1.2.1 Interfaccia per la conversione D/A

Vediamo un'interfaccia parallela per l'operazione di un convertitore D/A. Sul lato di uscita non si avranno handshake (sola uscita) ma il convertitore D/A stesso.

rete

1.3 Convertitore A/D

Descriviamo un particolare tipo di convertitori A/D detto convertitore ad **approssimazioni successive**:

circuito

descrizione

Il cuore di un convertitore di questo tipo è una rete sequenziale detta **SAR** (Successive Approximation Register). L'uscita del SAR viene fatta passare attraverso un convertitore D/A dello stesso tipo dell'A/D, e confrontata attraverso un **comparatore** con l'ingresso corrente in modo da migliorare la previsione, in quella che è effettivamente una **ricerca logaritmica** (o *binaria* o *dicotomica*). In particolare, ad ogni iterazione della ricerca si ricava il valore di un singolo bit, per cui n bit richiedono n iterazioni. Lato processore, il SAR dovrà implementare inoltre un handshake *soc/eoc*.

Una descrizione in verilog della SAR potrebbe essere:

descrizione verilog corsini

Il problema di questa descrizione, supponendo questa sia perfettamente chiara, è che abbiamo bisogno di un nuovo stato per ogni iterazione di aggiornamento di RBR. Una soluzione alternativa potrebbe essere allora:

descrizione verilog stea

dove si introduce un registro COUNT e una rete combinatoria per il calcolo di RBR.

1.3.1 Interfaccia per la conversione A/D

Vediamo un'interfaccia parallela per l'operazione di un convertitore A/D. Lato processore si implementerà un handshake *soc/eoc*.

rivedi