Capitolo 9

I CONVERTITORI ANALOGICO-DIGITALI

Introduzione

Un convertitore analogico-digitale (Analog to Digital Converter *ADC*) converte i valori che un segnale analogico $x(\cdot)$, ossia continuo nel tempo e nelle ampiezze, assume in determinati istanti temporali in una successione numerica $x_c[\cdot]$. Tale blocco deve pertanto eseguire le tre seguenti operazioni:

- la discretizzazione dei tempi (campionamento). Tale operazione, a partire dal segnale analogico di ingresso $x(\cdot)$, fornisce un segnale discreto nel tempo, ma continuo nelle ampiezze $x_s[\cdot]$. Nel caso molto diffuso di campionamento uniforme, ossia in istanti equispaziati, si ha percio'

$$x_s[\cdot] = \{x(nT),\ n = ...-1,\ 0,\ 1,...\}$$

dove T rappresenta il cosidetto periodo di campionamento.

- la discretizzazione delle ampiezze (quantizzazione). In questo caso, a partire dal segnale a tempo discreto e ampiezze continue $x_s[\cdot]$, si ottiene il segnale numerico $x_q[\cdot]$, ossia un segnale discreto sia nel tempo, sia in ampiezza.

- la rappresentazione del valore dei campioni quantizzati mediante quantita' numeriche (codifica). A partire dal segnale numerico $x_q[\cdot]$ tale operazione fornisce una successione di numeri $x_c[\cdot]$ rappresentati in una opportuna base numerica.

In fig.1 e' riportato lo schema funzionale a blocchi di un *ADC* ideale. In fig.2 e' invece riportato un esempio di conversione *A/D* (campionamento, quantizzazione e codifica) di un generico segnale.

Come e' stato appena affermato, un *ADC* ideale esegue una operazione di campionamento ideale. In realta', poiche' il tempo richiesto per la conversione del segnale di ingresso $x(\cdot)$ in un valore numerico non puo' mai essere nullo, e' necessario che le variazioni temporali di tale segnale

siano sufficientemente lente da poter supporre l'ingresso dell'*ADC* costante durante l'intera durata della conversione.

Quando questa ipotesi non puo' essere soddisfatta, all'*ADC* viene premesso un amplificatore sample-and-hold (*SHA*); questo blocco ha il compito di campionare il segnale analogico negli istanti prefissati (fase di sample) e di fornire in uscita il valore campionato mantenendolo per tutto il tempo necessario all'esecuzione della conversione (fase di hold).

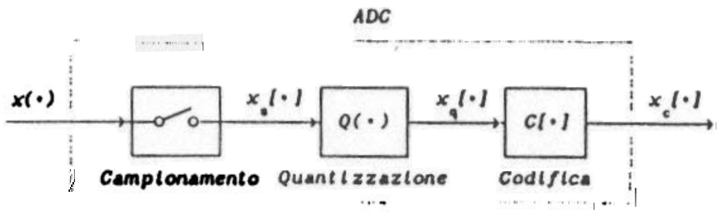


Fig.1. Schema funzionale a blocchi di un *ADC* ideale.

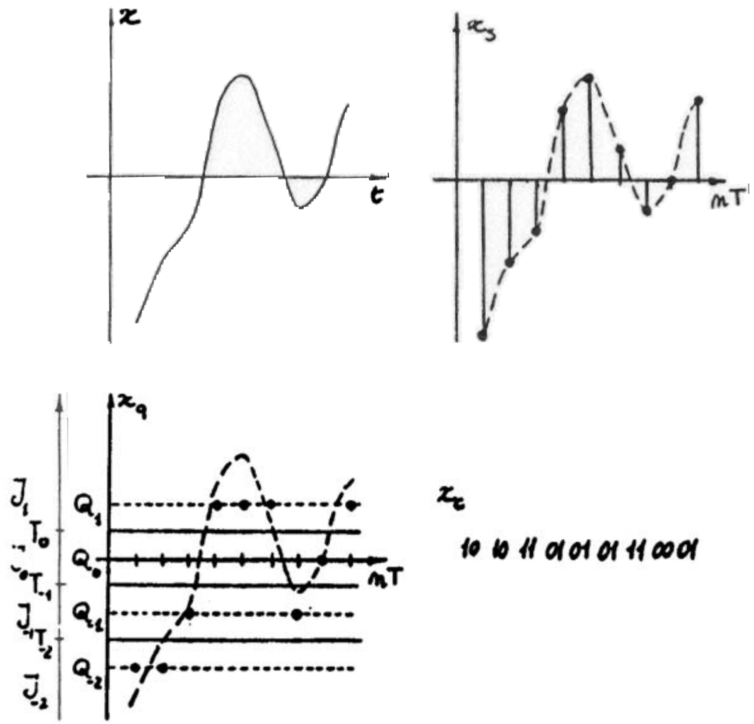


Fig.2. Campionamento, quantizzazione e codifica di un generico segnale.

Nel seguito si potra' pertanto supporre che l'ingresso dell'*ADC* sia costante durante l'intero processo di conversione; in tale ipotesi, inoltre, l'operazione di campionamento insita nella conversione potra' essere trascurata; sara' inoltre fornito solo qualche cenno sull'operazione di codifica. L'attenzione sara' pertanto posta sull'operazione di quantizzazione.

L'Operazione di Quantizzazione

Un quantizzatore converte l'intervallo continuo $\mathcal{I}$ di possibili valori di ingresso in un insieme finito di valori di uscita:

$$Q = \{ Q_k, \, k \in K \} \qquad (1)$$

Il generico elemento Q_k di tale insieme viene detto **livello di quantizzazione**; il numero totale di livelli utilizzati sara' inoltre indicato con *B*. L'operazione di quantizzazione consente pertanto di convertire un valore continuo x_s in un valore discreto x_q.

La relazione ingresso-uscita di un quantizzatore e' quindi ottenuta divendo l'intervallo continuo $\mathcal{I}$ di possibile variazione dell'ingresso in sotto-intervalli $\mathcal{I}_k$, a ognuno dei quali viene associato lo stesso livello Q_k. La caratteristica di quantizzazione e' percio' rappresentata da una funzione $Q(\cdot)$ costante a tratti definita in $\mathcal{I}$ e a valori in Q:

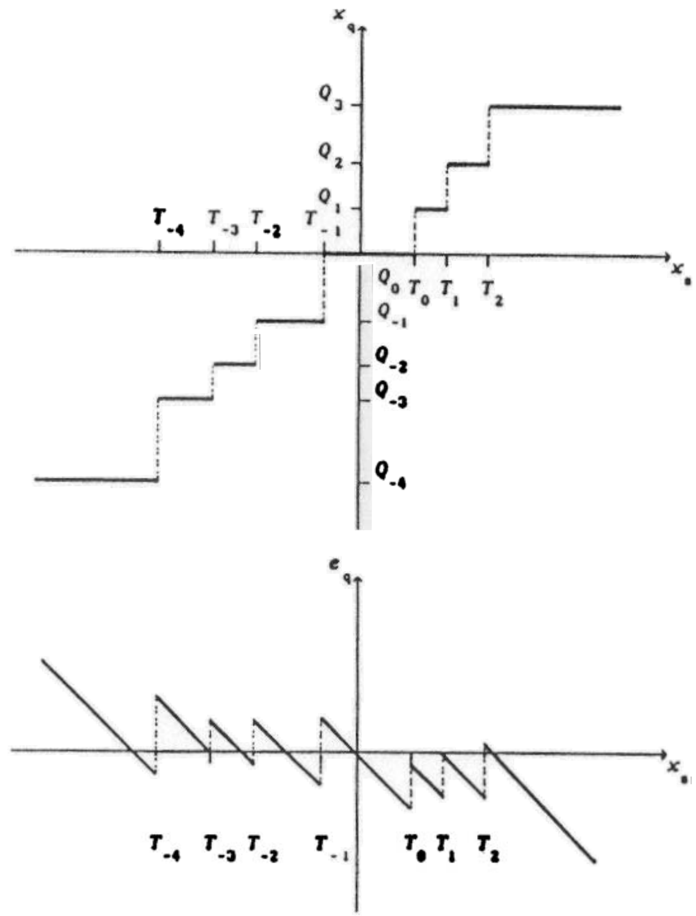
$$x_q = Q(x_s) \qquad x_q \in Q \qquad (2)$$

I *B-1* valori dell'ingresso ai quali si ha la transizione dell'uscita da un livello di quantizzazione a quello adiacente, ossia gli estremi dei sotto-intervalli in cui e' suddiviso l'insieme $\mathcal{I}$, sono detti **livelli di soglia** (o livelli di transizione) T_k.

Il campo di ingresso (input range) C_I di un *ADC* e' in genere leggermente piu' ampio dell'intervallo compreso tra i due livelli di soglia estremi. Se C_I e' contenuto solamente nel semiasse reale positivo, oppure in quello negativo, il convertitore viene detto **unipolare**; al contrario, se il campo di ingresso contiene valori di entrambi i segni, l'*ADC* e' detto **bipolare**. Per convertitori unipolari viene generalmente fornito solo l'estremo superiore del campo di ingresso, denominato fondo scala dell'*ADC*. Per gli *ADC* bipolari, invece, viene definito un valore di fondo scala per

ciascun dei due segni dell'ingresso; in questo caso, inoltre, C_I e' spesso costituito da un intervallo simmetrico rispetto all'origine, ossia i due fondo scala hanno lo stesso valore assoluto.

In fig.3(a) e' riportata una caratteristica di quantizzazione corrispondente a livelli di quantizzazione Q_k e di soglia T_k del tutto generici.



(a)

(b)

Fig.3. Generica caratteristica di quantizzazione (a) e di errore (b).

L'errore introdotto dall'operazione di quantizzazione e' poi definito dalla seguente relazione:

$$e_q(x_s) = x_q - x_s = Q(x_s) - x_s \qquad (3)$$

In particolare, quando x_s appartiene al campo di ingresso dell'*ADC* si parla di **errore di granularita'** o, piu' genericamente, di errore di quantizzazione. In caso contrario, ossia quando x_s supera i valori di fondo scala, l'uscita coincide con il valore corrispondente al livello di soglia estremo e la (3) fornisce il cosidetto **errore di sovraccarico** (overload error).

La caratteristica di errore $e_q(\cdot)$ corrispondente alla caratteristica di quantizzazione di fig.3(a) e' riportata in fig.3(b).

Nel seguito verra' utilizzata la relazione:

$$x_s \underset{q}{=} x_q \qquad (4)$$

per indicare che la differenza tra x_s e x_q e' dovuta solamente all'errore di quantizzazione.

L'Operazione di Codifica

Come gia' affermato, il valore dei livelli di quantizzazione viene in generale codificato in forma binaria; in particolare, utilizzando b cifre binarie e' possibile codificare fino a $B = 2^b$ livelli. La struttura che realizza la mappa vettoriale:

$$C : Q \rightarrow B^b \qquad con\ B = (0,1) \qquad (5)$$

viene detta codificatore.

Livello Quant.	x_q / $\overline{FS}$	MS x_c	C2 x_c
Q_{-4}	-1	-	100
Q_{-3}	-3/4	111	101
Q_{-2}	-2/4	110	110
Q_{-1}	-1/4	101	111
Q_{-0}	-0	100	-
Q_{+0}	0	000	000
Q_{+1}	1/4	001	001
Q_{+2}	2/4	010	010
Q_{+3}	3/4	011	011

Tab.I. Codifica di valori quantizzati su 3 bit: notazioni modulo e segno e complemento a due.

Usualmente i valori di ingresso sono supposti normalizzati rispetto al fondo scala dell'*ADC*; il campo di ingresso e' pertanto costituito dall'intervallo $C_I = [0,1)$ o dall'intervallo $C_I = [-1,1)$ a seconda che si il convertitore considerato sia unipolare o bipolare.

Nel caso di *ADC* bipolari, in particolare, il valore dei livelli di quantizzazione puo' essere codificato utilizzando diversi tipi di notazione. Si puo' ad esempio impiegare la notazione in complemento a due (*C2*), oppure la notazione in modulo e segno (*MS*); in questi casi, se si hanno a disposizione solamente tre bit ($B=2^3=8$) e si suppone che tutti i sotto-intervalli in cui viene suddiviso il campo di ingresso C_I siano eguali, la mappa $C: Q \rightarrow B^3$ e' definita dalla tab.I.

In generale, al livello Q_k di valore $k/(\frac{1}{2}B)$, viene fatto corrispondere il numero k espresso in forma binaria con b cifre, cioe':

$$Q_k \longrightarrow (k_1 k_2 ... k_b) \qquad k_m \in (0,1) \quad m = 1,...,b \qquad (6)$$

Pertanto, se il valore quantizzato x_q coincide con il livello Q_k, la regola di codifica consiste nell'assegnare a esso la parola binaria di b bit $(k_1 k_2 ... k_b)$:

$$x_q = Q_k \xrightarrow{C} x_c = (k_1 k_2 ... k_b) \qquad (7)$$

In generale, si scrivera' percio':

$$x_c = C[x_q] = (x_{c1} x_{c2} ... x_{cb}) \qquad (8)$$

Si noti che l'operazione di codifica non comporta alcuna perdita di informazione; e' cioe' possibile decodificare il valore ottenuto risalendo al corrispondente livello di quantizzazione. La regola di decodifica e' semplicemente:

$$x_c = (k_1 k_2 ... k_b) \xrightarrow{C^{-1}} x_q = Q_k \qquad (9)$$

dove k e' il numero avente rappresentazione binaria (in *MS* o *C2*) pari a $(k_1 k_2 ... k_b)$.

Esempio.
Se $b = 3$ e $x_q = -0.5 = -2/4$, in C2 si ha x_c=110. Viceversa, se x_c=110 si ottiene $x_q = -2/4 = -0.5$.

1.7

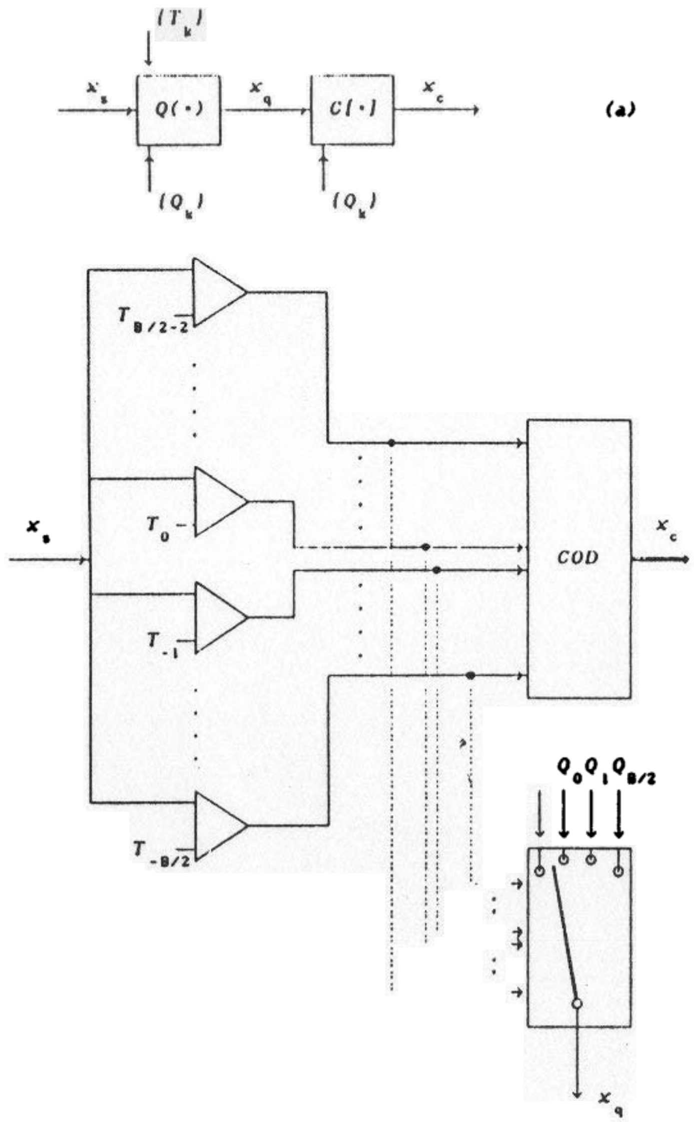


Fig.4. Schema di principio di un *ADC* (a) e una possibile realizzazione funzionale (b).

In fig.4(a) e' riportato lo schema di principio di un *ADC* ideale. La quantizzazione avviene eseguendo il confronto tra il valore di ingresso x_s e i B-1 livelli di soglia T_k; in uscita dal quantizzatore viene quindi fornito il valore del livello di quantizzazione Q_k risultante dal confronto. Come mostrato in fig.4(a), l'uscita del quantizzatore e' infine trasformata in un

valore numerico mediante un codificatore; questo blocco fornisce in uscita l'indice, normalmente rappresentato da una parola binaria, del livello di quantizzazione associato al valore x_s di ingresso.

Come si vedra' successivamente analizzando alcuni tipi di *ADC* in commercio, le operazioni di quantizzazione e di codifica sono spesso eseguite contemporaneamente; cio' significa che le cifre delle parole di codice vengono ottenute esaminando direttamente il valore continuo di ingresso anziche' quello quantizzato.

Da un punto di vista funzionale, le precedenti operazioni possono essere realizzate come mostrato in fig.4(b). Come si puo' vedere da tale figura, il confronto implicito nell'operazione di quantizzazione viene realizzato impiegando B-1 comparatori; ogni comparatore fornisce un'uscita che specifica se il valore del campione di ingresso e' maggiore o minore del livello di soglia a esso associato. Si ottiene cosi' un'insieme di B-1 segnali binari che rappresenta l'ingresso x_s con una notazione non posizionale di tipo a "termometro"; l'ultima uscita attiva rappresenta pertanto il livello di quantizzazione che corrisponde al valore di ingresso. Come mostrato in fig.4(b), l'informazione associata alle B-1 uscite binarie dei comparatori viene infine rappresentata mediante una parola binaria utilizzando un codificatore (digitale).

Si noti che lo schema funzionale ora illustrato non fornisce il valore quantizzato x_q; comunque, qualora fosse di interesse, questo valore puo' essere generato a partire dall'uscita dei comparatori e dai valori dei livelli di quantizzazione, come mostrato schematicamente nella stessa fig.4(b).

La Quantizzazione Uniforme

La legge di quantizzazione piu' diffusa suppone che tutti i livelli di soglia e tutti i livelli di quantizzazione dell'*ADC* siano equispaziati; la struttura cosi' ottenuta prende il nome di quantizzatore uniforme. Generalmente la caratteristica di quantizzazione e' supposta di pendenza "media" unitaria per cui la distanza (costante) tra due livelli di soglia adiacenti coincide con la distanza (anch'essa costante) tra due livelli di quantizzazione adiacenti; tale distanza viene denominata passo di quantizzazione Δ.

In generale, se b e' il numero di cifre binarie utilizzate per la rappresentazione del valore quantizzato, nel caso di *ADC* unipolari i $B = 2^b$ livelli di quantizzazione assumono i seguenti valori:

$$Q: \quad 0,\ \Delta,\ 2\Delta,\ \dots\ ,\ 1-\Delta \tag{10}$$

e il passo di quantizzazione Δ vale:

$$\Delta = 1/B = 2^{-b} \tag{11}$$

Nel caso di convertitori bipolari, invece, con la notazione *C2* i livelli di quantizzazione diventano:

$$Q: \ -1,\ -1+\Delta,\ \dots,\ -\Delta,\ 0,\ \Delta,\ \dots\ ,\ 1-\Delta \tag{12.a}$$

mentre con la notazione *MS* si ottengono i seguenti valori:

$$Q: \quad -1+\Delta,\ \dots,\ -\Delta,\ -0,\ 0,\ \Delta,\ \dots\ ,\ 1-\Delta \ ^{(1)} \tag{12.b}$$

Nelle (11) il passo di quantizzazione Δ vale:

$$\Delta = 2/B = 2^{-(b-1)} \tag{13}$$

Per quanto riguarda la legge di quantizzazione, se $x_s = k\Delta$, k intero, si pone $x_q = k\Delta$, mentre per

$$(k-1)\Delta < x_s < k\Delta \tag{14}$$

vengono utilizzate due diverse tecniche dette rispettivamente di arrotondamento e di troncamento, che differiscono tra loro per la posizione dei livelli di soglia.

Tecnica dell'arrotondamento

Questa tecnica consiste nell'associare a x_s il livello di quantizzazione piu' prossimo. Si ha pertanto:

$$Q_R(x_s) = k\Delta \qquad per \quad (k-\tfrac{1}{2})\Delta \le x_s < (k+\tfrac{1}{2}) \tag{15}$$

La caratteristica ingresso-uscita di un quantizzatore ad arrotondamento e' riportata in fig.5(a). In fig.5(b) e' riportata invece la caratteristica

---

[1] Si noti che l'intervallo di ingresso corrispondente al livello di valore 0 ha ampiezza doppia, per cui i livelli di quantizzazione distinti sono in realta' B-1.

dell'errore di quantizzazione $e_R(x_s) = Q_R(x_s) - x_s$. Tale figura mostra che, per qualsiasi valore dell'ingresso x_s, si ha:

$$-\tfrac{1}{2}\Delta < e_R(x_s) \le \tfrac{1}{2}\Delta \tag{16}$$

Si noti che, nel caso della tecnica di arrotondamento, entrambe le caratteristiche sono indipendenti dalla notazione utilizzata, *C2* o *MS*.

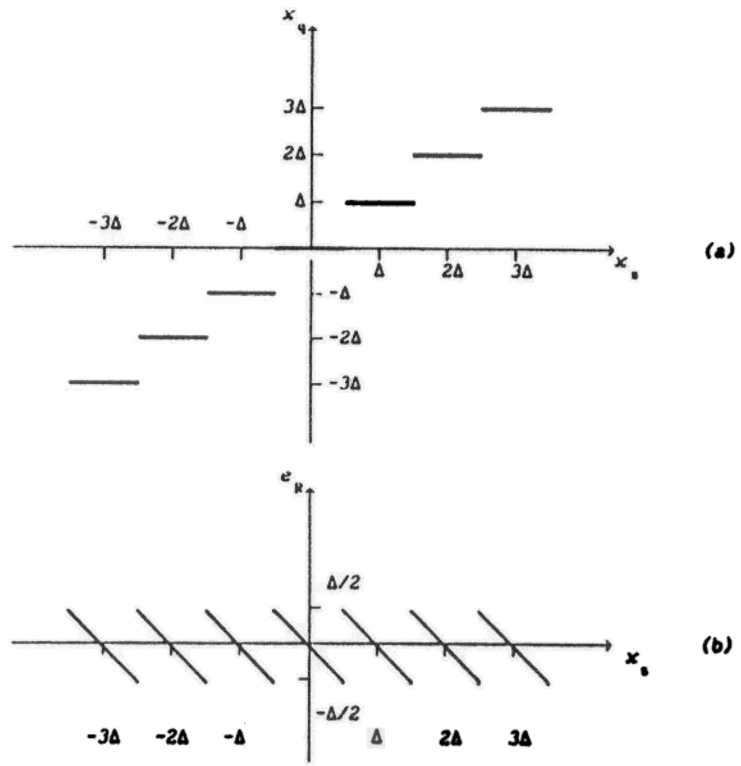


Fig.5. Caratteristica di I/O (a) e di errore (b) nel caso di quantizzazione ad arrotondamento.

Tecnica del troncamento

Questa tecnica consiste nel considerare solo i *b* bit piu' significativi di x_s. Nel caso di valori in notazione *MS* questo risultato e' ottenuto mediante la seguente relazione:

$$Q_T(x_s) = \begin{cases} (k-1)\Delta & x_s > 0 \\ k\Delta & x_s < 0 \end{cases} \qquad per \quad (k-1)\Delta < x_s < k\Delta \tag{17}$$

che corrisponde alla usuale nozione di troncamento; in base *10*, infatti, l'applicazione della (17) consente di quantizzare il numero *-3.4* nel numero *-3*, mentre il numero *+3.4* viene troncato in *+3*.

La caratteristica $Q_T(\cdot)$ di un quantizzatore a troncamento in notazione *MS* e' riportata in fig.6(a); da questa figura si vede che $|Q_T(x_s)| \le |x_s|$ per ogni valore di x_s. La caratteristica dell'errore di quantizzazione $e_T(x_s) = Q_T(x_s) - x_s$ e' riportata invece in fig.6(b); si puo' notare che:

$$-\Delta < e_T(x_s) \le 0 \qquad per \quad x_s > 0 \qquad (18.a)$$

$$0 \le e_T(x_s) < \Delta \qquad per \quad x_s < 0 \qquad (18.b)$$

per cui $|e_T(x)| < \Delta$. Si osservi in particolare che il segno dell'errore e' sempre opposto al segno del valore di ingresso.

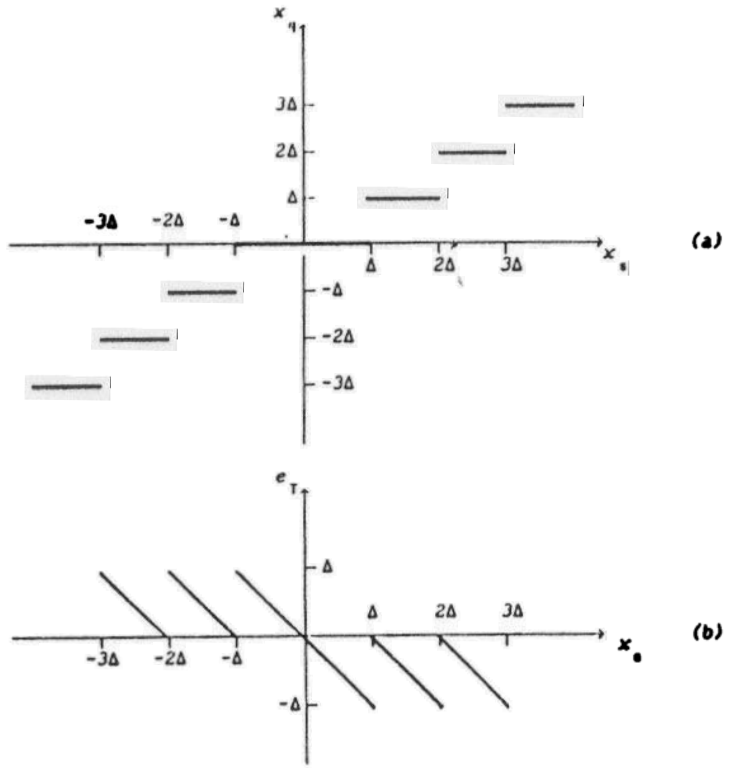


Fig.6. Caratteristica di I/O (a) e di errore (b) nel caso di quantizzazione a troncamento in notazione *MS*.

Nel caso di valori in notazione *C2* la quantizzazione a troncamento e' espressa dalla seguente relazione:

$$Q_T(x_s) = (k-1)\Delta \qquad per \qquad (k-1)\Delta < x_s < k\Delta \qquad (19)$$

La corrispondente caratteristica di quantizzazione $Q_T(\cdot)$ e' riportata in fig.7(a); come e' evidente dalla definizione, si ha $Q_T(x_s) \le x_s$, per ogni valore di x_s. La caratteristica dell'errore di quantizzazione $e_T(x_s) = Q_T(x_s)-x_s$ e' riportata invece in fig.7(b); come si puo' notare, il segno dell'errore e' indipendente da x_s e vale la relazione:

$$-\Delta < e_T(x_s) \le 0 \qquad per\ ogni \quad x_s \qquad (20)$$

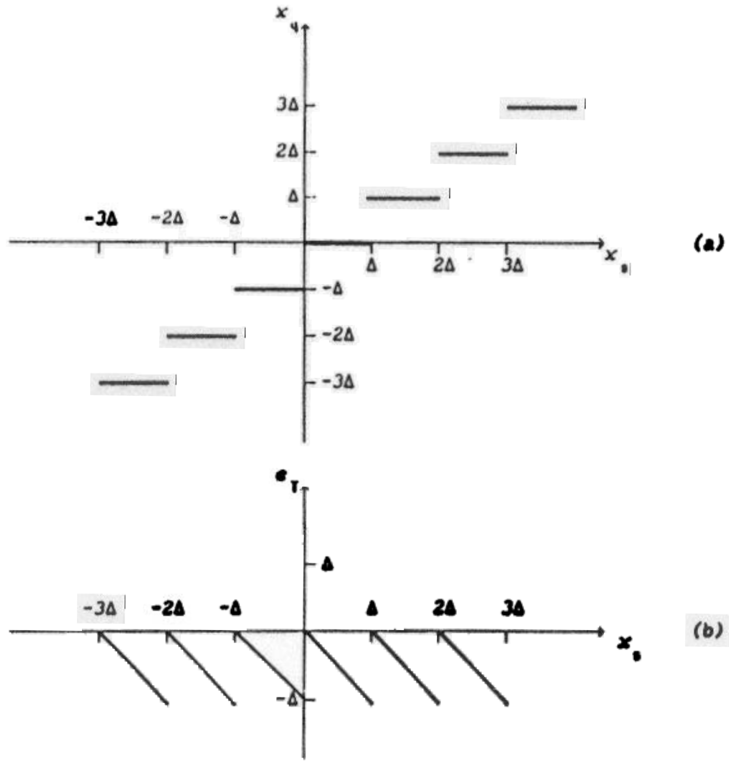


Fig.7. Caratteristica di I/O (a) e di errore (b) nel caso di quantizzazione a troncamento in notazione *C2*.

5f

Dall'esame delle precedenti figure risulta che la caratteristica di quantizzazione relativa all'arrotondamento dipende solamente da Δ, mentre quella relativa al troncamento dipende anche dalla notazione usata. Si ha inoltre:

$$|e_R(x_s)| \le \frac{1}{2}\Delta \qquad |e_T(x_s)| < \Delta \qquad (21)$$

Per ridurre l'errore di quantizzazione e' pertanto conveniente utilizzare un numero di bit elevato (infatti $\Delta = 2^{-b}$) e la tecnica dell'arrotondamento. In ogni caso, comunque, un aumento del numero di bit comporta un aumento del costo dell'*ADC* e del tempo necessario per la conversione. Nelle usuali applicazioni sono generalmente sufficienti *ADC* a *8÷12* bit; per applicazioni di precisione possono comunque essere richiesti *ADC* a *16* bit o piu'.

51