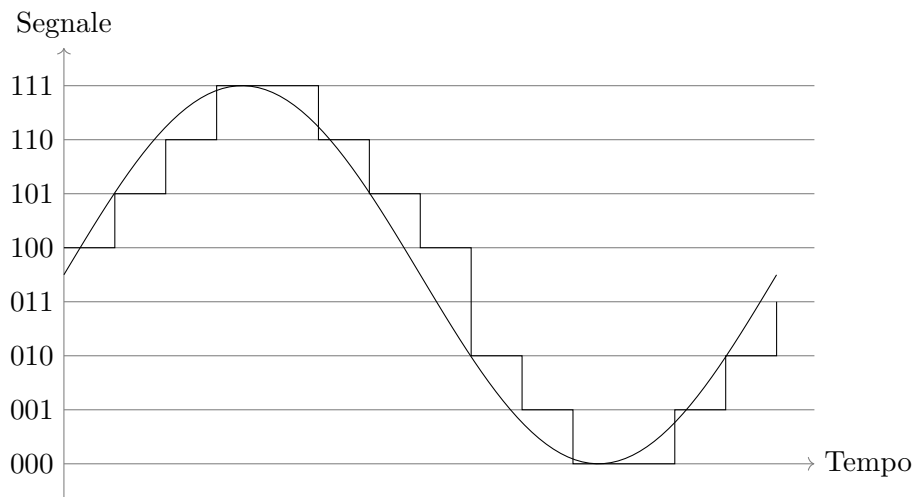


Conversione tra segnali analogici e digitali

1 Risoluzione di un segnale digitale

Quando un segnale analogico viene rappresentato in forma digitale, si usano una serie di parole binarie (in sostanza numeri interi) per rappresentarne il valore in diversi istanti di tempo:



Le parole binarie rappresentano però valori discreti, mentre un segnale analogico assume valori continui: si ha allora inevitabilmente un certo grado di approssimazione, che introduce di fatto un errore nella conversione tra analogico e digitale. All'aumentare del numero di bit che compongono una parola aumenta (esponenzialmente) il numero di valori diversi rappresentabili, cioè la **risoluzione**, diminuendo l'errore dovuto all'approssimazione. Perciò, la risoluzione degli ADC e dei DAC è indicata dalla dimensione della parola binaria su cui operano.

Ad esempio, un ADC a 4 bit può distinguere solo $2^4 = 16$ livelli di tensione in ingresso, che traduce nelle parole da 0 (0000) a 15 (1111), mentre qualunque livello "intermedio" viene approssimato; ogni livello copre il 6.25% dell'intervallo di tensioni in input. Invece, un ADC a 16 bit è in grado di distinguere $2^{16} = 65\,536$ livelli di input, ciascuno dei quali copre circa lo 0.0015% dell'intervallo — una risoluzione molto più alta rispetto a quella dell'ADC a 4 bit.

2 Tensione di riferimento

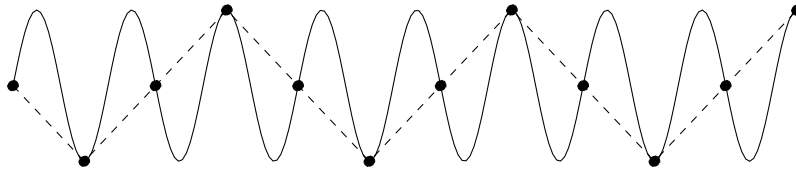
La tensione più alta che un ADC è in grado di leggere, o che un DAC è in grado di emettere, e che corrisponde alla parola $111 \dots 1$,¹ prende il nome di **tensione di riferimento**, poiché tutti gli altri livelli di tensione vengono calcolati relativamente a essa.

La tensione di riferimento è in genere fissa (o comunque limitata a un certo valore massimo / intervallo di valori) per un determinato ADC o DAC. Ad esempio, un tipico Arduino ha un ADC che opera a una tensione di riferimento massima pari alla tensione di alimentazione del microcontrollore, cioè $+3.3\text{ V}$ o $+5\text{ V}$.

3 Frequenza di campionamento

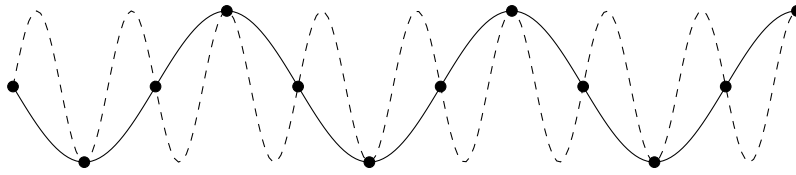
Siccome un segnale analogico varia nel tempo, un ADC deve misurarne il valore periodicamente, e abbastanza spesso da descrivere accuratamente la forma del segnale. Allo stesso modo, un DAC deve aggiornare il valore di tensione in uscita abbastanza spesso da riuscire a ricostruire la forma del segnale analogico rappresentato in forma digitale.

Il processo di misurazione periodica del segnale è chiamato **campionamento**, e il numero di campioni acquisiti (misurazioni effettuate) nell'unità di tempo è detto **frequenza di campionamento** (generalizzando anche al caso della conversione da digitale ad analogico, si parla di *frequenza di conversione*). Se questa è troppo bassa, il segnale digitale ottenuto dalla conversione può diventare un **alias** del segnale originale, cioè un segnale di fatto completamente diverso, che assume gli stessi valori di quello originale solo negli istanti in cui è avvenuto il campionamento. Ad esempio:



Infatti, riconvertendo questo segnale digitale in uno analogico, si otterrebbe un risultato del genere:

¹La tensione associata alla parola $111 \dots 1$ non è sempre esattamente uguale alla tensione di riferimento: alcuni ADC/DAC (tra cui l'ADC integrato nel microcontrollore delle schede Arduino UNO) associano invece a tale parola la tensione $(1 - 1/2^n)V_{\text{ref}}$, dove n è il numero di bit di risoluzione. Ad esempio, con $V_{\text{ref}} = 5\text{ V}$ e $n = 4$, il livello di tensione più alto sarebbe $15/16 \cdot 5\text{ V} = 4.6875\text{ V}$.

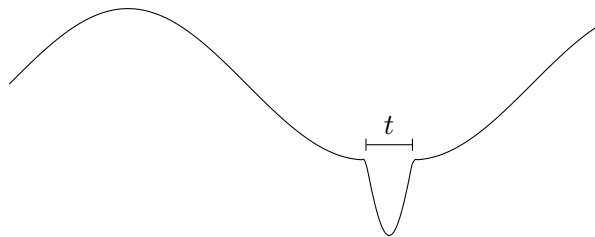


(qui la linea continua rappresenta il segnale riconvertito, mentre quella tratteggiata è il segnale originale).

Il **teorema di Nyquist** afferma che, per evitare di ottenere un alias, la frequenza di campionamento/conversione f_{conv} deve essere superiore al doppio della frequenza massima del segnale, f_{max} :

$$f_{\text{conv}} > 2f_{\text{max}}$$

Per un segnale sinusoidale, la frequenza massima è semplicemente la frequenza della sinusoide, cioè il numero di ripetizioni dell'onda nell'unità di tempo, ovvero il reciproco del periodo (durata) di un'onda. Invece, per un segnale "irregolare", la frequenza massima è determinata sostanzialmente dal periodo dei "dettagli più piccoli" presenti nel segnale:



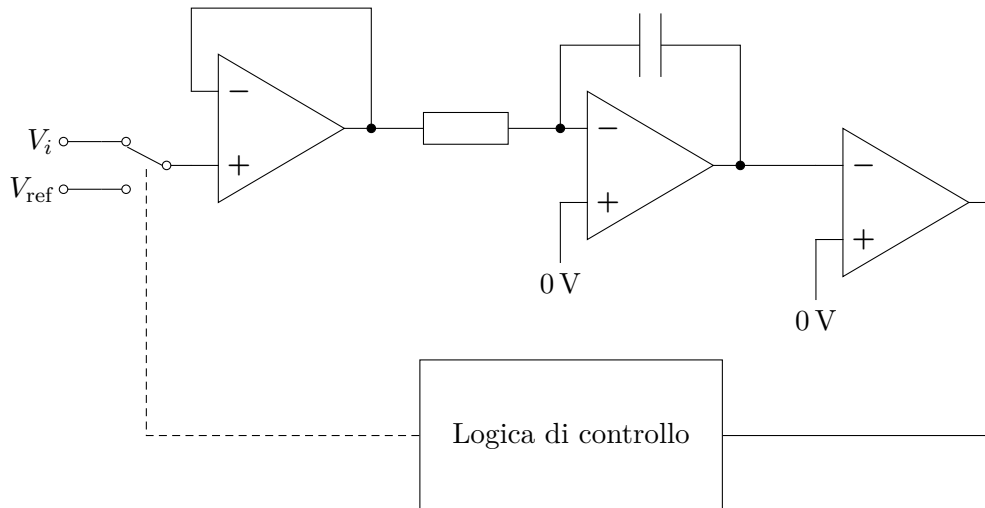
4 Funzionamento degli ADC

Esistono tanti diversi tipi di ADC, ma tutti sono caratterizzati da una certa complessità, perché la conversione di un segnale da analogico a digitale è intrinsecamente difficile.

In seguito, verranno brevemente illustrati due tipi di ADC: quelli a **doppia rampa**, molto usati ad esempio nei tester, e quelli ad **approssimazioni successive**, impiegati ad esempio nelle tipiche schede Arduino.

4.1 ADC a doppia rampa

Un ADC di tipo a doppia rampa (dual-slope) ha la seguente struttura (semplificata):

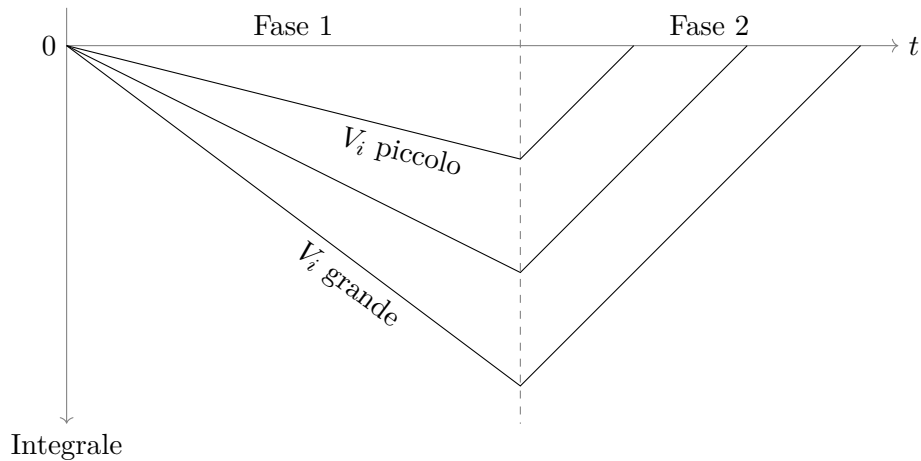


In questo circuito sono presenti tre amplificatori operazionali, che, andando da sinistra a destra, sono configurati come:

1. un voltage follower, per ottenere un'elevata resistenza in ingresso;
2. un integratore, che calcola l'integrale (invertito di segno) del segnale nel tempo;
3. un *comparatore*, che confronta le tensioni sui suoi due ingressi (qui quella sull'ingresso non invertente è fissata a 0 V), emettendo un output positivo o negativo per indicare quale delle due tensioni sia maggiore.

Per misurare il segnale analogico in ingresso, il convertitore (comandato dalla sua logica di controllo interna):

1. integra il segnale in ingresso per un periodo di tempo di durata prefissata, durante il quale l'integrale (l'output dell'integratore) aumenta con una velocità proporzionale al valore del segnale (se il segnale è positivo, l'output dell'integratore aumenta in direzione negativa);
2. collega all'integratore una tensione di riferimento V_{ref} — che è di polarità opposta alla tensione del segnale, e fa tornare l'integrale verso zero a una velocità nota — poi misura il tempo che passa finché il comparatore non indica che l'output dell'integratore è tornato a zero.

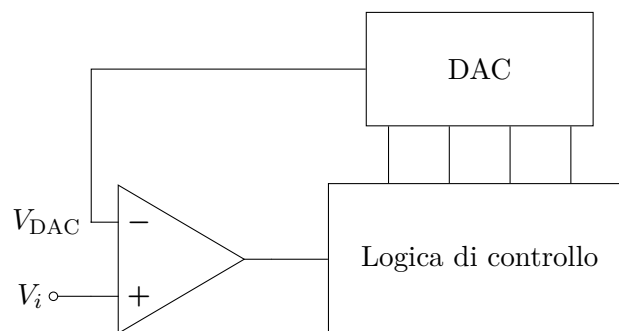


Siccome la durata della fase 1 è costante, il valore finale raggiunto dall'output dell'integratore dipende solo dalla velocità con cui tale output è aumentato, ovvero dalla tensione del segnale analogico. Viceversa, nella fase 2, la velocità di variazione dell'output è costante (fissata da V_{ref}), dunque il tempo misurato dipende dal valore massimo dell'integrale. Allora, da questo tempo, che è misurato in forma digitale mediante un apposito contatore, si può ricavare la rappresentazione digitale del valore del segnale analogico.

Questo tipo di ADC ha una risoluzione tanto maggiore quanto più è il tempo dedicato alla misurazione, a scapito però della velocità, e quindi della massima frequenza di campionamento possibile.

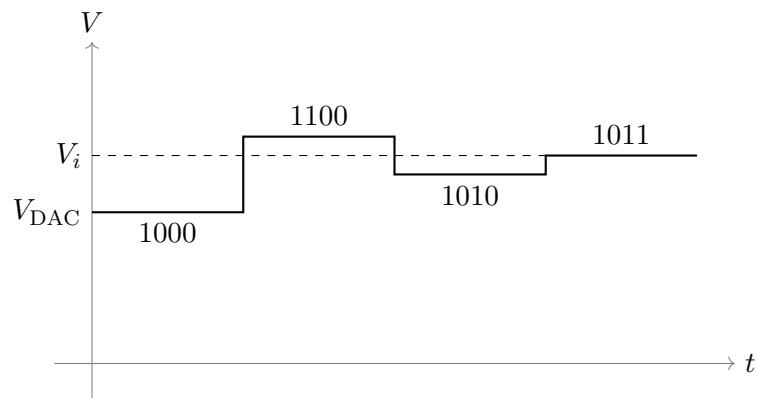
4.2 ADC ad approssimazioni successive

Un ADC ad approssimazioni successive è composto si basa su un convertitore digitale-analogico (DAC) e un comparatore:



Esso confronta il segnale analogico in ingresso con una tensione generata dal DAC, e varia la parola binaria mandata in input al DAC, secondo un qualche algoritmo di ricerca,

finché questo non produce una tensione il più possibile vicina (idealmente uguale) a quella in input.



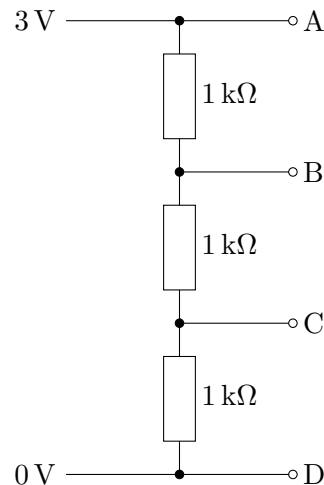
A questo punto, la parola di input del DAC corrisponde al valore digitale del segnale.

5 Funzionamento dei DAC

Anche i DAC possono essere di vari tipi, ma la conversione da digitale ad analogico è in genere più “facile” rispetto a quella nel verso opposto.

5.1 DAC a scala di resistenze

Un tipo di DAC semplice e ad alte prestazioni (adatto, per esempio, ad applicazioni audio) è quello basato su una “**scala di resistenze**”. Ad esempio, una scala di resistenze per un DAC a 2 bit (4 possibili livelli di tensione in uscita) con una tensione di riferimento $V_{ref} = 3\text{ V}$ (per comodità di calcolo) è fatta in questo modo:



La corrente che scorre nelle tre resistenze è

$$I = \frac{V_{\text{ref}}}{R_{\text{tot}}} = \frac{3 \text{ V}}{3 \text{ k}\Omega} = 1 \text{ mA}$$

dunque le tensioni ai quattro terminali A–D, misurate rispetto a D (cioè a 0 V), sono:

$$V_{AD} = IR_{AD} = 1 \text{ mA} \cdot 3 \text{ k}\Omega = 3 \text{ V}$$

$$V_{BD} = IR_{BD} = 1 \text{ mA} \cdot 2 \text{ k}\Omega = 2 \text{ V}$$

$$V_{CD} = IR_{CD} = 1 \text{ mA} \cdot 1 \text{ k}\Omega = 1 \text{ V}$$

$$V_{DD} = 0 \text{ V}$$

Così, il DAC può generare quattro livelli di tensione in uscita semplicemente selezionando quale dei quattro terminali collegare all’output, in base alla parola digitale ricevuta in ingresso (ad esempio, il terminale A viene usato per la parola 11 e D viene usato per la parola 00).

5.2 Pulse Width Modulation

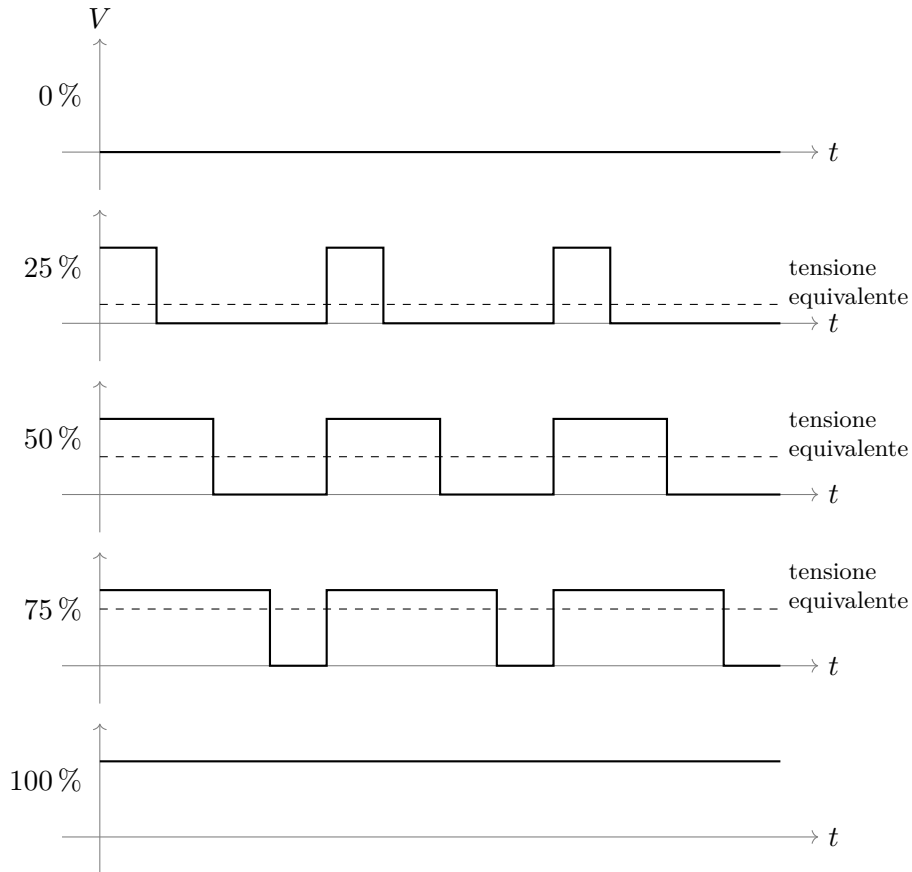
Un altro metodo di conversione da digitale ad analogico è la tecnica chiamata **Pulse Width Modulation (PWM)**, che è meno adatta per applicazioni che hanno bisogno di segnali analogici precisi (come l’audio), ma ottima ad esempio per variare l’intensità di una luce, la velocità di un motore elettrico, ecc.

Essa si basa su un output digitale che viene ripetutamente “acceso” (portato alto, a 1) e “spento” (portato basso, a 0). Se la frequenza a cui ciò avviene è abbastanza alta, si ha un effetto analogo a quello di un output a tensione variabile, perché, ad esempio:

- una luce che si accende e spegne abbastanza velocemente viene percepita dall’occhio umano come una luce costante a luminosità media;

- un motore elettrico acceso e spento abbastanza velocemente tende, per inerzia, a stabilizzarsi a una velocità media.

Variando la proporzione tra il tempo in cui l'output è alto e basso (cioè “modulando la larghezza degli impulsi”, da cui il termine *Pulse Width Modulation*), si può “simulare” qualunque tensione di uscita compresa tra i valori di tensione digitale alta e bassa:



Il grosso vantaggio di questa tecnica è che un segnale PWM può facilmente essere usato, tramite un transistor, per pilotare carichi anche grandi (come appunti lampade, motori elettrici, ecc.), mentre l'output di altri tipi di DAC dovrebbe essere mandato a un amplificatore di potenza per poter comandare grossi carichi.

È importante che un transistor usato in questo modo sia molto più veloce rispetto alla frequenza del segnale PWM. Infatti, un transistor dissipa poca energia quando è “completamente” acceso o spento, e ne dissipa invece molta di più durante le transizioni acceso-spento e spento-acceso, che quindi devono avere la durata minima possibile. Se l'energia dissipata fosse troppa, si ridurrebbe l'efficienza del sistema, e si rischierebbe di bruciare il transistor.