

UNIVERSITÀ DEGLI STUDI DI PADOVA  
DIPARTIMENTO DI TECNICA E GESTIONE DEI SISTEMI  
INDUSTRIALI

Corso di Laurea Triennale in Ingegneria Meccatronica

**Stato dell'arte della conversione D/A in  
applicazioni HiFi**

Relatore:

**Prof. Paolo Magnone**

Laureando:

**Mario de Privitello**

**1220309**

---

ANNO ACCADEMICO 2021/2022



# Indice

<b>Introduzione</b>	<b>1</b>
<b>1 I fondamenti della conversione</b>	<b>3</b>
1.1 La conversione D/A . . . . .	3
1.2 La transcaratteristica . . . . .	5
1.2.1 Errore di guadagno . . . . .	6
1.2.2 Errore di offset . . . . .	6
1.2.3 Errore di non linearità . . . . .	7
1.3 Schemi di conversione . . . . .	9
1.3.1 DAC a resistenze pesate . . . . .	9
1.3.2 DAC a traliccio R-2R . . . . .	11
1.3.3 DAC a conversione seriale . . . . .	12
1.3.4 Utilizzo della DPWM . . . . .	13
1.4 Innovazioni nella conversione . . . . .	13
1.4.1 Il Re-DAC . . . . .	14
1.4.2 Utilizzo della DDPM . . . . .	15
<b>2 La conversione D/A e la riproduzione audio</b>	<b>17</b>
2.1 Caratteristiche dei segnali audio . . . . .	17
2.2 Schema generale per la conversione audio . . . . .	17
2.2.1 Il sovracampionatore . . . . .	18
2.2.2 Il DAC . . . . .	24
2.3 Le architetture bitstream . . . . .	26
2.4 Fattori di downgrade alla risoluzione . . . . .	28
2.4.1 Il jitter . . . . .	28
2.4.2 Non linearità e passaggio per lo zero . . . . .	30
<b>3 Convertitori a confronto</b>	<b>33</b>
3.1 Sonic frontiers SFD-2 . . . . .	35
3.2 Rockna Wavedream DAC . . . . .	36
3.3 Texas Instruments PCM2704C . . . . .	38
3.4 Confronto prestazionale . . . . .	39
<b>Conclusioni</b>	<b>41</b>



# Introduzione

Il legame tra conversione D/A e riproduzione audio nasce nella prima metà degli anni '80, per l'esattezza il 1° ottobre 1982, giorno in cui l'album "52nd street" di Billy Joel venne commercializzato, primo fra tutti, in formato CD. Da quel momento, complice la rapida diffusione del supporto, si ebbe una forte spinta evolutiva nel settore, sia nell'ottica di contenimento dei costi che squisitamente prestazionale. Il DAC, divenuto componente fondamentale della catena di riproduzione, non fu da meno: la ricerca si concentrò sia sul mercato "mainstream", proponendo soluzioni poco costose e facilmente integrabili, che su soluzioni "no compromise", in cui le più piccole accortezze venivano implementate per il miglioramento della conversione. Si vennero inizialmente a creare due grosse categorie di convertitori: nella fascia "medio-bassa"<sup>1</sup> spopolavano i convertitori *bitstream*, architetture a singolo bit poco costose e facilmente integrabili, mentre nella fascia più alta<sup>2</sup> i *multibit* rimanevano l'opzione predominante. Come spesso accade, questa divisione sfumò col tempo, sia per quanto riguarda il prezzo (esistono ora soluzioni bitstream hi-end come multibit a basso costo), sia per quel che riguarda le architetture stesse (furono sviluppate in seguito architetture su base bitstream a numero limitato, ma non unitario, di bit).

Obiettivo di questo testo sarà quello di approfondire l'utilizzo dei convertitori D/A nel sopracitato ambito, le loro possibili architetture e la loro evoluzione nel tempo. Si partirà quindi con il trattare la conversione D/A in se e per se, analizzandone il funzionamento teorico e alcuni possibili schemi circuitali con cui questa può essere attuata. Si andrà poi a parlare dei DAC per la riproduzione audio, esponendo lo schema che li accomuna e le problematiche che più incidono sulle loro prestazioni. Nel terzo capitolo, infine, si confronteranno tre convertitori reali di età e fasce di prezzo differenti, in modo da vedere come tali circuiti siano evoluti nel tempo.

---

<sup>1</sup>Intendendo per "medio-bassa" dispositivi il cui prezzo poteva tranquillamente raggiungere le £5.000.000

<sup>2</sup>Qua i prezzi per il solo convertitore potevano variare dai 10 ai 15 milioni di lire



# Capitolo 1

## I fondamenti della conversione

Essendo questa, come è ormai facile intuire, una trattazione sull'uso dei DAC<sup>1</sup> in campo audiofilo, risulta opportuno fare una breve, ma per quanto possibile esaustiva, introduzione alla conversione in sé per sé. In questo primo capitolo si cercherà quindi di formare una minima base teorica utile alla comprensione degli argomenti trattati nella sezioni successive, andando ad analizzare la conversione "su carta", la transcaratteristica ideale che ne deriva, e come le non idealità la modifichino nel caso reale. Si andrà infine ad analizzare come avviene la conversione a livello circuitale, prendendo in considerazione alcuni basilari schemi di conversione.

### 1.1 La conversione D/A

Come intuibile dal nome, la conversione D/A è la tecnica che permette di trasformare una grandezza digitale, campionata nel dominio del tempo e quantizzata in ampiezza, in una grandezza analogica, continua sia nel dominio del tempo che in quello delle ampiezze. Quest'ultima, benché la grandezza digitale di partenza possa assumere qualunque significato fisico, risulta sempre di natura elettrica, dovendo essere dunque ritrasformata nella forma originaria attraverso un opportuno trasduttore.<sup>2</sup>

Prendiamo in considerazione la grandezza digitale di partenza: essa è formata da una serie di "parole" dette *campioni*, ciascuna composta da un numero prefissato di bit, tutte equidistanziate temporalmente tra loro. I bit di ciascun campione, essendo la notazione binaria una notazione posizionale, assumono peso diverso a seconda della posizione che occupano, andando da  $2^0$  per il bit più a destra, detto *least significant bit* (LSB), fino a  $2^{n-1}$  per l'ennesimo bit. Fissato il fondo scala  $FS$  del convertitore, ovvero la massima tensione che questo potrà produrre in uscita, e il

---

<sup>1</sup>Digital to Analog Converter

<sup>2</sup>Nel caso di interesse, il segnale elettrico in uscita dal lettore, e quindi dal DAC, dopo essere condizionato dall'amplificatore, viene trasformato in onde sonore dai diffusori, il trasduttore della nostra catena di riproduzione.

numero  $n$  di bit a cui si andrà a lavorare, risulta immediato calcolare

$$Q = FS/(2^n) \quad (1.1)$$

ove  $Q$  viene detto *passo di quantizzazione*, o *quanto*, equivalente alla minima differenza di tensione producibile dal convertitore in questa configurazione.<sup>3</sup> Una volta ottenuto  $Q$ , la conversione è immediata, e consiste nell'associare ad un generico campione  $D_{n-1}D_{n-2}...D_2D_1D_0$  la tensione in uscita

$$V = Q \cdot (D_{n-1} \cdot 2^{n-1} + D_{n-2} \cdot 2^{n-2} + \dots + D_2 \cdot 2^2 + D_1 \cdot 2 + D_0) \quad (1.2)$$

Due cose saltano subito all'occhio analizzando questa equazione:

- La tensione di uscita può assumere solo determinati valori, corrispondenti ai multipli interi di  $Q$  da 0 fino a  $V_{aMAX}$ , il che genererà un andamento del segnale "a gradini". Questo effetto può essere smorzato diminuendo  $Q$  attraverso l'aumento del numero di bit utilizzati (si veda la fig 1.1), al costo logicamente di una maggior difficoltà nell'implementazione.<sup>4</sup>
- La tensione massima  $V_{aMAX}$  raggiungibile non corrisponde a quella di fondo-scala  $FS$ , bensì ad  $FS - Q$ . La cosa risulta intuitiva: dalla (1.1) è immediato scrivere

$$FS = 2^n \cdot Q \quad (1.3)$$

mentre

$$V_{aMAX} = Q \cdot (1 \cdot 2^{n-1} + 1 \cdot 2^{n-2} + \dots + 1 \cdot 2^2 + 1 \cdot 2 + 1) \quad (1.4)$$

la quale differisce in difetto da  $FS$  appunto per  $Q$ .

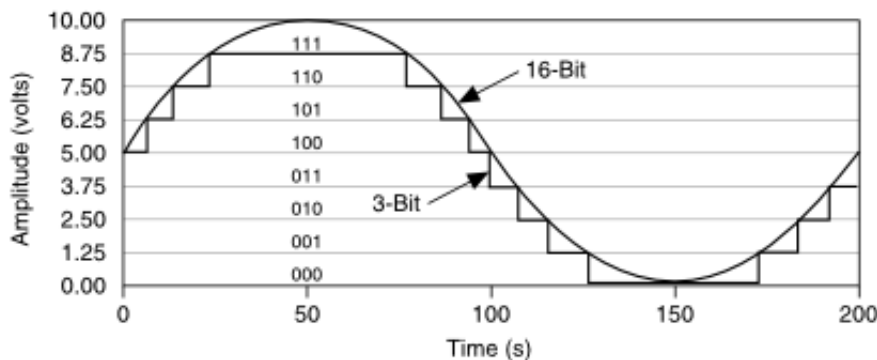


Figura 1.1: Confronto fra un segnale convertito a 3 bit e uno convertito a 16 bit. [1]

<sup>3</sup>Risulta intuitivo comprendere come  $Q$  sia pure il valore di tensione assunto dal LSB

<sup>4</sup>Vedremo in seguito come il grosso della "ripulitura" del segnale viene effettuata in stadi di condizionamento successivi attraverso l'utilizzo di appositi filtri di ricostruzione.



## 1.2 La transcaratteristica

Si definisce *transcaratteristica* la funzione di trasferimento che mette in relazione la grandezza di ingresso ad un dispositivo con la relativa grandezza in uscita. Se ne riporta di seguito un esempio riferito ad un DAC ideale a 3 bit.

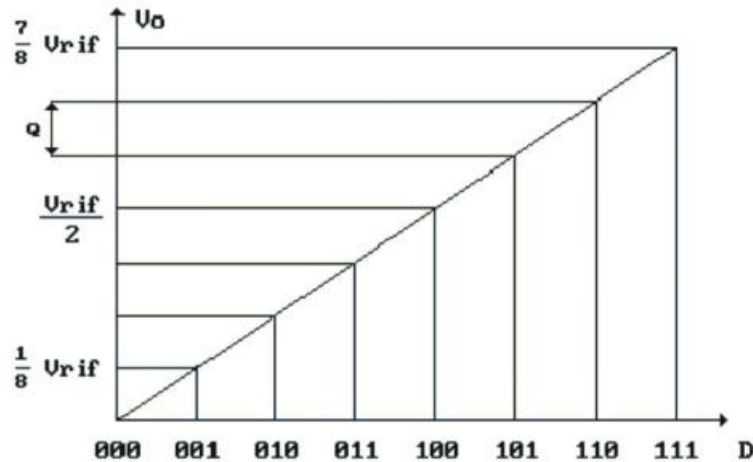


Figura 1.2: Transcaratteristica ideale di un DAC a 3 bit. [2]

E' facile, dopo una veloce analisi, verificare quanto detto prima:

- Ad ogni campione in ingresso corrisponde un solo possibile valore di tensione in uscita.
- Le tensioni ottenibili differiscono l'un l'altra per una quantità  $Q = 1/8FS$ .
- La tensione  $V_{MAX}$  raggiungibile non corrisponde ad  $FS$ , bensì a  $7/8FS = FS - Q$ .

Sfortunatamente, tutta questa semplicità è riscontrabile solo nel caso ideale: le non idealità caratterizzanti il funzionamento dei componenti alla base del convertitore, derivate soprattutto dagli amplificatori operazionali, impattano inevitabilmente sulla sua transcaratteristica, che si discosterà più o meno ampiamente dal suo andamento teorico. I tre maggiori tipi di scostamento dalla curva ideale, detti *errori*, sono i seguenti:

- Errore di guadagno
- Errore di offset
- Errore di non linearità

Si andrà ora ad analizzarli nel dettaglio, comprendendone le cause, gli effetti e le eventuali soluzioni.

### 1.2.1 Errore di guadagno

L'errore di guadagno, detto anche di *gain*, è dato dall'andamento variabile che assume questo parametro all'interno degli amplificatori operazionali. Questa variazione tende ad aumentare con la crescita del codice di ingresso, generando una diversa inclinazione della curva reale rispetto a quella attesa (si faccia riferimento alla fig.1.3).

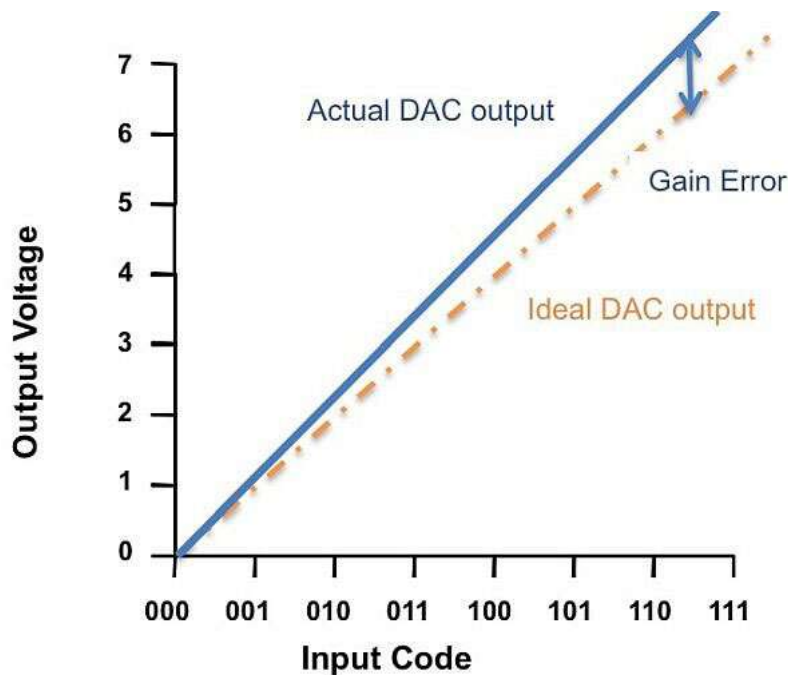


Figura 1.3: Effetto dell'errore di gain sulla transcaratteristica ideale di un DAC. [3]

L'errore viene definito quantitativamente come la differenza assunta dall'uscita  $V$  nel caso reale rispetto al valore teorico, avendo posto in ingresso un codice con tutti i bit pari a 1.<sup>5</sup> Questo può essere espresso indifferentemente come frazione del valore di fondoscala, o relazionato al valore dell'LSB, ed è facilmente correggibile tramite opportune operazioni di taratura.

### 1.2.2 Errore di offset

L'errore in questione è generato dalla tensione di offset  $V_0$  sempre presente in uscita all'amplificatore operazionale. L'equazione della tensione in uscita passerà quindi dalla (1.2) a

$$V = Q \cdot N + V_0 \quad (1.5)$$

<sup>5</sup>Potremmo dire ovviamente che l'errore è dato dalla differenza della  $V_{MAX}$  reale rispetto a quella teorica

con  $N = \sum D_i \cdot 2^i$ . L'effetto immediatamente visibile sulla curva è una traslazione rigida di entità  $V_0$ , come riscontrabile in fig.1.4.

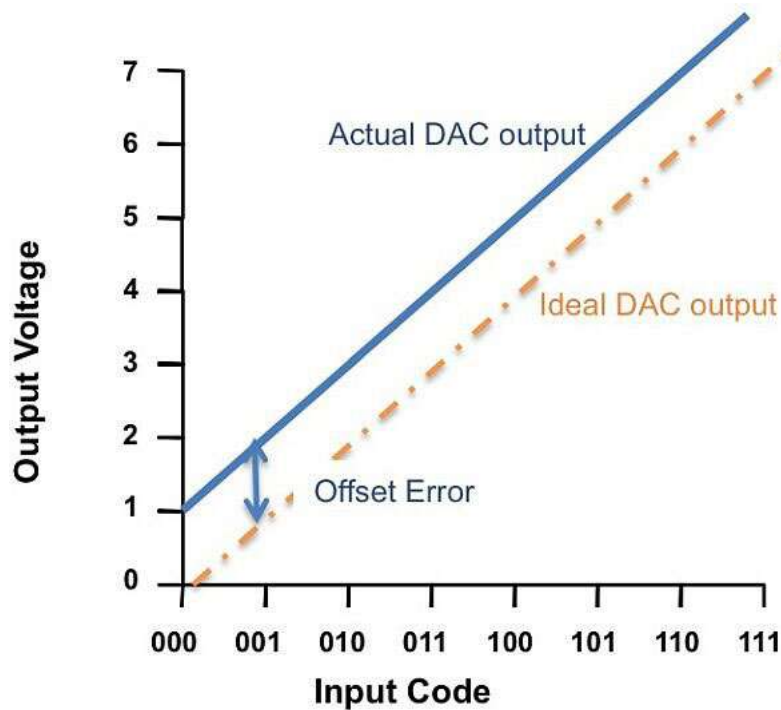


Figura 1.4: Errore di offset su un DAC ideale a 3 bit. [3]

Come per il caso precedente, l'errore è esprimibile come frazione di FS o in relazione all'LSB, ed è correggibile tramite una semplice taratura.

### 1.2.3 Errore di non linearità

L'errore di non linearità, il più difficile da analizzare e correggere, prende in considerazione, per l'appunto, la generale *non linearità* della transcaratteristica reale. Esso si può scomporre in due diverse tipologie:

- non linearità *integrale*
- non linearità *differenziale*

#### Non linearità integrale

L'errore di non linearità integrale, anche detto *INL*, è individuato come la massima deviazione fra la transcaratteristica ideale e quella reale a parità di codice di ingresso. Come si può vedere in fig.1.5, esso individua lo scostamento della curva reale rispetto ai valori teorici, una volta compensate le componenti di offset e di gain. L'entità dell'errore, è ormai superfluo ripeterlo, può essere espresso in frazioni di FS o posto in relazione all'LSB.

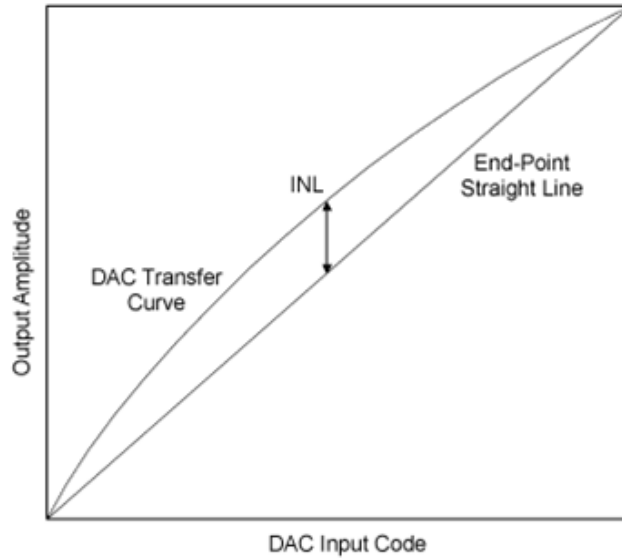


Figura 1.5: Errore di non linearità integrale [4]

### Non linearità differenziale

L'errore di non linearità differenziale (DNL) è definito come la massima deviazione tra la differenza di tensione generata da due ingressi adiacenti<sup>6</sup> e il passo di quantizzazione teorico  $Q$ .

In formule, possiamo scrivere che, per un generico  $n$ :

$$DNL_n = (V_n - V_{n-1}) - Q \quad (1.6)$$

L'andamento di questo errore risulta un ottimo indice per monitorare le variazioni di pendenza che affliggono la transcaratteristica reale, permettendo inoltre di rilevare l'eventuale *non-monotonicità* della stessa.<sup>7</sup> Quest'ultima si verifica ogniqualvolta il  $DNL$  risulti minore di  $-Q$ , generando un'uscita  $V_n > V_{n+1}$ . Tale fenomeno è da evitare categoricamente nel caso il DAC venga impiegato in sistemi di controllo a catena chiusa (il cui corretto funzionamento è legato anche alla complementarità delle conversioni effettuate da ADC e DAC) o in particolari schemi di conversione A/D<sup>8</sup>, onde comprometterne seriamente il funzionamento. Nell'ambito di nostro interesse, un DNL marcato può portare alla cosiddetta *distorsione da passaggio per lo zero*, che analizzeremo in maniera più approfondita nel secondo capitolo.

<sup>6</sup>Ovvero che differiscono fra di loro per un LSB

<sup>7</sup>Si parla di *non-monotonicità* quando la pendenza di curva non mantiene segno costante nell'intervallo di interesse.

<sup>8</sup>Ad esempio, negli ADC a gradinata la presenza di non monotonicità nel DAC genera il cosiddetto errore di *missing code*.

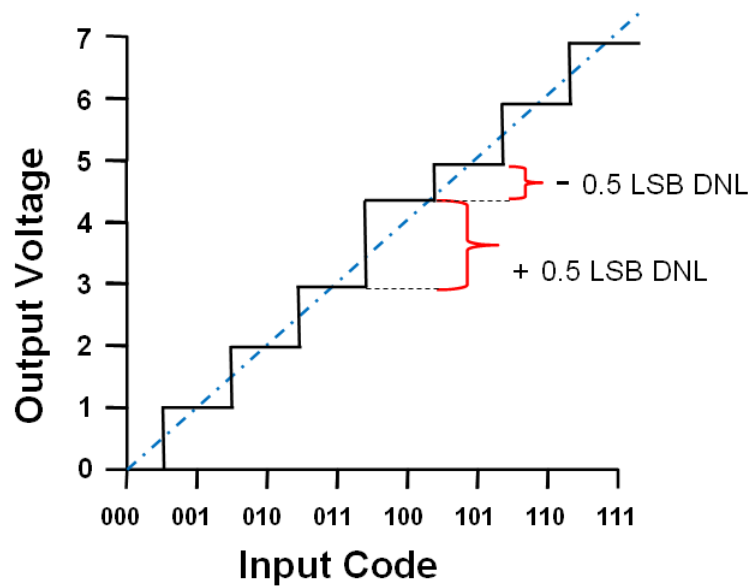


Figura 1.6: Errore di non linearità differenziale. [5]

## 1.3 Schemi di conversione

Andremo ora ad analizzare alcuni semplici circuiti per la conversione D/A. Seppur questi possano risultare molto basilari nel loro funzionamento, costituiscono la base delle architetture che verranno studiate nei capitoli successivi. Si noti come tutti i convertitori riportati presentino al loro interno un amplificatore operazionale, a riprova dall'importanza che è stata precedentemente attribuita a questo componente per la determinazione delle non idealità.

### 1.3.1 DAC a resistenze pesate

Si riporta in fig.1.7 lo schema del DAC preso in esame. I valori scelti per le resistenze seguono, a partire dal valore  $R$ , l'andamento delle potenze del 2. In questo modo la corrente  $I_i = V_{REF}/R_i$  che scorre in un generico ramo sarà sempre in rapporto fisso rispetto a quelle dei rami adiacenti, il doppio di quella precedente ( $I_{i-1}$ ) e la metà di quella successiva ( $I_{i+1}$ ). Il comportamento dei deviatori è invece determinato dal valore di ciascun bit presente in ingresso: se il generico bit in posizione  $i$  è pari a 1, l'interruttore  $D_i$  collegherà l'ingresso invertente dell'amplificatore operazionale alla tensione  $V_{REF}$ , in caso contrario la connessione verrà effettuata col ramo a tensione nulla. Si nota subito ed intuitivamente che le correnti, grazie al particolare rapporto generato fra di loro, simulano il peso di ciascun bit nel codice completo, con la posizione del deviatore a stabilire l'effettiva presenza di tale contributo sulla tensione finale.

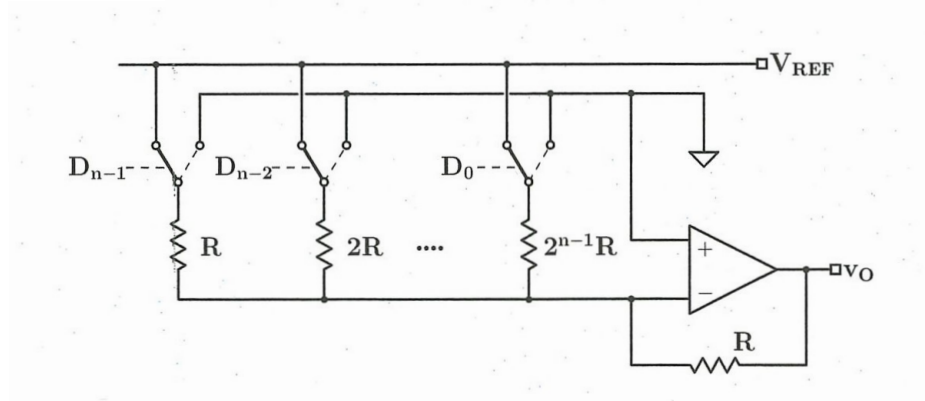


Figura 1.7: DAC a resistenze pesate [6]

Andando però ad analizzare nel dettaglio il circuito, si può dire che, per il fenomeno della sovrapposizione degli effetti, sul ramo inferiore del circuito verrà a formarsi una corrente

$$I = \sum_{i=0}^{n-1} D_i \cdot \left( \frac{V_{REF}}{R_i} \right) \quad (1.7)$$

con  $R_i = R \cdot 2^{n-1-i}$ . Questa, passando attraverso la resistenza (anch'essa di valore  $R$ ) posta in parallelo all'amplificatore operazionale, genererà una caduta di potenziale<sup>9</sup>

$$V_o = -(R \cdot I) = -V_{REF} \cdot \sum_{i=0}^{n-1} \frac{D_i}{2^{n-1-i}} \quad (1.8)$$

pari alla tensione d'uscita cercata.

Per quanto semplice, l'utilizzo di tale soluzione risulta pressoché impossibile. Questo per tre motivi principali:

- Per il funzionamento è necessaria la presenza resistenze i cui valori possono aumentare vertiginosamente all'aumentare del numero di bit. Per dare un'idea concreta, un DAC a 16 bit richiede un rapporto  $R_0/R_{n-1} = 32768$ , generando serie difficoltà di implementazione.
- Risulta particolarmente difficile ottenere in maniera precisa i rapporti necessari al corretto funzionamento del circuito. Piccole variazioni sui valori teorici delle resistenze possono comportare errori di non linearità nella transcaratteristica, come quelli visti al paragrafo precedente.
- Il valore della corrente erogata dal generatore di tensione che fornisce  $V_{REF}$  varia al variare delle combinazioni di bit in ingresso. Questo rende difficoltoso il mantenimento costante della  $V_{REF}$  stessa, generando oscillazioni sul guadagno dell'amplificatore operazionale.

<sup>9</sup>Si omette per semplicità il passaggio intermedio in cui si è sostituito a  $I$  il secondo membro della 1.7 e si è semplificato  $R$  a numeratore e denominatore.

### 1.3.2 DAC a traliccio R-2R

Questo convertitore si basa, a livello di funzionamento, su quanto detto per il circuito precedente, andando al contempo a risolvere molte delle problematiche che ne sfavorivano l'utilizzo pratico. Si può notare dalla fig.1.8 come il circuito rimanga

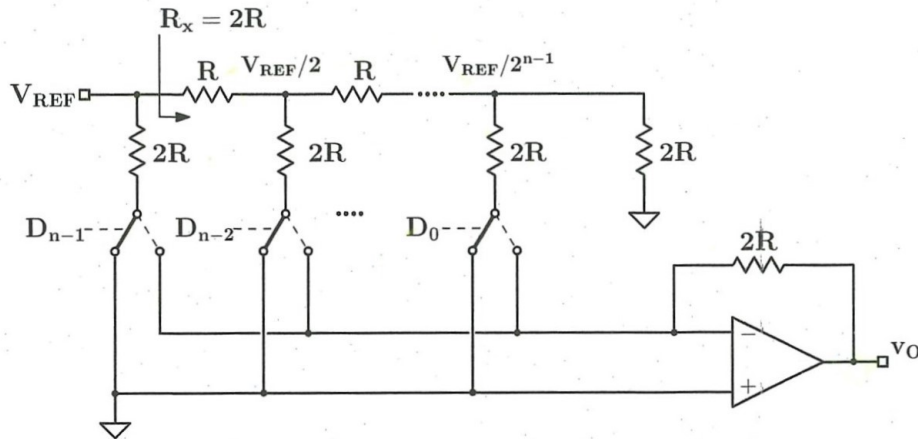


Figura 1.8: DAC a traliccio R-2R [6]

sostanzialmente invariato, con l'unica differenza riscontrabile nella sostituzione delle resistenze pesate con altre di valore  $R$  e  $2R$ . La particolare topologia del circuito fa sì che la resistenza equivalente a destra di ciascun deviatore sia sempre pari a  $2R$  e che si venga dunque ad avere sempre un parallelo tra due resistenze di ugual valore. La corrente  $I_i = V_i/R$  passerà per metà nel ramo del deviatore, verso l'amplificatore operazionale o la terra a seconda del valore di  $D_i$ , e per metà nella parte a destra del circuito, generando una caduta di tensione  $(I_i/2) \cdot R = V_i/2$ . Si nota quindi che ad ogni intersezione fra il ramo superiore e i deviatori, la tensione dimezza in valore, generando correnti che saranno sempre in rapporto pari a potenze del 2. In parole povere l'effetto finale sarà il medesimo di quello visto nel DAC a resistenze pesate, ma verrà ottenuto tramite il dimezzamento, in punti progressivi del circuito, della  $V_{REF}$  imposta. Logicamente si può scrivere di nuovo

$$V_o = -V_{REF} \cdot \sum_{i=0}^{n-1} \frac{D_i}{2^{n-1-i}} \quad (1.9)$$

con ormai ovvio significato dei simboli.

Analizziamo ora come i problemi che affliggevano il circuito precedente vengano risolti da questa piccola modifica:

- I valori delle resistenze presenti si limitano a  $R$  e  $2R$  per qualunque numero di bit, rimediando alla presenza di resistenze a ordini di grandezza differenti nel medesimo circuito.

- Il rapporto  $1/2$  tra le resistenze, necessario al funzionamento del circuito, è molto più semplice da ottenere rispetto alle potenze del 2 richieste dal DAC precedente, favorendo così la linearità della transcaratteristica.
- Essendo le resistenze  $2R$  posizionate "prima" del deviatore, il generatore di tensione vedrà sempre una resistenza equivalente di valore  $R$ , stabilizzando quindi la corrente generata e il guadagno dell'amplificatore operazionale.

L'unico costo di questa modifica consiste nel dover impiegare il doppio delle resistenze rispetto al circuito originale, trascurabile visto quanto di guadagnato.

Per quanto possa sembrare semplice nel funzionamento, questo circuito rimane alla base delle architetture di conversione audio più prestanti attualmente in commercio.

### 1.3.3 DAC a conversione seriale

Un DAC molto economico, dal funzionamento totalmente differente rispetto a quanto visto in precedenza, è riportato in fig.1.9.

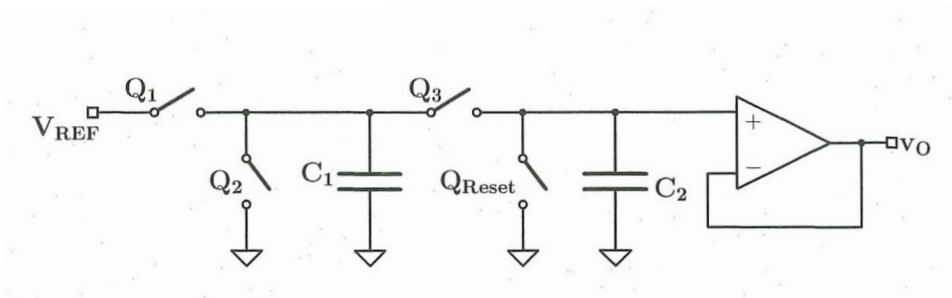


Figura 1.9: DAC seriale a capacità commutate [6]

In questo circuito, come è intuibile dal nome, i campioni non vengono convertiti nella loro interezza, bensì bit per bit, cominciando da quello meno significativo. Gli interruttori  $Q_1$  e  $Q_2$  sono controllati in base al valore del bit da convertire: un bit pari a 1 farà chiudere  $Q_1$ , determinando la carica del condensatore  $C_1$ , un bit pari a 0 farà chiudere invece  $Q_2$ , con la conseguente scarica del condensatore a terra. Ad ogni carica o scarica viene chiuso per un breve intervallo di tempo l'interruttore  $Q_3$ , mettendo in comunicazione  $C_1$  con la capacità  $C_2$ , detta *buffer*. In questo modo si ottiene ad ogni passo la redistribuzione di carica fra le due capacità, e una tensione finale ai capi di  $C_2$  proporzionale alla sequenza di bit di ingresso.

Data la scarsità di componenti, l'implementazione di questo circuito risulta relativamente semplice ed economica, al prezzo però di una velocità di conversione molto ridotta. Unico scoglio risiede nella necessità di utilizzare due capacità ( $C_1$  e  $C_2$ ) il più possibile di ugual valore, pena l'insorgenza di non linearità della transcaratteristica.



### 1.3.4 Utilizzo della DPWM

L'ultimo schema che viene preso in considerazione sfrutta la DPWM<sup>10</sup> quale controllo per la tensione in uscita (fig. 1.10). Un modulatore DPWM di base altro non è che un contatore integrato ad un comparatore: impostato a  $N$  il riferimento da convertire e  $2^n - 1$  il conteggio massimo del counter, esso fornirà un'uscita logica alta fintantoché  $N > cont$ , commutando a bassa al verificarsi del caso contrario. Al valore alto corrisponde l'erogazione della tensione di alimentazione  $V_{REF}$ , a quello basso l'uscita viene collegata a terra. Preso ora in considerazione un intervallo di tempo pari al periodo del contatore  $T = 2^n \cdot T_{clk}$  la tensione di uscita assume un valore medio

$$V_{out} = \frac{N}{2^n} V_{REF} \quad (1.10)$$

equivalente all'equazione caratteristica di un generico DAC.

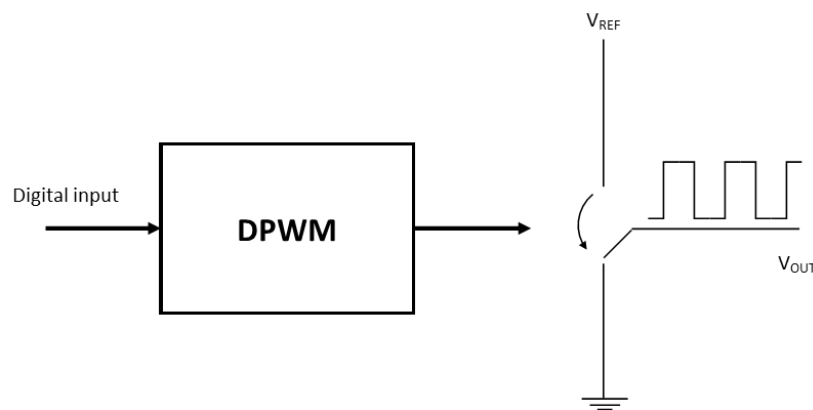


Figura 1.10: Schema DAC a DPWM.

Due fattori distinguono questa architettura da quanto visto precedentemente:

- La tensione di uscita acquisisce significato solo se integrata nel tempo. A tale scopo viene spesso posto in serie al circuito un filtro di ricostruzione RC.
- Il circuito si basa quasi totalmente su componenti digitali, la parte analogica è sostanzialmente costituita dal solo deviatore. Questo fattore ne permette l'implementazione economica su bassissimi fattori di scala.

## 1.4 Innovazioni nella conversione

Si riportano in questo ultimo paragrafo alcune possibili innovazioni nell'ambito della conversione D/A attualmente discusse dalla comunità scientifica, a riprova di come i concetti e le architetture viste finora stiano alla base di soluzioni tutt'ora adottate.

<sup>10</sup>Digital Pulse Width Modulation

### 1.4.1 Il Re-DAC

Il *Relaxation-DAC* si basa su una rielaborazione del convertitore a capacità commutate visto poco sopra (fig. 1.9). Lo schema è riportato in fig 1.11.

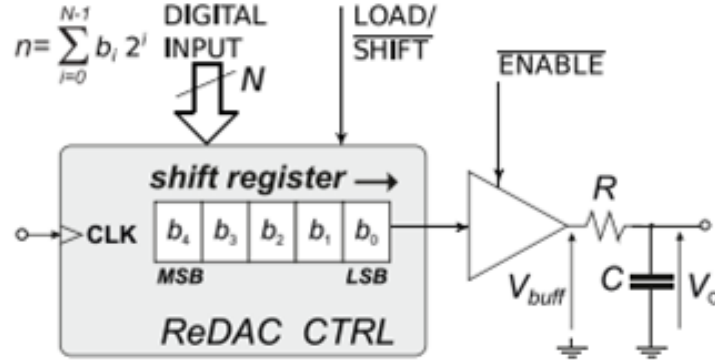


Figura 1.11: Schema di un Relaxation DAC [7]

Come si può vedere, il campione viene convertito bit per bit a partire da quello meno significativo. A seconda del valore del bit in ingresso, sulla capacità verrà applicata per un tempo  $T$  la tensione  $V_{REF}$ , corrispondente al valore alto, o  $V_0 = 0$ , corrispondente al valore basso. Il transitorio della tensione  $V_C(t)$  all'interno dell' $i$ -esimo intervallo di conversione è quindi esprimibile come

$$V_{C,i}(t) = D_i V_{REF} (1 - e^{-\frac{t-iT}{\tau}}) + V_C(iT) e^{-\frac{t-iT}{\tau}} \quad (1.11)$$

Dopo  $n$  periodi di conversione, con  $n$  pari al numero di bit del campione, otterremo

$$V_C(nT) = V_{REF} (1 - e^{-\frac{T}{\tau}}) \cdot \sum_{n=0}^{n-1} b_i e^{-\frac{(n-1-i)T}{\tau}} \quad (1.12)$$

che, ponendo  $\tau = \frac{T}{\log(2)}$ , diviene [7]

$$V_C(nT) = \frac{V_{REF}}{2^n} \cdot \sum_{n=0}^{n-1} D_i 2^i \quad (1.13)$$

del tutto equivalente alla (1.2) espressa per un generico DAC.

È immediato comprendere che tale circuito funziona solo se  $T$  e  $\tau$  rispettano la relazione da noi supposta, fattore abbastanza semplice da ottenere considerando che  $T$  corrisponde molto spesso al  $T_{clk}$  del registro d'ingresso, facilmente calibrabile in base al circuito RC con cui si sta operando. Di converso notiamo che a differenza delle architetture a capacità commutate il Re-DAC necessita la presenza di un solo condensatore, eliminando la difficoltà di implementare due capacità a valori esattamente uguali, pena l'insorgenza di non linearità nella transcaratteristica, fattore che ne favorisce l'implementazione anche a bassissimi valori d'area occupata.

### 1.4.2 Utilizzo della DDPM

Abbiamo analizzato poco sopra l'implementazione di convertitori D/A basati sulla modulazione DPWM. Caratteristica di questa modulazione è l'insorgenza di componenti spurie a bassa frequenza, che impone limiti stringenti al filtro analogico di ricostruzione.

Un'ipotesi recentemente discussa per risolvere tale problema risiede nell'utilizzo della DDPM, acronimo per *Diadyc Digital Pulse Modulation*. Tale modulazione presenta un funzionamento concettualmente complesso ma facilmente implementabile via hardware: ad ogni campione di valore  $N$  composto da  $n$  bit viene associato uno stream di bit

$$\Sigma_N = \sum_{i=0}^{n-1} D_i \cdot S_i(t) \quad (1.14)$$

ove  $D_i$  è il valore del bit alla posizione  $i$  e  $S_i$  è l'*ODBS*<sup>11</sup> associato a tale posizione. La rappresentazione di tali segnali è riportata in fig.1.12. Concettualmente, partendo

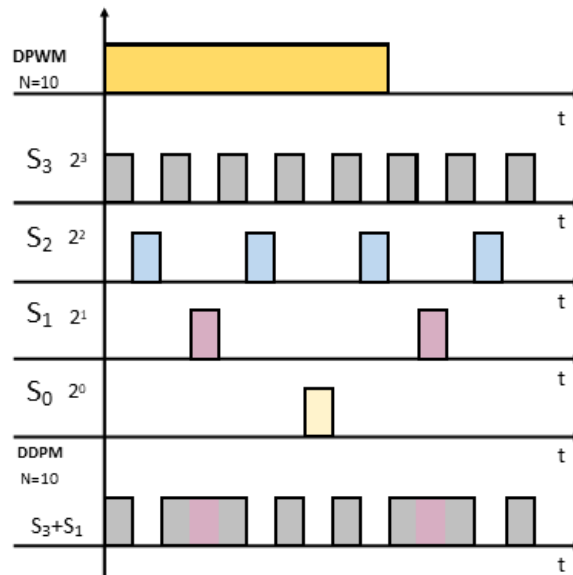


Figura 1.12: ODBS con  $n=4$  e  $N=10$ . [8]

dall'MSB, si ha che  $S_{n-1}$  sarà "alto" a cicli di clock alterni,  $S_{n-2}$  sarà alto per un periodo di clock ogni altra volta in cui  $S_{n-1}$  andrà basso,  $S_{n-3}$  sarà alto ogni altra volta in cui  $S_{n-1}$  e  $S_{n-2}$  andranno bassi e così via. È facile notare come, preso in considerazione un periodo  $T = 2^n \cdot T_{clk}$ ,  $S_i$  rimanga alto per una frazione di periodo pari al peso assunto dalla posizione del bit che rappresenta. Lo stream  $\Sigma_N$  sarà di conseguenza un segnale periodico con  $T = 2^n T_{clk}$  in cui figureranno le componenti date dai bit "alti" del campione. Grazie al fatto che i segnali  $S_i$ , per definizione, non si sovrappongono mai, si trova facilmente che la tensione media in un periodo

<sup>11</sup> *Orthogonal Dyadic Basis Signal*

vale [8]

$$V_{out} = \frac{N}{2^n} V_{REF} \quad (1.15)$$

del tutto equivalente a quanto ottenuto nella (1.10). Il vantaggio sta nel fatto che essendo il contributo dato dall'MSB di frequenza pari a  $2^{n-1}$  volte la frequenza del segnale, le sue armoniche saranno ben lontane dalla fondamentale, facilmente filtrabili dal filtro analogico.

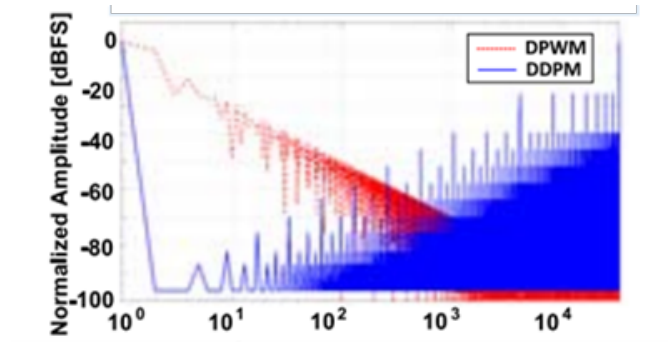


Figura 1.13: Contenuto spettrale di un segnale convertito tramite DDPM. [8]

## Capitolo 2

# La conversione D/A e la riproduzione audio

In questo capitolo andremo a vedere la struttura generale di un DAC per la riproduzione audio e come le architetture bitsream e multibit si basino su di essa, evidenziandone comunque le differenze fondamentali. Andremo a vedere poi alcune delle problematiche particolarmente sentite in questo ambito, come il *jitter* o il *passaggio per lo zero*, andando ad analizzare alcune possibili soluzioni.

### 2.1 Caratteristiche dei segnali audio

Prima di analizzare l'architettura di un DAC audio e il suo funzionamento, è utile soffermarsi sulle caratteristiche dei segnali che stiamo trattando. Tralasciando le formattazioni meno utilizzate, in questo testo si prenderà in considerazione segnali digitali in formato CD, caratterizzati da una codifica in 16 bit e una frequenza di campionamento di  $44kHz$ . Essendo la banda dell'udibile contenuta nell'intervallo  $20Hz - 20kHz$ , essa rispetta i limiti teorici imposti dal teorema di Nyquist, risultando però insufficiente in applicazioni pratiche. Vedremo in seguito come l'applicazione del sovracampionamento possa risolvere tale problema. Essendo poi la codifica a 16 bit, risulta teoricamente inutile impiegare DAC con un numero superiore di bit. Esistono però formati, ad esempio il cosiddetto "Hi-ResAudio", caratterizzati da una codifica a 24bit e frequenze di campionamento pari a  $192kHz$ , giustificando l'esistenza, come vedremo in seguito, di DAC molto prestanti ad elevata velocità di elaborazione, ben al di sopra di quanto richiesto dal formato CD.

### 2.2 Schema generale per la conversione audio

Come accennato poco sopra, la maggior parte dei DAC di nostro interesse può essere ricondotta ad uno schema fondamentale, qui sotto riportato (fig.2.1).

I blocchi fondamentali sono quindi tre:

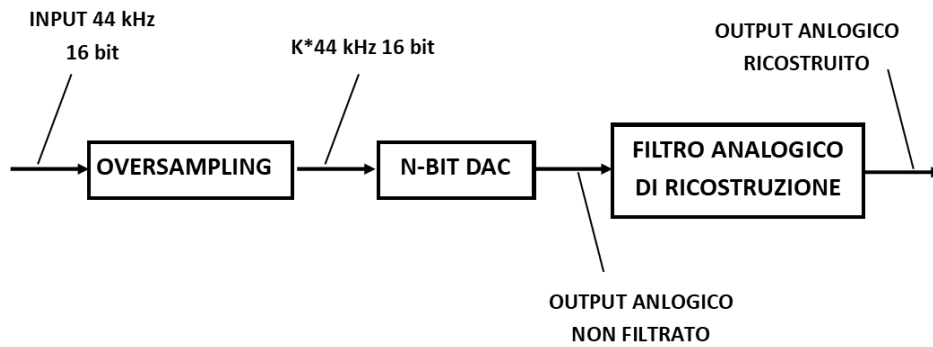


Figura 2.1: Schema generale per un DAC audio.

- Un sovracampionatore
- Il DAC vero e proprio
- Un filtro analogico per la ricostruzione finale del segnale

Questa base può essere poi integrata da altri accorgimenti, alcuni dei quali analizzeremo in seguito, per un generale miglioramento prestazionale.

### 2.2.1 Il sovracampionatore

Il sovracampionatore ha due funzioni fondamentali:

- agevolare le operazioni di ricostruzione attuate dal filtro analogico posto in uscita al DAC.
- aumentare la risoluzione del sistema incrementando l' $SNR$ .<sup>1</sup>

Per capire però come queste funzioni sono doverose fare una piccola digressione sul campionamento e il teorema di Nyquist.

La teoria base dei segnali ci insegna che un campionamento nel dominio del tempo a frequenza  $f_s$  genera una replicazione di egual passo  $f_s$  nel dominio delle frequenze. La cosa risulta più chiara analizzando la fig.2.2: sopra è riportato lo spettro del segnale continuo, contenuto nella banda di frequenze  $0 - w$ ,<sup>2</sup> sotto lo spettro dello stesso segnale campionato a frequenza  $f_s = 2w$ . La versione campionata è formata da infinite copie dello spettro originale, dette *alias*, centrate in  $kf_s$  con  $k \in Z$ . Si può allora comprendere facilmente quanto esposto dal teorema di Nyquist:

*È possibile ricostruire un segnale a partire dalla sua versione campionata solamente se la frequenza di campionamento  $f_s$  è pari almeno al doppio della banda del segnale campionato.*

Per ricostruire un segnale a partire dalla sua versione campionata, bisogna applicare

<sup>1</sup>*Signal Noise Ratio.*

<sup>2</sup>Si ricorda che ogni spettro è simmetrico rispetto all'asse delle ampiezze.

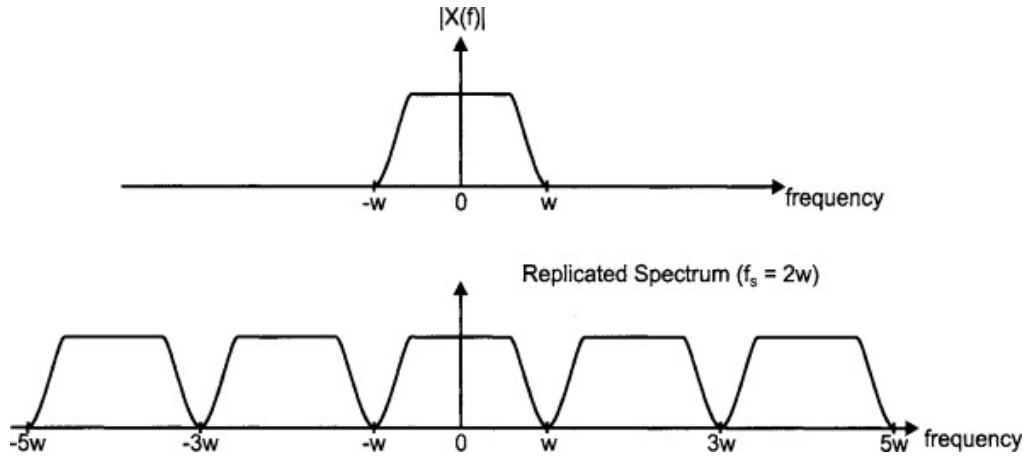


Figura 2.2: Effetto del campionamento sullo spettro delle frequenze. [9]

un filtro passa-basso, supposto ideale, con banda passante  $w$ , in modo da eliminare tutte le componenti armoniche dovute alle repliche spettrali. Questo è però possibile solamente se la banda di interesse è ben distinta dalle sue copie, ovvero solo per  $2w < f_s$ . In caso contrario lo spettro del segnale e il primo alias si sovrapporrebbero nel tratto di frequenze tra  $f_s - w$  e  $w$ , modificando irreparabilmente il contenuto in banda originario.

A livello totalmente teorico, il sistema di nostro interesse soddisfa pienamente le condizioni imposte dal teorema, avendo un segnale campionato a  $f_s = 44.1kHz$  il cui contenuto è limitato alla banda delle frequenze sonore, quindi con  $w \simeq 20kHz$ . L'enunciato è però valido solamente prendendo in considerazione un filtro ideale, identificabile con un finestra rettangolare a guadagno unitario di larghezza pari alla cosiddetta *banda passante*. Un filtro reale ha un andamento molto più graduale, attenuando in maniera sempre più marcata le frequenze superiori alla banda passante senza eliminarle bruscamente. (fig.2.3)

È a questo punto che entra in gioco il sovracampionatore: aggiungendo  $k - 1$  campioni nulli fra due campioni consecutivi e attuando un'interpolazione tramite un apposito filtro FIR<sup>3</sup>, esso porta a una nuova

$$f'_s = k \cdot f_s \quad (2.1)$$

in modo da separare nettamente gli alias dalla banda di interesse, permettendo anche ad un filtro reale di "recuperare" il segnale in maniera adeguata. Logicamente, più è alto  $k$ , più saranno distanziati gli alias, meno saranno elevate le prestazioni richieste al filtro per funzionare correttamente.

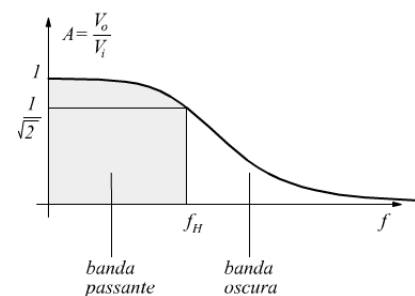


Figura 2.3: LPF reale. [10]

<sup>3</sup>Finite Impulse Response

Il filtro di interpolazione, come già accennato, è costituito da un filtro FIR. L'equazione all'uscita per  $N$  coefficienti è la seguente:

$$y(n) = h_0x(n) + h_1x(n-1) + h_2x(n-2) + \dots + h_{N-1}x_{n-N+1} \quad (2.2)$$

Il numero di coefficienti  $N$  è di solito scelto dispari in modo da sfruttare la simmetria nella risposta impulsiva del filtro, ovvero il fatto che  $h_n = h_{N-1-n}$ , per diminuire il numero di operazioni necessarie. Prendendo ad esempio un filtro con  $N = 5$ , la (2.2) diviene

$$y(n) = h_0x(n) + h_1x(n-1) + h_2x(n-2) + h_3x(n-3) + h_4x(n-4) \quad (2.3)$$

ma essendo

$$h_0 = h_4 \quad h_1 = h_3 \quad (2.4)$$

si può scrivere

$$y(n) = h_0[x(n) + x(n-4)] + h_1[x(n-1) + x(n-3)] + h_2x(n-2) \quad (2.5)$$

richiedendo due moltiplicazioni in meno rispetto alla (2.3). I coefficienti vengono solitamente calcolati tramite l'algoritmo iterativo di Parks McClellan, anche detto "metodo ottimale". Vengono presi in considerazione i cinque parametri che caratterizzano il filtro, ovvero:

- $N$  = numero di coefficienti utilizzati
- $f_p$  = banda passante del filtro
- $f_s$  = banda soppressa del filtro
- $\delta_1$  = ripple in banda passante<sup>4</sup>
- $\delta_2$  = ripple in banda soppressa

Di questi il progettista fissa il valore dei primi tre e del rapporto  $\delta_1/\delta_2$ . I coefficienti ottimali saranno quelli che minimizzano il valore massimo assoluto di

$$E(f) = W(f) \cdot [D(f) - H(f)] \quad (2.6)$$

ove  $E(f)$  è detta "funzione errore",  $D(f)$  è la risposta in frequenza desiderata e  $H(f)$  quella effettivamente ottenuta nell'attuale iterazione.  $W(f)$  è detto "peso dell'errore", pari a  $\delta_1/\delta_2$  all'interno della banda passante, 1 nella banda soppressa. [13]

Le operazioni descritte sopra hanno l'effetto non secondario di aumentare la risoluzione del segnale in fase di conversione. Per comprendere ciò dobbiamo però

---

<sup>4</sup>Indica la massima differenza fra la risposta in frequenza attesa e quella reale. Il parametro verrà analizzato meglio al capitolo 3.



scomodare l'SNR. Il *Signal Noise Ratio* viene definito come il rapporto fra la potenza  $P_x$  di un segnale sinusoidale di ampiezza  $FSR$ , e la potenza  $P_Q$  del rumore generato dall'errore di quantizzazione. Esso viene spesso utilizzato come indice per monitorare le variazioni nella risoluzione. Il rumore che prende in considerazione è infatti detto "di quantizzazione", ovvero di ampiezza pari al massimo errore di quantizzazione presente nel segnale ( $Q/2$ ). Essendo tale errore per definizione non legato alla grandezza di ingresso, è possibile equiparare un segnale quantizzato con passo  $Q$  alla sua controparte analogica sommata al rumore di quantizzazione. Diminuire l'ampiezza di tale rumore significherebbe diminuire  $Q$ , aumentando di conseguenza la risoluzione.

Essendo

$$P_x = \frac{V_x^2}{R} = \frac{FSR^2}{8R} \quad (2.7)$$

con  $V_x = \frac{FSR}{2\sqrt{2}}$  valore efficace della tensione e  $R$  resistenza equivalente di ingresso, e

$$P_Q = \frac{Q^2}{12R} \quad (2.8)$$

si ottiene

$$SNR = \frac{P_x}{P_Q} = \frac{12 FSR^2}{8 Q^2} = \frac{12}{8} 2^{2n} \quad (2.9)$$

che in decibel diventa

$$SNR_{dB} = 10 \log_{10} SNR = 6.02 n + 1.76 [dB] \quad (2.10)$$

Per una conferma intuitiva di quanto esposto, si noti che ad ogni bit aggiunto alla codifica l'SNR aumenta di 6 dB, corrispondente ad un fattore 2. Infatti, aggiungendo un bit l'LSB dimezza di valore, raddoppiando di conseguenza la risoluzione. Si riporta in fig.2.4 una rappresentazione grafica della DFT di un segnale campionato a frequenza  $f_s$ .<sup>5</sup> Senza soffermarsi sull'ovvia distinzione fra la componente data dal segnale e il rumore di fondo che la circonda, concentriamoci su altri due aspetti:

- Sono presenti  $N$  campioni, ognuno di larghezza  $F_Q = \frac{f_s/2}{N}$ , detto *quanto frequenziale*. Preso quindi il campione dato dal segnale in ingresso possiamo definire

$$p_x = \frac{P_x}{F_Q} \quad (2.11)$$

come la densità di potenza di tale segnale.

- Il rumore interessa solamente la banda  $0 - f_s/2$ . Anche in questo caso definiamo

$$p_Q = \frac{P_Q}{f_s/2} \quad (2.12)$$

come la densità di potenza del rumore di quantizzazione.

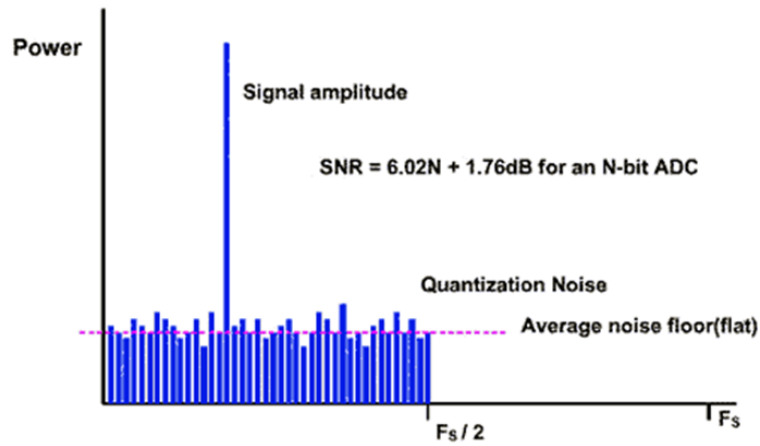


Figura 2.4: DFT di un segnale campionato a frequenza  $f_s$ . [11]

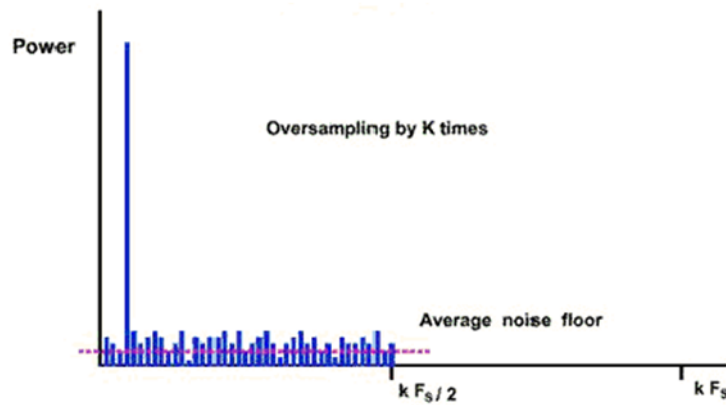


Figura 2.5: DFT di un segnale campionato a  $f'_s = k * f_s$ . [11]

Andiamo invece ora ad analizzare la DFT di un segnale campionato a frequenza  $f'_s = k \cdot f_s$ , riportata in fig.2.5. Si possono individuare alcune sottili differenze rispetto al grafico precedente:

- Il rumore interessa ora la banda  $0 - k f_s/2$ , portando a una nuova densità di potenza

$$p'_Q = \frac{P_Q}{k f_s/2} = \frac{p_Q}{k} \quad (2.13)$$

- Essendo il numero dei campioni in banda sempre  $N$ , il quanto frequenziale passerà da  $F_Q$  a

$$F'_Q = \frac{k f_s/2}{N} = k F_Q \quad (2.14)$$

---

<sup>5</sup>Utilizziamo la DFT per comodità nel maneggiare le densità di potenza, ma quanto detto può essere tranquillamente riportato alla trasformata di Fourier continua.

e di conseguenza la densità di potenza diverrà

$$p'_x = \frac{P_x}{F'_Q} = \frac{p_x}{k} \quad (2.15)$$

Le potenze  $P_x$  e  $P_Q$  rimangono però invariate, come di conseguenza l'SNR. Fin qua, a livello di risoluzione, non abbiamo ancora guadagnato nulla.

Andiamo però ora ad applicare un filtro digitale passa basso nella nostra banda di interesse, ovvero fino a  $f_s/2$ , come riportato in fig.2.6. Rimanendo invariata densità

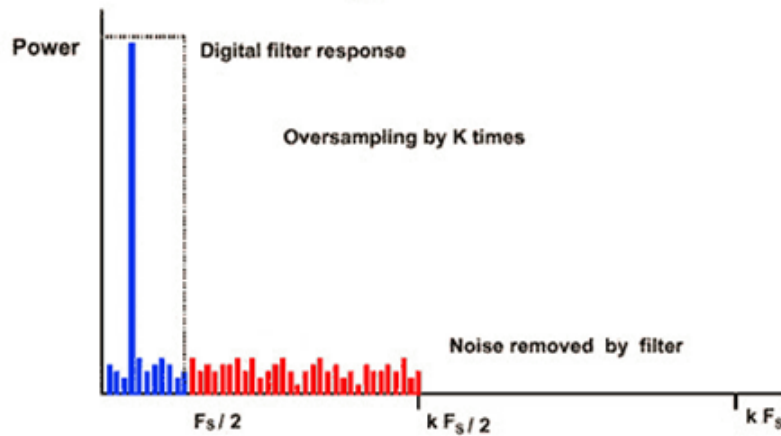


Figura 2.6: Applicazione di un LPF digitale alla banda 0 -  $f_s/2$ . [11]

$p'_Q$ , avremo che la potenza del rumore diverrà

$$P'_Q = p'_q \cdot f_s/2 = \frac{P_Q}{k} \quad (2.16)$$

e di conseguenza

$$SNR' = \frac{P_x}{P'_Q} = k \cdot SNR \quad (2.17)$$

Attraverso un banale filtraggio eseguito dopo il sovracampionamento abbiamo quindi aumentato l'SNR di un fattore  $k$ . Traducendo in decibel abbiamo [12]

$$SNR' = 6.02n + 1.76 + 10 \log_{10} k = 6.02n + 1.76 + 6.02 \log_4 k [dB] \quad (2.18)$$

Ad ogni aumento di  $k$  pari a un fattore 4 otteniamo un aumento della risoluzione di  $6dB$ , pari a quanto ottenuto tramite l'aggiunta di un bit. In astratto, potremmo pensare di diminuire il numero di bit utilizzati e compensare totalmente aumentando il fattore di sovracampionamento  $k$ .

Vedremo in seguito che i convertitori  $\Sigma - \Delta$  sfruttano questa base teorica per attuare la conversione tramite un DAC a singolo bit.

### 2.2.2 Il DAC

Sul circuito di conversione vero e proprio c'è ormai poco da aggiungere rispetto a quanto già detto nel primo capitolo. La quasi totalità dei DAC utilizzati in questo ambito, eccezion fatta per i bitstream, presenta una configurazione  $R - 2R$ , l'unica capace di sostenere le alte frequenze di conversione richieste. Il numero di bit impiegati varia di progetto in progetto: solitamente le architetture più costose tendono a puntare molto su un'elevata *bitness*<sup>6</sup>, ma non sono rari i casi in cui un DAC a pochi bit venga compensato da un forte sovracampionamento e un'adeguata interpolazione del segnale.

Vale la pena discutere i DAC a  $n > 16$  bit, abbastanza comuni nei dispositivi top di gamma. Tralasciando le recenti registrazioni musicali ad alta definizione, caratterizzate da un elevato numero di bit per campione ( $n = 24$ ), e prendendo quindi in considerazione il solo CD quale sorgente del nostro segnale, viene immediato chiedersi quale significato possa avere utilizzare convertitori a 18-20 bit per gestire una sorgente codificata in 16 bit. Le ragioni possono essere molteplici, e sono tutt'ora oggetto di discussione fra i tecnici e gli appassionati. Se ne riportano di seguito un paio:

- Un primo aspetto è legato banalmente ad una questione di marketing. Si ha infatti che la risoluzione *effettiva* del convertitore, data dall'SNR reale del circuito<sup>7</sup>, risulta ben diversa dalla banale risoluzione teorica del DAC. Utilizzare un convertitore a *bitness*  $> 16$  facilita il raggiungimento dei fatidici 98.08 dB di SNR, corrispondenti ad una risoluzione reale di 16 bit, ma difficilmente ne implica l'ottenimento effettivo. Se si utilizzasse come parametro l'SNR reale, e non il banale numero di bit usati nella conversione, il problema non sussisterebbe.
- Esistono però circuiti di conversione la cui risoluzione effettiva raggiunge i 17-18 bit, non rientrando quindi nel caso precedente. Per dare una ragione d'essere a tali architetture bisogna introdurre il cosiddetto *SBM*<sup>8</sup>, una tecnica ideata da Sony che consente di ottenere una risoluzione apparentemente maggiore dei 16 bit a cui è vincolato il supporto CD. La registrazione avviene inizialmente a 24 bit, ridotti successivamente a 16 in fase di elaborazione. La riduzione non avviene tramite un banale troncamento, ma i bit eccedenti di ogni campione vengono utilizzati per modellare il campione successivo, sulla falsariga di quanto avveniva già per alcune forme di *noise shaping*, di cui parleremo in seguito. L'effetto di tutto ciò è uno spostamento del rumore di quantizzazione verso le bande più alte dello spettro, a cui il nostro orecchio è meno sensibile, ottenendo, ad esempio, l'equivalente di 20 bit sulle frequenze nell'intorno dei 4-5 kHz (banda dei medi) e 14 bit sopra i 15 kHz (banda degli alti). La risoluzione "media" rimane identificata da 16 bit, i quali però vengono "concentrati" ove

---

<sup>6</sup>Quantità di bit con cui viene effettuata la conversione.

<sup>7</sup>In cui il rumore è dato sia dalla quantizzazione che dal contributo generato dalle componenti analogiche del sistema.

<sup>8</sup>Acronimo per *Super Bit Mapping*.

più sono utili. Per sfruttare questo fenomeno si dovrà però utilizzare un DAC "all'altezza", giustificando così l'esistenza di convertitori a 17-18 bit effettivi.

Altro aspetto su cui vale la pena soffermarsi riguarda la possibilità di combinare più DAC "in parallelo" allo scopo di migliorare la qualità complessiva della conversione. Un esempio di tale pratica giace nella cosiddetta configurazione in "push-pull", che nel capitolo 3 vedremo essere molto utilizzata in dispositivi hi-end.

### Configurazione push-pull

In tale configurazione vengono utilizzati due convertitori perfettamente identici posti in parallelo secondo lo schema a blocchi riportato in fig.2.7. Nel ramo superiore

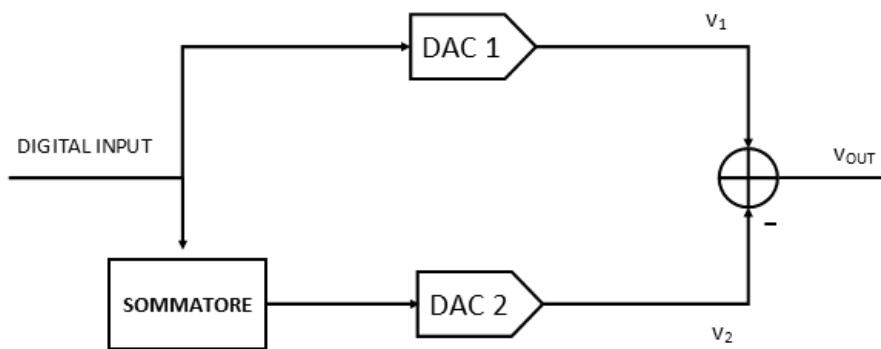


Figura 2.7: Configurazione DAC in push-pull.

il segnale viene convertito secondo quanto è stato già visto in precedenza. Nel ramo inferiore invece il segnale digitale viene per prima cosa elaborato tramite un apposito sommatore che sottrae a  $2^n - 1$  il campione di ingresso. In questo modo viene invertita la transcaratteristica del secondo DAC, che ad un ingresso  $N = 0$  risponderà con la tensione massima, ad un ingresso  $N = 2^n - 1$  risponderà con tensione pari a 0. La tensione  $V_2$  viene quindi sottratta alla tensione  $V_1$ . Come è possibile vedere dalla tabella sottostante, il range della tensione di uscita raddoppia. Ciò che non raddoppia è però il rumore di quantizzazione: quest'ultimo, essendo

	DAC 1	DAC2	DAC 1-DAC 2
$N = 2^n - 1$	10V	0V	10V
$N = 2^{n-1}$	5V	5V	0V
$N = 0$	0V	10V	-10V

Tabella 2.1: Associazione fra ingresso N e tensione di uscita preso FSR=10V.

definito da una distribuzione statistica, non sarà dato dalla somma fra le tensioni massime dei rumori iniziali (pari a  $Q/2$  per ciascuna uscita), bensì dalla somma dei loro valori quadratici medi. Si avrà quindi [24]

$$V_{N1+N2} = \sqrt{V_{N1}^2 + V_{N2}^2} = \sqrt{2 \cdot \frac{Q^2}{2}} = \frac{Q}{\sqrt{2}} \quad (2.19)$$

con  $V_N$  tensione massima del rumore generato dai due DAC. In astratto l'ampiezza dell'errore di quantizzazione finale è aumentata, passando da  $Q/2$  a  $Q/\sqrt{2}$ , ma essendo raddoppiato anche il range delle tensioni in uscita si ha, in proporzione, una riduzione di un fattore  $\sqrt{2}$ . Logicamente questo risultato è riscontrabile anche sull'SNR: ponendo nella (2.9)  $Q' = Q/\sqrt{2}$  e  $FSR' = 2 \cdot FSR$  si ottiene

$$SNR = \frac{12}{8} \frac{(2FSR)^2}{(Q/\sqrt{2})^2} = 2 \cdot \frac{12}{8} 2^{2n} = 3 \cdot 2^{2n} \quad (2.20)$$

che in dB diviene

$$SNR_{dB} = 6.02n + 1.76 + 3.01 \text{ [dB]} \quad (2.21)$$

guadagnando sostanzialmente mezzo bit sulla risoluzione finale. Come è facile comprendere, non sempre l'ottenimento di mezzo bit in risoluzione giustifica il costo di un secondo convertitore, relegando questo tipo di soluzioni ad una fascia molto ristretta (ed elitaria) di dispositivi.

## 2.3 Le architetture bitstream

Come già brevemente accennato, le architetture bitstream hanno fin da subito dominato la fascia medio-bassa del mercato grazie alla relativa semplicità costruttiva, che ne consentiva un prezzo finale non troppo elevato, unita a prestazioni abbastanza elevate. Ciò è dovuto ad un forte utilizzo del sovracampionamento quale mezzo principale per raggiungere un'elevata risoluzione, permettendo l'utilizzo di DAC a 1 bit<sup>9</sup> pur conservando nonostante tutto buone prestazioni.

Andando però con ordine, in fig. 2.8 è riportata la struttura di un classico convertitore bitstream. Notiamo immediatamente che non presenta nulla di parti-

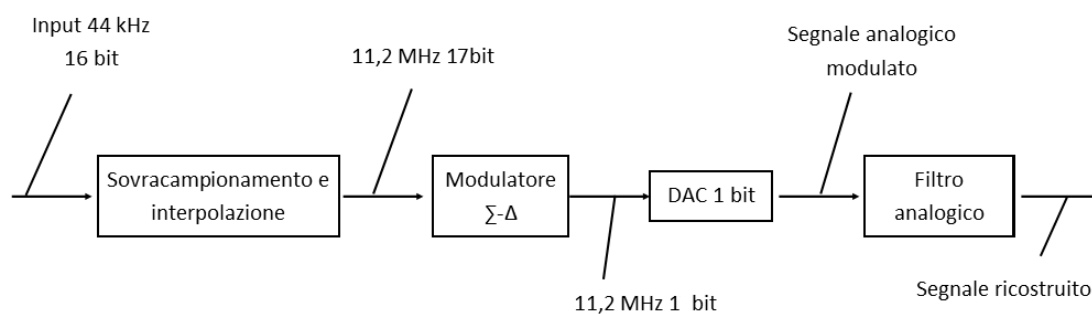


Figura 2.8: Schema generale di un convertitore bitstream.

colarmente nuovo rispetto a quanto già visto: un sovracampionamento con annessa interpolazione per aumentare l'SNR, la conversione tramite DAC e infine la ricostruzione del segnale analogico. La peculiarità dello schema sta però nei dettagli.

<sup>9</sup>A tutti gli effetti un comparatore.

Per prima cosa notiamo che, nel caso riportato, il segnale sovracampionato passa dai 44 kHz di base a 11.2 MHz, corrispondenti ad un fattore  $k \simeq 256$ . Nel caso di conversione a singolo bit, questo forte sovracampionamento, unito alla necessaria interpolazione, porterebbe ad un

$$SNR = 6.02 + 1.76 + 6.02 \log_4 256 \simeq 32 \text{ dB} \quad (2.22)$$

corrispondenti circa a 5 bit, risoluzione abbastanza scarsa ma non di meno impressionante se pensiamo al singolo bit da cui siamo partiti.

Il grosso del lavoro del lavoro è svolto però dal blocco successivo, quello che viene impropriamente chiamato *modulatore*  $\Sigma - \Delta$ . Nonostante sia un circuito totalmente digitale, costituito di solito da un filtro a retroazione IIR del secondo ordine, esso viene chiamato così per la funzione che svolge, analoga all'effettivo modulatore  $\Sigma - \Delta$  presente nelle architetture A/D. Come la sua controparte analogica, infatti, detiene il ruolo fondamentale di convertire il segnale in entrata a 17 bit in una serie di campioni a bit singolo, attuando per l'appunto una modulazione. Se il suo operato si fermasse qui, la sua presenza sarebbe totalmente superflua, potremmo dare il segnale a 17 bit direttamente in ingresso al DAC e l'effetto sarebbe il medesimo. Il circuito è però progettato per attuare l'elaborazione detta di *noise shaping*: un po' come avviene per i CD registrati in SBM, i 16 bit meno significativi non vengono scartati, bensì sommati, con eventuali fattori moltiplicativi, ai due campioni successivi. Uno schema d'esempio è riportato in fig. 2.9, ove si riporta un modulatore utilizzato nelle macchine Philips: I 16 bit "in eccesso" del campione in uscita vengono sommati ai due campioni successivi dopo essere stati moltiplicati per i rispettivi coefficienti ( $k_{n+1} = 2$  e  $k_{n+2} = -1$ ). Lo scopo di questa elaborazione è quello di spostare

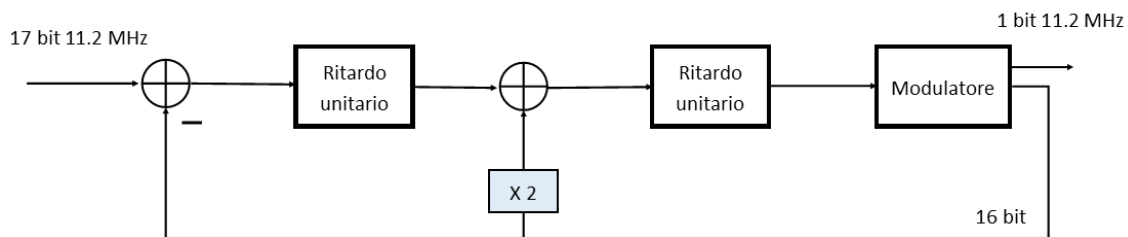


Figura 2.9: Schema di modulazione  $\Sigma - \Delta$  utilizzato nel convertitore SAA7320 di Philips. [13]

il rumore di quantizzazione il più possibile alle alte frequenze, fuori dalla banda  $0 - f_s/2$ , in modo che nella banda passante del filtro non ne rimanga quasi traccia. L'effetto sullo spettro è visibile in fig. 2.10, da confrontare con la fig. 2.5: nella banda  $f_s/2 - kf_s/2$  il rumore diviene quasi la metà del segnale stesso, mentre in  $0 - f_s/2$  esso risulta quasi impercettibile, raggiungendo l'equivalente di 15-16 bit di risoluzione.

Tirando le fila del discorso, possiamo vedere come nonostante tutti i processi di cui abbiamo discusso in questo paragrafo siano di natura puramente digitale e la

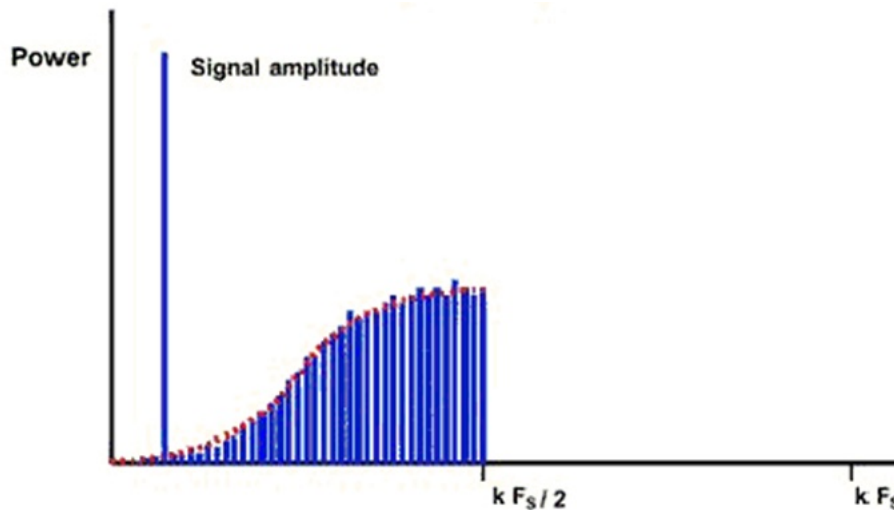


Figura 2.10: Effetto del noise shaping sul rumore di quantizzazione. [11]

componente analogica si trovi a livelli minimi, le prestazioni ottenute si attestino comunque su ottimi livelli (15-16 bit con un buon algoritmo di noise shaping). Non risulta quindi difficile comprendere come tali architetture siano divenute predominanti in tutte le applicazioni caratterizzate da una forte integrazione dei componenti.

## 2.4 Fattori di downgrade alla risoluzione

Come già detto parlando dei DAC a più di 16 bit, la risoluzione effettiva di un circuito di conversione può essere ben diversa dal valore teorico dato dal semplice numero di bit impiegati. Fattori come il *jitter*, i *ripple* sulla banda passante dei filtri digitali e la poca linearità ai bassi livelli contribuiscono in maniera più o meno marcata al peggioramento prestazionale del circuito, andando a plasmare quel "SNR effettivo" che spesso viene ricercato sul banco di misura. In chiusura a questo capitolo andremo ad analizzare alcuni di questi fattori, presentando possibili soluzioni sviluppate nel tempo.

### 2.4.1 Il jitter

Il problema del jitter nasce dal non perfetto sincronismo riscontrabile fra il circuito di elaborazione presente nel lettore e quello presente nel convertitore. Esso si fa sentire soprattutto nelle architetture di alto livello, ove, per disaccoppiare la componente elettronica di lettura da quella di conversione, si tende molto spesso a gestire le due operazioni tramite dispositivi fisicamente separati, posti anche a distanza di metri l'uno dall'altro.<sup>10</sup> In questi casi il clock generato dal lettore deve essere "recuperato"

<sup>10</sup>Si ricorda come "fun fact" il caso del Wadia 2000, in cui il convertitore era collegato al lettore tramite una fibra ottica lunga 15 metri, in modo da non generare interferenza fra i due dispositivi.



dal convertitore tramite appositi circuiti detti  $PLL^{11}$ : essendo i dati trasmessi per via seriale in "blocchi" di 32bit (16 di intestazione e informazioni aggiuntive e 16 di dato vero e proprio), il circuito ha il compito, leggendo l'intestazione, di rilevare il clock a cui si sta trasmettendo e rapidamente comunicarlo ai blocchi di elaborazione e sovracampionamento (si veda la fig. 2.11 come esempio). Questo "recupero"

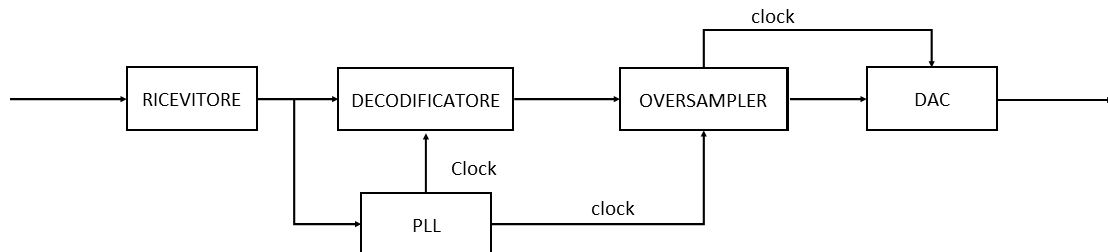


Figura 2.11: Esempio di circuito dotato di PLL.

risente però di una certa incertezza di fondo, dovuta alla qualità del mezzo fisico di trasmissione e alle non idealità del PLL stesso, portando ad un'instabilità temporale nell'uscita e generando componenti spurie molto spesso non correlate al segnale di partenza, particolarmente fastidiose all'udito. Se poi, come spesso accade, il jitter assume un andamento totalmente aleatorio, si verifica un generale aumento del rumore, con conseguente diminuzione della risoluzione. Una rappresentazione di tale fenomeno è riportata in fig.2.12.

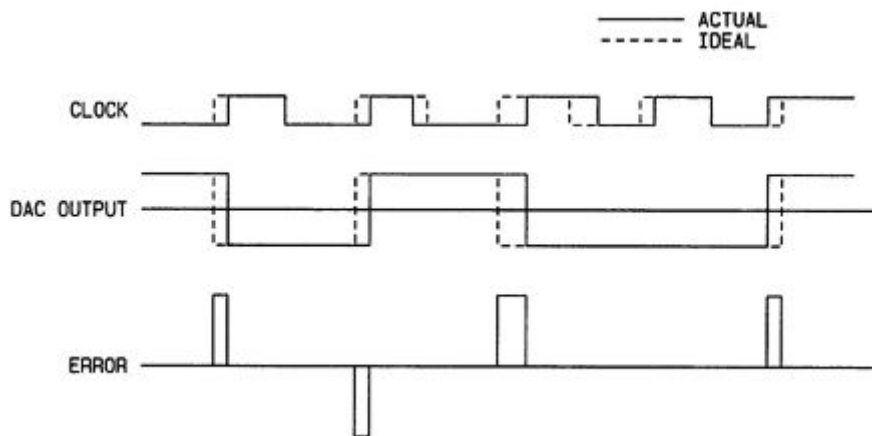


Figura 2.12: Effetto del jitter su un segnale a un bit. [14]

La soluzione più immediata al problema risulta un miglioramento generale nella qualità del sistema, ad esempio attraverso l'utilizzo della fibra ottica più che dei classici cavi seriali o l'implementazione di differenti PLL all'interno dello stesso circuito. Un'alternativa molto valida risulta però essere l'implementazione di un architettura ove venga invertito il rapporto "master-slave" tra lettore e convertitore (fig.2.13):

<sup>11</sup> *Phase Locked Loop*.

il PLL viene infatti sostituito da un generatore di clock, di solito un oscillatore al quarzo, che pilota sia il lettore che il convertitore. In questo modo tutti gli errori dati dal PLL vengono eliminati, e il umore generato cala visibilmente.

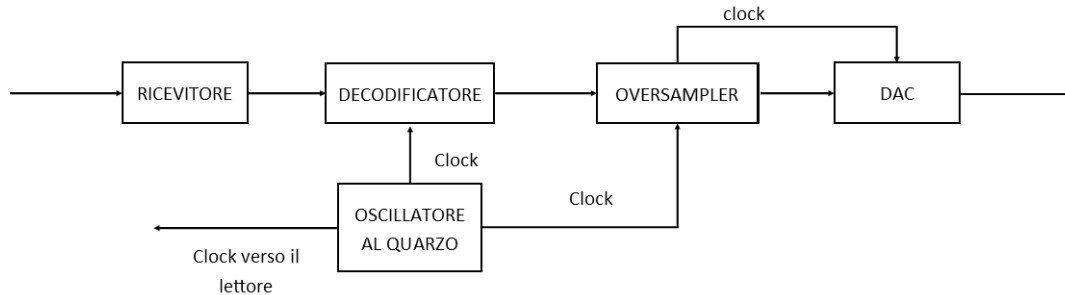


Figura 2.13: Esempio di architettura con generatore di clock esterno

## 2.4.2 Non linearità e passaggio per lo zero

Un altro problema molto influente sulla resa della conversione è la scarsa linearità presentata ai bassi livelli, o, detto in gergo, *al passaggio per lo zero*. Come già visto nel primo capitolo, la caratteristica di un DAC può essere afflitta da non linearità differenziali, che, in casi estremi, possono generare dei punti di non monotonicità. Essendo l'ampiezza di tali errori in relazione percentuale rispetto al peso del bit che vanno ad affliggere, è evidente che l'incertezza massima ricadrà sull'MSB, e che tanto maggiore è il numero di bit, tanto maggiore sarà il rapporto  $E_{MSB}/LSB$ .<sup>12</sup> Quantitativamente, prendendo in considerazione campioni a 16 bit, che ormai sappiamo essere il formato della nostra sorgente, già un piccolo errore del 0.006% sull'MSB pesa quanto 2 LSB. Nel caso questo pesi in negativo sul primo bit, si ha che al passaggio  $0111111111111111 \rightarrow 1000000000000000$ ,<sup>13</sup> corrispondente all'aumento di un LSB, la tensione, invece di aumentare di un passo Q, diminuisce (fig.2.14). Si riporta in fig.2.15 l'effetto di una non linearità su un segnale sinusoidale: nello spettro compaiono fastidiose componenti spurie in tutte le bande di frequenza, difficilmente filtrabili.

Purtroppo non esiste una "panacea" a questo problema. Sicuramente una superiore qualità costruttiva, caratterizzata da minori incertezze sul valore delle resistenze, garantisce maggior stabilità. Una soluzione che viene spesso adottata consiste nell'elaborare i campioni in modo tale da non avere mai una commutazione dell'MSB, aggirando di fatto il problema. Le architetture bitstream, non avendo al loro interno resistenze pesate, sono strutturalmente immuni a questa problematica, fattore che le aiuta nel raggiungimento delle loro ottime prestazioni.

<sup>12</sup>Con  $E_{MSB}$  errore di non linearità sull'MSB.

<sup>13</sup>Il fatidico "passaggio per lo zero".

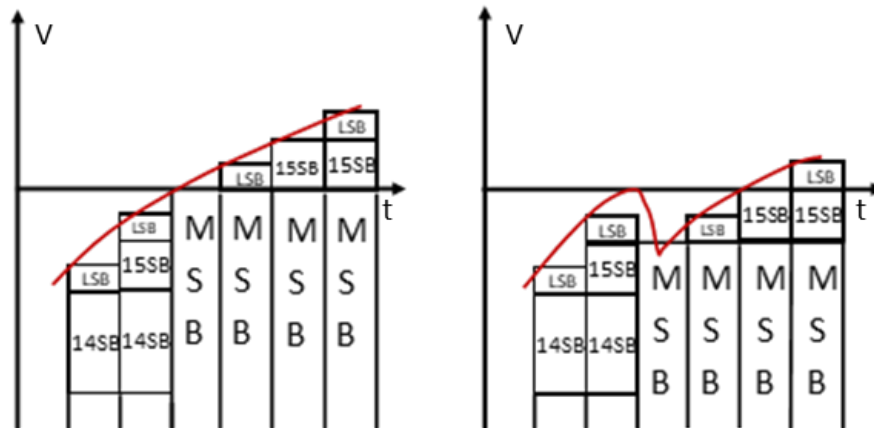


Figura 2.14: Transcaratteristica lineare (a sx) e transcaratteristica non lineare ai bassi livelli (a dx). [15]

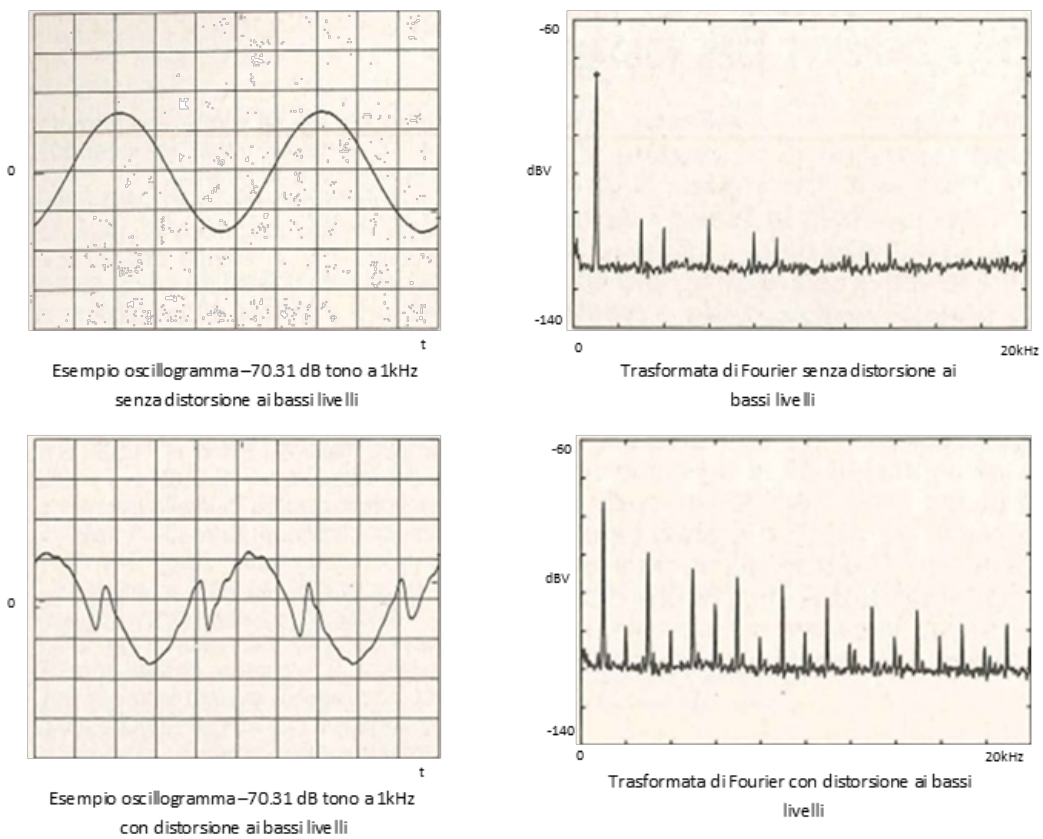


Figura 2.15: Ricostruzione di una sinusoide senza errori di non linearità (sopra) e sinusoide ricostruita con errori di non linearità (sotto). [15]



# Capitolo 3

## Convertitori a confronto

In questo capitolo verranno descritti alcuni esempi di convertitori DAC per riproduzione audio. Andremo quindi a confrontare 3 convertitori, uno dei primi anni 90 e due tutt'ora in commercio <sup>1</sup>, per vedere come i valori prestazionali di queste macchine siano cambiati in quasi trent'anni.

### Parametri per il confronto

Prima di effettuare il succitato confronto è utile fare una breve introduzione sui parametri che verranno utilizzati, ripassando brevemente quelli già visti e analizzando quelli non ancora discussi.

- Numero di bit: Rappresenta banalmente il numero di bit utilizzato nella conversione.
- Sovracampionamento: Il fattore di sovracampionamento utilizzato nell'architettura, secondo quanto già visto nel capitolo 2.
- SNR: Il signal noise ratio già discusso precedentemente, è indice della risoluzione raggiungibile tramite un determinato circuito DAC.
- Range Dinamico: Indica la differenza in dB fra l'ampiezza massima e l'ampiezza minima delle frequenze erogate in uscita distinguibili nello stesso intervallo di tempo. É infatti possibile che il rumore generato dalle componenti analogiche del convertitore, proporzionale all'ampiezza del segnale in uscita, superi il minimo valore convertibile definito dal passo di quantizzazione  $Q$ , imponendo un'ampiezza minima più elevata rispetto a quella teorica. Logicamente, qualora il rumore fosse sempre inferiore al passo di quantizzazione  $Q$ , il range dinamico diverrebbe equivalente all'SNR.
- THD+N: La distorsione armonica totale (Totale Harmonic Distorsion, THD) è un parametro che indica la distorsione armonica inserita in un segnale dal

---

<sup>1</sup>Si prenda come periodo di riferimento per la stesura di questa tesi l'anno 2021/22

dispositivo. Viene misurata ponendo in ingresso una sinusoide a frequenza prefissata (per convenzione  $f = 1kHz$ ) e analizzando lo spettro del segnale in uscita. Il THD sarà pari al rapporto fra la potenza delle armoniche generate e quella della fondamentale. Si ha quindi che

$$THD = \frac{\sqrt{V_2^2 + V_3^2 + V_4^2 + \dots}}{V_1} \quad (3.1)$$

con  $V_n$  valore efficace in tensione dell' $N$ -esima armonica. Il THD+N include nel calcolo anche il rumore bianco inserito dai circuiti, generalizzando il significato del parametro al grado di distorsione complessivo introdotto dal dispositivo.

- **Jitter intrinseco:** Il jitter intrinseco indica il massimo valore di jitter introdotto dal convertitore nel segnale in ingresso. Minore è tale valore, più fedele sarà il segnale in uscita.
- **Attenuazione in banda soppressa:** Questo parametro indica la differenza in dB fra il valore minimo nel guadagno riscontrabile in banda passante e quello massimo presente in banda soppressa inerenti al filtro digitale di interpolazione.
- **Ripple in banda passante:** Indica lo scostamento massimo fra la risposta in frequenza ideale del filtro digitale e quella effettivamente rilevata all'interno della banda passante. Essendo lo scopo del filtro quello di preservare il più possibile le caratteristiche del segnale di interesse, minore sarà il ripple maggiore sarà la qualità del filtraggio. Discorsi totalmente analoghi possono essere fatti per quanto riguarda il ripple in banda soppressa. Può essere espressa in percentuale, anche se solitamente è riportata in dB.

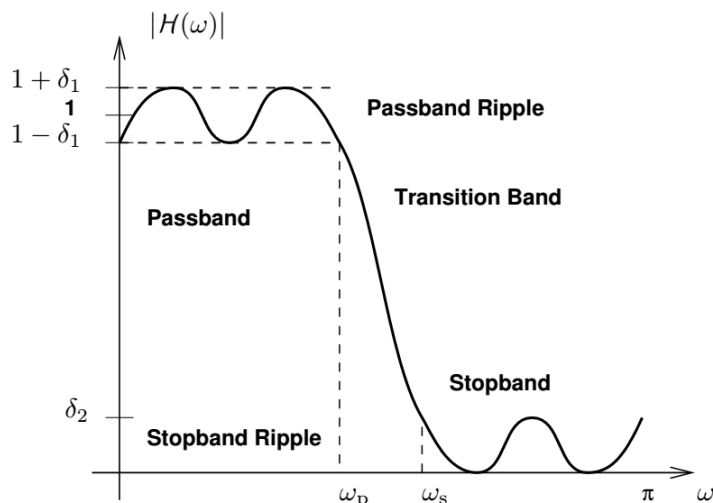


Figura 3.1: Ripple in banda passante e soppressa [16]



Figura 3.2: Sonic Frontiers SFD-2. [17]

### 3.1 Sonic frontiers SFD-2

In figura 3.2 è riportato un esemplare di Sonic Frontiers SFD-2, prodotto canadese commercializzato in Italia a partire da settembre 1993 al costo di £13.000.000. Nonostante l'aspetto poco appariscente, caratterizzato da linee squadrate e controlli ridotti all'osso, viene tutt'ora ricordato come uno dei migliori DAC hi-end mai costruiti. Le ottime prestazioni raggiunte, che all'epoca definirono lo stato dell'arte, sono dovute in gran parte all'utilizzo di componenti Ultra Analog, azienda californiana ai tempi leader in ambito di conversione D/A: la macchina si appoggia infatti su un'interfaccia digitale AES20 e su una coppia di DAC D20400A, entrambi prodotti della succitata casa, al tempo appena usciti.

Andando però ad analizzare nel dettaglio le componenti principali troviamo:

- Un decodificatore digitale Ultra Analog AES20. Il suo scopo, come già accennato nel secondo capitolo, è quello di ricavare il dato "effettivo" in ingresso, "ripulendolo" di tutti i bit accessori, e di rilevare il clock con cui questo viene trasmesso, generando la minor quantità possibile di jitter. A tale scopo vengono utilizzati due PLL distinti, uno più rapido, ma a forte generazione di jitter, e uno più lento, che in serie al primo vada a "riagganciare" il clock dando un bassissimo jitter finale. Il primo fornisce il clock al decoder, componente in testa alla catena di elaborazione, il secondo, sfruttando il tempo di calcolo necessario alla decodifica, lo fornisce ai circuiti successivi. Si ottiene così un jitter massimo di  $40ps$ , risultato record ai tempi.
- L'elaborazione digitale è affidata ad un sovracampionatore NPC SM5803 APT, con fattore moltiplicativo 8x e attenuazione in banda soppressa  $>110$  dB.
- I convertitori, come detto, sono una coppia di Ultra Analog D20400A a 20 bit, uno per ciascun canale. Particolarità di questi convertitori è la loro struttura "ibrida": il loro funzionamento è fondamentalmente quello di una scala R-2R, ma dei 20 bit che la compongono gli 8 più significativi, più sensibili alle incertezze, sono controllati tramite componenti discreti, più costosi e precisi, mentre i rimanenti sono implementati su circuiti integrati. Altro aspetto interessante giace nell'utilizzo di un'organizzazione in "push-pull", analoga a quanto già analizzato nel secondo capitolo.

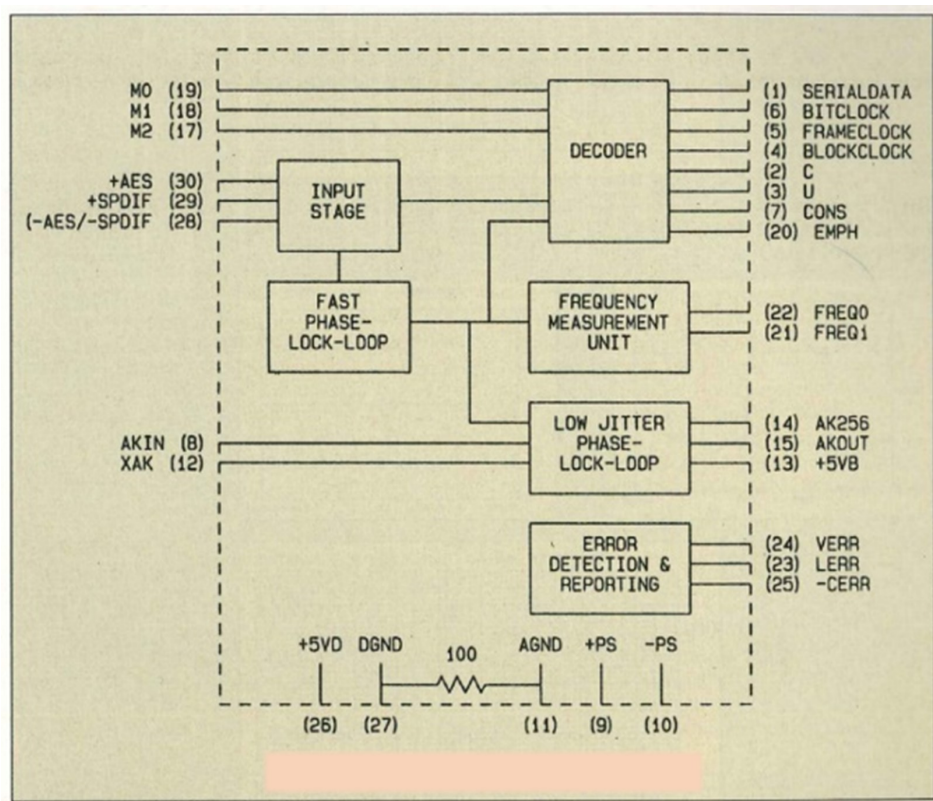


Figura 3.3: Schema a blocchi di un decodificatore AES20. [17]

Per concludere, si riportano alcune delle specifiche principali [17]:

Numero bit	20
Sovracampionamento	8x
SNR	110 dB
Range dinamico	99.9 dB
THD+N	0.05%
jitter intrinseco	40ps
Attenuazione in banda soppressa	-110 dB
Ripple in banda passante	$\pm 0.00005$ dB

## 3.2 Rockna Wavedream DAC

Quello presentato ora è il Wavedream DAC di casa Rockna, prodotto di fascia hi-end uscito nel 2020 al prezzo di € 9900. Come fatto poc'anzi, andiamo ad analizzare nel dettaglio gli aspetti più salienti di tale macchina:

- La gestione del clock è affidata ad un sistema proprietario di nome *Femtovox*. Per quanto non sia ancora possibile reperirne informazioni dettagliate, si sa





Figura 3.4: Rockna Wavedream DAC. [18]

che a differenza di quanto visto in fig. 2.11, in cui il clock generato dal PLL viene inviato alle componenti di elaborazione digitale e solo da lì al circuito di conversione, il sincronizzatore invia il riferimento sia agli elaboratori che direttamente al DAC, in modo che il clock venga modificato il meno possibile. L'ampiezza massima di jitter dichiarata dal costruttore è di  $300fs$ .

- L'elaborazione digitale è gestita da un' *FPGA*<sup>2</sup> su cui giacciono 58 DSP, che garantiscono circa 30 GFlops di potenza computazionale. Il fattore di oversampling è fissato a 16x con un filtraggio digitale operato tramite filtri FIR progettati secondo l'algoritmo di Parks-McClellan visto in precedenza.
- La conversione vera e propria è affidata a quattro DAC organizzati a coppie in configurazione "push-pull", similmente a quanto visto prima. Tali convertitori, di progettazione interna, basano il loro funzionamento sulla classica scala R-2R, in questo caso con architettura a 27 bit. Interessante il fatto che, a differenza di quanto visto con l'SFD-2, il circuito sia implementato esclusivamente tramite componenti discrete, nelle quali le resistenze, di provenienza aeronautica, sono caratterizzate da una tolleranza massima dello 0.001%.

Si riportano, come prima, le specifiche principali [18]:

---

<sup>2</sup>Field Programmable Gate Array, un circuito hardware le cui connessioni possono essere modificate tramite software.



Figura 3.5: Interno del Rockna Wavedream DAC. [18]

Numero bit	27
Sovracampionamento	16x
SNR	132 dB
Range dinamico	132 dB
THD+N	0.0008%
jitter intrinseco	300 fs
Attenuazione in banda soppressa	-145 dB
Ripple in banda passante	$\pm 0.0001$ dB

### 3.3 Texas Instruments PCM2704C

Ultimo dispositivo che andiamo ad analizzare è il Texas Instruments PCM2704, chip a bassissimo costo (indicativamente 3€ per la vendita ai privati) impiegato in piccoli convertitori D/A USB. A differenza dei convertitori visti in precedenza, questo si basa su una architettura bitstream con modulatore  $\Sigma - \Delta$ , caratterizzata da un fattore di oversampling 128x che garantisce un valore di SNR pari a 98 dB, l'equivalente di 16 bit effettivi. Il range dinamico, supposta la temperatura di utilizzo compresa fra i 20°C e i 30°C e la frequenza di campionamento  $f_s = 44.1kHz$ , presenta i medesimi valori dell'SNR. Il datasheet non presenta nessuna informazione riguardo al jitter, è specificata la presenza di un PLL all'interno del circuito ma null'altro. Il filtraggio digitale, infine, è la componente più carente, con un filtro a doppia retroazione caratterizzato da un'attenuazione in banda passante di 50dB e un ripple di 0.04dB. Si riporta di seguito un sunto di quanto appena detto. [19]



Figura 3.6: Texas Instruments PCM 2704. [19]

Risoluzione	16
Sovracampionamento	128x
SNR	98 dB
Range dinamico	98 dB
THD+N	0.012%
jitter intrinseco	N/D
Attenuazione in banda soppressa	-50 dB
Ripple in banda passante	$\pm 0.04$ dB

### 3.4 Confronto prestazionale

Di seguito è riportata una tabella riepilogativa di quanto visto finora.

Risulta immediato constatare come il Rockna, grazie ai quasi trent'anni che lo separano dall'SFD-2, risulti prestazionalmente migliore in praticamente ogni aspetto. È interessante però fare un paio di considerazioni alla luce di quanto visto. Innanzitutto, è curioso notare come in tre decenni vi sia stata un'evoluzione sul piano prestazionale ma non un'equivalente evoluzione "strutturale". Possiamo vedere infatti come il DAC vero e proprio continui a basarsi su scale R-2R, o comunque schemi da esse derivanti, senza che ci sia stata una vera e propria "rivoluzione" in quest'ambito. Curioso è pure notare come già nel '90 la spinta verso l'integrazione portasse allo sviluppo architetture ibride con l'obiettivo di spostarsi gradualmente verso l'utilizzo dei soli circuiti integrati. Oggi i migliori circuiti si basano su componenti totalmente discrete, controcorrente rispetto al resto dell'elettronica.

Ultimo aspetto interessante riguarda il confronto tra il top degli anni '90 e il convertitore base attuale: benché l'SFD-2 rimanga prestazionalmente superiore, ve-

	Sonic Frontiers SFD-2	Rockna Wavedream DAC	PCM2074
Architettura	Multibit	Multibit	Bitstream
Numero bit	20	27	1
Sovracampionamento	8x	16x	128x
SNR	110 <i>dB</i>	132 <i>dB</i>	98 <i>dB</i>
Range dinamico	99.9 <i>dB</i>	132 <i>dB</i>	98 <i>dB</i>
THD+N	0.05%	0.0008%	0.012%
Jitter intrinseco	40 <i>ps</i>	300 <i>fs</i>	N/D
Attenuazione B.S.	-110 <i>dB</i>	-145 <i>dB</i>	-50 <i>dB</i>
Ripple in B.P.	$\pm 0.00005$ <i>dB</i>	$\pm 0.0001$ <i>dB</i>	$\pm 0.04$ <i>dB</i>

diamo come in certi parametri la differenza non sia poi così ampia, ottenendo valori pressoché equivalenti per quanto concerne il range dinamico. La distorsione armonica risulta addirittura minore nel PCM2074, probabilmente grazie alla natura totalmente integrata di quest'ultimo. Stranamente, ciò su cui il PCM2074 perde il passo è il filtro digitale, dalle prestazioni nettamente inferiori a quanto presentato dagli altri due. Rimane comunque abbastanza impressionante (anche se l'evoluzione tecnologica ci ha abituato in questo senso) vedere come la funzione un tempo svolta da un dispositivo di 15kg e dall'ingombro non indifferente possa oggi essere svolta tramite un piccolo chip contenuto in una chiavetta USB.

# Conclusioni

In questa tesi è stato analizzato lo stato dell'arte nello sviluppo di convertitori D/A per il settore HiFi. Dopo una breve ma si auspica esaustiva introduzione alle fondamenta teoriche della conversione, si è esaminata la struttura "generale" di un DAC per la riproduzione audio, constatando come ogni convertitore si basi sostanzialmente su un unico schema fondamentale. Si sono inoltre evidenziate le differenze che separano le architetture multibit dai convertitori  $\Sigma - \Delta$ , le due grandi "famiglie" in cui è possibile dividere il mercato odierno. È stato infine proposto un "confronto generazionale" fra alcuni dispositivi, concludendo che le prestazioni più elevate vengono ad oggi ancora raggiunte tramite architetture  $R - 2R$  discrete, evoluzioni sul piano prestazionale, ma non strutturale, di quanto visto alla nascita della musica in formato digitale. E' stato constatato inoltre come il miglioramento più marcato giaccia nella riduzione del rumore e delle distorsioni inserite dal convertitore, evoluzione non riscontrabile invece per quanto riguarda l'elaborazione digitale.

Le architetture bitstream rappresentano un'importante componente per il futuro: il grande passo non sarà quello di migliorare prestazionalmente gli ingombranti dispositivi sopra descritti, quanto quello di raggiungere le medesime prestazioni tramite un DAC inseribile in uno smartphone o in un laptop. In un mercato dove auricolari e piccoli diffusori in generale presentano risposte in frequenza sempre più lineari, e soprattutto un suono sempre più gradevole, è importante che alla base ci sia una conversione efficace che riesca a sfruttare le potenzialità di questi dispositivi.



# Bibliografia

- [1] <https://zone.ni.com/reference/en-XX/help/370466AH-01/measfunds/resolution/>
- [2] <http://www.elemania.altervista.org/adda/parametri/param2a.html>
- [3] <https://www.edn.com/trimming-a-digital-to-analo/bibg-converter-to-improve-accuracy/>
- [4] <https://www.maximintegrated.com/en/design/technical-documents/app-notes/4/4159.html>
- [5] <https://training.ti.com/system/files/docs/Precision-20DAC-20-20DC-20Specs-0.pdf>
- [6] S. Buso, *Introduzione alle applicazioni industriali di Microcontrollori e DSP*, Società editrice Esculapio, 2020, Bologna, ISBN 978-888-7488935-8.
- [7] R. Rubino, P. S. Crovetti and F. Musolino, "FPGA-Based Relaxation D/A Converters With Parasitics-Induced Error Suppression and Digital Self-Calibration", in *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 68, no. 6, pp. 2494-2507, June 2021, doi: 10.1109/TCSI.2021.3064419.
- [8] P. Toledo, R. Rubino, F. Musolino and P. Crovetti, "Re-Thinking Analog Integrated Circuits in Digital Terms: A New Design Concept for the IoT Era", in *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 68, no. 3, pp. 816-822, March 2021, doi: 10.1109/TCSII.2021.3049680.
- [9] <https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-341-discrete-time-signal-processing-fall-2005/lecture-notes/lec15.pdf>
- [10] <https://www.edutecnica.it/elettronica/filtrip/filtrip.htm>
- [11] <https://www.microst.it/Tutorial/dac-char-2.htm>
- [12] M. Bertocco, A. Sona, *Introduzione alle misure elettroniche*, Lulu press, 2010.
- [13] R. Lucchesi, *La conversione a 1 bit*, *Audioreview* n°84, giugno 1989.
- [14] <https://www.stereophile.com/content/jitter-digital-interface-page-2>

- [15] R. Lucchesi, *Il passaggio per lo zero e le non linearità ai bassi livelli*, Audioreview n°74, luglio 1988.
- [16] <https://www.allaboutcircuits.com/technical-articles/design-examples-of-fir-filters-using-window-method/>
- [17] T. De Marchi, *Sonic Frontiers SFD-2*, Audioreview n°136, marzo 1994.
- [18] <https://www.rockna-audio.com/products/wavedream-dac>
- [19] <https://www.sengoku.co.jp/mod/sgk-cart/detail.php?code=3DXT-REMF>
- [20] P. M. Aziz, H. V. Sorensen and J. van der Spiegel, "An overview of sigma-delta converters," in IEEE Signal Processing Magazine, vol. 13, no. 1, pp. 61-84, Jan. 1996, doi: 10.1109/79.482138.
- [21] R. Lucchesi, *L'influenza del jitter sulle prestazioni dei convertitori D/A*, Audioreview n°120, ottobre 1992.
- [22] <https://dsp.stackexchange.com/questions/70779/how-is-signal-to-noise-ratio-actually-measured-by-receiver-equipment>
- [23] <https://hifiplus.com/articles/rockna-wavedream-dac/>
- [24] <https://patents.justia.com/patent/4972188>



# Ringraziamenti

Un ringraziamento speciale va sicuramente all'Avv. Zeno Garibba, le cui conoscenze in materia, unite alla tonnellata di materiale cartaceo gentilmente messi a disposizione, hanno plasmato gran parte dei contenuti di questo testo.

Vorrei inoltre ringraziare la mia Debora, che lungo tutti i numerosi anni di studio ha deciso di stare al mio fianco nonostante le difficoltà e la frustrazione pressoché perenne.

Ultimi, ma non per importanza, vorrei ringraziare i miei genitori, per aver gettato le fondamenta culturali e materiali grazie alle quali ho potuto intraprendere questo percorso, e per avermi sempre fornito quella serenità familiare che troppo spesso è mancata in altri ambiti.