

Scheda di laboratorio n.02

Risposta in frequenza e FFT

In questa seconda esperienza ci focalizzeremo nello studio della risposta in frequenza di semplici circuiti lineari passivi, di nuovo focalizzandoci più sugli strumenti e sull'analisi che sui circuiti che resteranno relativamente semplici. In particolare, vedremo come tale risposta possa essere misurata facendo una spazzata in frequenza sul segnale di ingresso (con una misura che potremmo chiamare *continuous wave* o CW) oppure sfruttando un qualche segnale impulsato e la trasformata di Fourier (con una misura che potremmo chiamare *pulsed*). Facciamo notare che questa dualità di approccio trova spazio in moltissimi ambiti di misura, anche al di là dell'elettronica, per esempio nella NMR (*Nuclear Magnetic Resonance*) o altre tecniche ancora. Inoltre, durante questa settimana inizieremo a fare pratica con l'ambiente di simulazione numerica di circuiti elettronici TINA.

1 La funzione di trasferimento

Nei futuri circuiti, ci troveremo spesso a confrontarci con la configurazione illustrata in Fig. 1.1, dove un qualche network¹ genera un voltaggio $V_{\text{out}}(t)$ in funzione di un certo $V_{\text{in}}(t)$. Qui faremo due semplificazioni fondamentali: la prima è che studieremo la risposta a circuito aperto, ossia dimenticando quelli che possono essere gli effetti connessi alla presenza di una qualche resistenza di uscita finita; la seconda è che assumeremo la risposta lineare. Nel nostro caso avremo un circuito *intrinsecamente lineare*, dato che ci occuperemo di reti passive composte da condensatori, resistenze e induttori. Tuttavia, in generale, qualsiasi piccola oscillazione attorno ad un punto di lavoro di un circuito può essere analizzata in maniera simile.

Chiarito questo, in termini molto generali, ci aspettiamo che la risposta $V_{\text{out}}(t)$ sia legata a $V_{\text{in}}(t)$ attraverso una funzione di risposta $G(t)$ secondo la convoluzione

$$V_{\text{out}}(t) = (G \star V_{\text{in}})(t) = \int_0^\infty G(\tau) V_{\text{in}}(t - \tau) d\tau \quad (1.1)$$

dove ricordiamo che $G(t)$ è anche la risposta ad un ingresso impulsivo $V_{\text{in}}(t) = \delta(t)$. Come noto, dopo una trasformata di Fourier l'equazione diventa una semplice moltiplicazione nello spazio delle frequenze

$$\hat{V}_{\text{out}}(\omega) = \mathcal{H}(\omega) \hat{V}_{\text{in}}(\omega) \quad (1.2)$$

dove la trasformata di Fourier $\mathcal{H}(\omega) = \mathcal{F}[G(t)](\omega)$ è detta **funzione di trasferimento**. Un modo semplice per misurare la risposta consiste quindi nell'impostare un ingresso sinusoidale $V_{\text{in}}(t) = A_{\text{in}} \cos \omega t$, che qui descriveremo in formalismo complesso come un fasore² $A_{\text{in}} e^{j\omega t}$. Dalla (1.2) ci attendiamo una uscita $\mathcal{H}(\omega) A_{\text{in}} e^{j\omega t} = A_{\text{out}} e^{j\varphi} e^{j\omega t}$, quindi la $\mathcal{H}(\omega) = (A_{\text{out}}/A_{\text{in}}) e^{j\varphi}$ può essere misurata facendo una scansione in frequenza e andando a misurare:

- il **guadagno**³ (in tensione in questo caso), che è uguale al modulo $|\mathcal{H}(\omega)| = A_{\text{out}}/A_{\text{in}}$ e indica il rapporto di ampiezza fra le oscillazioni in ingresso e in uscita.
- lo **sfasamento** fra ingresso e uscita $\varphi = \arg[\mathcal{H}(\omega)]$; se la fase di $\mathcal{H}(\omega)$ è positiva (negativa), diremo che il segnale in uscita è in anticipo (ritardo) sul segnale in ingresso.

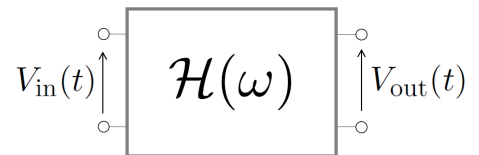


Figura 1.1: Filtro passivo

¹Tecnicamente viene detto "quadrupolo"; nella pratica per noi i due riferimenti in basso delle due porte saranno sempre lo stesso elettrodo di terra, ma in generale potrebbero essere diversi.

²Notate la convenzione di segno $e^{+j\omega t}$, usata in matematica ed elettronica. In fisica tipicamente usiamo $e^{-i\omega t}$.

³Notare che qui il "guadagno" tipicamente sarà inferiore a uno ossia è un'attenuazione.

Un modo tipico per rappresentare queste due quantità è di usare un **diagramma di Bode**, che consiste nel tracciare due curve in funzione della frequenza: una per il **guadagno** e una per lo **sfasamento**. Questi diagrammi vengono rappresentati usando una scala logaritmica per le frequenze e il guadagno. Questo si fa in primis per coprire un'ampia gamma di frequenze, che comprende diversi ordini di grandezza; per esempio, i dispositivi per la riproduzione o la rivelazione di segnali acustici vanno da 20 Hz a 20 kHz e una scala lineare che arrivi a 20 kHz comprimerebbe in una zona troppo ridotta la regione delle basse frequenze, mentre una scala logaritmica dà lo stesso spazio ad ogni decade. La seconda ragione è che le funzioni di risposta delle reti passive hanno sempre una forma del tipo

$$\mathcal{H}(\omega) = \frac{P(j\omega)}{Q(j\omega)} \quad (1.3)$$

dove P e Q sono dei polinomi a coefficienti reali⁴. La scala bilogaritmica nel guadagno è utile per evidenziare regioni dello spettro dove dominano alcune leggi di potenza in $\mathcal{H}(\omega)$: in figura, fino a circa 10 Hz chiaramente $\mathcal{H}(\omega)$ cresce come ω , mentre in seguito il guadagno rimane costante; in seguito, poco dopo 1 kHz, il guadagno decade come $1/\omega$. Tutti questi trend sono lineari in scala bilogaritmica

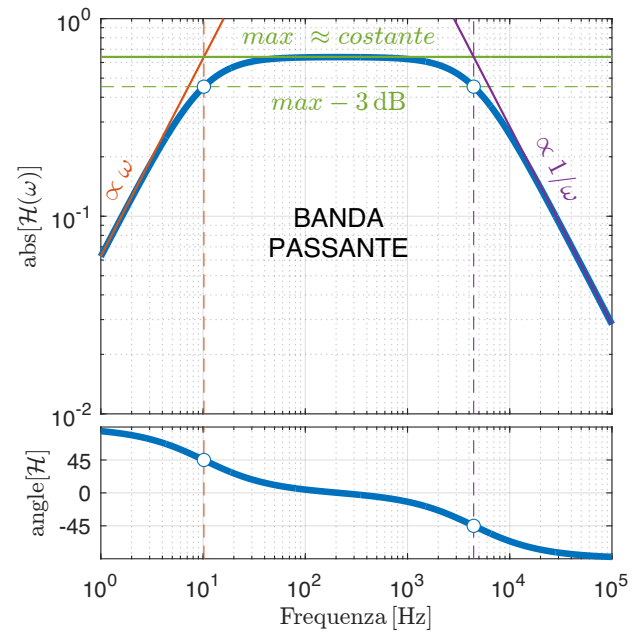


Figura 1.2: Esempio di diagramma di Bode.

2 Studio di rete passiva

Inizieremo sfruttando la scheda DAQ per misurare il diagramma di Bode di un circuito lineare, nella fattispecie di un *filtro passivo*. Dato che avete già fatto pratica sull'acquisizione, alzeremo un po' il livello sull'elettronica realizzando un passa-banda componendo un CR e un RC.

Lo schema del circuito con le connessioni da effettuare è illustrato in Fig. 2.1: un segnale viene applicato all'ingresso del filtro a sinistra e siamo interessati a capire come questo propaga fino all'altro lato a destra, in maniera da estrarre $\mathcal{H}(\omega)$. Come visibile in figura avremo bisogno delle seguenti connessioni:

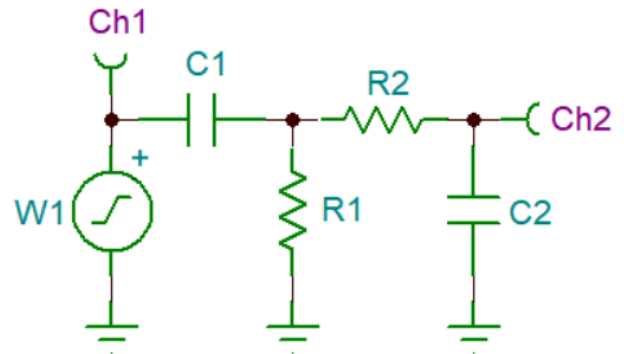


Figura 2.1: Passa banda CRRC.

- l'uscita W1, ossia la sorgente del segnale;
- l'ingresso Ch1 che useremo per misurare il voltaggio in ingresso nel filtro V_{in} ;
- l'ingresso Ch2 che useremo per misurare il voltaggio in uscita dal filtro V_{out} .

IMPORTANTE. Ricordiamo che – dove non esplicitamente indicato – misureremo dei segnali rispetto alla terra. Questo significa che Ch#+ è l'elettrodo di misura da collegare al punto indicato per semplicità come Ch# nello schema, mentre Ch#- va collegato a terra. Questa configurazione può essere detta RSE (modalità *(Ground) Referenced Single-Ended*), in contrasto per esempio con DIFF (modalità differenziale) in cui potremmo connettere sia Ch#+ che Ch#- a voltaggi diversi dalla terra.

Task 1 Costruite il filtro e connettete la strumentazione. Usate indicativamente i seguenti componenti: $R_1 = 100\text{ k}\Omega$, $C_1 = 100\text{ nF}$, $R_2 = 1\text{ k}\Omega$ e $C_2 = 56\text{ nF}$. Non tutti hanno questi componenti quindi se li variante anche fino al 50% va bene lo stesso. Alternativamente, possiamo anche fornire componenti mancanti dal nostro stock. Una volta montato (in maniera ordinata!), chiamateci per un check.

⁴Il motivo è che la risposta in frequenza deriva da una soluzione delle ODE lineari delle reti passive, che usando i fasori diventano semplici equazioni algebriche e inversioni di matrice dove gli elementi sono polinomi a coefficienti reali in $j\omega$.

Studio del comportamento atteso

Come mostrato a lezione, la funzione di risposta ha in questo caso la forma funzionale

$$\mathcal{H}(\omega) = \frac{j\omega\tau_1}{(1+j\omega\tau_1)(1+j\omega\tau_2) + j\omega R_1 C_2} = \frac{j\omega\tau_1}{(1+j\omega\tau_A)(1+j\omega\tau_B)} \quad (2.1)$$

dove $\tau_1 = R_1 C_1$ e $\tau_2 = R_2 C_2$. Le costanti di tempo τ_A e τ_B si possono risolvere con le solite formule per le radici di un polinomio di grado due, e non sono riportate esplicitamente. Le frequenze angolari $\omega_A = 1/\tau_A = 2\pi f_A$ e $\omega_B = 1/\tau_B = 2\pi f_B$ sono quelle a cui il Bode cambia pendenza. Noi useremo tipicamente dei valori numerici, ma spesso le curve di guadagno di un filtro come quello che stiamo per studiare si scrivono nella scala logaritmica dei **decibel** che può essere calcolata dal rapporto fra le ampiezze in ingresso e in uscita come

$$20\log_{10}(A_{\text{out}}/A_{\text{in}}). \quad (2.2)$$

Le pendenze $\propto \omega$ e $\propto 1/\omega$ vengono a questo punto espresse tipicamente in “decibel per decade”: in entrambi i casi abbiamo ovviamente 20 decibel per decade, ma per filtri più “ripidi” (di ordine superiore, dove la parola “ordine” è direttamente connessa con l’ordine del polinomio $Q(\omega)$), con magari una soppressione $\propto 1/\omega^2$, è possibile avere pendenze superiori. Chiaramente il cambio di pendenza alle frequenze f_A e f_B è graduale, ma è facile verificare che a quelle frequenze si ha una soppressione di un fattore $1/\sqrt{2}$ corrispondente a -3 dB. È facile mostrare anche queste frequenze sono anche quelle a cui si incrociano le estrapolazioni dei regimi lineari del filtro. Il punto in cui cambia il comportamento della funzione di risposta viene spesso detto “gomito” o “corner” del filtro, e la soppressione viene tipicamente descritta come “roll off” del filtro.

ATTENZIONE. Prima di iniziare a plottare qualsiasi cosa, mettiamo in guardia su quanto sia facile fare confusione fra frequenze angolari (ω) e frequenze “standard” ($f = \omega/2\pi$). Qualcuno finisce sempre per confondersi e – dato che il fattore 2π non è esattamente trascurabile – ottenere risultati completamente sbagliati. Per minimizzare i rischi, quando riportate i risultati, plot, fit, eccetera, siete obbligati a:

- dare **etichette chiare** agli assi^a magari indicando oltre all’eventuale testo qualche simbolo inequivocabile come ω , f oppure ν ;
- anche se un po’ artificioso, usare **unità diverse** per le due quantità: sempre e solo rad/s per le frequenze angolari; sempre e solo Hz per le frequenze.

^aL’opzione di non etichettare gli assi ovviamente non è nemmeno presa in considerazione: non ci provate...!!!

Task 2 Generate numericamente con MATLAB un grafico di Bode per la funzione di risposta in (2.1). Nel farlo evitate di usare i cicli for ma piuttosto costruite un vettore delle frequenze con il comando `logspace`^a e poi procedete con delle operazioni elemento per elemento. Confezionate un grafico, ma – per organizzazione dei vostri tempi – valutate se sia davvero necessario sistemare i dettagli finì mentre siete in H1/Lab2: probabilmente sono cose che potete aggiustare tranquillamente anche a casa. Per ora è sufficiente che verifichiate almeno in linea di massima che cosa aspettarvi dalla misura.

^ase avete dubbi su come usarlo scrivete `help logspace` e tutto sarà chiaro!

Task 3 Ricostruite il filtro in TINA e simulate la risposta in frequenza. Scaricare i dati su file, plottarli con MATLAB insieme alla simulazione numerica ottenuta nel precedente punto. Salvate il tutto nel grafico cumulativo `Simulazioni_CRRC.pdf`.

Primi check sperimentali “manuali”

Partiamo qualche verifica preliminare usando `dac+adc.ipynb`, che è una versione un poco più avanzata di quello scritto da voi la scorsa settimana e che alcune funzionalità interattive di base come:

- l’aggiornamento continuo o meno della presa dati (tasto “space”);
- il salvataggio su file (tasto “x” di *eXport*).

Task 4 Sbirciate nel codice e verificare che vi sia tutto chiaro. L'interazione utente non è cruciale (ma neanche incredibilmente complessa, date pure una occhiata e chiedete nel caso). Più utile la sezione di configurazione, che dovrete/dovreste capire nei minimi dettagli. Nella configurazione dell'oscilloscopio (qui chiamato *scope*) c'è una nuova chiamata al metodo *trig*, che gestisce il *trigger*. Esempi:

```
# Attivazione trigger a livello 0V (default) su canale Ch1 (default)
scope.trig(True)
# Attivazione trigger a 1V con isteresi 0.1V, su canale Ch2
scope.trig(True, level = 1, hist = 0.1, sour = tdwf.trigsrcCh2)
```

Suggeriamo di guardare la documentazione inclusa nel comando per i dettagli.

Task 5 Fare pratica cambiando manualmente i vari parametri di frequenza, ampiezza, forma d'onda e misurando con `dac+adc.ipynb`. Verificare la risposta del filtro “a campione”, per esempio impostando la frequenza a 10Hz, 100Hz, 1kHz e 10kHz. Verificare se/quando succede qualcosa di interessante, in particolare in termini di guadagno e sfasamento. Prendete pure nota delle cose rilevanti sul *logbook* ma per ora non è necessario fare misure quantitativamente esatte. Vogliamo solo verificare qualitativamente che tutto stia funzionando a dovere (sempre saggio prima di lanciarsi in misure infinite).

Risposta in frequenza: studio CW

Ovviamente non vogliamo cambiare a mano n frequenze e segnarci con carta e penna i guadagni e gli sfasamenti. Le misure automatiche in *continuous wave* possono essere fatte con il codice `sweepfreq.ipynb`, che ha una struttura simile a quella di `sweepbias.ipynb` ma con alcune piccole modifiche per venire incontro ad alcune nuove esigenze:

1. la scansione dei valori da f_0 a f_1 è spaziata logicamente;
2. il codice studia n_{per} periodi al variare della frequenza, ma nel confrontare il segnale in ingresso e in uscita modula il campionamento per cercare di acquisire più punti possibili entro il limite indicato n_{pt} ;
3. fase e ampiezza vengono ottenute con un fit⁵. Questo può facilmente portare a fit con ampiezze negative e/o a discontinuità nella differenza di fase, mentre noi vorremmo un risultato che ricada sempre nell'intervallo $[-\pi, +\pi]$.

Task 6 Aprite il codice e al solito verificate se capite bene che fa, individuando come sono stati risolti i problemi elencati, suggeriamo di prendere nota sul *logbook*, come sempre per futura memoria.

Task 7 Misurate la $\mathcal{H}(\omega)$ del filtro già montato, scegliendo l'intervallo di frequenze, il numero di punti e gli altri parametri che avete a disposizione... Fare un fit della ampiezza di $\mathcal{H}(\omega)$ usando la funzione di fit già scritta e verificate se il risultato è consistente con i valori dichiarati dei componenti e le loro tolleranze. Quali sono i valori f_A e f_B della banda passante? Quale è il guadagno a centro banda? Per quanto riguarda gli errori dei dati, siete liberi di scegliere la vostra strategia.⁶ Salvate il risultato finale su `Misura_CRRC.pdf`.

⁶Diamo alcuni possibili esempi: (1) quantificate gli errori facendo a campione delle misure a qualche frequenza fissata (quindi impostando $f_0=f_1$ nel VI) e verificare quanto è precisa la misura di fase e guadagno; (2) tentare di propagare l'errore da qualche incertezza nota (ma premettiamo che potrebbe non essere banale); (3) rifare l'intera misura Bode varie volte e calcolare varianza e media punto per punto.

Nel seguito aggiungeremo un diverso ingrediente, ossia l'**analisi di Fourier**. Partiremo richiamando i concetti teorici di base, proseguiremo con l'individuazione delle difficoltà tecniche e concluderemo con qualche esempio applicativo.

⁵Una delle possibilità, altri approcci sono possibili

3 Trasformate di Fourier

L'analisi di Fourier si basa sulla decomposizione di segnali periodici e non in somme o integrali di funzioni sinusoidali. I segnali possono essere discreti o continui nel tempo, e periodici o aperiodici, da cui possiamo definire quattro tipi di analisi:

- Per i segnali **aperiodici continui**, si usa la **trasformata di Fourier**;
- Per i segnali **periodici continui**, si usa la **serie di Fourier**;
- Per i segnali **aperiodici discreti**, si usa la **trasformata di Fourier a tempo discreto**;
- Per segnali **periodici discreti**, si usa la **serie (o trasformata) di Fourier discreta** o DFT.

Per un fisico sperimentale, i segnali sono ottenuti in un processo di misura che genera dati necessariamente discreti e limitati nel tempo; ci concentreremo quindi sull'ultimo caso illustrato.

Trasformata di Fourier Discreta

Sia x una grandezza fisica continua campionata per N volte a tempi discreti separati da un intervallo di campionamento Δt e frequenza di campionamento $f_s = 1/\Delta t$ (Fig. 3.1), con una durata totale di acquisizione $T_{\text{tot}} = N\Delta t$. Gli N valori misurati verranno qui indicati come $x_n = x(t_n) = x(n\Delta t)$, con n che va da 0 a $N-1$. La DFT è una operazione lineare biunivoca definita come segue (inclusa inversa)⁶

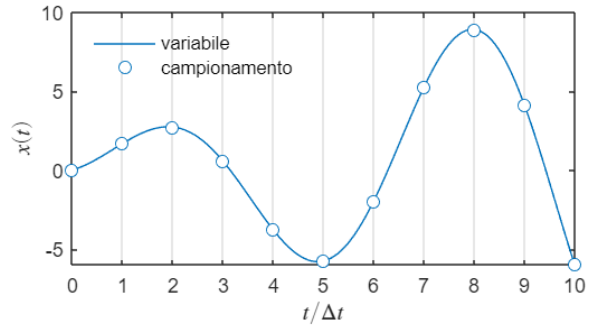


Figura 3.1: Campionamento.

$$X_k = \sum_{n=0}^{N-1} x_n \exp(-j\omega_k t_n) = \sum_{n=0}^{N-1} x_n \exp(-2j\pi nk/N) \quad (3.1)$$

$$x_n = \sum_{k=0}^{N-1} \frac{X_k}{N} \exp(+j\omega_k t_n) = \sum_{k=0}^{N-1} \frac{X_k}{N} \exp(+2j\pi nk/N) \quad (3.2)$$

la cui validità si può dimostrare facilmente dalla nota somma $\sum_{k=0}^{N-1} \exp(2j\pi nk/N)$, che si annulla sempre eccetto quando k è zero o un multiplo di N e la fase sono tutte nulle. Partendo dalla inversa (3.2), è chiaro che le nuove X_k non sono altro che le ampiezze complesse dello sviluppo di Fourier (discreto), per l'armonica a frequenza angolare $\omega_k = 2\pi k f_s/N$, ossia a frequenza $k f_s/N$. Un aspetto gradevole rispetto a trasformate e serie di Fourier è che trattandosi di un problema finito non bisogna mai preoccuparsi di divergenze, casi limiti o altro: la trasformazione funziona sempre. Tuttavia, la discretizzazione porta con sé un serie di altre questioni piuttosto subdole. Alcune osservazioni:

- **Risoluzione in frequenza.** La minima distanza fra le componenti spettrali nella DFT è banalmente $\Delta f = f_s/N = 1/T_{\text{tot}}$ quindi una prima regola banale (ma che a volte sfugge): per avere una alta risoluzione spettrale bisogna misurare a lungo! Tutto il resto è abbastanza irrilevante.
- **Repliche spettrali.** La trasformazione soddisfa $X_k = X_{k+N}$, ossia la DFT è costruttivamente non in grado di distinguere un segnale a frequenza f_k da $f_{k+N} = f_k + f_s$. Il motivo è semplice e ineludibile: a tempi discreti due sinusoidi a queste due frequenze appariranno identiche. Tutte le frequenze della DFT sono quindi note solo a meno di una traslazione di n volte f_s . Abbiamo scoperto un **primo artefatto** della discretizzazione, detto **aliasing**⁷.
- **Frequenze positive o negative?** Dato il punto precedente, parlare di frequenze positive o negative è un poco arbitrario. Potremmo dire che le componenti della DFT vanno da frequenza 0 fino a $(N-1)f_s/N$, ma più tipicamente fra tutte le repliche si prende quella più piccola in modulo. In questo caso considereremo le frequenze con $0 \leq k \leq N/2$ positive con $f_k = k f_s/N \geq 0$, mentre per le altre⁸ $f_k = (k-N)f_s/N < 0$, andando a coprire così l'intervallo $[-f_s/2, +f_s/2]$.

⁶Sono possibili altre normalizzazioni, per esempio moltiplicando tutto per $1/\sqrt{N}$, ma ci atteniamo a quella che si trova in genere implementata nei codici numerici, e in particolare in MATLAB. Notate anche la convenzione di segno per la fase.

⁷Ci sono vari modi per venire a patti con tutto questo: (1) è molto sensato, come possono N ampiezze un numero infinito di frequenze come nella serie di Fourier?; (2) così come la periodicità nello spazio dei tempi corrisponde alla presenza di armoniche discrete nella serie di Fourier, la discretizzazione dei tempi corrisponde ad una periodicità nella trasformata; (3) a ben guardare, lo spazio delle frequenze è in realtà un insieme di punti su in cerchio, in cui X_{N-1} dovremmo pensarla come "vicino" a X_0 .

⁸Se N è pari il caso $k = N/2$ in realtà fa un poco caso a sé, un poco come $k = 0$.

Qualche regola: i segnali reali e la densità spettrale

Quando le x_n sono reali, come sempre per noi, c'è una ulteriore semplificazione: $X_{N-k} = X_{-k} = X_k^*$ e conoscere le ampiezze complesse sull'intervallo $[0, f_s/2]$ sono sufficienti a definire del tutto il segnale. In questo caso è quindi sufficiente considerare le frequenze positive, ossia focalizzarsi su uno **spettro unilaterale** su $[0, f_s/2]$. Nel caso dei segnali reali, per una data X_k abbiamo la componente

$$\frac{X_m e^{j2\pi f_m t} + X_{-m} e^{-j2\pi f_m t}}{N} = \frac{2\text{Re}(X_m)}{N} \cos(2\pi f_m t) - \frac{2\text{Im}(X_m)}{N} \sin(2\pi f_m t)$$

ossia la parte reale e immaginaria di X_m sono connesse con le ampiezze in coseno e seno del segnale a frequenza f_k . Avremo quindi una ampiezza di picco $2|X_k|/N$ e una ampiezza quadra media $\langle x^2 \rangle = 2|X_k|^2/N^2$.

Fast Fourier Transform - FFT

La (3.2) non è particolarmente efficiente da un punto di vista computazionale. Nel 1965 Cooley e Tukey⁹ hanno trovato una brillante soluzione a questo problema, dimostrando un algoritmo molto più efficiente per calcolare la DFT, oggi noto come *Fast Fourier Transform*. Per i dettagli sulla FFT rimandiamo alla letteratura specifica, ma sostanzialmente l'algoritmo sfrutta uno schema iterativo per passare dalle $\approx N^2$ operazioni necessarie nella formula in (3.2) ad un approccio dove servono solo $\approx N \log_2 N$ operazioni. Va sottolineato che la DFT e la FFT producono risultati *identici*, quindi non si tratta di una approssimazione ma di una velocizzazione.

CONSIGLI PRATICI. Seguono consigli per il campionamento consapevole e per l'uso della FFT.

1. L'aliasing è assolutamente problematico quando il segnale contiene frequenze superiori a $f_s/2$. In base al **criterio di Nyquist** la frequenza di campionamento deve soddisfare almeno

$$f_s > 2.5 f_{\max},$$

(ma pure $f_s > 4 f_{\max}$...) dove f_{\max} è la massima frequenza rilevante contenuta nel segnale. Per questo motivo spesso nei sistemi di acquisizione si introducono dei *filtri anti-aliasing*, che tagliano le componenti spettrali troppo rapide prima di fare il campionamento.

2. In teoria l'efficienza della FFT è massima quando il numero di campionamenti N è una potenza di 2. Tuttavia, normalmente non sarà necessario usare N di questo tipo ed esistono diverse ricette per gestire in maniera efficiente N generici, già implementate nella `fft` di MATLAB.
3. Suggeriamo di usare il codice che segue per ottenere le ampiezze spettrali unilaterali $2|X_k|/N$:

```
XTv = fft(xv);
fv = (0:(N/2))*fs/N;
Av = abs(XTv(1:N/2+1))/N;
Av(2:end-1) = 2*Av(2:end-1);
```

dove xv è il vettore dei dati di N (qui multiplo di 2) valori, Av è il vettore delle ampiezze spettrali, fv il vettore delle frequenze delle frequenze, mentre fs è chiaramente la frequenza di campionamento.

Dimostrazione sperimentale dell'aliasing

Capire l'*aliasing* è fondamentale per fare un corretto uso della DFT, quindi ne metteremo in evidenza la fenomenologia rompendo tutte le regole che ci siamo dati in modo da capire a che problemi si va incontro. Studieremo un caso specifico ma significativo: **un segnale a dente di sega**, che ha un ricco sviluppo di Fourier

$$f(t) = \sum_{m=0}^{\infty} -\frac{2A}{\pi m} \sin(m f_0 t) \quad (3.3)$$

dove A è l'ampiezza e f_0 la frequenza.

⁹Cooley-Lewis-Welch, Historical Notes on the FFT, IEEE, Trans. on Audio and Electroacoustics, Vol-AU-15, 2, 1967.

Task 8 Usare `dac+adc.ipynb` per generare e misurare un dente di sega con rampa a salire di ampiezza 1 V e frequenza $f_0 = 10$ kHz. Collegare W1 a Ch1 (più terre) e salvare un dato, calcolare la trasformata di Fourier e plottare il segnale nel tempo e il modulo della DFT per le seguenti configurazioni:

1. Usare $f_s = 1$ MSa/s $\gg f_0$ e verificare qualitativamente la presenza delle armoniche di (3.3).
2. Usare ora $f_s = 38$ kSa/s: riuscite a dare un significato ai (vari) picchi che vedete?
3. Usare $f_s = 10$ kSa/s: che cosa osservate? Ha senso?
4. Usare $f_s = 10.5$ kSa/s: dovreste misurare un segnale apparentemente insensato, lo sapete spiegare?

Fare un grafico rappresentativo dell'aliasing e salvatelo su `Aliasing.pdf`.

Un altro artefatto: il leakage spettrale

Supponiamo di avere un segnale sinusoidale ad una data frequenza f_0 e supponiamo che il campionamento a f_s sia tale che nessuna delle varie $f_m = m f_s / N$ corrisponda a f_0 . Immagiamoci di arrivare al caso particolarmente sfortunato in cui f_0 casca fra due componenti della DFT. Che cosa possiamo aspettarci dalla trasformata? La risposta è... il **leakage spettrale**.

Facciamo notare che dato che in genere la trasformata si fa per vedere il contenuto spettrale di un segnale, f_0 in genere non sarà noto e quindi è del tutto probabile che f_s e N siano scelti in maniera tale che nessuna delle f_k sia esattamente uguale a f_0 . Ergo è del tutto normale trovarsi nelle condizioni di cui sopra. Qui riporteremo solo la risposta matematica calcolando la DFT di una sinusoide a f_0 con un campionamento generico. La trasformata vale

$$X_k = \sum_{n=0}^{N-1} \sin(2\pi f_0 t_n) e^{i2\pi f_k t_n} = \sum_{n=0}^{N-1} \frac{e^{i2\pi(f_k+f_0)t_n} - e^{i2\pi(f_k-f_0)t_n}}{2i}. \quad (3.4)$$

che possiamo riscrivere usando $t_n = n\Delta t$ e $\Delta t = T_{\text{tot}}/N$ come

$$X_m = \frac{1}{2i} \left[\frac{1 - e^{i2\pi(f_m+f_0)N\Delta t}}{1 - e^{i2\pi(f_m+f_0)\Delta t}} - \frac{1 - e^{i2\pi(f_m-f_0)N\Delta t}}{1 - e^{i2\pi(f_m-f_0)\Delta t}} \right]. \quad (3.5)$$

Finché f_0 è sufficientemente "lontano" da zero, praticamente i due termini non si sovrappongono in maniera apprezzabile e possiamo semplificare questa formula orribile il qualcosa di più intellegibile

$$\frac{2|X_m|}{N} \simeq \frac{1}{N} \cdot \left| \frac{\sin[\pi(f_m - f_0)N\Delta t]}{\sin[\pi(f_m - f_0)\Delta t]} \right|. \quad (3.6)$$

I due casi limite (quando $f_0 = 1$ kHz è esattamente una delle f_k oppure quando è fra due frequenze successive della DFT) sono riportati in Fig. 3.2. Si verifica facilmente che il caso ottimale a sinistra si ottiene quando $f_0 = k f_s / N = k / T_{\text{tot}}$ per un qualche k intero, ossia quando la misura ha una durata T_{tot} uguale a un multiplo del periodo del segnale $T_0 = 1/f_0$. Negli altri casi, e in particolare quando si misura un numero semi-intero di periodi, la trasformata si "sparpaglia" sulle componenti spettrali limitrofe.

Task 9 Continuando ad usare lo stesso segnale a dente di sega, cambiare i parametri di campionamento in maniera da evidenziare o sopprimere il fenomeno del leakage. Riuscite verificare, almeno approssimativamente, la correttezza dell'ampiezza dell'armonica in presenza di leakage? Facciamo notare che, con una sola sinusoide e nell'approssimazione di (3.6) le somme in quadratura delle ampiezze nei due grafici di Fig. 3.2 sono uguali. Fare un grafico sperimentale che evidenzi il fenomeno del leakage e salvatelo su `Leakage.pdf`.

Funzione di risposta: studio impulsato

Tornando al nostro filtro CRRC, ripartiamo dall'affermazione secondo cui $\mathcal{H}(\omega)$ non è altro che la trasformata della funzione di risposta $G(t)$, che è a sua volta la risposta ad un segnale di ingresso impulsivo $V_{\text{in}}(t) = \delta(t)$. Un segnale impulsivo è in effetti molto ricco in frequenza: la trasformata di $V_{\text{in}}(t) = \delta(t)$ ha infatti una distribuzione piatta che si estende fino a frequenze infinite. In realtà non è possibile generare un vero impulso a delta di Dirac, ma solamente funzioni impulsive finite, che possono comunque contenere una grande quantità di frequenze.

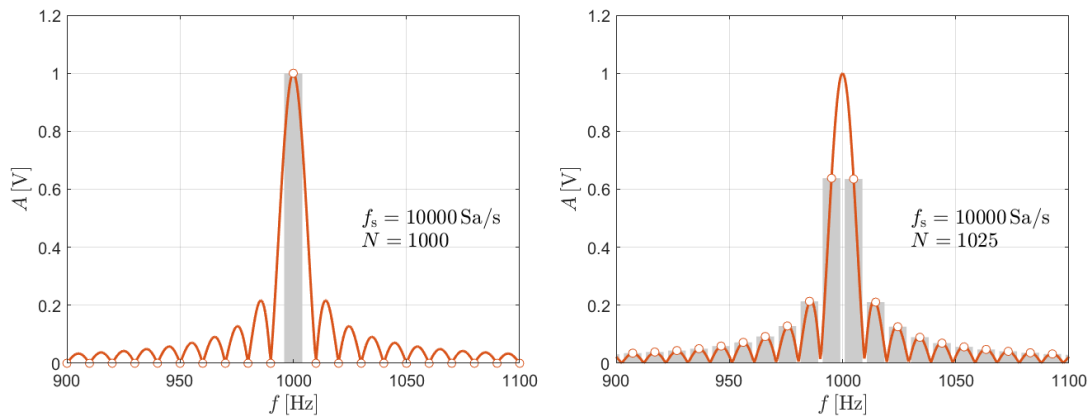


Figura 3.2: *Leakage spettrale nella DFT di una sinusoide a 1 kHz. La curva rossa corrisponde a (3.6).*

Task 10 Usare `dac+adc.ipynb` per generare un impulso^a ripetuto con una frequenza a piacere e misurare sia l'impulso stesso $V_{in}(t)$ che la risposta $V_{out}(t)$. Così come la forma dell'impulso, il campionamento è libero... ma prima di lanciarsi nelle misure suggeriamo di riflettere *in primis* su quale sia la trasformata dell'impulso ripetuto che inviate al circuito, in funzione della forma dell'impulso, della sua frequenza di ripetizione e del modo in cui lo campionate (quanti punti per periodo? quanti periodi?). Una volta effettuata una presa dati che vi soddisfa, calcolare la funzione di risposta implementando le seguenti operazioni sui vettori dei dati V_{in} e V_{out}

```
Hv = fft(Vout)./fft(Vin);
Hv = Hv(1:N/2+1);
fv = (0:N/2)*fs/N;
```

A questo punto la coppia f_v e H_v conterrà la funzione richiesta: usare questi vettori per generare un primo Bode, che sperabilmente (ma non è detto) ha qualche somiglianza con quelli già ottenuti in CW.

A questo punto ottimizzare i parametri di misura, inclusa l'eccitazione e il campionamento, per ottenere un Bode. Nella sezione in seguito si riporta qualche consiglio, molto probabilmente ne avrete bisogno. Generare un grafico finale `MisuraImpulsata_CRRC.pdf`. Non è obbligatorio fare un fit, data la relativa complessità del dato risultante, del suo rumore e possibili artefatti.

^aA volta scelta, potrebbe essere una onda quadra con un basso *duty cycle* e un opportuno offset e ampiezza (in maniera che oscilli fra una base 0 V e un qualche valore massimo), oppure una funzione a vostro piacere usando la modalità `func=tdwf.funcCustom` e un vettore di valori libero, per esempio una gaussiana!

Osservazioni e consigli. La procedura proposta pone alcune criticità, fra cui:

1. **Poco segnale.** Ad alcune (o magari molte) frequenze `fft(Vin)` contiene poco segnale o perfino solo rumore, generando vari problemi nel calcolo del rapporto tra le trasformate.
2. **Leakage spettrale.** Sconsigliamo vivamente di fare acquisizioni a frazioni di periodo di ripetizione dell'impulso, per evitare problemi connessi al *leakage*.
3. **Cattiva copertura.** Un segnale a frequenza ν contiene solo componenti a multipli di ν , inoltre non potete sperare di estrarre informazioni su frequenze superiori/prossime a quella di campionamento.

Tutte queste criticità si risolvono misurando un segnale strettissimo (e altissimo altrimenti in trasformata rimane poco), misurando molto a lungo e con un campionamento molto veloce (ma questo richiede moltissimi punti di misura, che non è detto che abbiamo). Purtroppo non si può. Qualche consiglio:

- Ribadiamo di riflettere su quale sia il contenuto spettrale della funzione di test usata.
- Non è detto che il Bode debba uscire da una sola misura: si può farne varie su vari range spettrali.
- Le misure con poco segnale si possono "eliminare" dai plot con `Hv(abs(fft(Vin))<0.x)=NaN`.
- Misurare a $n > 8192$ punti è possibile se il campionamento è sufficientemente lento da permettere il trasferimento dati (consigliamo massimo 0.5 MSa/s o il sistema potrebbe fallire la misura).