



Università degli Studi di Messina

Dipartimento di Ingegneria

Corso di Laurea magistrale in Ingegneria Elettronica per
l'Industria

Sistema di misura delle prestazioni di cuffie commerciali con ANC

Studenti:

Aldo Fonseca

Marco Gitto

Vincenzo Bucaria

ANNO ACCADEMICO 2023/2024

Indice

Indice	I
1 Introduzione	1
1.0.1 Obiettivi	1
1.0.2 Cenni teorici	2
2 Materiali e metodi	3
2.0.1 Struttura di isolamento del sistema di misura	4
2.0.2 Dispositivi e software per la generazione e la diffusione dei rumori di prova	6
2.0.3 Acquisizione ed elaborazione dei dati	8
2.0.4 Software e metodo di misura	9
3 Risultati sperimentali	18
3.0.1 Realme Buds	18
3.0.2 Huawei Free Buds Pro	19
3.0.3 Nothing Ear 2	20
4 Discussione e conclusioni	22
5 Reference	24

CAPITOLO

1 | Introduzione

Il progetto sviluppato si pone l'obiettivo di realizzare un sistema a basso costo in grado di caratterizzare i sistemi ANC (Active Noise Cancelling) ponendo l'attenzione sugli aspetti di seguito elencati.

1.0.1 Obiettivi

- Possibilità di valutare l'efficacia dell'isolamento passivo e della modalità ANC di diverse cuffie al variare del tipo di rumore e dell'intensità rumorosa
- Analisi spettrale dei dati acquisiti: al termine della misura il sistema permette all'utente di visualizzare lo spettro del rumore soppresso dal sistema ANC
- Permettere il paragone quantitativo tra cuffie ANC appartenenti a diverse fasce di prezzo
- Sistema di misura progettato per essere riproducibile a basso costo

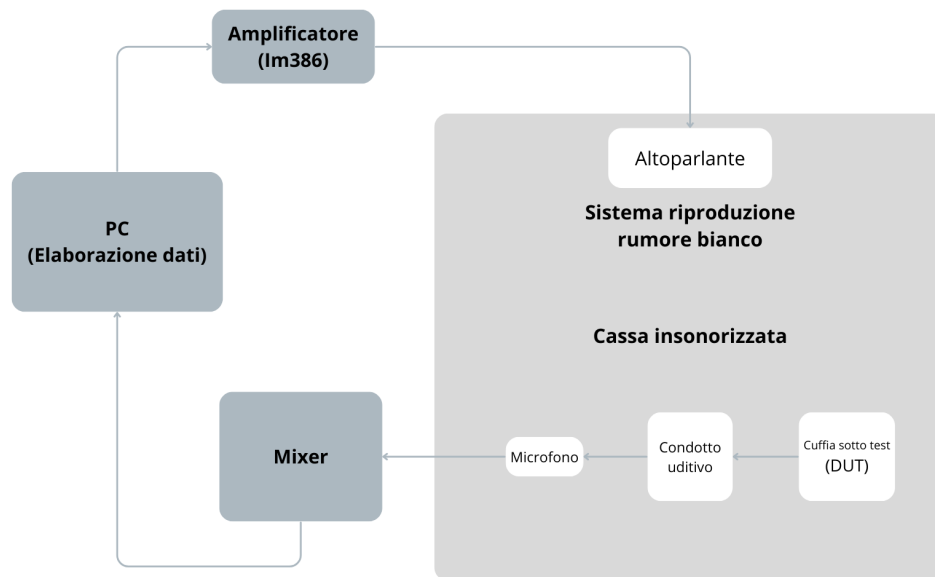


Figura 1.1: Schema a blocchi del sistema di misura

1.0.2 Cenni teorici

L'ANC, acronimo di Active Noise Cancellation, è una tecnologia sviluppata allo scopo di attenuare e, possibilmente, eliminare i rumori indesiderati provenienti dall'esterno durante l'ascolto in cuffia. Il suo funzionamento si serve di microfoni, opportunamente incorporati nei dispositivi di riproduzione che rilevano il rumore ambientale; un processore interno analizza quindi il segnale di rumore e genera un'onda sonora inversa, con fase opposta a quella del segnale di rumore rilevato. Questa onda sonora, definita "antifase", viene quindi emessa dagli altoparlanti del dispositivo. Quando l'onda antifase si combina con il rumore ambientale, si verifica una cancellazione parziale o totale, a seconda della qualità e dell'efficacia del sistema ANC in esame, del segnale di disturbo. La cancellazione attiva del rumore, dunque, consente di isolare l'utente da suoni indesiderati, migliorando l'esperienza di ascolto, riducendo la fatica uditiva e consentendo una maggiore concentrazione in ambienti rumorosi.

2

Materiali e metodi

I materiali e i dispositivi impiegati ai fini della realizzazione del progetto sono riportati nel seguente elenco:

- Vari modelli di cuffie in-ear dotati di Active-Noise-Canceling (Huawei Free Buds Pro, Nothing Ear 2 e Realme Buds)
- Struttura di isolamento del sistema di misura dall'ambiente esterno realizzato grazie all'impiego di:
 - Pannelli di legno compensato
 - Materiali fonoassorbenti
- Dispositivi e software per la generazione e diffusione dei rumori di prova:
 - Diffusore acustico con diametro di 7 cm
 - Circuito amplificatore autoprodotta basato sull'IC LM386
 - Software autoprodotta in Python
- Elettronica, software e materiali per l'acquisizione, condizionamento ed elaborazione dei dati:
 - Elemento simulativo di un padiglione con canale uditivo umano artificiale realizzato in gel di silice con rapporto 1:1

- Mixer audio
- Microfono cardiode low-cost
- PC e software autoprodotta in Python

In questo capitolo si discute di come i materiali dell'elenco sono stati impiegati, il loro ruolo e le scelte progettuali effettuate.

2.0.1 Struttura di isolamento del sistema di misura

Al fine di poter testare la qualità del sistema di cancellazione del rumore attiva, tra differenti modelli di cuffie (in-ear), ci si è avvalsi di un ambiente adatto al rilevamento il più possibile in assenza di disturbi, ovvero tramite l'isolamento dall'ambiente esterno.

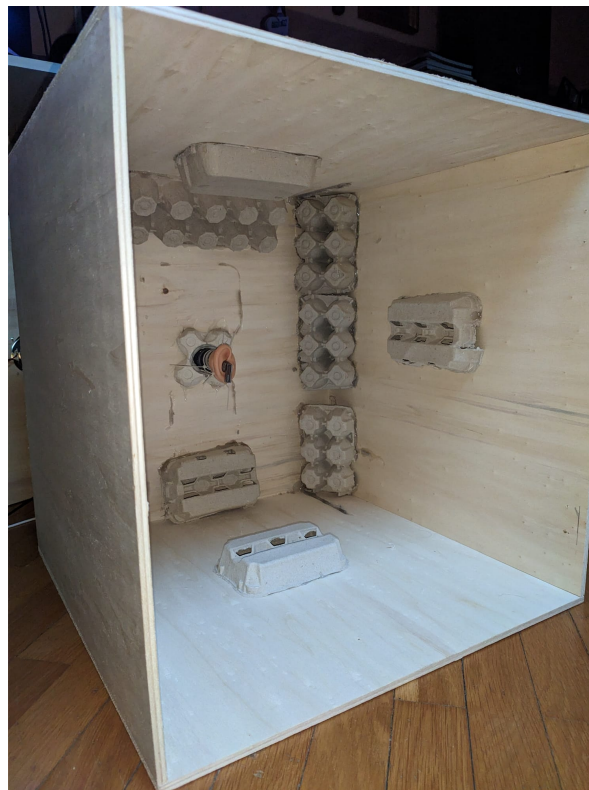


Figura 2.1: Dettaglio della struttura di isolamento

Si è dunque optato per una soluzione insonorizzante, entro il quale il comportamento del dispositivo possa risultare inalterato dall'influsso di variabili esterne. In una delle pareti è

presente la cuffia stessa, inserita nell'orecchio in silicone che simula il canale uditivo. All'altro capo dell'orecchio, è posizionato il microfono che effettua l'acquisizione. Questi elementi sono chiusi all'interno di una box insonorizzante. Per la struttura di test si è scelto di realizzare dunque una box cubica con 6 pannelli in legno compensato, di misura 50x50 e spessore 0.8 cm, aggiungendo un rivestimento ove possibile con materiali che possano abbattere la riflessione acustica all'interno della stessa, con l'obiettivo di poter effettuare una schermatura per il dispositivo in analisi. Il rivestimento è stato effettuato tramite l'utilizzo di un materiale molto comune e facilmente reperibile, ovvero il cartone che costituisce le vaschette contenitore per le uova trovate in commercio. La fibra in cellulosa che costituisce quest'ultime, è rinomata per essere adatta all'utilizzo in campo audio e musicale, come soluzione low cost per effettuare l'insonorizzazione. Il punto di forza di questi contenitori sta proprio nella forma degli stessi, che aiuta all'abbattimento delle onde acustiche. Il posizionamento delle vaschette è stato effettuato principalmente nei punti più critici della box, quali gli angoli.



Figura 2.2: Dettaglio della struttura di isolamento

Per fissare i pannelli e dare solidità e stabilità alla struttura, sono stati utilizzati alcuni

chiodi a testa piatta, posti agli spigoli delle tavole, tre per ogni lato. Per permettere invece di poter operare con la struttura stessa e poter effettuare i vari test, senza dover smontare l'intera box, la "porta" anteriore è stata fissata tramite due cerniere poste al lato sinistro del pannello. Sono stati inoltre effettuati due fori a due lati opposti del cubo; nel primo è stato posizionato il diffusore acustico, parte integrante della sezione di simulazione del rumore; dall'altro lato, invece, il foro è stato effettuato per permettere il passaggio delle onde sonore in direzione della cella microfonica, posta a contatto con il padiglione simulativo in silicone. Per ridurre quanto più possibile l'eventualità di incorrettezze nell'acquisizione dati tra il microfono e le cuffie in-ear in analisi, si è scelto di fissare il microfono al padiglione stesso. Tuttavia, dal momento che il silicone si presta poco all'interazione con materiali collanti, per via delle sue caratteristiche, la soluzione adoperata è stata quella di effettuare il contatto tra microfono e padiglione per pressione, utilizzando le comuni fascette stringicavo utilizzate in ambito elettrico. Sono state dunque poste rispettivamente ad anello attorno al padiglione ed al microfono, e successivamente poste a pressione l'una contro l'altra tramite altri due anelli fatti passare tra gli anelli precedenti, ponendoli dunque in tiro. La naturale forma del padiglione che all'esterno presenta un'incavo, è risultata adatta a questa scelta.

2.0.2 Dispositivi e software per la generazione e la diffusione dei rumori di prova

Al fine di rendere semplice, veloce e automatica la misura, il software in Python da noi prodotto è in grado di funzionare da sorgente e permette di riprodurre uno dei diversi modelli di rumore messi a disposizione.

Più nel dettaglio, la porzione di codice responsabile di ciò è racchiusa nel file *record.py* il quale si occupa di riprodurre i suoni di test e registrare il rumore percepito dal microfono e condizionato dal mixer. Nel seguente listato si può osservare la porzione di codice dedicata alla riproduzione dei rumori di test.

Listing 2.1: Esempio di codice Python

```

path_white_noise = os.path.join("noise",type_of_noise+elements+".wav" )

for elements in ["0dB", "-3dB", "-6dB", "-9dB"]: #Ciclo for per
    fare 4 misure con 4 livelli di rumore differenti
type_of_noise = "plane_" #Variabile che permette di scegliere il
    tipo di rumore riprodotto

winsound.PlaySound(path_white_noise,winsound.SND_ASYNC |
    winsound.SND_ALIAS ) #Riproduce il rumore di test
#Ho introdotto 1 secondo di delay per assicurarmi che la
    registrazione avvenga sicuramente dopo l'inizio della
    riproduzione dell'audio di test

...

...

winsound.PlaySound(None, winsound.SND_ASYNC)

```

Su sistemi Windows la riproduzione è possibile grazie alla libreria (o modulo) *winsound*, la quale permette mediante le api del sistema operativo di gestire i dispositivi di riproduzione sonora del pc.

E' possibile scegliere la tipologia di rumore da riprodurre agendo sulla variabile *type_of_noise*, scegliendo uno tra i seguenti valori possibili: *plane_*, *autobus_*, *white_noise_*

Per osservare il funzionamento dei sistemi delle cuffie al variare dell'intensità rumorosa, si è pensato, per ogni modalità di funzionamento delle cuffie, di effettuare quattro misure a quattro diverse intensità di rumore (0dB, -3dB, -6dB e 9dB).

Per essere riprodotto dall'altoparlante dal diametro di 7 cm, il segnale audio proveniente

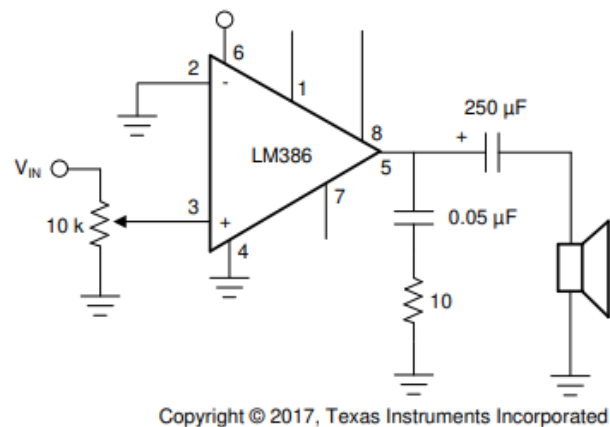


Figura 2.3: Schema dell'amplificatore

dal PC deve essere amplificato.

Tenendo conto che la distanza tra l'altoparlante e il microfono è contenuta e che l'altoparlante utilizzato possiede una buona efficienza, si è optato per un amplificatore a bassa potenza (1 Watt) basato sull'IC LM386.

In figura 2.4 è presente lo schema del circuito che abbiamo realizzato su una scheda millefori in un primo stadio di realizzazione.

Successivamente, poiché l'amplificatore risultava molto rumoroso, a causa dei disturbi introdotti dall'alimentatore utilizzato, è stato aggiunto un condensatore da 220 micro Farad a cavallo tra Vcc e Gnd, questo agisce da filtro e ci ha permesso di eliminare i disturbi accusati precedentemente.

Infine, l'altoparlante è fissato sulla parete opposta a quella del microfono ad una distanza, da quest'ultimo, di circa 45 cm.

2.0.3 Acquisizione ed elaborazione dei dati

Il trasduttore impiegato è un microfono cardioide low-cost per impiego canoro, un microfono più performante avrebbe comportato dei costi di realizzazione eccessivi.

Poiché la pressione sonora generabile da cuffie in-ear è di piccolissima entità, è stato necessario condizionare il segnale proveniente dal trasduttore con un amplificatore per microfoni ad alto guadagno, nel nostro caso incorporato nel mixer da noi utilizzato.

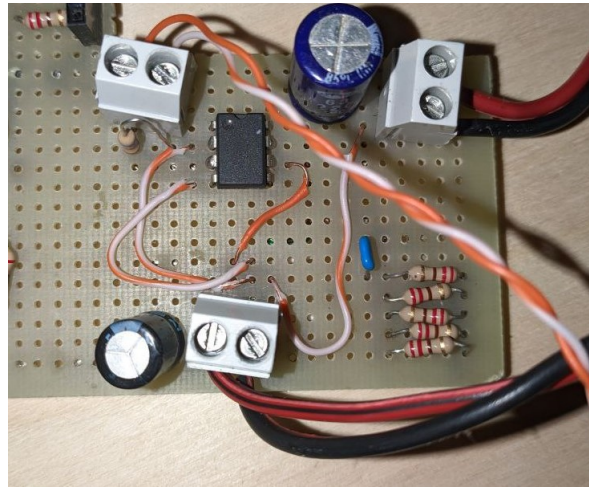


Figura 2.4: Circuito realizzato su scheda millefori

Realizzare autonomamente un amplificatore a basso rumore e ad altissimo guadagno sarebbe stato eccessivamente difficoltoso; in particolar modo per l'impossibilità di realizzare il circuito su un PCB. Il mixer, inoltre, ci ha permesso di equalizzare correttamente il microfono al fine di ottenere una risposta in frequenza quanto più piatta possibile, specificando che il microfono da noi usato opera su frequenze superiori ai 50Hz e inferiori ai 13kHz.

2.0.4 Software e metodo di misura

La procedura utilizzata per l'analisi del dispositivo si compone dei seguenti step:

- Un rumore campione genera la componente di disturbo, emessa dal diffusore acustico verso l'interno della box.
- Il microfono posto a contatto con il padiglione, trasduce l'abbattimento del rumore effettuato dalle cuffie ANC.
- I dati acquisiti dal microfono vengono inviati al software che ne effettua, per ognuna delle 3 modalità, la FFT, per permettere l'analisi spettrale e dare una stima precisa dell'effettivo funzionamento auspicato.

Al fine di rendere semplici le operazioni di misura, il software, che costituisce il centro di controllo del sistema di misura, è dotato di interfaccia grafica.

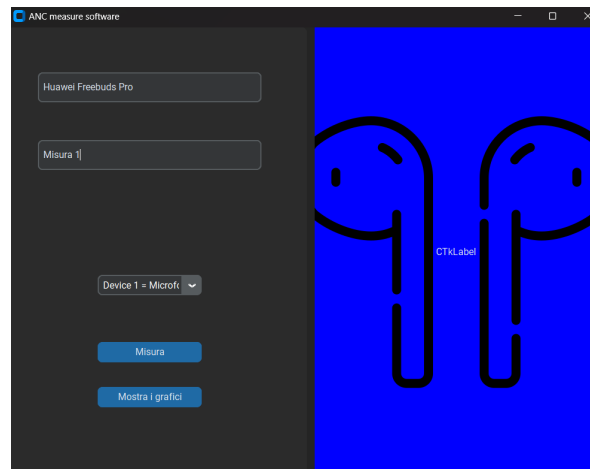


Figura 2.5: GUI di accoglienza

In figura è illustrata la prima schermata del software, dove è possibile indicare il nome delle cuffie ANC su cui effettuare la misura e il nome della misura stessa. Attraverso un menù è inoltre possibile scegliere il dispositivo di registrazione audio da utilizzare.

Il primo dei due pulsanti permette di dare inizio alla serie di misure, mentre il secondo pulsante permette di visualizzare a schermo una misura già effettuata in precedenza.

Come già accennato precedentemente, verrà valutata ogni modalità canonica delle moderne cuffie ANC, ovvero trasparenza (o alternatively si può eseguire la misura senza inserire le cuffie), off (isolamento passivo) e ANC. Per ciascuna di queste modalità verranno effettuate quattro misure a quattro diversi livelli di intensità sonora, ottenendo un totale di 12 misure.

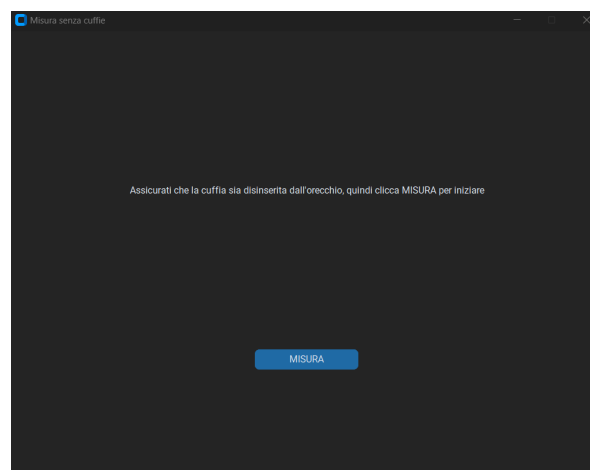


Figura 2.6: GUI di istruzione

Al cambiare di ogni modalità, il software istruirà l'utente sulle operazioni che deve compiere prima di procedere con la misura vera e propria cliccando sul pulsante "MISURA".

Per consentire un'esperienza utente reattiva e fluida il software è stato reso multi thread. Il main thread si occupa della GUI mentre per ogni misurazione viene lanciato un thread. In tal maniera la GUI può continuare a rispondere anche se contemporaneamente il software sta eseguendo la misura.

In questa fase non viene effettuata nessuna elaborazione, bensì il software si limita a riprodurre, come detto prima, a differenti intensità sonore, il rumore di test scelto e registrare il segnale acquisito dal microfono, salvandolo per la successiva elaborazione.

Il suono da riprodurre viene prelevato dalla cartella *noise*, mentre le registrazioni vengono salvate nella cartella **measures**, i nomi dei file registrati contengono il nome delle cuffie e il nome della misura: ciò consente all'utente di poter visualizzare a proprio piacimento una misura effettuata in precedenza.

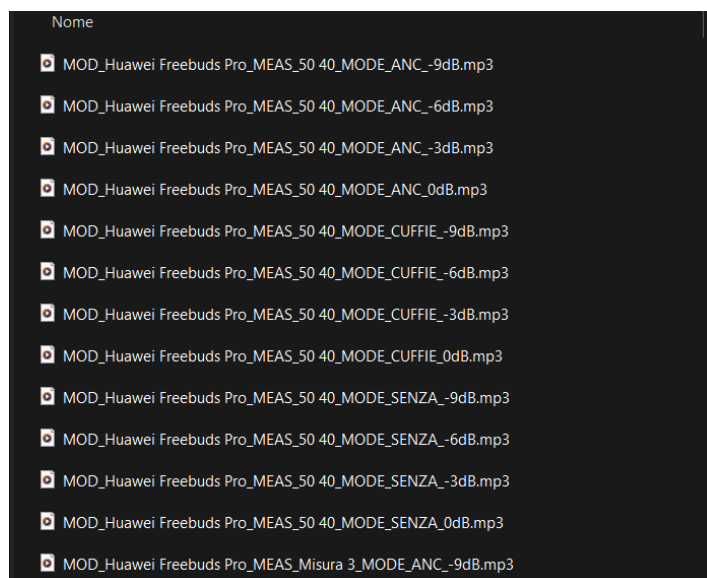


Figura 2.7: File di misure

Durante questa fase il codice ad intervenire è quello salvato nei file `measure.py` e `record.py`.

`Measure.py` è in realtà il "main" del software, mentre la riproduzione e la registrazione sono a carico di `record.py` di cui si mostra il codice.

Listing 2.2: record.py

```
chunk = 1024
sample_format = pyaudio.paInt16 #Numero di bit per campione
channels = 1
fs = 44100 # Frequenza di campionamento
seconds = 5 #Durata della registrazione

type_of_noise = "plane_"

def get_devices(): #Restituisce la lista dei dispositivi audio

    device_list = list()

    p = pyaudio.PyAudio()
    device_count = p.get_device_count()
    for i in range(0, device_count):
        info = p.get_device_info_by_index(i)
        device_list.append("Device {} = {}".format(info["index"],
            info["name"]))
    return device_list

def record_audios(filename, device_index):

    for elements in ["0dB", "-3dB", "-6dB", "-9dB"]: #Ciclo for per fare 4
```

```
misure con 4 livelli di rumore differenti

path_white_noise =
    os.path.join("noise",type_of_noise+elements+".wav" )

winsound.PlaySound(path_white_noise,winsound.SND_ASYNC |
    winsound.SND_ALIAS ) #Riproduce il rumore di test
#Ho introdotto 1 secondo di delay per assicurarmi che la
    registrazione avvenga sicuramente dopo l'inizio della
    riproduzione dell'audio di test

p = pyaudio.PyAudio()

stream = p.open(format=sample_format,
                channels=channels,
                rate=fs,
                input_device_index=device_index,
                frames_per_buffer=chunk,
                input=True)

frames = []

for i in range(0, int(fs / chunk * seconds)):
    data = stream.read(chunk)
    max = audioop.max(data, 2)
    print(20*math.log10(max)-91.28, max)
    frames.append(data)

stream.stop_stream()
```



```
stream.close()

p.terminate()

measure_path =
    os.path.join("measures", filename+"_"+elements+".mp3")
print(measure_path)
wf = wave.open(measure_path, 'wb')
wf.setnchannels(channels)
wf.setsampwidth(p.get_sample_size(sample_format))
wf.setframerate(fs)
wf.writeframes(b''.join(frames))
wf.close()
winsound.PlaySound(None, winsound.SND_ASYNC)
time.sleep(2)
```

Terminata la serie di misurazioni, la GUI informa l'utente il quale può, presso la schermata principale, istruire il software a elaborare i dati e stampare a schermo i grafici risultanti.

L'elaborazione vera e propria è a carico del codice salvato nel file *fft.py*, di seguito illustrato e di cui se ne spiega brevemente il funzionamento.

Listing 2.3: Esempio di codice Python

```
volumi = ["_0dB.mp3", "_-3dB.mp3", "_-6dB.mp3", "_-9dB.mp3"]

def calcola_le_fft(modello_cuffie, filename_1, filename_2, filename_3):

    mode_vector = ["Trasparenza", "ANC OFF", "ANC ON"]
    color_vector = ["green", "red", "blue"]
```

```
for volume in volumi:
    plt.figure(volumi.index(volume),figsize=(12, 6))

    print(filename_2)

    for i in range(1, 4):

        if(i==1):
            y, sr = librosa.load(filename_1+volume, sr=None)
        if(i==2):
            y, sr = librosa.load(filename_2+volume, sr=None)
        if(i==3):
            y, sr = librosa.load(filename_3+volume, sr=None)

    fft_result = np.fft.fft(y)
    frequencies = np.fft.fftfreq(len(fft_result), 1/sr)
    fft_result_db = librosa.amplitude_to_db(np.abs(fft_result))

    positive_frequencies = frequencies[:len(frequencies)//2]
    positive_fft_result = 2.0/len(y) *
        np.abs(fft_result[:len(fft_result)//2])
    positive_fft_result_db =
        librosa.amplitude_to_db(np.abs(positive_fft_result))
    freq_mask = (positive_frequencies >= 100) &
        (positive_frequencies <= 1000)
    low_freq_positive_fft_result_db =
        positive_fft_result_db[freq_mask]
    mean_value_low_freq = np.mean(low_freq_positive_fft_result_db)
```

```

plt.semilogx(positive_frequencies, positive_fft_result_db,
             label=mode_vector[i-1], color=color_vector[i-1])
if(mode_vector[i-1] != "Trasparenza"):
    plt.axhline(y=mean_value_low_freq, linestyle='--',
               label='Valore medio (LF) '+mode_vector[i-1], color=
               color_vector[i-1])

plt.title(modello_cuffie+volume)
plt.xlabel('Frequenza (Hz)')
plt.ylabel('Ampiezza(dB)')
plt.xlim([70, 20000])
plt.legend()

plt.tight_layout()
plt.show()

```

Per ogni registrazione audio afferita alla misura, il modulo *librosa* consente di caricare il file audio .mp3 in un vettore facilmente elaborabile.

Il modulo numpy mette a disposizione i metodi per computare la fft a partire dal vettore caricato da librosa.

Poiché lo spettro di un segnale reale è simmetrico rispetto l'asse delle frequenze, si è scelto di mostrare solamente il risultato relativo alle frequenze positive.

Una volta effettuata la computazione, i risultati vengono mostrati a schermo grazie al modulo matplotlib. Viene mostrato il grafico risultante dalla computazione della FFT, nonché le medie, una per ciascuna modalità, limitate ai valori compresi tra 100Hz e 10000Hz (tenendo conto che, il microfono da noi adoperato, risultava inadeguato al rilevamento delle bassissime e delle altissime frequenze).

Il software provvede quindi a stampare quattro grafici, uno per ogni intensità di rumore. Su ogni grafico sono a loro volta illustrati i risultati dovuti alle 3 modalità di funzionamento delle cuffie, nonché le media citate precedentemente.

Si noti bene che i risultati non sono espressi in termini di dB SPL ma sono riferiti al

livello di segnale audio nella catena di acquisizione, dove 0dB rappresenta il massimo livello gestibile.

3

Risultati sperimentali

Nelle seguenti sotto-sezioni si provvede a discutere i risultati ottenuti con ciascuna cuffia, con target di prezzo crescente; in particolare vengono mostrati i grafici riguardanti gli spettri in frequenza dei vari modelli al variare delle modalità dell'ANC e dell'intensità di rumore ambientale diffuso nella cassa di test.

N.B: Nel presente elaborato vengono riportati, per motivi di spazio, solo i grafici relativi al segnale di disturbo a 0dB.

I colori impiegati per evidenziare l'andamento dello spettro nelle diverse modalità di funzionamento dell'ANC sono:

- Rosso: modalità OFF
- Blu: modalità ON
- Verde: modalità trasparenza

3.0.1 Realme Buds

Le Realme Buds sono le cuffie più economiche tra quelle testate, si anticipa sin da subito che sono anche quelle con le peggiori prestazioni ANC.

L'isolamento passivo è buono, tuttavia la cancellazione attiva è inesistente alle basse frequenze e non brilla neanche alle medio basse frequenze. Alle medio alte frequenze si ha una

buona soppressione del rumore, nell'intorno di 2700Hz si registrano picchi di soppressione che variano tra 8dB e 12dB. Tra 2800Hz e 5000Hz il funzionamento dell'ANC è ancora evidente anche se non ottimale, per frequenze superiori a 5000Hz non si riesce più a distinguere il contributo dell'ANC da quello dell'isolamento passivo.

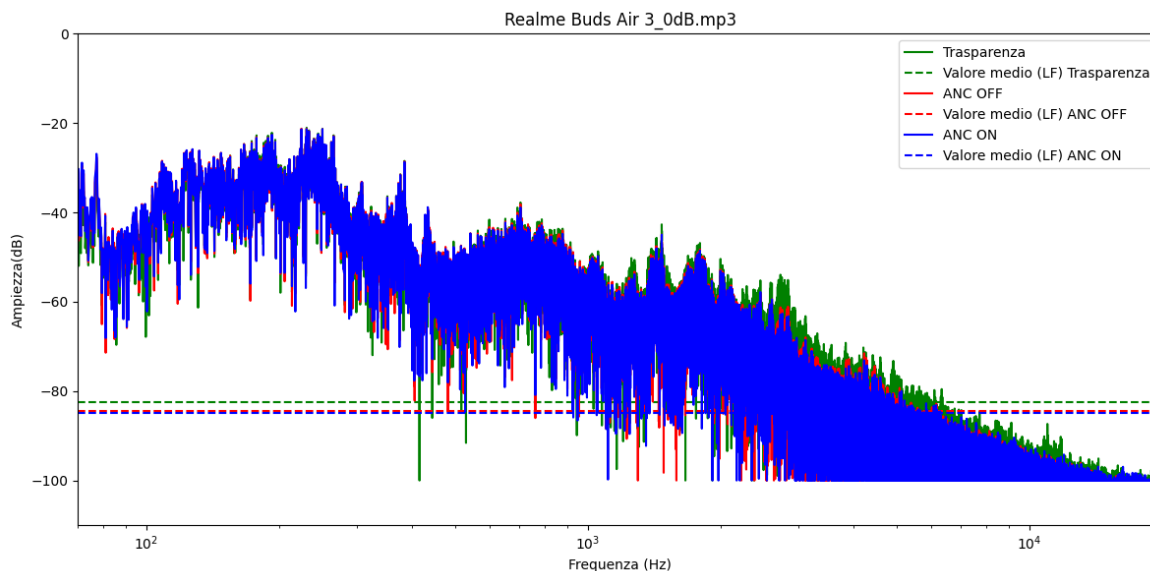


Figura 3.1: Analisi spettrale delle cuffie Realme Buds

3.0.2 Huawei Free Buds Pro

Il primo aspetto che è possibile notare è il buon funzionamento dell'isolamento passivo, rilevato in particolar modo alle medio alte frequenze.

Alle basse frequenze il comportamento dell'ANC è a nostro avviso buono, nei pressi di 100Hz registriamo dei picchi di soppressione compresi tra 8dB e 10dB. Tra i 200Hz e i 300Hz l'ANC non eccelle ed è influente, a partire dalla frequenza di 350Hz ritorna ad essere efficace. La soppressione alle medio-basse frequenze è notevole, nell'intervallo tra 440Hz e 520Hz si registrano picchi di soppressione che variano tra 6dB e 8dB, mentre nei pressi di 1000Hz si registra una soppressione di picco di 12dB. A partire da 1000Hz è evidente il buon funzionamento dell'isolamento passivo, cui permette, insieme a una buona performance dell'ANC, di registrare una soppressione di picco di 7dB a 1200Hz e di 12dB a 2600Hz. A 4000Hz l'ANC si comporta male, tra 4000Hz e 4200Hz è presente uno spot in cui l'ANC non

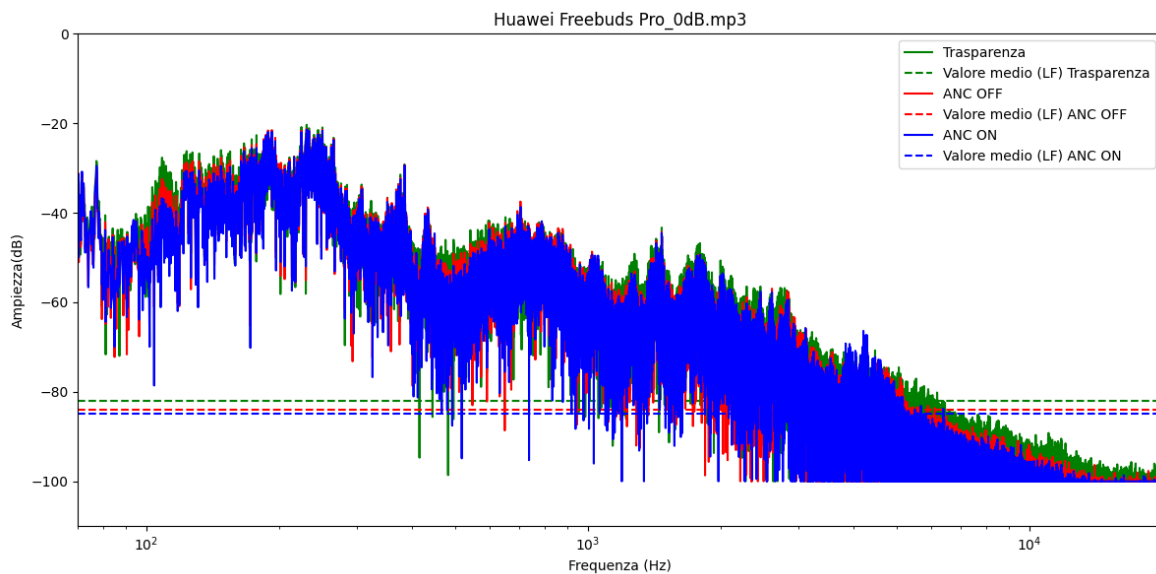


Figura 3.2: Analisi spettrale delle cuffie Huawei FreeBuds pro

sopprime minimamente il rumore. Superati i 4500Hz, l'ANC riprende a funzionare in maniera ottimale, sopprimendo in media di 8dB.

3.0.3 Nothing Ear 2

La misura con questo modello di cuffie è stata effettuata giorni prima rispetto agli altri modelli e con una configurazione del sistema non ancora ultimata, la maggior differenza riguarda la diversa equalizzazione del microfono.

Le cuffie mostrano un ottimo isolamento passivo, sicuramente migliore delle altre cuffie testate.

In tabella 3.1 si può evincere come nelle basse frequenze (20-250Hz) il dispositivo si dimostra in grado di effettuare un'attenuazione di circa 9 dB, alle medie (250Hz-2kHz) di 18 dB ed alle alte (2kHz- 20 kHz) di 6 dB. È evidenziato dai valori come le Nothing Ear 2 siano meglio preposte ad attenuare maggiormente le basse e le medio basse frequenze rispetto alle altre cuffie provate, mentre le Huawei sopprimono meglio alle medio alte e alte frequenze.

	Basse frequenze	Medie frequenze	Alte frequenze
0dB	9dB	15dB	6dB
-3dB	8dB	17dB	7dB
-6dB	8dB	19dB	4dB
-9dB	8dB	22dB	7dB

Tabella 3.1: Tabella riassuntiva dei risultati di attenuazione ottenuti con le Nothing Ear 2

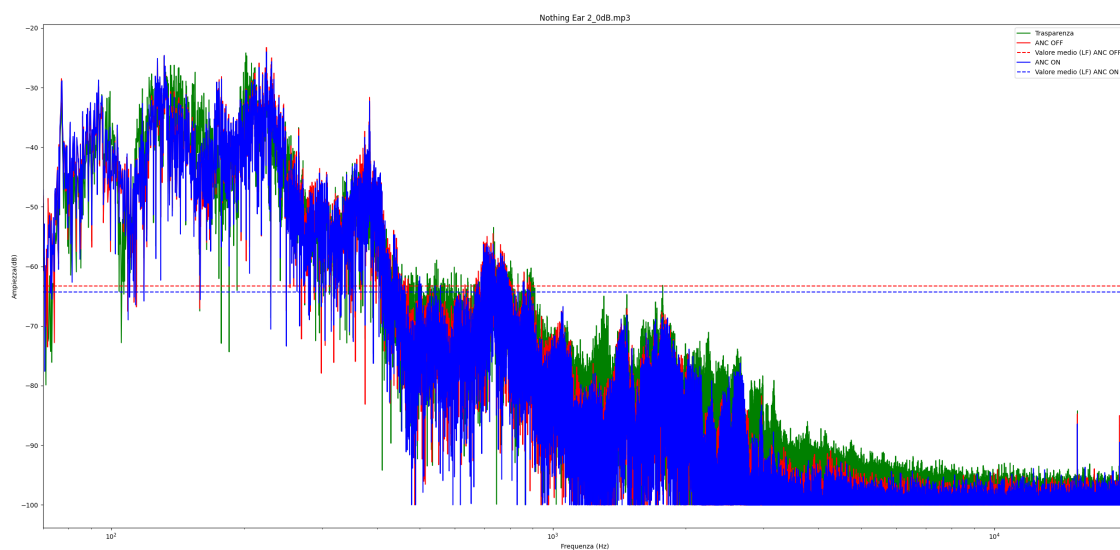


Figura 3.3: Analisi spettrale delle cuffie Nothing Ear 2 con segnale a 0dB

4

Discussione e conclusioni

In conclusione, l'esperienza di misura ha prodotto risultati soddisfacenti, rispecchiando correttamente le aspettative attese. Per ogni test condotto su ogni diverso dispositivo è stato possibile stimare quantitativamente l'effettivo funzionamento della tecnologia ANC. Nonostante gli ottimi risultati ottenuti dall'operazione di analisi precedentemente descritta si è voluto comunque effettuare un ulteriore confronto con dei risultati effettuati da terzi reperiti on-line nel settore della stima della qualità del prodotto. Nello specifico il lavoro condotto dal team di RTINGS, sugli auricolari Nothing Ear 2, basava le sue misurazioni su test svolti con l'impiego di suoni diffusi di natura strettamente sinusoidale. Di conseguenza questo tipo di analisi si basa sulla risposta in frequenza, differentemente da quella proposta nel presente elaborato che effettua un più realistico studio dello spettro delle frequenze all'applicazione di un rumore bianco generico.

Infatti, la tecnologia ANC restituisce le migliori prestazioni quando il suono da cancellare segue un andamento prevedibile e il più possibile costante, in queste condizioni chiaramente una sinusoide rappresenta il caso migliore in cui una cuffia ANC può operare.

Ben diverso è il comportamento delle cuffie nelle nostre prove, in quanto i rumori da sopprimere si rivelano essere sicuramente più complessi da gestire rispetto ad uno sweep sinusoidale; a tutto ciò va aggiunto che gli strumenti impiegati per la realizzazione del progetto non sono comparabili a quelli utilizzati dal team di RTINGS. In particolar modo, un microfono specifico per il test di auricolari e cuffie sarebbe stato troppo costoso, d'altra parte la più elevata sensibilità ci avrebbe permesso di ridurre il guadagno e una più determinata direzionalità ci

avrebbe consentito di osservare in maniera più netta la soppressione dovuta all'isolamento passivo, consentendoci di conseguenza di registrare soppressioni, in generale, più elevate.

Sottolineando che le grandezze protagoniste dell'elaborato sono difficili da misurare con strumenti di uso comune e a basso costo, ci riteniamo comunque soddisfatti del lavoro svolto in quanto, nonostante la difficoltà, siamo riusciti a visualizzare in maniera grafica come l'ANC riesca ad attenuare il rumore ambientale e come i risultati ottenuti siano comparabili a quelli ottenuti da RTINGS con attrezzature di gran lunga più costose.

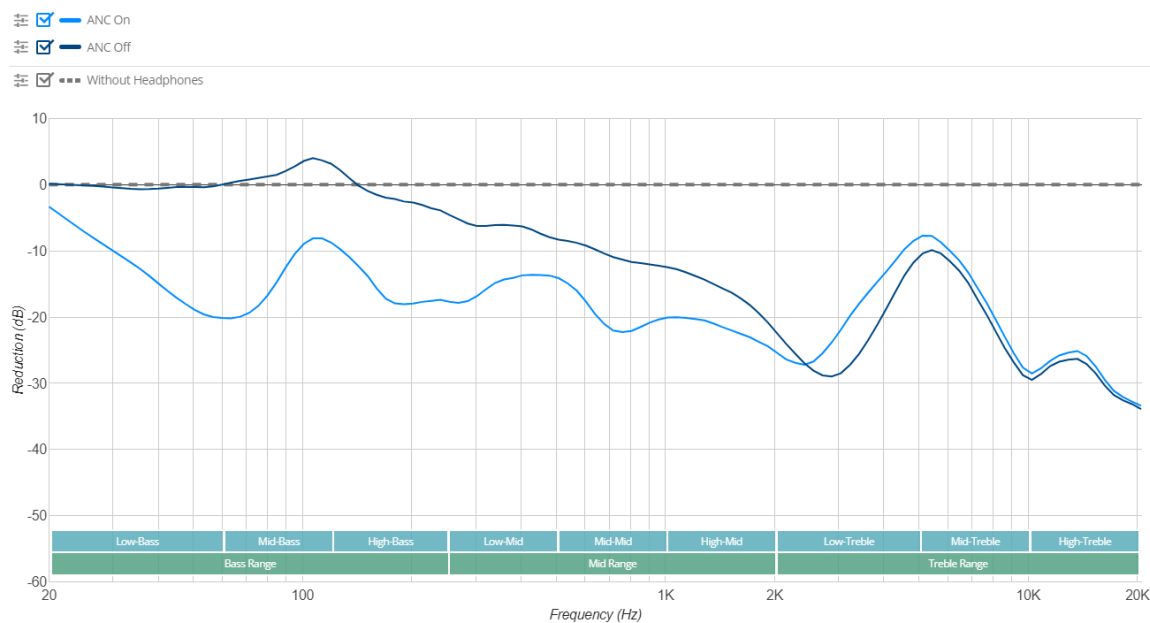


Figura 4.1: Risposta in frequenza svolta dal team di RTINGS delle Nothing Ear 2

In figura viene riportato il risultato di RTINGS dove è possibile apprezzare un'attenuazione alle medie frequenze di circa 18 dB rispetto a quella vista precedentemente nel corso dell'elaborato dove è stato ottenuto un risultato di 15 dB (con rumore di fondo a 0 dB).

CAPITOLO

5 | **Reference**

RTINGS-Nothing Ear (2) Truly Wireless

URL <https://www.rtings.com/headphones/reviews/nothing/ear-2-truly-wireless>