

Istruzioni – elaborazione Bode da CSV con GNU Octave

Lo script `bode_batch_macro_armoniche.m` serve a elaborare automaticamente una serie di acquisizioni esportate dall'oscilloscopio, ricavando per ogni frequenza **modulo e fase** del rapporto tra i due canali.

La convenzione utilizzata è:

- **CH1 = VSENSE**
- **CH2 = VOUT**
- **$H = CH2 / CH1 = VOUT / VSENSE$**

Lo script può inoltre gestire una seconda serie di acquisizioni, indicata come **B**, ottenuta invertendo i terminali del secondario del trasformatore di iniezione.

Se la misura B è disponibile, nella versione attuale viene calcolata la media tra A e B; se non è disponibile viene utilizzata solamente la misura A.

Nota importante

La combinazione A/B fa parte della procedura sperimentale che sto attualmente caratterizzando e non deve essere considerata una correzione universalmente valida o una calibrazione definitiva dello strumento.

1. File necessari

Per utilizzare lo script è consigliabile creare una cartella contenente:

```
bode_batch_macro_armoniche.m
tabella_bode.csv
bode.txt
30k_A.csv
50k_A.csv
80k_A.csv
...
30k_B.csv
50k_B.csv
80k_B.csv
...
```

I file della serie B sono **opzionali**.

Il file `bode.txt` contiene invece il diagramma esportato da LTspice ed è necessario solamente se si desidera effettuare il confronto con la simulazione.

Per semplicità conviene tenere tabella, acquisizioni e file LTspice nella stessa cartella. Lo script accetta comunque anche percorsi relativi o assoluti.

2. Preparazione della tabella delle misure

Il file `tabella_bode.csv` associa a ogni frequenza i corrispondenti file di acquisizione.

Esempio:

```
frequenza,file_A,file_B
30kHz,30k_A.csv,30k_B.csv
50kHz,50k_A.csv,50k_B.csv
80kHz,80k_A.csv,80k_B.csv
100kHz,100k_A.csv,100k_B.csv
120kHz,120k_A.csv,120k_B.csv
150kHz,150k_A.csv,150k_B.csv
```

La misura **A è obbligatoria**, mentre il campo relativo alla misura B può essere lasciato vuoto:

```
80kHz,80k_A.csv,
```

In questo caso, per quel punto verrà utilizzata solamente la misura A.

Lo script ordina automaticamente i punti in frequenza.

Le frequenze possono essere indicate, per esempio, come:

```
1000
1000Hz
30kHz
30k
1MHz
1M
```

È importante che la frequenza riportata nella tabella corrisponda a quella effettivamente utilizzata durante l'acquisizione, perché il fit viene eseguito a **frequenza nota**.

3. Formato dei CSV dell'oscilloscopio

La routine di importazione è stata scritta per i file CSV generati dagli **oscilloscopi Rigol** che utilizzo.

Lo script legge dall'header i parametri temporali `Start` e `Increment` e successivamente importa, per ciascun campione:

```
indice,CH1,CH2
```

Da indice e incremento temporale viene ricostruito l'asse dei tempi.

Oscilloscopi di altri produttori possono utilizzare header, separatori o organizzazioni differenti e quindi **non è garantita la compatibilità diretta**.

È inoltre consigliabile non modificare o ris salvare i CSV con programmi che possano cambiare automaticamente separatori o formato dei numeri.

Chi volesse utilizzare lo script con un oscilloscopio differente può inviarmi **l'header e alcune righe di esempio** di un CSV prodotto dal proprio strumento. Se il formato è sufficientemente semplice, posso provare ad adattare la routine di importazione.

4. Parametri principali dello script

Nella parte finale di `bode_batch_macro_armoniche.m` sono raccolti i parametri che normalmente è necessario modificare.

Tabella delle acquisizioni

```
file_tabella = "tabella_bode.csv";
```

Indica il file contenente frequenze e nomi dei CSV.

File LTspice

```
file_ltspice = "bode.txt";
```

Indica il file contenente il diagramma esportato da LTspice.

Se non si desidera effettuare il confronto con LTspice:

```
file_ltspice = "";
```

Nome della prova

```
nome_prova = "prova_bode";
```

Questo nome viene utilizzato come prefisso per i file generati.

Per esempio:

```
nome_prova = "R3k3";
```

produrrà file del tipo:

```
R3k3_risultato.txt  
R3k3_misure_A.txt  
R3k3_misure_B.txt  
R3k3_misure_finali.txt  
R3k3_modulo.png  
R3k3_fase.png
```

Cartella dei risultati

```
cartella_output = "output_bode";
```

Se la cartella non esiste viene creata automaticamente.

5. Numero di cicli da analizzare

Il parametro:

```
numero_cicli = 0;
```

stabilisce quanta parte dell'acquisizione viene utilizzata.

Con:

```
numero_cicli = 0;
```

viene analizzato **l'intero CSV**.

Impostando invece, per esempio:

```
numero_cicli = 100;
```

vengono letti solamente i campioni necessari a coprire approssimativamente i primi 100 cicli alla frequenza specificata.

Questa possibilità può essere utile con acquisizioni molto lunghe, perché i CSV esportati dall'oscilloscopio possono raggiungere dimensioni considerevoli.

6. Fit della fondamentale e delle armoniche

La versione attuale permette di includere nel fit anche le armoniche del segnale.

I parametri sono:

```
usa_armoniche_fit = true;  
numero_armoniche_fit = 3;
```

Con questa impostazione vengono considerate:

- fondamentale;
- seconda armonica;
- terza armonica;
- componente continua.

Impostando, per esempio:

```
numero_armoniche_fit = 5;
```

il modello comprenderà la fondamentale e le armoniche dalla seconda alla quinta.

Per disabilitare le armoniche:

```
usa_armoniche_fit = false;
```

In questo caso il fit contiene solamente fondamentale e offset.

Il diagramma di Bode viene comunque calcolato sempre e soltanto dalla fondamentale.

Le armoniche aggiuntive servono a descrivere meglio i segnali acquisiti e a limitare la contaminazione della stima della fondamentale, in particolare quando la finestra analizzata non contiene un numero intero di periodi.

Nel report vengono inoltre riportati i livelli delle armoniche considerate, il THD calcolato sulle armoniche incluse e il residuo RMS del fit.

7. Misure A e B

La procedura che utilizzo attualmente prevede:

1. effettuare l'intera serie di frequenze con il secondario del trasformatore collegato in un verso;
2. salvare queste acquisizioni come serie **A**;
3. invertire fisicamente i due terminali del secondario;
4. ripetere la stessa serie di frequenze;
5. salvare le nuove acquisizioni come serie **B**.

Lo script effettua l'unwrap delle fasi delle due serie e, dove sono presenti entrambe le acquisizioni, calcola attualmente:

```
modulo finale = (modulo A + modulo B) / 2  
fase finale   = (fase A + fase B) / 2
```

La media del modulo viene effettuata direttamente in dB, mentre quella della fase viene eseguita dopo l'unwrap e l'allineamento delle due serie.

Se B manca, viene utilizzata A.

Nota

Questa procedura è ancora oggetto di caratterizzazione sperimentale. In alcune condizioni ha prodotto risultati molto buoni, ma non deve essere considerata una correzione definitiva valida per qualsiasi impedenza o frequenza.

8. Confronto con LTspice

Se viene specificato `file_ltspice`, lo script importa la curva esportata dal simulatore.

Il parser cerca righe contenenti dati nel formato prodotto dall'esportazione testuale di LTspice, riconducibili a:

```
frequenza      (modulo_dB, fase)
```

La curva LTspice viene poi confrontata con le frequenze effettivamente misurate.

Poiché i punti della simulazione e quelli sperimentali normalmente non coincidono, i valori simulati vengono interpolati alle frequenze della misura.

Lo script può quindi confrontare direttamente:

- modulo misurato;
- modulo LTspice;
- errore in dB;
- fase misurata;
- fase LTspice;
- errore in gradi.

9. Fit opzionale poli/zeri

Lo script contiene anche una funzione sperimentale per il fit razionale della risposta:

```
esegui_fit_pz = true;  
numero_poli_fit = 3;  
numero_zeri_fit = 1;  
mostra_fit_pz = true;
```

Se non interessa questa elaborazione è possibile utilizzare:

```
esegui_fit_pz = false;
```

Il fit poli/zeri è un'elaborazione aggiuntiva e **non è necessario per ottenere modulo, fase e confronto con LTspice**.

10. Avvio dello script

Avviare GNU Octave e portarsi nella cartella contenente lo script e la tabella.

Per esempio, sotto Windows:

```
cd('C:\percorso\della\cartella')
```

quindi eseguire:

```
bode_batch_macro_armoniche
```

Lo script analizzerà in sequenza tutti i file indicati in `tabella_bode.csv`.

11. File generati

Al termine vengono prodotti nella cartella di output:

```
<nome_prova>_risultato.txt  
<nome_prova>_misure_A.txt  
<nome_prova>_misure_B.txt  
<nome_prova>_misure_finali.txt  
<nome_prova>_modulo.png  
<nome_prova>_fase.png
```

Il report completo contiene, per ciascuna acquisizione:

- frequenza;
- incremento temporale;
- numero di campioni utilizzati;
- durata analizzata;
- numero di cicli effettivi;
- ampiezza della fondamentale di CH1 e CH2;
- armoniche incluse nel fit;
- THD;
- residuo RMS del fit;
- offset dei due canali;
- rapporto CH2/CH1;
- modulo in dB;
- fase in gradi.

I tre file delle misure contengono invece una forma compatta:

```
frequenza, modulo, fase
```

utile per eventuali elaborazioni successive.

12. Note e limiti

Questo software nasce come **strumento di laboratorio personale** e viene pubblicato principalmente per permettere di riprodurre o modificare la procedura descritta nell'articolo.

In particolare:

- l'importazione CSV è attualmente specifica per il formato Rigol utilizzato durante le mie prove;
- è necessario rispettare la convenzione **CH1 = VSENSE** e **CH2 = VOUT**, oppure modificare opportunamente lo script;
- la frequenza indicata nella tabella deve corrispondere a quella utilizzata durante l'acquisizione;

- la procedura A/B è ancora in fase di caratterizzazione;
- il fit poli/zeri è opzionale e non è necessario per la misura di base;
- i risultati prodotti dallo script non sostituiscono la verifica che il circuito e la catena di misura stiano lavorando in condizioni lineari e prive di saturazione o distorsione significativa.

Non vengono forniti i CSV utilizzati nelle mie prove perché possono essere molto grandi: con le impostazioni che utilizzo **un singolo file può occupare circa 16 MB** e una caratterizzazione completa può richiedere alcune decine di acquisizioni.

Per ripetere la procedura sono sufficienti lo script, una tabella `tabella_bode.csv` correttamente compilata e le acquisizioni esportate dal proprio oscilloscopio.