

GX-TXT

Manuale operativo dei software GNU Octave per l'analisi dinamica di HOLD e LED

GXTXT_singola_v6 - GXTXT_batch_v7 - GXTXT_batch_impulsi_v8 - GXTXT_analisi_globale_v1

Flusso operativo consigliato



La guida segue il flusso operativo: acquisizione, importazione, analisi, verifica degli output e confronto globale.

Scopo della guida - permettere di utilizzare e adattare i programmi senza dover studiare gli algoritmi interni. Vengono descritti preparazione dei dati, configurazione, avvio, struttura degli output, compatibilità tra versioni e problemi pratici più comuni.

Versione documento 1.0 - 28 agosto 2026

Progetto GX-TXT-v3 - software GNU Octave

Indice

1. Panoramica e scelta del programma
2. Requisiti e preparazione dell'ambiente
3. Segnali, nomi e mappatura del proprio circuito
4. Preparare i CSV dell'oscilloscopio
5. Uso di GXTXT_singola_v6
6. Uso di GXTXT_batch_v7
7. Uso di GXTXT_batch_impulsi_v8
8. Capire e controllare gli output
9. Uso di GXTXT_analisi_globale_v1
10. Adattamento rapido a un altro circuito o oscilloscopio
11. Risoluzione dei problemi
12. Checklist operativa e riferimenti

1. Panoramica e scelta del programma

La raccolta comprende strumenti con ruoli diversi. Non è necessario usarli tutti: la scelta dipende da quante acquisizioni si devono elaborare e dal tipo di confronto finale desiderato.

Programma	Quando usarlo	Ingresso principale	Output principale
GXTXT_singola_v6	Una misura alla volta; messa a punto e controllo manuale	1 CSV Rigol + metadati nello script	Cartella prova con risultati, tabella HOLD e 9 PNG
GXTXT_batch_v7	Campagna HOLD/LED con struttura storica v7	prove.txt + molti CSV	risultati_batch.csv + cartelle per prova + log
GXTXT_batch_impulsi_v8	Campagna con numero di impulsi variabile e V_RSENSE per prova	prove_impulsi.csv + molti CSV	Master arricchito, stato batch, cartelle per prova, log
GXTXT_analisi_globale_v1	Confronto globale della campagna v7	risultati_batch.csv prodotto dalla v7	3 grafici globali + 3 CSV dati + log

Compatibilità importante - GXTXT_analisi_globale_v1 è stato scritto per il master della v7 e cerca esplicitamente la colonna vgen_V. L'output della v8 non va quindi passato direttamente a questa versione dell'analisi globale.

Per una nuova campagna conviene partire da una singola acquisizione, verificare che i canali siano mappati correttamente, poi passare al batch. In questo modo eventuali errori di formato o di assegnazione dei segnali vengono scoperti prima di elaborare decine di file.

2. Requisiti e preparazione dell'ambiente

2.1 Software necessario

- GNU Octave. Gli script non dichiarano nel codice una versione minima specifica; è consigliabile utilizzare una release recente e stabile.
- I file .m pubblicati nella pagina software.
- I CSV esportati dall'oscilloscopio nel formato atteso oppure adattati tramite il parser.
- Spazio libero sufficiente per CSV, cartelle di output, log e grafici PNG.

La documentazione ufficiale di GNU Octave ricorda che le funzioni vengono cercate nel current directory e nel load path. Per questo il metodo più semplice è lavorare in una cartella dedicata e impostarla come directory corrente prima di eseguire i programmi.

2.2 Struttura consigliata della cartella di lavoro

```
GXTXT_analisi/  
  GXTXT_singola_v6.m  
  GXTXT_batch_v7.m  
  GXTXT_batch_impulsi_v8.m  
  GXTXT_analisi_globale_v1.m  
  prove.txt  
  prove_impulsi.csv  
  acquisizioni/  
    misura_001.csv  
    misura_002.csv  
  ...
```

I batch cercano prima il nome del CSV così come scritto nell'indice; se non lo trovano, provano a risolverlo rispetto alla cartella che contiene il file indice. Per ridurre gli errori conviene mantenere indice e acquisizioni nella stessa directory oppure usare percorsi relativi semplici.

2.3 Nome del file e nome della funzione

Per i file che iniziano con function è buona pratica far coincidere il nome del file con il nome della prima funzione. GNU Octave cerca infatti i function file usando il nome della funzione come nome base del file.

Funzione	Nome file consigliato
GXTXT_batch_v7()	GXTXT_batch_v7.m
GXTXT_batch_impulsi_v8()	GXTXT_batch_impulsi_v8.m
GXTXT_analisi_globale_v1()	GXTXT_analisi_globale_v1.m

Nota sulla v8 - se si parte dal file storico GXTXT_batch_impulsi_v8_CH3_CH4.m, rinominarlo GXTXT_batch_impulsi_v8.m rende l'avvio diretto coerente con la funzione contenuta nel file. Il suffisso CH3_CH4 può essere conservato nella documentazione o nel changelog.

2.4 Avvio rapido da Octave

```
cd 'percorso/della/cartella/GXTXT_analisi'  
  
GXTXT_singola_v6  
% oppure  
GXTXT_batch_v7  
% oppure  
GXTXT_batch_impulsi_v8  
% oppure  
GXTXT_analisi_globale_v1
```

GXTXT_singola_v6 è uno script; gli altri tre file principali qui descritti sono function file. In entrambi i casi, con la cartella corretta selezionata, si avviano digitando il nome base senza estensione .m.

3. Segnali, nomi e mappatura del proprio circuito

I nomi presenti nei programmi descrivono il circuito GX-TXT, ma non sono un requisito elettrico universale. Per riutilizzare gli script su un altro circuito bisogna identificare segnali equivalenti e assegnarli alle variabili attese.

Nome usato	Ruolo operativo	Cosa deve fornire il proprio circuito
V_HOLD / vHold	Tensione del nodo di mantenimento	Un segnale che descrive la carica/scarica o la grandezza di stato da correlare al LED
N_LED_ARANCIO / vLed	Nodo da cui viene ricavata la corrente LED	La tensione sul lato del resistore serie coerente con la formula usata nello script
VCC	Riferimento del ramo LED	Nello script viene stimato dal valore pretrigger di vLed
RLED	Resistenza serie del LED	Deve essere impostata al valore reale del circuito
P_GAIN	Regolazione di guadagno	Metadato; può rappresentare una diversa regolazione equivalente
P_SENS	Regolazione di sensibilità	Metadato e parametro di raggruppamento
V_RSENSE / RSENSE	Misura dello stimolo nella v8	Opzionale come concetto generale, ma necessario se si vuole usare isense_uA come previsto dalla v8
numero_impulsi	Durata discreta dello stimolo	Numero di periodi/burst associato alla prova nella v8

Regola pratica - prima di modificare criteri o parametri di elaborazione, adattare soltanto tre cose: colonne del CSV, RLED/RSENSE e metadati. Se i grafici risultano coerenti, lasciare invariati gli altri parametri per mantenere confrontabili le prove.

3.1 Mappatura minima

```
% Esempio concettuale: il proprio oscilloscopio esporta
% colonna 2 = LED_SENSE
% colonna 3 = CAP_HOLD

seq = dati(:,1);
vLed = dati(:,2); % LED_SENSE -> vLed
vHold = dati(:,3); % CAP_HOLD -> vHold
```

Non è necessario rinominare i nodi nel CAD o nel circuito. Basta mappare correttamente le colonne lette dal CSV alle variabili vLed e vHold.

4. Preparare i CSV dell'oscilloscopio

Il parser originale è stato scritto per CSV Rigol con due righe iniziali di intestazione. Dalla seconda riga vengono letti Start e Increment; i campioni successivi sono interpretati come Sequence più due canali analogici. Le acquisizioni v6/v7 e v8 usano un ordine dei canali differente.

Versione	Seconda colonna dati	Terza colonna dati
v6 / v7	V_HOLD (CH2 nelle misure originali)	N_LED_ARANCIO (CH3)

v8	N_LED_ARANCIO (CH3)	V_HOLD (CH4)
----	---------------------	--------------

4.1 Blocchi da controllare se il CSV è diverso

Per un oscilloscopio differente, in genere è sufficiente intervenire nella zona che apre il file, ricava il passo temporale e assegna le colonne. Il resto del programma può rimanere invariato.

```

riga1 = fgetl(fid);
riga2 = fgetl(fid);

campi = strsplit(riga2, ',');
tStart = str2double(campi{4});
dt      = str2double(campi{5});

dati = fscanf(fid, '%f,%f,%f,', [3, Inf]);

seq  = dati(:,1);
vLed  = dati(:,2);
vHold = dati(:,3);

t = tStart + seq .* dt;

```

Se il proprio strumento esporta direttamente una colonna tempo, si può costruire t da quella colonna e ignorare Start/Increment. Se usa un separatore diverso, va modificato sia strsplit sia il formato fscanf. Se esporta più canali, si devono leggere tutte le colonne necessarie e poi scegliere quelle corrette.

4.2 Controlli da fare prima del batch

1. Aprire il CSV con un editor di testo e verificare quante righe di header precedono i campioni.
2. Controllare il separatore: virgola, punto e virgola o tabulazione.
3. Verificare se il tempo è esplicito oppure ricavato da Start e Increment.
4. Verificare l'ordine reale dei canali.
5. Usare il punto come separatore decimale nei file indice, come negli esempi forniti.
6. Eseguire prima una singola prova e controllare visivamente 01_hold_fit.png e 02_n_led_completo.png.

Acquisizione consigliata - includere un tratto pretrigger stabile. I programmi stimano i livelli di riferimento dal pretrigger; se questo manca, alcune versioni usano una porzione iniziale della traccia come ripiego. Una finestra con qualche margine prima dello stimolo rende più robusto l'uso pratico.

5. Uso di GXTXT_singola_v6

La v6 è il punto di partenza più comodo per verificare una nuova acquisizione o adattare il parser. I parametri da cambiare a ogni misura sono raccolti all'inizio del file.

5.1 Parametri da impostare

```

filename = 'time_led.csv';

testID = 'burst100_vgen305_gain5_sens9432';
RGAIN_ohm = 5.0;
RSENS_ohm = 9432.0;

frequenza_Hz = 38000;

```

```

numeroImpulsi = 100;
VGEN_V = 3.05;
dutyCycle_pct = 50.0;
IPD_uA = NaN;

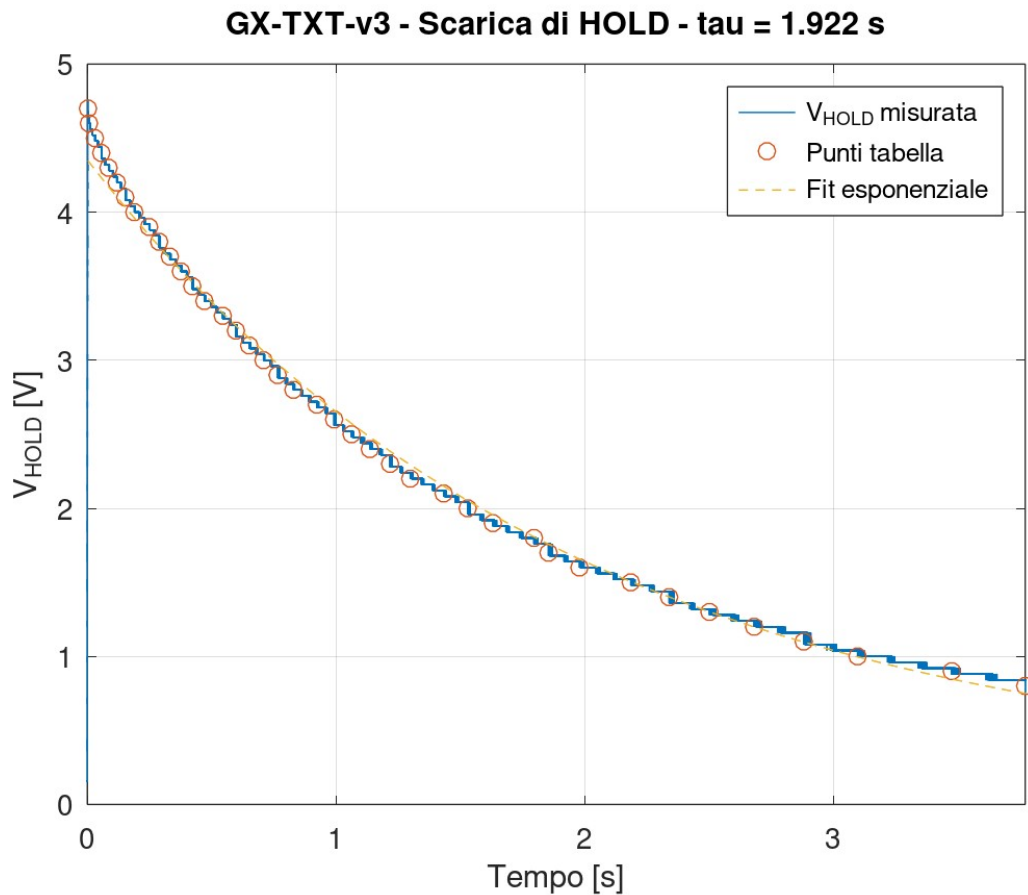
outputRoot = 'calibrazione script';
RLED = 680;

```

testID diventa il nome della sottocartella della prova. Conviene usare un identificatore corto ma descrittivo e non riutilizzarlo per acquisizioni diverse, in modo da evitare confusione tra risultati.

5.2 Procedura operativa

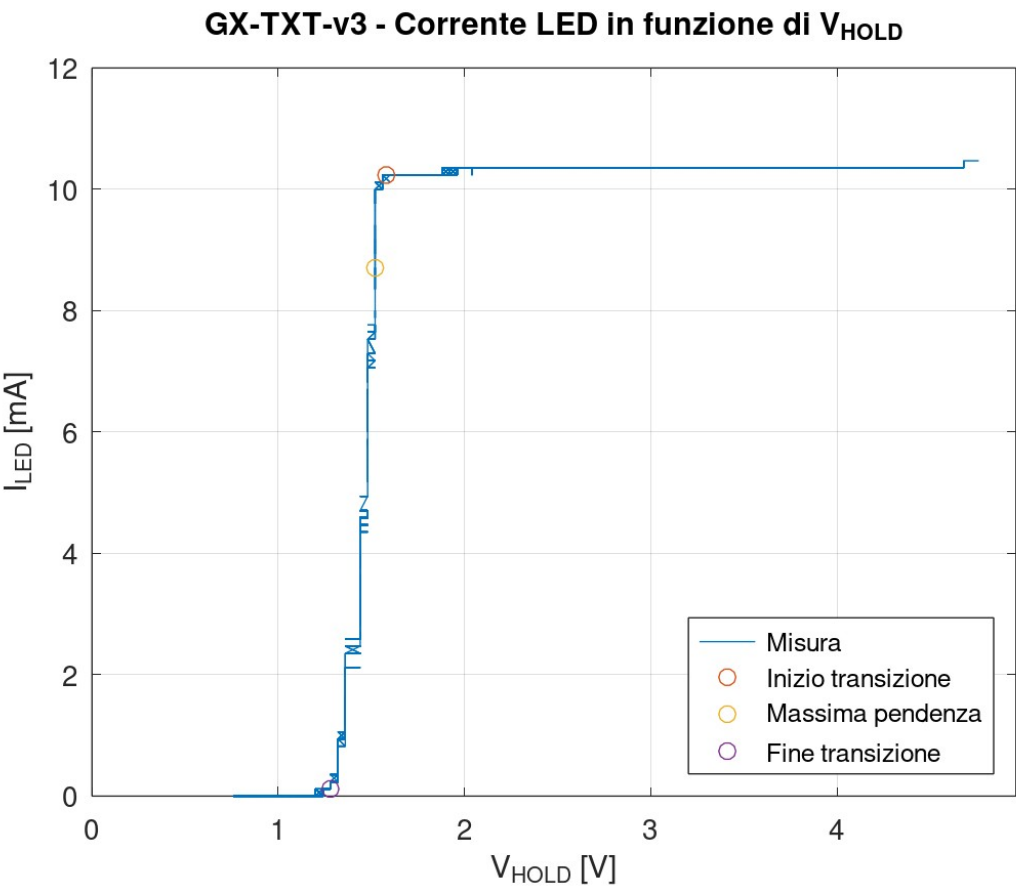
1. Copiare il CSV nella cartella di lavoro oppure indicarne il percorso in filename.
2. Impostare testID e i metadati reali della prova.
3. Controllare RLED e, se si è modificato l'hardware, i parametri elettrici iniziali.
4. Avviare GXTXT_singola_v6.
5. Leggere i messaggi a console e verificare che le prime righe siano coerenti con i valori attesi.
6. Aprire i PNG generati e verificare forma di HOLD e segnale LED.
7. Controllare risultati.txt e risultati.csv prima di aggiungere la prova a una campagna.



Esempio di output 01_hold_fit.png. Per l'uso del software interessa soprattutto verificare che il segnale HOLD letto dal CSV sia quello corretto e che la finestra temporale contenga la scarica attesa.

5.3 Output della singola prova

File	Uso pratico
metadati.txt	Registro delle condizioni impostate per la prova.
risultati.txt	Riepilogo leggibile dei risultati numerici.
risultati.csv	Riga strutturata della singola acquisizione.
tabella_hold.csv	Dati HOLD/LED campionati ai livelli usati dal programma; utile per elaborazioni successive.
GXTXT_risultati_completi.csv	Database cumulativo aggiornato dalle esecuzioni della v6.
01...09 PNG	Controllo visivo dell'acquisizione e dei risultati.



Esempio di grafico della singola acquisizione: relazione tra corrente LED ricavata e tensione HOLD. E' utile come controllo di plausibilit  prima di automatizzare una campagna.

6. Uso di GXTXT_batch_v7

La v7 automatizza l'analisi di molte acquisizioni con parametri comuni di campagna. E la versione da utilizzare quando si vuole produrre un master compatibile con GXTXT_analisi_globale_v1.

6.1 Configurazione comune

```
indexFile = 'prove.txt';
outputRoot = 'calibrazione script';

cfg.RLED = 680;
cfg.frequenza_Hz = 38000;
cfg.numeroImpulsi = 100;
cfg.dutyCycle_pct = 50;
```

Questi valori valgono per tutta la campagna v7. Se il numero di impulsi cambia da prova a prova, usare la v8 invece di duplicare batch differenti salvo esigenze specifiche.

6.2 Formato del file prove.txt

```
p_gain,p_sens,vgen,nome_file_csv
500,9459,1,500R_10000R_time_led_1.csv
500,9459,2,500R_10000R_time_led_2.csv
500,7527,1,500R_7500R_time_led_1.csv
```

- p_gain e p_sens sono espressi in ohm.
- vgen e il livello di generatore usato dalla campagna v7.
- nome_file_csv può essere un nome nella stessa cartella dell'indice oppure un percorso risolvibile.
- Le righe vuote vengono ignorate e le righe che iniziano con # possono essere usate come commenti.
- Il parser dell'indice accetta virgola oppure punto e virgola, scegliendo il delimitatore dall'header.

6.3 Esecuzione e controllo

1. Preparare prove.txt e controllare almeno due righe a mano.
2. Verificare che tutti i CSV esistano e che abbiano lo stesso ordine di canali previsto dalla v7.
3. Avviare GXTXT_batch_v7.
4. A fine esecuzione aprire stato_batch.csv: consente di individuare rapidamente eventuali prove fallite.
5. Aprire batch.log se una prova ha dato errore o se i risultati sembrano anomali.
6. Controllare risultati_batch.csv: e il master che verrà usato dall'analisi globale.

Buona pratica - non cambiare i parametri di elaborazione a metà campagna. Se è necessario correggerli, rieseguire l'intero batch in una nuova cartella timestampata così da mantenere una storia riproducibile.

7. Uso di GXTXT_batch_impulsi_v8

La v8 mantiene il flusso generale della v7 ma è stata preparata per campagne in cui il numero di impulsi cambia per ogni prova e in cui si registra anche V_RSENSE. Inoltre produce un master più ricco e gestisce in modo più esplicito alcune condizioni sperimentali non patologiche, come LED non acceso o fit HOLD non disponibile.

7.1 Configurazione comune

```
indexFile = 'prove_impulsi.csv';
outputRoot = 'calibrazione script';

cfg.RLED = 680;
cfg.RSENSE = 51e3;
cfg.frequenza_Hz = 38000;
cfg.dutyCycle_pct = 50;
```

RLED e RSENSE devono essere adeguati all'hardware reale. Il numero di impulsi non è più globale: viene letto dall'indice per ogni acquisizione.

7.2 Formato di prove_impulsi.csv

```
p_gain,p_sens,vgen_vpp,v_rsense_vpp,n_impulsi,nome_file_csv
500,5048,2.10,1.02,20,time_led_N020.csv
500,5048,2.10,1.02,40,time_led_N040.csv
500,5048,2.10,1.02,60,time_led_N060.csv
500,5048,2.10,1.02,100,time_led_N100.csv
```

Campo	Significato pratico
p_gain	Regolazione di guadagno associata alla prova.
p_sens	Regolazione di sensibilità associata alla prova.
vgen_vpp	Ampiezza nominale del generatore, Vpp.
v_rsense_vpp	Ampiezza misurata su RSENSE, Vpp.
n_impulsi	Numero di impulsi del burst della singola prova.
nome_file_csv	CSV da elaborare.

7.3 Cosa controllare nel master v8

Oltre ai campi già noti, la v8 aggiunge v_rsense_vpp, isense_uA e holdFitValid. Per l'uso pratico sono utili per distinguere una prova valida senza fit da un vero errore e per associare direttamente lo stimolo misurato alla singola acquisizione.

Campo	Interpretazione operativa
holdFitValid	1/true: fit disponibile; 0/false: prova elaborata ma fit HOLD non valido o non disponibile.
ledTransitionValid	Indica se è stata riconosciuta una transizione LED utilizzabile.
isense_uA	Corrente di stimolo ricavata da V_RSENSE e RSENSE.
status in stato_batch.csv	OK, OK_PLOT_ERROR oppure ERROR_ANALYSIS nella v8.

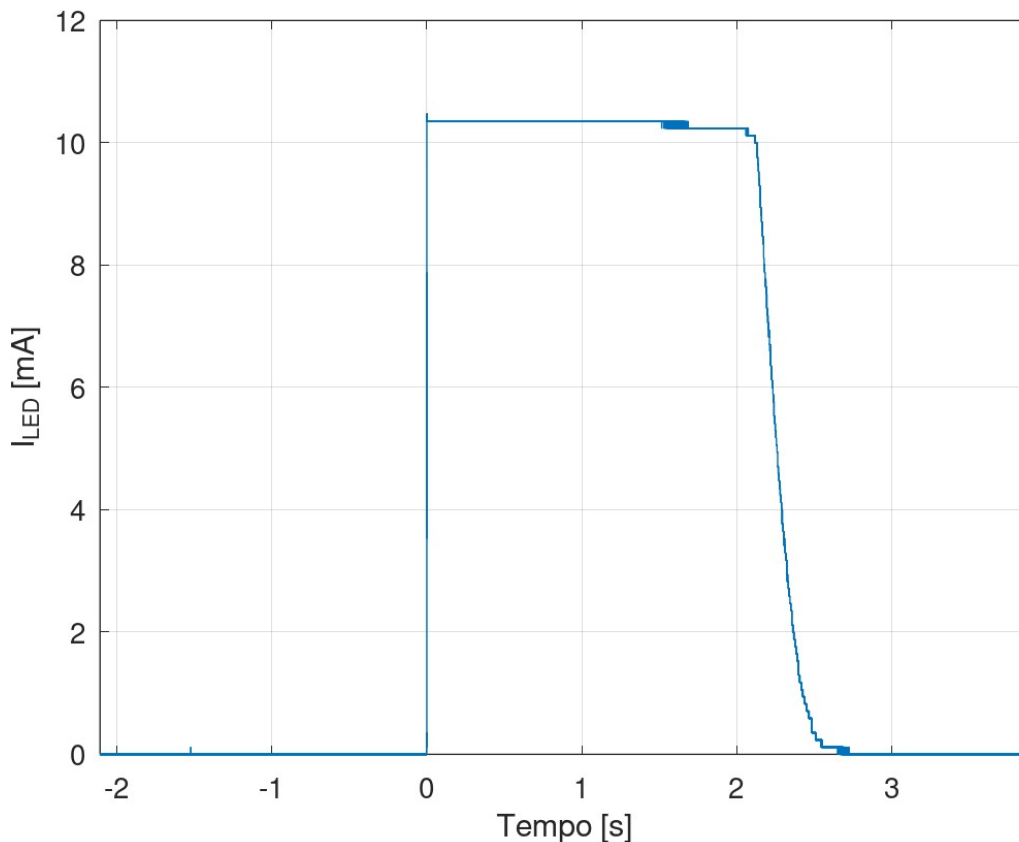
8. Capire e controllare gli output

Gli output v6, v7 e v8 sono pensati per due livelli di controllo: file numerici per elaborazioni successive e grafici per verificare rapidamente che l'acquisizione sia stata interpretata nel modo atteso. Non è necessario analizzare ogni PNG in dettaglio, ma alcuni controlli visivi sono molto utili.

PNG	Controllo pratico consigliato
01_hold_fit.png	La traccia HOLD è corretta e copre una porzione sufficiente della dinamica.
02_n_led_completo.png	Il canale assegnato a vLed è realmente il nodo LED e non è scambiato con HOLD.

03_accensione_led_zoom.png	La finestra attorno alla transizione contiene dati e non è tagliata ai bordi.
04_corrente_led_tempo.png	La corrente ricavata ha scala e segno plausibili rispetto a RLED e VCC.
05_derivata_led_transizione.png	Serve soprattutto a diagnosticare transizioni ambigue; non va interpretato come un dato elettrico autonomo.
06-08 HOLD	Controlli temporali della carica/scarica; utili se il risultato numerico sembra anomalo.
09_corrente_led_vs_hold.png	Controllo sintetico della relazione tra i due segnali della prova.

GX-TXT-v3 - Corrente del LED arancio nel tempo



Esempio di 04_corrente_led_tempo.png. Una scala incompatibile con il circuito e spesso il primo indizio di RLED errata, canale invertito o livello di riferimento non corretto.

8.1 File da conservare sempre

- CSV originale dell'oscilloscopio: e il dato grezzo da cui tutto può essere rigenerato.
- File indice realmente usato: il batch ne salva una copia come indice_prove.csv.
- batch.log e stato_batch.csv: servono per ricostruire cosa e successo durante l'elaborazione.
- risultati_batch.csv: master della campagna.
- Per l'analisi globale v1, conservare anche le cartelle delle singole prove con tabella_hold.csv.

9. Uso di GXTXT_analisi_globale_v1

L'analisi globale v1 è un programma successivo al batch v7. Non legge direttamente i CSV dell'oscilloscopio: parte dal master risultati_batch.csv e, per alcune elaborazioni, dalle tabella_hold.csv presenti nelle cartelle delle singole prove.

9.1 Impostazioni da verificare prima dell'avvio

```
batchDir = fullfile( ...  
    'calibrazione script', ...  
    'batch_20260823_181139');  
  
masterFile = fullfile(batchDir, 'risultati_batch.csv');  
outputDir = fullfile(batchDir, 'analisi_globale');
```

Il primo intervento è sostituire batch_20260823_181139 con il batch reale che si vuole analizzare.

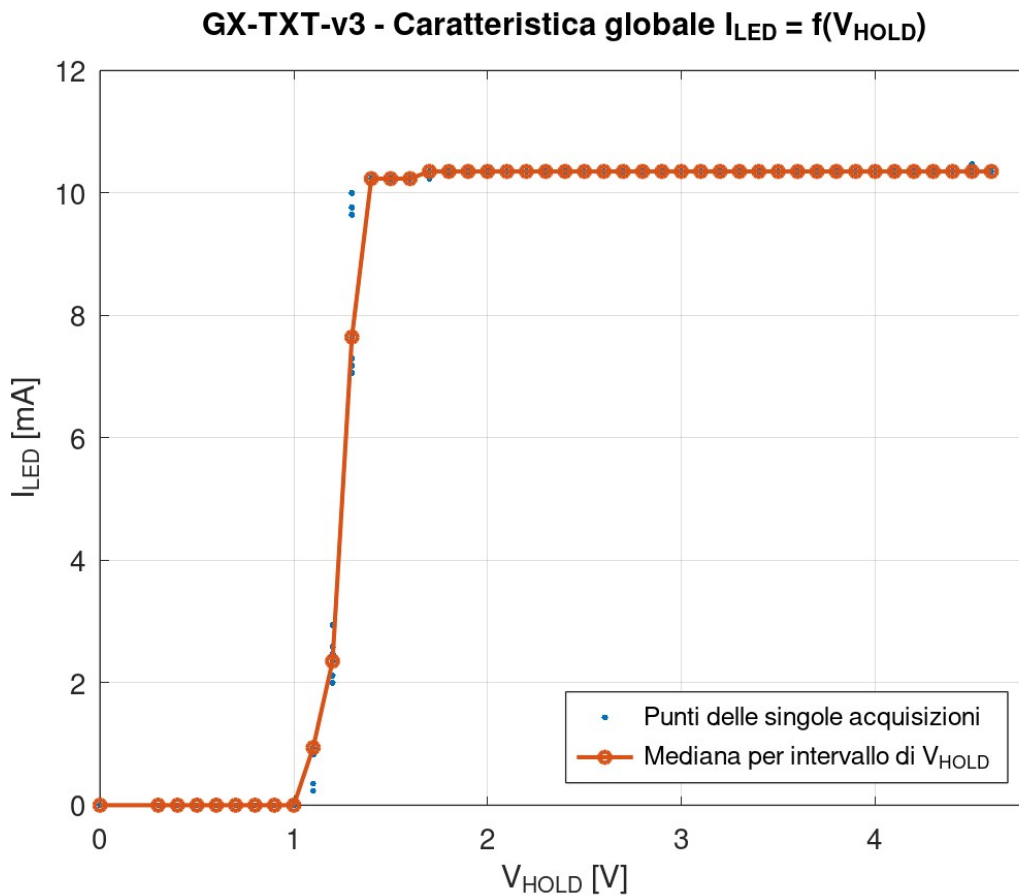
9.2 Parametri specifici della campagna originale da rivedere

Blocco	Quando modificarlo
vgenLevels / isenseLevels_uA	Se la corrispondenza tra livello VGEN e corrente di stimolo non coincide con quella della campagna originale.
psensSeries	Se si vogliono confrontare valori di P_SENS diversi da quelli elencati nello script.
extraPoints	Se i punti manuali previsti dalla campagna originale non appartengono ai propri dati; possono essere aggiornati o rimossi.
holdBin_V	Solo se si vuole cambiare la granularità dei dati globali; per confronti ripetibili è preferibile non variarlo tra campagne.

Compatibilità v7 - il parser del master cerca le colonne testID, p_gain_ohm, p_sens_ohm, vgen_V, VholdMax_V, ledTransitionValid e tempoLedAcceso_s. La colonna vgen_V è la ragione principale per cui il master v8 non è intercambiabile senza modifica del programma.

9.3 Output globale

File	Contenuto
10_iled_vs_vhold_globale.png	Caratteristica globale I_LED in funzione di V_HOLD.
11_tled_vs_vholdmax_globale.png	Tempo LED in funzione del massimo di HOLD.
12_tled_vs_isense_psens.png	Famiglie di T_LED rispetto a I_SENSE per differenti P_SENS.
dati_iled_vhold_globali.csv	Dati numerici del grafico 10.
dati_tled_vholdmax.csv	Dati numerici del grafico 11.
dati_tled_isense.csv	Dati numerici del grafico 12.
analisi_globale.log / .txt	Traccia dell'esecuzione e riepilogo.



Esempio di output globale. Il grafico riunisce le singole acquisizioni della campagna; prima di usarlo per conclusioni sperimentali è opportuno verificare che il master contenga soltanto prove coerenti e correttamente etichettate.

10. Adattamento rapido a un altro circuito o oscilloscopio

Gli script non sono legati in modo inevitabile alla GX-TXT. La parte che richiede attenzione è l'interfaccia tra il proprio dato grezzo e le variabili interne. Un adattamento prudente si può fare per livelli, senza toccare l'algoritmo.

10.1 Livello 1 - cambiare solo i metadati

- RLED e, nella v8, RSENSE.
- Frequenza e duty cycle se diversi.
- P_GAIN e P_SENS usati come etichette della campagna.
- Nomi dei file e percorsi di output.

10.2 Livello 2 - rimappare i canali

Se il CSV ha lo stesso formato Rigol ma i canali sono differenti, basta invertire o cambiare le assegnazioni di `dati(:,2)` e `dati(:,3)`. Questa è la modifica più comune.

```
% Caso A: colonna 2 = HOLD, colonna 3 = LED
vHold = dati(:,2);
vLed = dati(:,3);
```

```
% Caso B: colonna 2 = LED, colonna 3 = HOLD
vLed = dati(:,2);
vHold = dati(:,3);
```

10.3 Livello 3 - adattare il parser

Se cambia il formato del CSV, costruire esplicitamente tre vettori coerenti: t, vHold e vLed. Appena questi tre vettori sono corretti, il resto dello script può normalmente rimanere invariato.

```
% Obiettivo del parser personalizzato:
% t      -> vettore tempo in secondi
% vHold  -> tensione del nodo HOLD equivalente
% vLed   -> tensione del nodo da cui ricavare la corrente LED

assert(length(t) == length(vHold));
assert(length(t) == length(vLed));
```

10.4 Livello 4 - cambiare il significato elettrico

Se il proprio circuito non usa un LED con resistore verso VCC come nella GX-TXT, il calcolo della corrente non è più direttamente applicabile. In tal caso sostituire soltanto la conversione da vLed alla grandezza di interesse, mantenendo invariato il resto del flusso se i dati temporali rimangono compatibili.

Metodo consigliato - dopo ogni modifica del parser o della mappatura, confrontare a mano alcuni valori del CSV con i primi campioni stampati dal programma e con i PNG. E molto più sicuro di giudicare la correttezza soltanto dal fatto che lo script termina senza errori.

11. Risoluzione dei problemi

Sintomo	Controlli consigliati
File o funzione non trovata	Verificare current directory, nome del file .m e corrispondenza con il nome della funzione.
Impossibile ricavare Start o Increment	Il CSV non ha l'header Rigol previsto oppure Start/Increment non sono nelle posizioni 4 e 5 della seconda riga.
Numero di colonne inatteso	Il formato fscanf non corrisponde al numero/ordine di colonne del CSV.
HOLD e LED sembrano scambiati	Controllare l'ordine dati(:,2)/dati(:,3), soprattutto nel passaggio v7 -> v8.
Corrente LED negativa o con scala assurda	Controllare RLED, polarità del nodo, riferimento VCC e scelta del canale.
Nessuna transizione LED ma la prova è valida	Nella v8 può essere un risultato sperimentale normale: verificare ledTransitionValid e il grafico N_LED.
Fit HOLD assente nella v8	Controllare holdFitValid. La prova può essere stata salvata correttamente anche senza fit.
OK_PLOT_ERROR	I risultati numerici sono stati salvati ma uno o più grafici non sono stati generati; leggere batch.log.
Analisi globale: colonna vgen_V non trovata	Probabile uso del master v8. GXTXT_analisi_globale_v1 richiede il master v7.

Il batch sembra usare un vecchio file .m	Cambiare directory o eseguire clear sul nome della funzione; Octave può mantenere definizioni già caricate durante una sessione.
--	--

11.1 Diagnosi in tre minuti

1. Aprire il CSV e leggere manualmente header e prime cinque righe.
2. Verificare che il nome del CSV nell'indice sia corretto.
3. Controllare che vHold e vLed non siano scambiati.
4. Aprire batch.log e cercare la prima occorrenza di ERROR oppure del nome della prova.
5. Aprire stato_batch.csv e isolare le sole righe non OK.
6. Rieseguire una delle prove problematiche con GXTXT_singola_v6, adattando i canali se necessario.

12. Checklist operativa e riferimenti

12.1 Checklist prima dell'acquisizione

- Segnali HOLD e LED collegati ai canali che verranno poi mappati nello script.
- Pretrigger presente e sufficientemente stabile.
- Finestra temporale abbastanza lunga da contenere l'evento di interesse.
- RLED e RSENSE annotati con i valori effettivamente montati.
- Nomi dei CSV univoci e coerenti con il file indice.

12.2 Checklist prima di lanciare un batch

- Una singola acquisizione e già stata elaborata con successo.
- File indice controllato: numero di colonne, punto decimale, nomi file.
- Parametri comuni di campagna verificati nello script.
- OutputRoot scelto e spazio disco disponibile.
- Versione corretta: v7 per compatibilità con analisi globale v1; v8 per numero impulsi variabile/V_RSENSE.

12.3 Checklist dopo il batch

- stato_batch.csv non contiene errori inattesi.
- batch.log è stato conservato.
- risultati_batch.csv contiene il numero atteso di prove.
- Sono stati aperti alcuni PNG di prove rappresentative e di condizioni limite.
- La copia indice_prove.csv salvata nel batch coincide con l'indice voluto.

12.4 Riferimenti

Software e dati di riferimento: sorgenti GX-TXT-v3 utilizzati per questa guida: GXTXT_singola_v6, GXTXT_batch_v7, GXTXT_batch_impulsi_v8 e GXTXT_analisi_globale_v1, inclusi gli esempi di file indice e gli output generati dalle campagne sperimentali.

GNU Octave - documentazione ufficiale: <https://docs.octave.org/> . In particolare sono utili le sezioni Function Files, Script Files e Current Working Directory per comprendere come Octave individua ed esegue i file .m.

Progetto GX-TXT e articoli di laboratorio: <https://giux-lab.it/> .

Confine di questa guida - questo manuale descrive come usare, configurare, adattare e controllare i programmi. I criteri matematici con cui vengono ricavati fit, transizioni e altre grandezze non sono trattati qui; sono documentati nel codice e nella documentazione tecnica separata.