

Laboratorio 6

Programmazione modulo DAQ per la generazione segnali

In questa esercitazione il vettore di campioni del segnale ECG acquisito tramite modulo DAQ (vedi LAB 5) viene, grazie a una conversione digitale-analogica eseguita dal DAQ, trasformato in una tensione variabile che si può osservare all'oscilloscopio, prelevando il segnale tramite una sonda dal canale di uscita del modulo DAQ.

1) Opzioni per ottenere il vettore di campioni da generare

A seconda che vogliate o meno rivedere parte del laboratorio precedente avete varie opzioni:

1) Semplicemente scaricare il vettore già fornito nella pagina del corso "filtered_ECG", salvarlo nella stessa cartella dove state lavorando e caricarlo con load.

2) Sfruttare lo stesso vettore acquisito la volta scorsa, nel caso aveste provveduto a salvarlo.

3) Utilizzare lo script preparato durante la scorsa esercitazione per acquisire una porzione di ECG e salvarla in un vettore 'data' (o altro nome a piacimento).

Per comodità vi riporto qui di seguito le linee di codice da inserire nello script (da far precedere come già visto dall'operazione preliminare di ricerca del device tramite `daq.getDevices` digitato in command window che ritorna le informazioni utili circa il vendor 'ni e il Device ID es. 'MyDAQ1'. Una volta che si sono ottenute queste informazioni meglio non inserire più `daq.getDevices` nello script):

```
s=daq.createSession('ni');
Fs=250;
Tw=5;
s.Rate=Fs;
s.DurationInSeconds=Tw;
ch=addAnalogInputChannel(s, 'myDAQ1', 'ai0', 'Voltage');
data=startForeground(s);
N_samples=round(Fs*Tw);
t=[0:N_samples-1]*(1/Fs);
figure(1)
plot(t,data)
```

I dati di interesse saranno in questo esempio quelli salvati in 'data'.

A questo punto è possibile preparare un nuovo script (facendo attenzione chiaramente a salvare il vettore data nella cartella) oppure solo una nuova sezione

nello stesso script (si ricorda che per iniziare una nuova sezione è sufficiente iniziare con un doppio %%).

Fare attenzione che non sia rimasta aperta nessuna sessione di acquisizione per non interferire con la creazione della nuova sessione necessaria per riprodurre in uscita il segnale filtrato. Per assicurarsi di questo utilizzare il comando:

`delete(s)`

all'inizio del nuovo script oppure della nuova sezione.

Per poter procedere simultaneamente con l'acquisizione e la visualizzazione del segnale proveniente dal generatore vi consiglio di utilizzare un adattatore a T in modo da inviare il segnale prodotto dal generatore sia al modulo DAQ che all'oscilloscopio.

2. Creazione della sessione, definizione dei parametri di generazione, collegamenti e definizione del canale di uscita

Verificare che il modulo DAQ sia ancora effettivamente presente e riconosciuto dal sistema utilizzando l'istruzione: `daq.getDevices`

(Se questa operazione è già stata svolta per la sezione di acquisizione non dovrebbe essere necessario ripeterla, il **vendor** e il **device ID** che serviranno saranno gli stessi utilizzati in fase di acquisizione.)

Si procede poi alla **creazione di una nuova sessione** tramite la quale verrà gestita l'interfaccia con il DAQ durante la generazione (nel caso abbiate voluto riacquisire, attenzione ad aver cancellato con `delete(s)` la sessione creata in precedenza).

L'istruzione è:

```
s = daq.createSession('ni');
```

dove la stringa di caratteri 'ni' identifica il costruttore del dispositivo hardware da controllare (in questo caso, 'ni' = National Instruments, rilevabile alla voce Vendor della lista sopra ricordata), mentre s è un nome di variabile (handle) che identifica la sessione ed ovviamente può essere scelto liberamente.

I **parametri di generazione** andranno scelti tenendo conto del segnale salvato in fase di acquisizione (*se si volesse creare un segnale a tavolino bisognerebbe tenere conto del fatto che la massima ampiezza del segnale in uscita è ± 10 V, in questo caso non serve specificarlo perchè sicuramente il requisito è rispettato essendoci lo stesso limite ± 10 V in fase di acquisizione*):

- La **frequenza di campionamento** utilizzata per la generazione dei campioni in uscita è associata alla sessione e viene definita tramite l'istruzione `s.Rate = Fs`; dove Fs è la frequenza di campionamento espressa in Hz. Va tenuto presente che la massima frequenza di campionamento raggiungibile dal dispositivo è 250 kHz per ciascun canale. Per riprodurre lo stesso segnale con le stesse caratteristiche nel tempo va mantenuta la stessa Fs utilizzata in fase di acquisizione

Dopo aver creato la sessione e stabilito i parametri di generazione, si deve determinare **l'insieme delle linee fisiche interessate** e, successivamente, le loro caratteristiche.

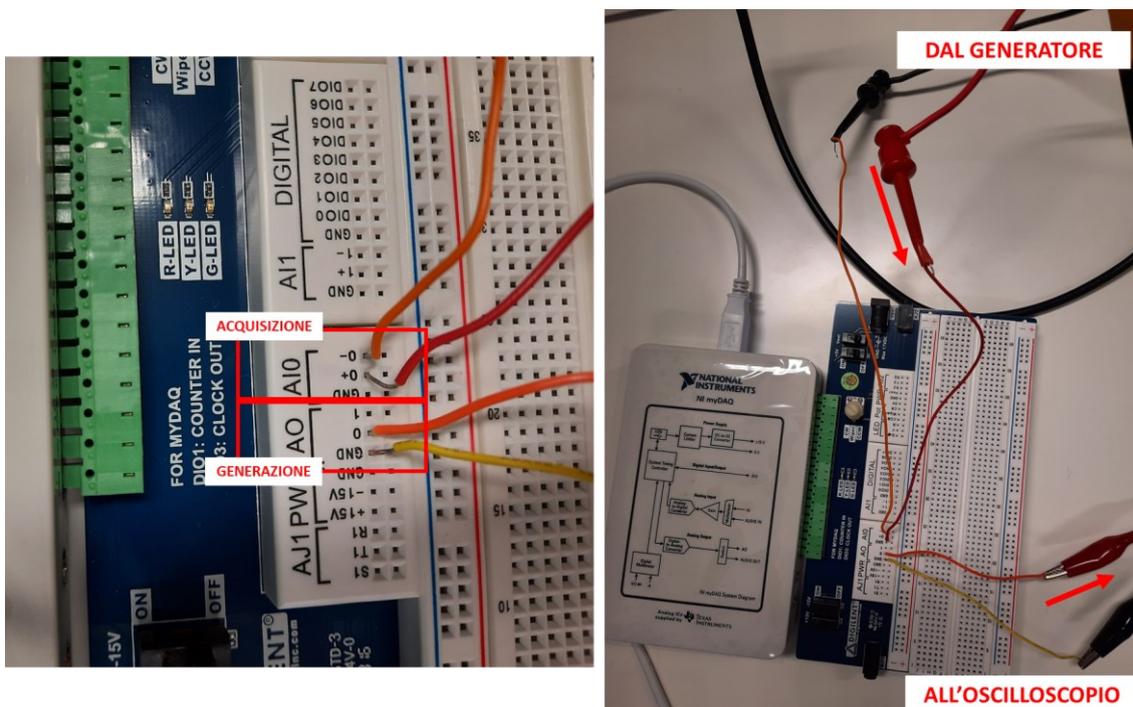
L'istruzione che associa un canale fisico di uscita alla sessione ha la forma:

```
ch=addAnalogOutputChannel(s, deviceID, channelID, measurementType);
```

dove s è la handle alla sessione e deviceID è la stringa che identifica il dispositivo interessato (ossia il modulo DAQ), riportata come Device ID nell'elenco visualizzato in precedenza.

Il terzo elemento identifica il canale utilizzato, ad esempio '0' (oppure 'ao0') si riferisce al canale di uscita analogico AO0. L'ultimo campo infine indica la grandezza generata ed in questo caso 'Voltage' viene usato per indicare che il segnale generato sarà una tensione. In questa esercitazione si fa uso di una sola linea di uscita.

Per esempio per utilizzare il canale 0 andranno inseriti due fili nella breadboard in corrispondenza di GND e 0 nella sezione AO0, come mostrato in figura. A tali fili andranno collegati rispettivamente: al GND la pinzetta nera e allo 0 quella rossa del cavo collegato all'oscilloscopio.



N.B. LA FIGURA MOSTRA IL CASO PIU' COMPLESSO IN CUI SI ACQUISISCE E POI GENERA IN SEQUENZA. SE SI UTILIZZANO VETTORI GIA' PRONTI NON SERVIRANNO I CAVETTI RELATIVI ALL'ACQUISIZIONE.

3) FASE 1: Generazione in foreground del segnale e visualizzazione singola sull'oscilloscopio

Prima di procedere con la generazione, i dati generati vanno trasferiti alla coda di ingresso al DAQ mediante l'istruzione:

```
queueOutputData(s, data);
```

(sostituire a 'data' un eventuale nome diverso dato al vettore di campioni e **prestare attenzione che il vettore che viene passato sia un vettore colonna**. Nel caso in cui sia un vettore riga è sufficiente trasporlo utilizzando l'apice es. `ecg'`).

In questa sezione iniziamo con l'eseguire una **generazione finita del vettore di campioni in foreground**, in cui tutte le altre attività nell'ambiente MatLab sono sospese fino al completamento della sua esecuzione.

La generazione del segnale in uscita viene avviata semplicemente con il comando:

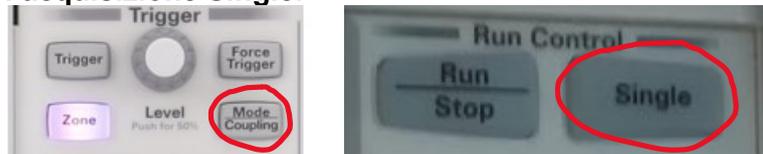
```
startForeground(s) ;
```

che attiva la generazione del segnale una sola volta (a meno che l'istruzione non venga posta entro un ciclo iterativo).

Per poter **visualizzare il segnale sull'oscilloscopio**, oltre ad assicurarsi che i collegamenti siano corretti, è necessario predisporre adeguatamente lo strumento ad una **acquisizione singola**, e **non ripetitiva** come quelle che di solito sono state eseguite nelle scorse esercitazioni.

Per questo, **prima di compilare il codice**, è necessario:

- Essendo il segnale non ripetitivo le **condizioni di trigger vanno predefinite**: nota l'ampiezza del segnale, risulta facile stabilire un corretto livello di trigger (es. se avete generato un segnale di ampiezza $V_{pp}=100\text{mV}$ potete collocare il livello di trigger a piacere tra i 30 e gli 80 mV).
- Va scelta la **modalità di trigger Normal** (effettuando l'apposita selezione nel menù che si attiva attivando mode-coupling nella sezione trigger) e **l'acquisizione Single**.



Differenze tra modalità AUTO e NORMAL del trigger

AUTO rappresenta la modalità utilizzata di default e sta per "modalità automatica". In questa modalità l'oscilloscopio sincronizzerà il segnale se viene riconosciuta una condizione di trigger, ma nel caso in cui la condizione di trigger non venga rilevata, dopo un certo intervallo di tempo l'oscilloscopio inizierà a generare un trigger automatico che non sarà però sincronizzato con il segnale sotto esame, solo per procedere con l'acquisizione. Questo provocherà una traccia instabile, causa di più tratti acquisiti non sovrapposti tra loro.

NORMAL rappresenta la modalità utilizzata di solito quando si sa a priori che l'acquisizione che si sta per svolgere sarà singola oppure che l'evento utile avviene non frequentemente. Quando in modalità NORMAL infatti l'oscilloscopio sincronizza e acquisisce solo e soltanto se viene riconosciuta una condizione di trigger nel segnale. In alternativa non viene acquisito e mostrato a schermo nulla.

Ref materials difference between normal and auto trigger mode

<https://edadocs.software.keysight.com/kkbopen/oscilloscope-triggering-when-is-normal-not-so-normal-589742415.html>

- Attenzione a impostare correttamente i valori di **fattore di scala orizzontale** (s/div) e **verticale** (V/div) in base all'ampiezza e alla durata del segnale rappresentato nel plot su Matlab. (es. se avete rappresentato su matlab 5 s di tracciato, per visualizzarli tutti nelle 10 divisioni orizzontali impostare il fattore orizzontale attorno a un valore di 500 ms/div; se avete un segnale di ampiezza $V_{pp}=100\text{ mV}$, per visualizzarlo sul massimo numero di divisioni possibili impostare il fattore verticale attorno a un valore di 15-20 mV/div)

Verificare se la frequenza o l'ampiezza della sinusoide generata corrispondono ai valori impostati nel programma Matlab.

Osservare come varia il segnale generato sull'oscilloscopio quando viene modificata la frequenza di campionamento, senza cambiare i parametri del segnale.

Quando la sessione non è più necessaria può essere rimossa utilizzando il comando `delete(s)` ;

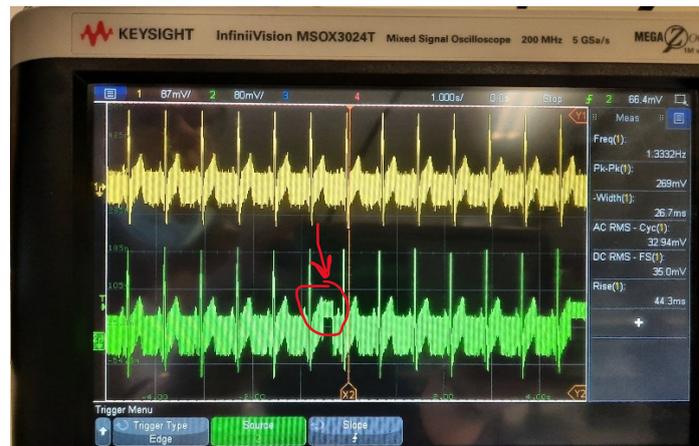
4) FASE 2: Generazione continua del segnale e osservazione sull'oscilloscopio

La generazione di una forma d'onda continua utilizzando un vettore finito di campioni richiede una particolare modalità di interazione tra il modulo DAQ e l'ambiente MatLab.

Si può verificare, infatti, che la semplice esecuzione di un ciclo iterativo:

```
for i=1:10
queueOutputData(s, samples);
startForeground(s);
end
```

produce semplicemente una sequenza di segmenti di segnale, che sono però intervallati da tempi di attesa di lunghezza variabile (*es. porzione evidenziata in rosso nella figura sotto*), causati dalle interazioni con il sistema operativo del PC, proprie della modalità di esecuzione dello script in foreground finora utilizzata.



Per poter generare e osservare il segnale in maniera continua va utilizzata la modalità **background** che ottimizza l'interazione tra modulo DAQ e ambiente MatLab per consentire una generazione continua dei dati.

A questo scopo si ricorre alla rilevazione dell'evento '**DataRequired**', forma di segnalazione che indica quando sta per essere terminata la lettura del vettore di campioni presente nella memoria del modulo DAQ. La rilevazione ha luogo attraverso una funzione di ascolto denominata listener, che al verificarsi dell'evento provvede ad accodare nuovi dati all'uscita. Rispetto all'esempio precedente la programmazione è leggermente più complessa e richiede le seguenti istruzioni:

```
s.IsContinuous = true; %% serve per impostare il DAQ in modalità
continua, di default questo valore è impostato su false
queueOutputData(s, samples); %% serve per accodare i dati pronti
```

```
lh = addlistener (s, 'DataRequired', @(src,event) src.queueOutputData(samples));
%% serve per segnalare l'evento Datarequired alla scheda ogni
volta sta per terminare il vettore di campioni.
startBackground(s); %% serve per iniziare la generazione in
background
```

Si tenga presente che, in assenza di un comando esplicito dato attraverso l'istruzione `s.stop()`, l'esecuzione in background non viene terminata.
La sequenza di istruzioni:

```
pause(30)
s.stop()
delete(lh)
```

può essere inserita nello script per terminare la sessione dopo 30 s dall'invio del comando `startBackground` (istruzione `pause(30)` – la durata si può modificare) e cancellare il listener (istruzione `delete(lh)`).

Con la generazione in background l'oscilloscopio va utilizzato nella normale modalità di acquisizione, dato che il segnale è ripetuto con continuità per tutto il tempo prescelto. Si tenga presente però che la ripetizione riguarda l'insieme dei campioni generati. **Se alle estremità del vettore di campioni vi sono delle discontinuità, queste si ritrovano e possono essere osservate nella forma d'onda ricostruita, che rappresenterà infatti una periodicizzazione del vettore 'data' stesso.**

Può essere utile come passo conclusivo confrontare il segnale generato dal modulo DAQ con quello direttamente generato dal generatore, collegando su due canali diversi dell'oscilloscopio rispettivamente il cavo coassiale proveniente dal generatore e quello proveniente dal DAQ. Per poter eseguire un confronto spostare le due tracce utilizzando la manopola piccola relativa all'offset nella sezione vertical e regolare ad un valore adeguato e uguale tra i due canali il fattore di scala verticale.

