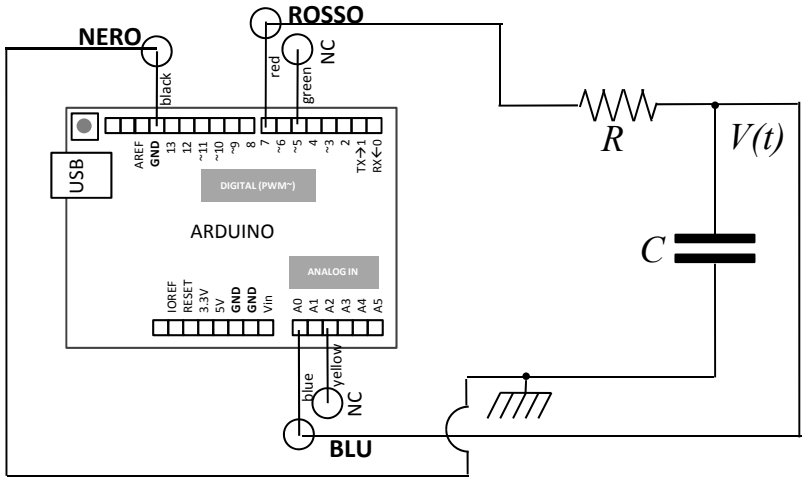
Nome e Cognome:	☐LUN ☐MAR ☐GIO Data: 9 ARDU

Carica/scarica condensatore (con Arduino)

Obiettivo dell'esperienza è realizzare un esperimento con presa dati automatizzata via Arduino finalizzato a studiare il processo di carica e scarica di un condensatore C attraverso una resistenza R . Nell'esperimento la carica/scarica è comandata automaticamente attraverso la d.d.p. generata su comando dalla porta `7` di Arduino, mentre il segnale $V(t)$ è campionato dalla porta `A0` di Arduino; è necessario riferire i potenziali alla linea di massa (piedino `GND`) di Arduino.
Lo schema dell'esperimento è mostrato in figura: fate attenzione a realizzare correttamente i collegamenti con la scheda (dovete usare tre boccole di distinti colori) e controllate preliminarmente che le connessioni con i vari piedini della scheda siano corretti. Fate riferimento al foglio di istruzioni che trovate sul banco per ulteriori avvertenze e informazioni generali.



1. Dimensionate opportunamente i valori di R e C : per la scelta, tenete presente che nell'esperimento sono campionati 200 punti nella fase di carica e altrettanti in quella di scarica; le misure si susseguono con un intervallo di durata nominale Δt=(100-900)μs, aggiustabile a passi da 100μs tramite opportuno comando nello script di Python (vedi dopo). Un corretto dimensionamento garantisce che carica e scarica siano "ben riprodotte" nei dati. Evitate comunque di utilizzare R < 680ohm (nominali) per limitare la richiesta di corrente dalla porta di Arduino. Riportate R (misurata con tester) e C (nominale) nella tabella in basso assieme al valore τ_{att} .
2. Preliminarmente dovete eseguire l'upload dello sketch `carich.ino` nella memoria di Arduino utilizzando il programma Arduino (o Arduino IDE) nel computer di laboratorio.
3. Quindi dovete modificare a vostro piacimento lo script di Python (nome `caricascarica_v1.py`) che serve per gestire la comunicazione seriale via USB: i parametri che siete invitati a modificare secondo necessità sono l'intervallo di tempo Δt fra due campionamenti successivi e il nome (eventualmente la directory) dei files generati. Riportate il valore di Δt prescelto nella tabella in basso.
4. Il lancio dello script di Python fa partire l'acquisizione (la console dà indicazioni sul progresso) al termine della quale nel computer vengono registrati due files, con parte finale e estensione rispettivamente `_C.txt` e `_S.txt` , che contengono i dati del processo rispettivamente di carica e di scarica.
5. I files hanno 200 righe, corrispondenti a 200 misure, e due colonne: la prima riporta il tempo t in unità di μs, la seconda la d.d.p. ai capi del condensatore, $V(t)$, in unità arbitrarie di digitalizzazione (numeri interi); ricordate che la dinamica di digitalizzazione di Arduino è 10 bits. I tempi vengono nominalmente azzerati all'inizio di ognuna delle due fasi (carica e scarica).
6. Al termine, dovete graficare e analizzare i dati acquisiti. Allo scopo è necessario che: (i) individuiate e riportiate nei grafici le barre di errore opportune; (ii) realizziate best-fit secondo gli andamenti attesi per la fase di carica e di scarica. Si ricorda che è accettabile riportare i dati in unità arbitrarie di digitalizzazione; si consiglia, almeno in partenza, di eseguire i best-fit con un solo parametro libero, il tempo caratteristico τ . Riportate tutte le informazioni relative ai best-fit (funzioni usate, risultato, chi-quadro, etc.), assieme agli altri commenti che ritenete opportuni (accordo con le aspettative e accordo fra i tempi caratteristici delle due fasi, etc.), sul retro di questo foglio.

R [ ]	C [ ]	τ_{att} [ ]	Δt [μs]