

**POLITECNICO DI BARI**

**SISTEMI PROGRAMMABILI**

**L.M. INGEGNERIA INFORMATICA**

**A.A. 2010/11**

**“HOW TO ACQUIRE RESISTANCE VALUES OF THE PT100  
TEMPERATURE PROBE”**

**DI LEO CARLO** [ing.carlo88@gmail.com](mailto:ing.carlo88@gmail.com)

**SALATINO ANGELO ANTONIO** [a.salatino@studenti.poliba.it](mailto:a.salatino@studenti.poliba.it)

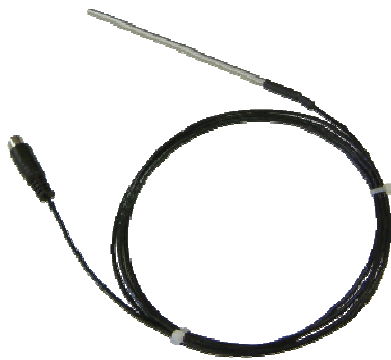
## Obiettivi

L'obiettivo di questo elaborato è quello di impartire una semplice dimostrazione sull'uso dei sistemi ATE (Automatic Test Equipment) in ambiente Matlab. L'esercitazione si prefigge di misurare la resistenza della sonda di temperatura al platino PT100 collegata al multimetro digitale in modalità 4-wires. In seguito verrà mostrato uno script contenente codici Matlab per effettuare l'acquisizione e successivamente una elaborazione dei dati.

## Apparato sperimentale

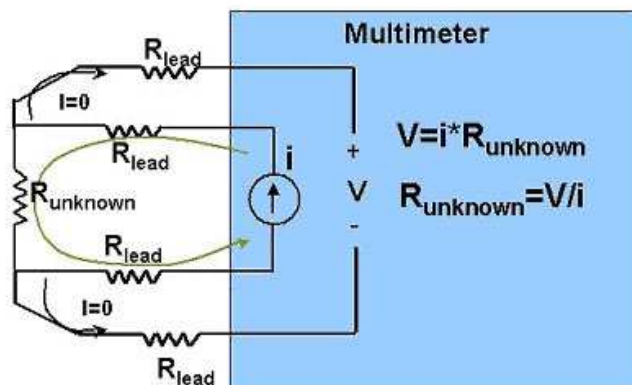
Strumenti e accessori: Multimetro digitale HP34401, Sonda termica PT100, Computer (con software Matlab e driver dello strumento).

La sonda in questione, mostrata nella figura in basso, possiede una resistenza interna che varia tra 100  $\Omega$  a 138 $\Omega$  rispettivamente a 0°C e 100°C come si evince dalla tabella.



| °C      | $R$    |
|---------|--------|
| -200.00 | 18.52  |
| -190.00 | 22.83  |
| -180.00 | 27.10  |
| -170.00 | 31.34  |
| -160.00 | 35.54  |
| -150.00 | 39.72  |
| -140.00 | 43.88  |
| -130.00 | 48.00  |
| -120.00 | 52.11  |
| -110.00 | 56.19  |
| -100.00 | 60.26  |
| -90.00  | 64.30  |
| -80.00  | 68.33  |
| -70.00  | 72.33  |
| -60.00  | 76.33  |
| -50.00  | 80.31  |
| -40.00  | 84.27  |
| -30.00  | 88.22  |
| -20.00  | 92.16  |
| -10.00  | 96.09  |
| 0.00    | 100.00 |
| 0.00    | 100.00 |
| 10.00   | 103.90 |
| 20.00   | 107.79 |
| 30.00   | 111.67 |
| 40.00   | 115.54 |
| 50.00   | 119.40 |
| 60.00   | 123.24 |
| 70.00   | 127.08 |
| 80.00   | 130.90 |
| 90.00   | 134.71 |
| 100.00  | 138.51 |
| 110.00  | 142.29 |
| 120.00  | 146.07 |
| 130.00  | 149.83 |
| 140.00  | 153.58 |
| 150.00  | 157.33 |

Trovandosi in condizioni di variazioni di resistenza relativamente basse, effettuare una misura diretta comporterebbe non pochi errori soprattutto per la presenza di cavi tra la sonda e l'ohmmetro. Una tecnica alternativa che consente una misura più accurata è quella del 4-wires che richiede una misura indiretta, dove si alimenta la sonda con un generatore di corrente andando così a misurare la d.d.p. ai suoi capi per poi ricavare il valore della resistenza attraverso la legge di Ohm. Con opportuni accorgimenti è ipotizzando che il voltmetro abbia un'impedenza interna pressoché infinita si capisce che gli effetti resistivi dei fili sono trascurabili. Il circuito per implementare un collegamento 4-wires è mostrato nella seguente figura:





Successivamente si deve impostare il valore di End Of String in modo tale che il Matlab interpreti e indichi i caratteri terminatori di riga per controllare e distinguere il flusso verso e da lo strumento.

```
set(multim, 'EOSMode', 'read&write')
set(multim, 'EOSCharCode', 'LF') % Line Feed -> ASCII = 10
```

## Reset dello strumento

```
fprintf(multim, '*RST');
```

## Acquisizione dati

In questa fase del codice si è pensato di inserire un ciclo `for` così da ripetere l'operazione ogni 0.5 secondi per 5 minuti così da effettuare 600 acquisizioni. Il valore resistito verrà inserito poi in un vettore.

```
% ciclo di lettura
for k = 1:600
    resistance(k)=str2double(query(multim, ':MEASure:FRESistance?'));
    pause(0.5); % ferma l'esecuzione per 0.5 secondi
end
```

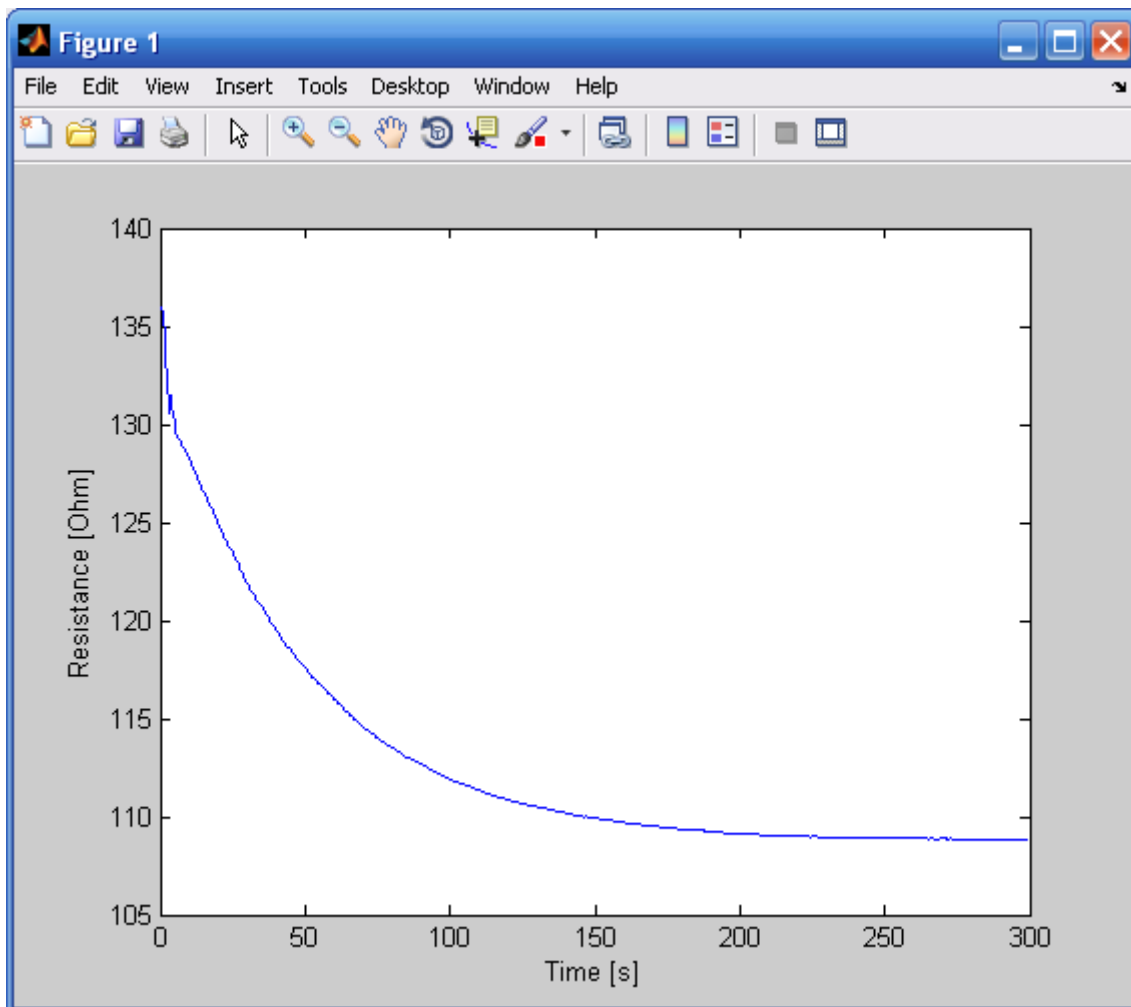
## Curva di tendenza

La sonda nella fase di raffreddamento ha un andamento esponenziale decrescente del primo ordine. Con i seguenti comandi si dovrà calcolare, da i punti precedentemente acquisiti, la funzione che regola il raffreddamento della sonda.

Si crea inizialmente il vettore `time` così da avere ogni singolo valore per il suo corretto tempo di acquisizione, successivamente si esegue il plot dei valori della resistenza.

```
time=linspace(0,299.5,600);
n = plot(time,resistance); % plotting dei dati sulla base dei tempi
xlabel('Time [s]');
ylabel('Resistance [Ohm]');
set(n, 'Color', 'blue', 'LineWidth', 1)
```

Si ottiene il seguente grafico:



Per i primi istanti dell'acquisizione ci sono dei valori con elevate variazioni per via della fase in cui si sta estraendo la sonda dalla fonte di calore (termos). Dopo essersi accertati del numero di campioni che subiscono quell'effetto si procederà alla loro cancellazione.

```
g = 8; %campioni da tagliare
res = resistance(g:600); %elimino le instabilità iniziali
```

Automaticamente anche il vettore `time` deve essere troncato per poi avere una giusta corrispondenza con `resistance` (stesse dimensioni).

```
time = time(1:600-g+1);
```

Plotting del nuovo vettore sulla vecchio grafico.

```
hold all, o = plot(time,res);
set(o,'Color','black','LineWidth',1)
```

Di qui in poi si sta usando una tecnica per la creazione della curva di tendenza. Essendo un sistema del primo ordine esso potrà essere approssimato con una funzione simile alla seguente:

$$A + B e^{-\frac{t}{\tau}}$$

Dove  $A$  sarà uguale al valore di regime ovvero il minimo tra i campioni presenti. Tutto ciò potrebbe portare in errore perché il valore di regime dipende dalla temperatura circostante e potrebbe non essere raggiunto nel tempo considerato.  $B$  sarà il valore compreso tra massimo e minimo ovvero l'intervallo di valori per cui

varia la resistenza, mentre  $\tau$  (tau) viene calcolato come il tempo necessario al sistema per raggiungere il 63,2 % del valore di regime.

```
mi = min(res);
ma = max(res);

A = mi %uguale al valore di regime
B = ma - mi %stato iniziale

% ricerca della costante tau
taustop = 0.632 * B;
resTau = ma - taustop;

w = size(res);

l = 0;
a = 0;
while l <= w(2) && a==0
    l = l + 1;
    if (res(l) <= resTau)
        a=1
    end
end

tau = l/2 % perché abbiamo effettuato una misura ogni 0.5 secondi
```

Calcolati i parametri A,B e  $\tau$  è possibile ora mettere a confronto i valori di resistenza acquisiti con i valori della funzione ricavata con un ulteriore plot dei dati sul precedente grafico.

```
% plotting

result = A + B*exp(-time/tau);

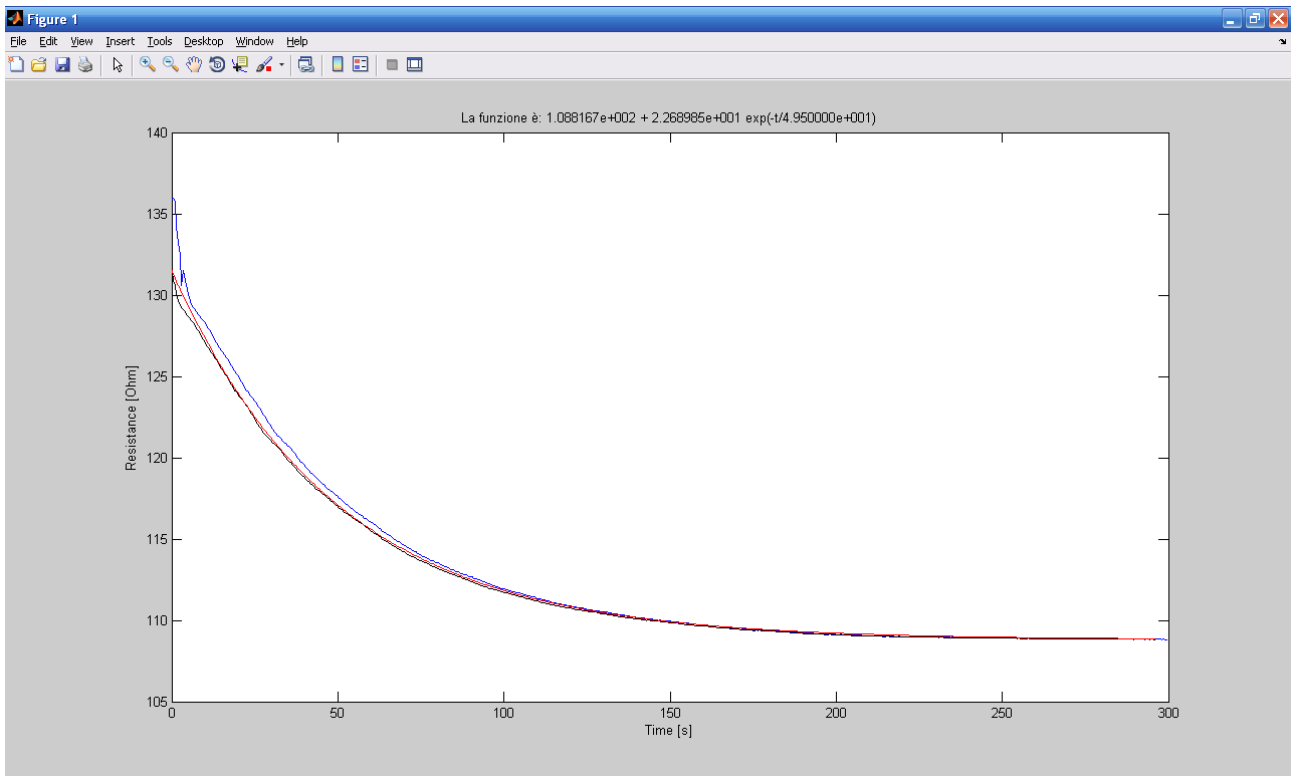
p = plot(time, result);
set(p, 'Color', 'red', 'LineWidth', 1)
title(str)
ylabel('Temperature [°C]')
xlabel('Time [s]')
```

Una dimostrazione dei valori plottati è mostrata nella seguente figura dove la curva in blu mostra l'andamento dei dati acquisiti dal multimetro, la curva in nero rappresenta i valori acquisiti a meno dei primi 8, mentre la curva rossa rappresenta la curva di tendenza calcolata.

Matlab terminato il calcolo restituisce anche:

```
str =
```

La funzione è:  $1.088167e+002 + 2.268985e+001 \exp(-t/4.950000e+001)$



## Chiusura e cancellazione oggetti

Qui di seguito, si trovano i comandi per chiudere le comunicazioni con gli strumenti e di conseguenza eliminarli dal workspace.

```
fclose(multim);  
delete(multim);
```

```
clear
```