



Misura di temperatura con sensore LM335 e scheda Arduino Uno



Obiettivo dell'esperienza

- Sviluppare un termometro digitale usando
 - ↳ un sensore elettronico di temperatura
 - ✓ LM335 (costo \approx 1 EUR)
 - ↳ una scheda Arduino
 - ✓ Arduino UNO (costo \approx 20 EUR)
- Progettare il circuito di condizionamento del sensore
- Stimare l'incertezza attesa

Il sensore



- National Semiconductor modello LM335
 - È un diodo Zener con tensione di *breakdown* proporzionale alla temperatura assoluta T (K)
 - Uscita riferita allo zero assoluto
 - ↳ $V_{out} = 0 \text{ V} @ 0 \text{ K} \equiv -273.15 \text{ }^{\circ}\text{C}$
 - Sensibilità nominale $S = 10 \text{ mV/K}$
 - ↳ Corrente inversa nel campo da $400 \text{ }\mu\text{A}$ a 5 mA

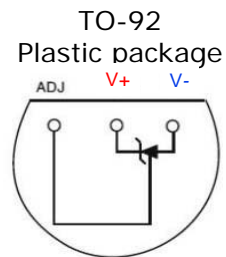
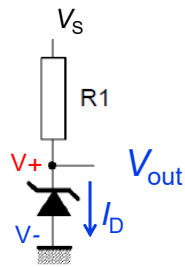
Il sensore



- National Semiconductor modello LM335
 - Campo di temperatura
 - ↳ $(-40 \div 100) ^\circ\text{C}$
 - Incertezza strumentale a temperatura ambiente (modello deterministico)
 - ↳ $\delta T = \pm 2 ^\circ\text{C}$
 - Resistenza termica: $165 ^\circ\text{C/W}$

Circuito di condizionamento

- Il diodo Zener deve essere polarizzato inversamente



Bottom view

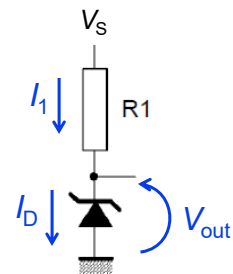
↪ Corrente I_D nel campo (0.4 ÷ 5) mA

Circuito di condizionamento

- Tensione di alimentazione fornita dalla scheda Arduino (uscita 5 V)
 - ↪ Verificare se l'uscita 5 V è in grado di alimentare il sensore
- Stimare il valore di R_1

$$I_D \approx I_1 = \frac{V_S - V_{out}}{R_1}$$

I_D dipende da V_{out} , che a sua volta dipende dalla temperatura in misura





Circuito di condizionamento

- Stimare il valore di R_1
 - ⇒ $V_S = 5\text{ V}$
 - ⇒ Campo di temperatura: $(5 \div 50)^\circ\text{C}$

$$I_{D,\min} = \frac{V_S - V_{\text{out,max}}}{R_1} > 0.4\text{ mA}$$

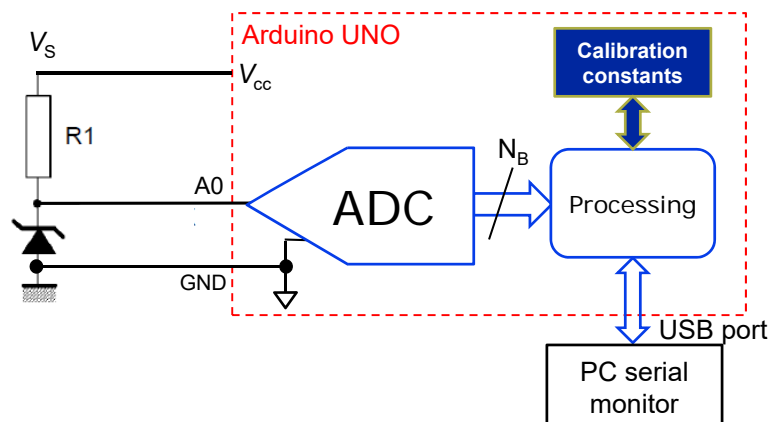
$$I_{D,\max} = \frac{V_S - V_{\text{out,min}}}{R_1} < 5\text{ mA}$$

- Tenere in considerazione l'autoriscaldamento



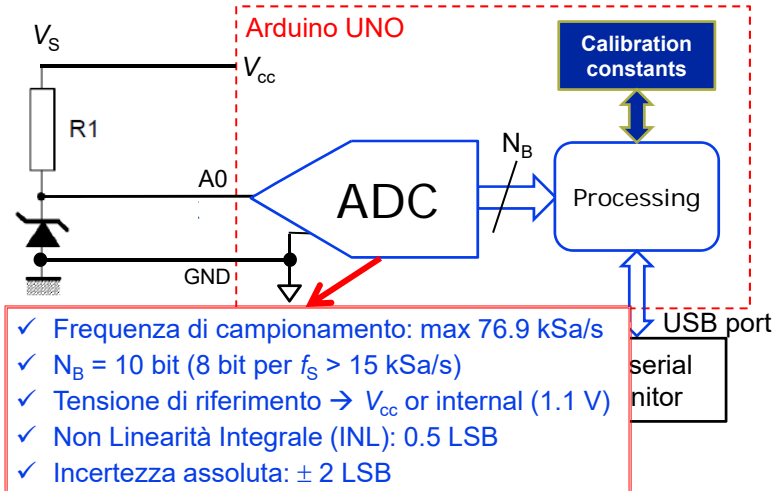
Circuito di condizionamento

- Collegamento alla scheda Arduino





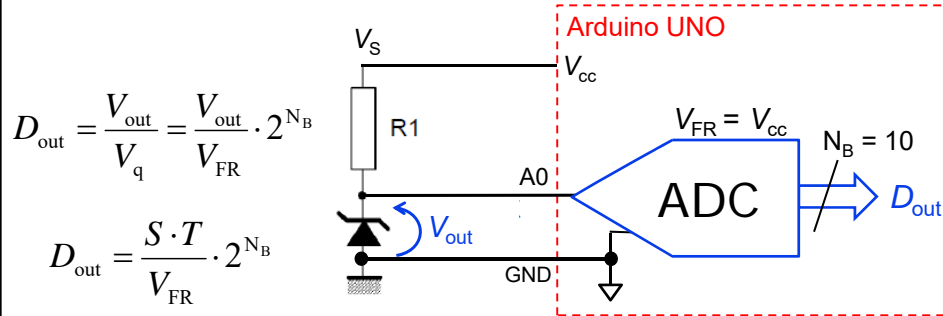
Schema di principio



Schema di principio

- Campo di temperatura: $(5 \div 50) ^\circ\text{C}$
 - $\approx (278 \div 323) \text{ K}$
 - ↳ $V_{\text{out}}: (2.78 \div 3.23) \text{ V}$
- Tensione di riferimento: V_{cc}
 - V_{cc} : dalla porta USB
 - ↳ USB 2.0 $\rightarrow V_{cc} = (5 \pm 0.25) \text{ V}$
 - ↳ USB 3.0 $\rightarrow V_{cc} = (4.45 \div 5.25) \text{ V}$

Schema di principio



FUNZIONE DI TARATURA (1)

$$T = D_{out} \cdot \frac{V_{FR}}{2^{N_B}} \cdot \frac{1}{S}$$

Stima dell'incertezza

$$T = D_{out} \cdot \frac{V_{FR}}{2^{N_B}} \cdot \frac{1}{S}$$

➤ Modello deterministico

$$\delta T = \left| \frac{\partial T}{\partial D_{out}} \right| \cdot \delta D_{out} + \left| \frac{\partial T}{\partial V_{FR}} \right| \cdot \delta V_{FR} + \left| \frac{\partial T}{\partial S} \right| \cdot \delta S$$

$$\delta T = \frac{V_{FR}}{S \cdot 2^{N_B}} \cdot \delta D_{out} + \frac{D_{out}}{S \cdot 2^{N_B}} \cdot \delta V_{FR} + \delta T^{\text{sensor}}$$

$$\delta V_{FR} = 0.25 \text{ V}$$

$$\delta T^{V_{FR}} \rightarrow 16.2 \text{ K !!! (per } D_{out, \max} = 662)$$

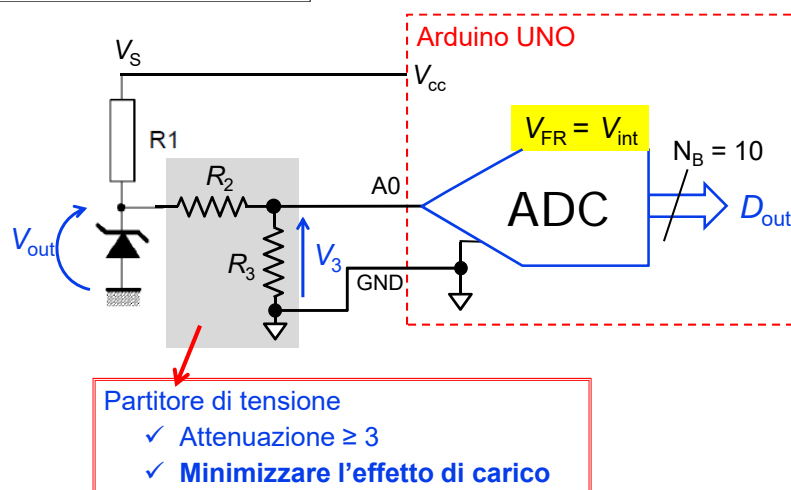


Stima dell'incertezza

- Possibili interventi per ridurre l'incertezza
 - ✓ Uso del riferimento interno dell'ADC (1.1 V)
 - ↳ $V_{int} = (1.1 \pm 0.1) V$
 - ↳ Tolleranza più alta di V_{CC}
 - ↳ Necessario attenuare il segnale di uscita del sensore
 - ✓ Misura della tensione di riferimento dell'ADC
 - ↳ V_{CC} oppure V_{int} ?
 - ↳ La prima scelta non richiede un partitore di tensione, **ma la costante di taratura è legata al PC usato** (attraverso la porta USB)



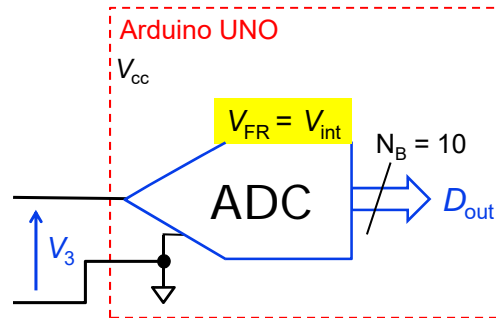
Schema modificato



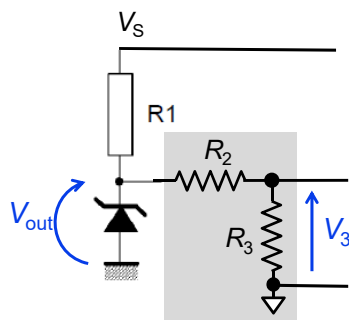


Schema modificato

$$D_{\text{out}} = \frac{V_3}{V_q} = \frac{V_3}{V_{\text{int}}} \cdot 2^{N_B}$$



Schema modificato

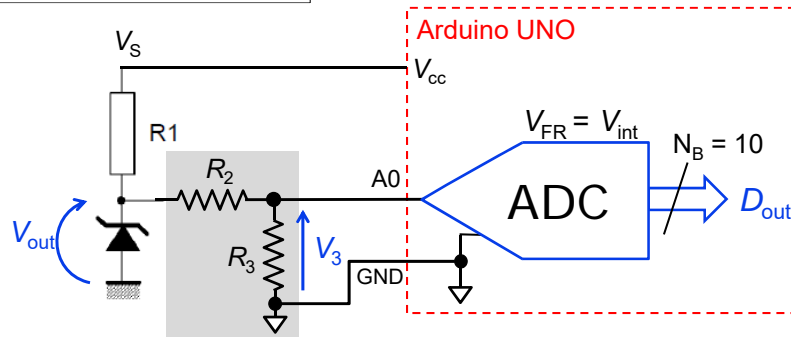


$$D_{\text{out}} = \frac{V_3}{V_q} = \frac{V_3}{V_{\text{int}}} \cdot 2^{N_B}$$

$$V_3 = V_{\text{out}} \cdot \frac{R_3}{R_3 + R_2} = S \cdot T \cdot \frac{R_3}{R_3 + R_2}$$

$$D_{\text{out}} = S \cdot T \cdot \frac{R_3}{R_3 + R_2} \cdot \frac{2^{N_b}}{V_{\text{int}}}$$

Schema modificato



FUNZIONE DI TARATURA (2)

$$T = D_{out} \cdot \frac{V_{int}}{2^{N_B}} \cdot \frac{1}{S} \cdot \left(1 + \frac{R_2}{R_3} \right)$$

Schema modificato

$$T = D_{out} \cdot \frac{V_{int}}{2^{N_B}} \cdot \frac{1}{S} \cdot \left(1 + \frac{R_2}{R_3} \right)$$

➤ Stima dell'incertezza

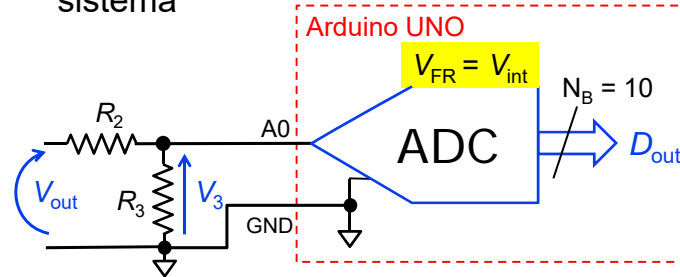
✓ Propagazione dei vari contributi

- ↗ Incertezza del sensore
- ↗ D_{out} (dalle specifiche del μ -C)
- ↗ R_2 ed R_3 (misurate con il DMM)
- ↗ V_{int} (misurata con il DMM tra i pin AREF and GND della scheda Arduino)
 - ↗ Solo quando la tensione di riferimento è impostata su "internal" o V_{CC}

Schema modificato

➤ Stima dell'incertezza

- ✓ Soluzione alternativa: caratterizzazione del sistema



$$D_{out} = V_{out} \cdot \frac{R_3}{R_3 + R_2} \cdot \frac{2^{N_B}}{V_{int}} + D_{OFF}$$

Schema modificato

➤ Stima dell'incertezza

- ✓ Soluzione alternativa: caratterizzazione del sistema

$$D_{out} = V_{out} \cdot \left(\frac{R_3}{R_3 + R_2} \cdot \frac{2^{N_B}}{V_{int}} \right) + D_{OFF}$$

G

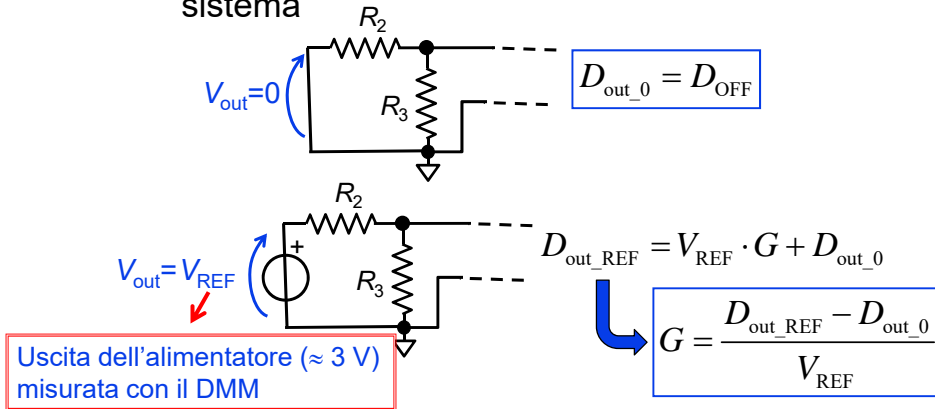
$$D_{out} = V_{out} \cdot G + D_{OFF}$$

↪ stima dei parametri G e D_{OFF}
della relazione I/O

Schema modificato

➤ Stima dell'incertezza

- ✓ Soluzione alternativa: caratterizzazione del sistema



Schema modificato

➤ Stima dell'incertezza

- ✓ Soluzione alternativa: caratterizzazione del sistema

$$V_{out} = S \cdot T = \frac{D_{out} - D_{OFF}}{G}$$



FUNZIONE DI TARATURA (3)

$$T = \frac{1}{S} \cdot \frac{D_{out} - D_{OFF}}{G}$$



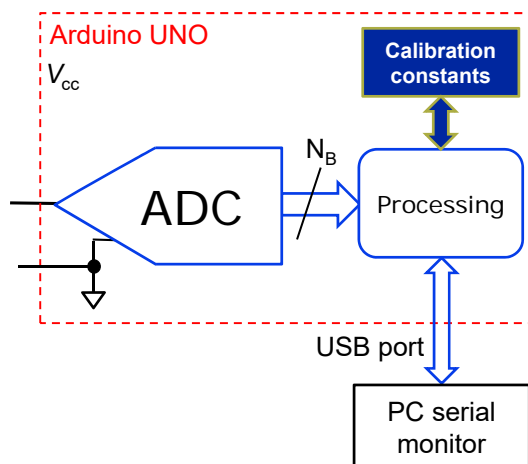
Schema modificato

$$T = \frac{1}{S} \cdot \frac{D_{\text{out}} - D_{\text{OFF}}}{G}$$

- Stima dell'incertezza
 - ✓ Propagazione dei vari contributi
 - ↪ Incertezza del sensore
 - ↪ D_{out} (INL e quantizzazione)
 - ↪ D_{OFF} e G ...



Firmware del micro-controllore



- Gestire la comunicazione seriale con il PC
- Configurare l'ADC
- Acquisire il segnale di tensione
- Implementare la funzione di taratura (1) e (2) oppure (3)
- Fornire le misure di temperatura



Firmware del micro-controllore

- Gestire la comunicazione seriale
 - Fissare il *baud rate*, usare la *built-in function* `serialEvent()`, ...
 - ↳ Vedere capitoli 3, 4 e 5 della guida *Using Arduino boards in Measurements for dummies*
- Configurare l'ADC
- Acquisire il segnale di tensione
 - *built-in functions* `analogReference()` e `analogRead()`
 - ↳ Vedere capitolo 6 della guida



Firmware del micro-controllore

- Implementare la funzione di taratura
 - Convertire il codice di uscita dell'ADC in misura di temperatura
 - ↳ Usare le opportune costanti di taratura (V_{cc} o V_{int} , fattore di attenuazione)
 - ↳ Stimare l'incertezza di misura
- Fornire le misure di temperatura
 - Inviare i risultati al *serial monitor* via USB



Caratterizzazione del sistema

- Specificare:
 - ✓ Campo di temperatura
 - ✓ Risoluzione
 - ✓ Incertezza
 - ↪ Funzioni di taratura (1) e (2) oppure (3)
 - ✓ Dimensione del *firmware*