

Come utilizzare il sensore di temperatura TMP36

La conversione di grandezze fisiche in elettriche è un aspetto che prima o poi viene affrontato quando iniziamo a realizzare piccoli esperimenti con Arduino. Oggi la sensoristica è ricca di dispositivi capaci di svolgere questa conversione.

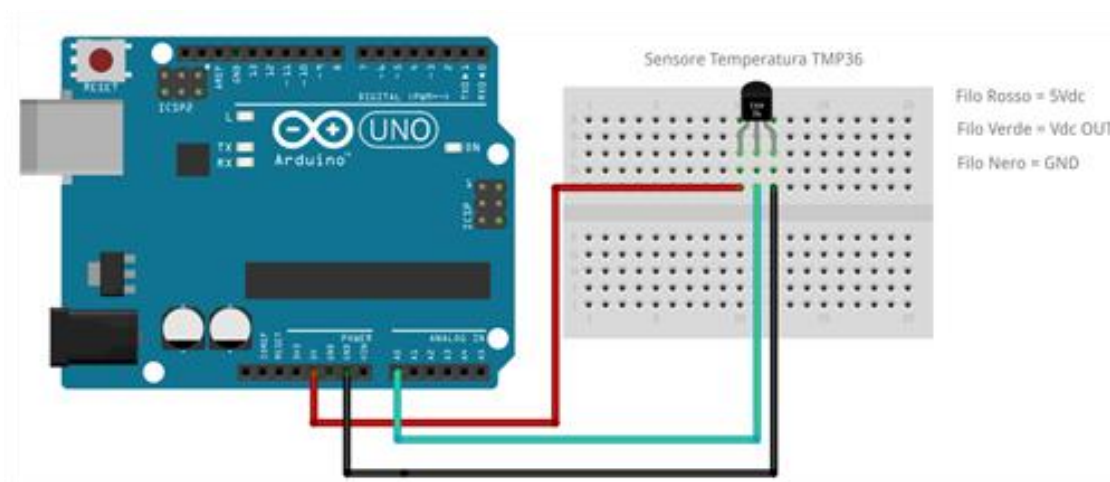
Nello specifico, per quanto riguarda la misura della temperatura abbiamo in commercio dispositivi di ogni sorta, da quelli ultrasensibili a quelli con precisioni dell'ordine del centesimo di grado, dai costi contenuti ai costi esorbitanti. A tal proposito ho in precedenza utilizzato sia il sensore di temperatura analogico MCP9700 sia il sensore digitale MCP9803.

Il segnale di conversione di questi sensori può essere sia analogico (variazione di tensione in funzione della variazione della temperatura) sia digitale (con convertitore analogico digitale ed invio dei dati su linea I2C, SPI o 1Wire).

Per i nostri esperimenti ci accontenteremo di un sensore molto diffuso, dal costo contenuto e dal semplice utilizzo, il **sensore TMP36** prodotto da Analog Device.

Il **TMP36** permette di acquisire **temperature** comprese nell'**intervallo** tra **-40°C e +125°C** restituendo in uscita valori di tensione lineari tra circa **0.1Vdc e 1.7Vdc**. Una variazione di grado produce una variazione della tensione di uscita pari a 10mV; alla temperatura di **0°C** il sensore eroga una tensione di **500mV**.

Il circuito che ho realizzato è molto banale, si limita a collegare il sensore direttamente ad **Arduino UNO** tramite la porta analogica **A0**:



guardando frontalmente il **TMP36** troviamo sul lato sinistro il **pin di alimentazione**, sul pin centrale il **segnale in uscita** e sul pin destro il **collegamento a massa**.

Il primo sketch di esempio legge dalla **porta analogica A0** il valore di tensione che eroga il **TMP36**:

```
1 void setup()
2 {
3   //Init Seriale
4   Serial.begin(9600)
5 }
6
7 void loop()
8 {
9   delay(500);
10  int val_ADC = analogRead(0);
11  //invio il dato acquisito al pc
12  Serial.println(val_ADC);
13 }
```

Il codice precedente non fa altro che convertire in forma digitale la tensione presente sul piedino A0, generata dal TMP36. Questo valore deve essere ora interpretato in modo da ottenere direttamente il valore in °C. Per far ciò abbiamo bisogno del datasheet del componente, e precisamente il grafico di conversione °C/Vdc, rappresentato anche nella figura seguente:

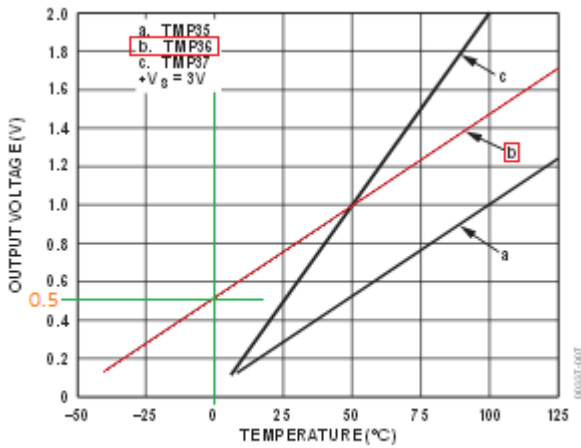


Figure 6. Output Voltage vs. Temperature

come possiamo osservare dal grafico (per il **TMP36** la linea b evidenziata in rosso) per una tensione di uscita di 0.5Vdc il sensore rileva la temperatura di 0°C. Questo dato ci permette subito di intuire che tensioni inferiori a 0.5Vdc indicano una temperatura sotto lo zero. Inoltre sappiamo che una variazione di grado si ripercuote con una variazione di tensione di 10mV. Quindi, passando agli esempi, se sul pin A0 sono presenti 510mV significa che il sensore sta rilevando una temperatura di 1°C (510mV – 500mV = 10 mV variazione di 1°C).

Non ci resta che interpretare i dati a nostra disposizione e scrivere uno sketch che restituisca il valore in °C:

```

1 //variabili globali
2 int val_Adc = 0;
3 float temp = 0;
4
5 void setup()
6 {
7   //init seriale
8   Serial.begin(9600);
9 }
10
11 void loop()
12 {
13   //leggo dalla porta A0
14   val_Adc = analogRead(0);
15   //converto il segnale acquisito in un valore
16   //espresso in gradi centigradi
17   temp = ((val_Adc * 0.00488) - 0.5) / 0.01;
18   //invio il dato sulla seriale
19   Serial.println(temp);
20   //ritardo di mezzo secondo
21   delay(500);
22 }

```

la formula che converte il valore acquisito in gradi centigradi è la seguente

$$^{\circ}\text{C} = ((\text{valoreADC} * \text{PrecisioneADC}) - \text{TensioneZeroGradi}) / \text{stepGradoTensione}$$

dove

°C = valore della temperatura in gradi centigradi

valoreADC = valore della conversione analogico digitale restituito da analogRead

PrecisioneADC = questo valore è ottenuto dividendo la tensione di riferimento dell'ADC (default 5Vdc) e il numero massimo restituito dalla conversione (1024). (5Vdc/1024 = 0.00488)

TensioneZeroGradi = indica la tensione di uscita dal sensore quando rileva una temperatura di 0°C

stepGradoTensione = indica la variazione di tensione per ogni variazione di grado (0.01 = 10 mV)

Come aumentare la risoluzione nelle letture

Eseguendo l'ultimo sketch ci accorgiamo che i valori di temperatura non sono proprio quelli che ci aspettiamo. Considerando la **precisione** del sensore **TMP36** che si attesta a **+/- 2°C** per tutta la scala, otteniamo comunque dei valori che differiscono di circa 5-7°C dal valore reale. Questo 'errore' dipende da diversi fattori tra cui imprecisione dell'ADC, rumore del segnale e approssimazione dei valori nei calcoli.

Per limitare questi errori possiamo adottare due tecniche, la prima è quella di usare il **pin AREF** dell'**Arduino UNO** per dare al convertitore analogico digitale dell'ATmega328 un riferimento di tensione più basso (di default è 5Vdc), l'altra tecnica è quella di effettuare diverse letture ed eseguire una media dei valori letti.

Nello schema seguente viene indicato come collegare il **pin AREF** al **pin power di 3.3Vdc**:


```
36 temp = ((val_Adc * 0.0032) - 0.5) / 0.01;  
37 //invio i dati al computer  
38 Serial.println(temp);  
39 }
```

in questo modo otterremo delle misure di temperatura che si avvicineranno molto al valore reale.