

**Compito Sistemi - 5 ETA – 30/05/2014**  
**Pilotaggio di una ventola con Arduino**

Si vuole progettare un sistema in grado di attivare il funzionamento del motore di una ventola di raffreddamento in funzione della temperatura dell'ambiente da controllare rilevata da un sensore.

La temperatura rilevata ricade in una di 4 fasce per le quali il motore deve ruotare con una determinata velocità che sarà proporzionale, com'è noto, al duty-cycle imposto.

In particolare il motore, supposto funzionante in c.c. con alimentazione a 5V, è pilotato in PWM (Pulse Width Modulation) attraverso un'onda quadra il cui duty-cycle è regolabile dal 20% all'80%. Il sistema prevede in uscita anche la presenza di 4 diodi LED distinti (verde, giallo, arancio e rosso) che hanno lo scopo di segnalare, con l'accensione di uno solo dei quattro, la fascia di temperatura rilevata. L'uscita in PWM, prelevata sul pin 9 della scheda Arduino, pilota la base di un transistor ed il motore è posto tra collettore e alimentazione. Nella seguente tabella si riassume il funzionamento del sistema.

Fascia di temperatura	Velocità del motore	Tasso di modulazione PWM in D%	LED indicatore acceso	Pin di Arduino
$t < 20^{\circ}\text{C}$	minima	20%	verde	13
$20^{\circ}\text{C} < t < 30^{\circ}\text{C}$	media	40%	giallo	12
$30^{\circ}\text{C} < t < 40^{\circ}\text{C}$	elevata	60%	arancio	11
$40^{\circ}\text{C} < t$	massima	80%	rosso	10

**Hardware utilizzato**

Si mostra nella fig.1 il montaggio su breadboard del circuito prototipale realizzato.

Il sensore di temperatura impiegato è un circuito integrato a tre terminali con sigla LM35 la cui uscita rispetta la formula:  $V_s = K * t$

ove K è una costante di valore  $K = 10\text{mV}/^{\circ}\text{C}$  e t rappresenta la temperatura rilevata espressa in  $^{\circ}\text{C}$ .

La sonda lavorerà in un ambiente la cui temperatura è sempre all'interno dell'intervallo  $0 - 50^{\circ}\text{C}$ .

Per tale motivo si avrà:

$V_s$  (minima) = \_\_\_\_\_

$V_s$  (massima) = \_\_\_\_\_

Poiché il segnale analogico di ingresso sul pin A0 della scheda Arduino accetta un range tra 0 e 5V, è conveniente procedere ad un'amplificazione del segnale  $V_s$  di un fattore \_\_\_\_\_ in modo da offrire alla scheda Arduino un ingresso analogico compreso tra 0 e 5V.

Si è optato per un amplificatore non invertente che utilizza un operazionale e che soddisfa la formula:  $A_v = \text{_____}$  (vedi fig.2).

Negli schemi di fig.1 e 2 si è sempre usata la tensione di alimentazione  $V_{cc}=+5\text{V}$  prelevata dalla scheda Arduino che a sua volta la preleva dalla interfaccia USB del PC.

In pratica conviene fornire un'alimentazione maggiore per l'operazionale (ad esempio 9V) ed un'alimentazione adeguata al motore della ventola che si intende utilizzare.

Per assicurare la saturazione del transistor quando l'uscita PWM è al livello logico alto, la corrente assorbita dalla ventola dovrà essere inferiore al valore  $I_C$  che si ricava dopo i seguenti passaggi:

$I_B = \text{_____}$

Supponendo 50 il valore di  $h_{FEmin}$  del transistor, la corrente di collettore, cioè quella che attraverserà il motore della ventola, sarà superiore al limite massimo dato dalla formula:

$I_C = h_{FE} * I_B = \text{_____}$

Se la ventola necessita di una corrente di intensità maggiore si dovrà rivedere, diminuendolo, il valore di  $R_5$ .

Ultima osservazione: è opportuno inserire un diodo in parallelo al motore con il Catodo verso l'alimentazione e l'Anodo verso il collettore per \_\_\_\_\_

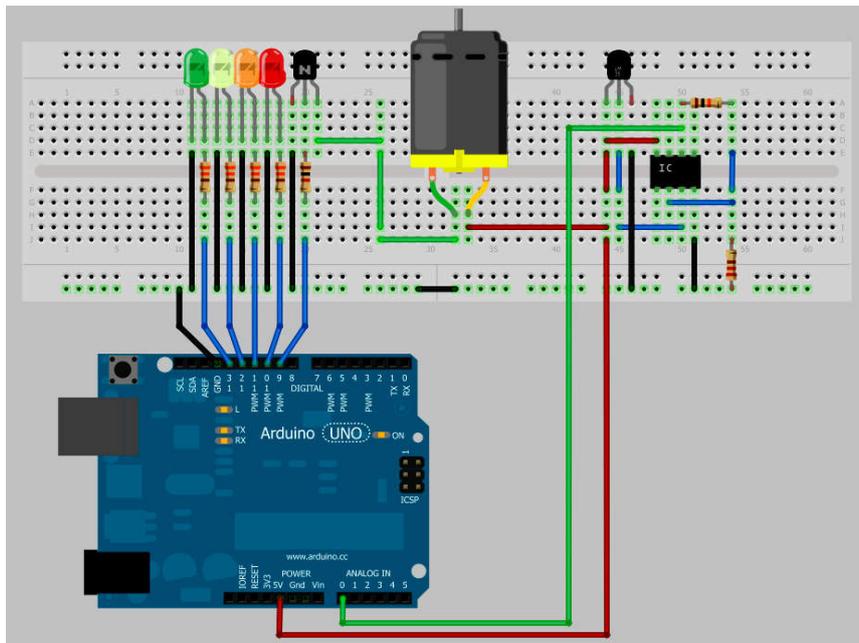


Fig. 1 – Schema su breadboard realizzato col software free “Fritzing” (<http://fritzing.org>). Nello schema si è supposto di utilizzare un transistor NPN con piedinatura EBC, un operazionale ad 8 terminali con alimentazione sui pin 8 e 4.

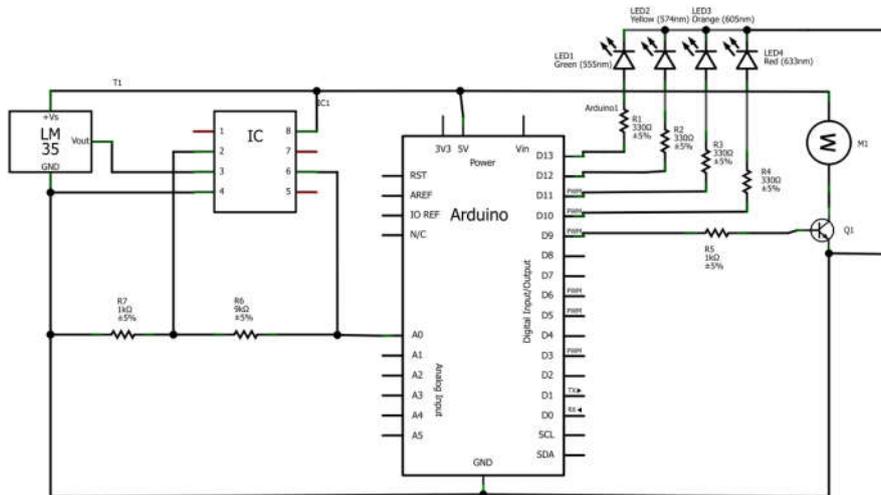


Fig.2 – Schema elettrico ricavato da quello di fig.1

<pre> int verde=13; int giallo=12; int arancio=11; int rosso=10; int pwm=9; int sensore= A0; int valore; int fascia; //valori:0, 1, 2, 3, 4, 5  void setup() {   pinMode(verde, OUTPUT);   pinMode(giallo, OUTPUT);   pinMode(arancio, OUTPUT);   pinMode(rosso, OUTPUT);   pinMode(pwm, OUTPUT);   Serial.begin(9600);   Serial.println("valore fascia"); } </pre>	<pre> void loop() {   valore=analogRead(sensore);   Serial.print(valore);   // fascia è compreso tra 0 e 5   fascia=map(valore, 0, 1023, 0, 5);   Serial.print(" ");   Serial.println(fascia);   if (fascia==0) fascia=1;   spegni();   digitalWrite(14-fascia, HIGH);   analogWrite(pwm, 51*fascia);   delay(500); }  // sottoprogramma che spegne i 4 LED void spegni() {   digitalWrite(verde,LOW);   digitalWrite(giallo,LOW);   digitalWrite(arancio,LOW);   digitalWrite(rosso,LOW); } </pre>
---	---

Completare i dati richiesti, discutere la soluzione e le istruzioni del software.

Cognome e nome \_\_\_\_\_