

## PRIMI PASSI CON ARDUINO

### 1) Accensione e spegnimento del led presente sulla scheda Arduino.

La scheda Arduino presenta, a bordo, alcuni led uno dei quali può essere controllato dal pin digitale 13; quando questo pin è HIGH agli estremi del led c'è una differenza di potenziale e il led si accende, quando è LOW la differenza di potenziale è zero e il led è spento.

Ecco lo sketch che accende e spegne il led per un secondo per 10 volte.

```
#define led 13 // viene dato un nome al pin 13, da questo punto in poi si chiamerà led
int i = 0; //viene definita una variabile di tipo int (intero) che si chiama i
int N = 10; // viene definita una variabile di tipo int (intero) che si chiama N e che vale 10
int b = 1; // ancora un'altra variabile di tipo int che si chiama b e che viene posta uguale a 1
```

```
void setup()
```

```
{
  pinMode(led, OUTPUT); // si stabilisce che il pin chiamato led un di OUTPUT
  digitalWrite(led, LOW); /* il pin chiamato led viene posto nello stato LOW (agli estremi del led non c'è tensione) */
}
```

```
void loop()
```

```
{
  delay(5000); // Arduino aspetta per 5000 ms cioè 5 s
  for (i=0; i<N; i++) /* inizia il ciclo for, per N volte (10) si ripetono le istruzioni che sono inserite tra parentesi graffe */
  {
    digitalWrite(led, HIGH); // il pin chiamato led viene posto nello stato HIGH, il led si accende
    delay(1000); // Arduino aspetta per 1000 ms cioè 1 s
    digitalWrite(led, LOW); // il pin chiamato led viene posto nello stato LOW, il led si spegne
    delay(1000); // Arduino aspetta per 1000 ms cioè 1 s
  }
  while(b==1){ /* questo ciclo serve per fermare l'esecuzione del void loop, fino a quando il valore della variabile b è uguale a 1 Arduino ripete quello che sta tra le parentesi graffe successive, cioè nulla e rimane fermo per sempre */
  }
}
```

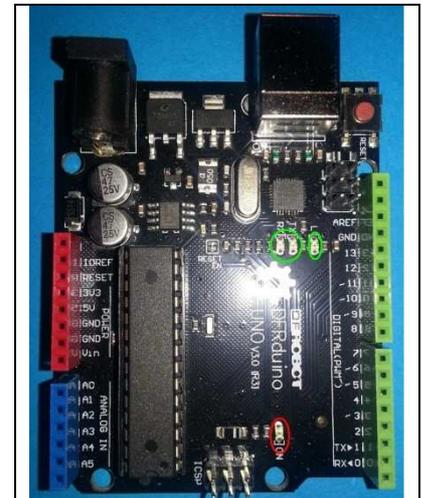


Figura 1 – I led della scheda Arduino. Sono uno rosso e 3 verdi. Accenderemo e spegneremo quello in alto a destra.

Ricorda che quello che segue // fino alla fine della riga e quello compreso tra /\* e \*/ sono commenti e nello sketch, hanno solo lo scopo di illustrare il significato dei vari comandi e possono essere tralasciati.

**Scrivi lo sketch ed eseguillo.**

**Varia il valori dei delay per far accendere e spegnere il led con una frequenza diversa.**

## 2) Accensione e spegnimento di un led esterno.

In questo esercizio faremo accendere e spegnere un led esterno alla scheda Arduino. Per prima cosa è necessario costruire un circuito elettrico (vedi figura 2) in cui sia presente un led<sup>[1]</sup> e una resistenza da 220  $\Omega$  (la tensione che fornisce Arduino è troppo elevata per il led e la presenza della resistenza la abbassa).

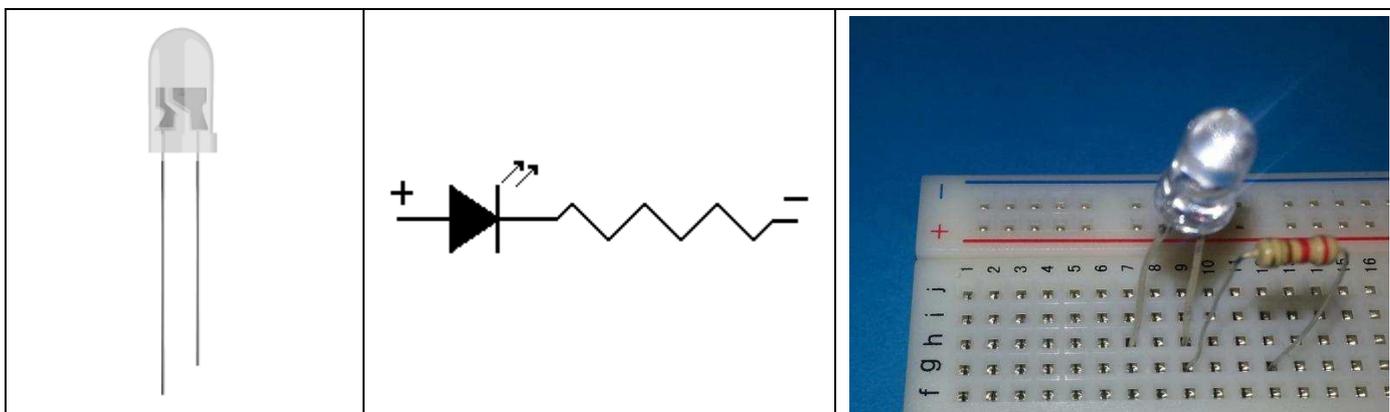


Figura 2 – A sinistra un led. Si ricordi che in un circuito il led va montato in modo che il piedino più corto sia collegato al – e quello più lungo al + del generatore. Al centro lo schema del montaggio del led e della resistenza e a destra il montaggio di led e resistenza sulla breadboard.

La tensione al circuito viene fornita da Arduino mediante l'accensione e lo spegnimento di un pin digitale (nello sketch qui sotto è stato scelto il 2, ma va bene uno qualsiasi). Il + va collegato quindi al pin 2 e il – a terra (al pin GRD) vedi figura 3).

```
#define led 2
int i = 0;
int N = 10;
int b = 1;

void setup()
{
  pinMode(led, OUTPUT);
  digitalWrite(led, LOW);
}
void loop()
{
  delay(5000);
  for (i=0; i<N; i++)
  {
    digitalWrite(led, HIGH);
    delay(1000);
    digitalWrite(led, LOW);
    delay(1000);
  }
}
```

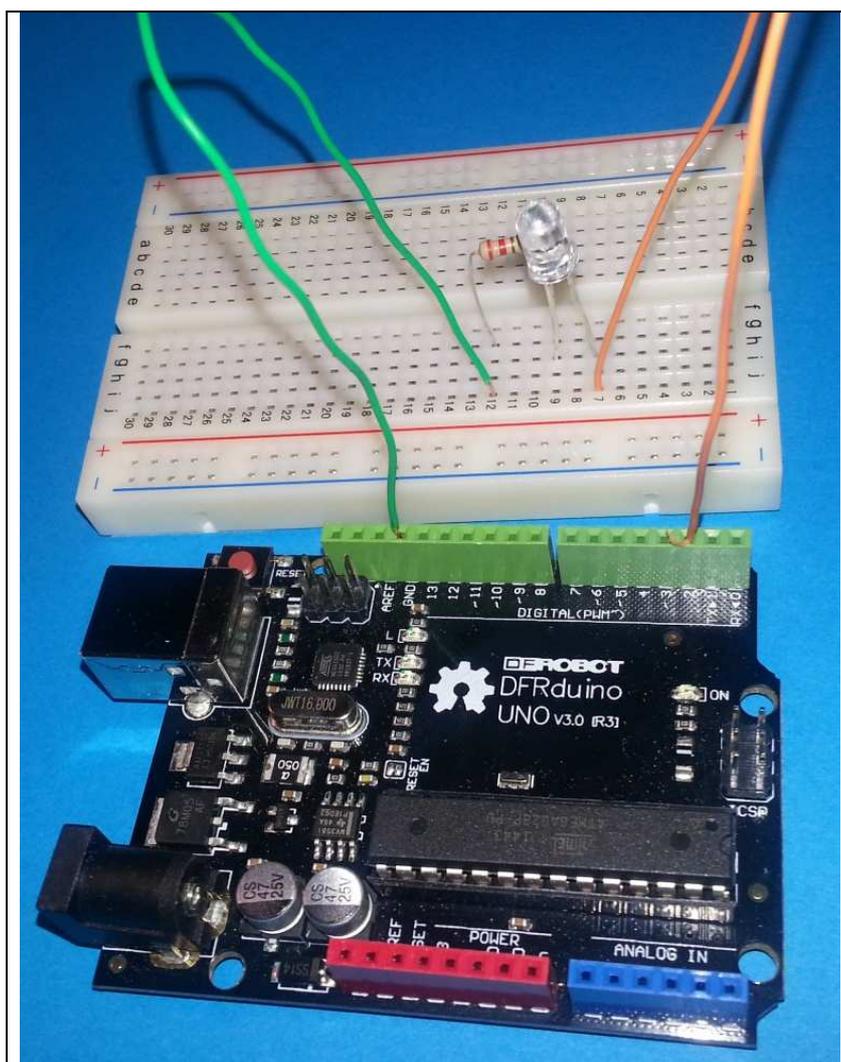


Figura 3 – Collegamento del circuito ad Arduino

[<sup>1</sup>] Nella figura il led è bianco, ma nella realtà, quando si accende è colorato (i colori disponibili sono rosso, verde, giallo, blu)

```
}  
while(b==1){  
}
```

Come si può osservare ad eccezione di una piccola, ma significativa variazione lo sketch è identico al precedente. La differenza sta nel fatto che in questo caso si utilizza il pin 2 invece che il 13.

**Scrivi lo sketch ed eseguillo.**

**Varia il valori dei delay per far accendere e spegnere il led con una frequenza diversa.**

**Varia il pin.**

### ***3) Accensione e spegnimento alternato di due led di colore diverso.***

---

**Da solo prova a realizzare il circuito e lo sketch che permettano di accendere e spegnere alternativamente due led di colore diverso.**

## CODICE DEI COLORI PER LE RESISTENZE DEI RESISTORI

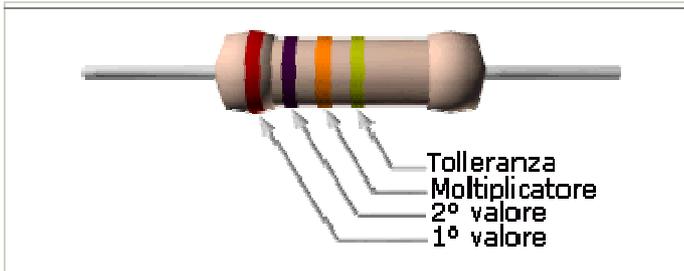
Il valore della resistenza ed altre caratteristiche di ciascun resistore sono spesso codificati usando dodici colori che, opportunamente posizionati in anelli, permettono di distinguere velocemente un resistore dall'altro (Vedi tabella a lato ed esempi sotto).

In realtà questo metodo di codifica è usato solo per i componenti "tradizionali" di minore potenza (in genere da 1/8 ad 1 W) e forma cilindrica.

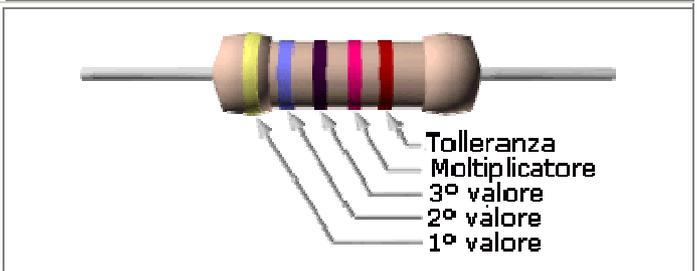
Esistono codifiche specifiche per i resistori a montaggio superficiale mentre sui componenti di grosse dimensioni l'indicazione è "in chiaro" o quasi.

Ci sono resistori con 4 o 5 strisce colorate, qui sotto due esempi.

Corrispondenza colore-valore			
Colore	Valore	Moltiplicatore	Tolleranza (%)
Nero	0	0	-
Marrone	1	1	±1
Rosso	2	2	±2
Arancio	3	3	±0.05
Giallo	4	4	-
Verde	5	5	±0.5
Blue	6	6	±0.25
Violetto	7	7	±0.1
Grigio	8	8	-
Bianco	9	9	-
Oro	-	-1	±5
Argento	-	-2	±10
Niente	-	-	±20

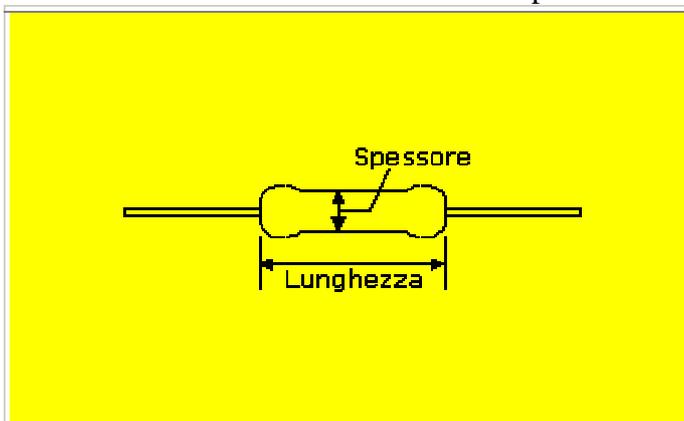


**Esempio 1**  
 (Marrone=1), (Nero=0), (Arancio=3)  
 $10 \times 10^3 = 10k \text{ ohm}$   
 Tolleranza (Oro) = ±5%



**Esempio 2**  
 (Giallo=4), (Violetto=7), (Nero=0), (Rosso=2)  
 $470 \times 10^2 = 47k \text{ ohm}$   
 Tolleranza (Marrone) = ±1%

Le dimensioni dei resistori ad impasto indicano la potenza dissipabile.



Potenza max (W)	Spessore (mm)	Lunghezza (mm)
1/8	2	3
1/4	2	6
1	3.5	12
2	5	15

### 3) *Accensione e spegnimento alternato di due led di colore diverso. SOLUZIONE*

#### **schetch**

```
#define colore1 2
#define colore2 4
int i = 0;
int N = 20;
int b = 1;

void setup()
{
  pinMode(colore1, OUTPUT);
  pinMode(colore2, OUTPUT);
  digitalWrite(colore1, LOW);
  digitalWrite(colore2, LOW);
}
void loop()
{
  delay(5000);
  for (i=0; i<N; i++)
  {
    digitalWrite(colore1, HIGH);
    delay(500);
    digitalWrite(colore1, LOW);
    digitalWrite(colore2, HIGH);
    delay(500);
    digitalWrite(colore2, LOW);
  }
  while(b==1){}
}
```