2019

Instroduzione all’utilizzo delle librerie

Paolo Weishaupt, Carlo pezzotti

SAM trevano |[Indirizzo della società]

INDICE

1. [Manuale d’uso delle librerie 2](#_Toc535408206)

[Led RGB: 2](#_Toc535408207)

[Come utilizzare il led RGB 2](#_Toc535408208)

[Come utilizzare la libreria del led RGB: 3](#_Toc535408209)

[Push button: 3](#_Toc535408210)

[Come utilizzare un bottone: 3](#_Toc535408211)

[Come utilizzare la libreria del bottone: 3](#_Toc535408212)

[Led: 4](#_Toc535408213)

[Come utilizzare un led: 4](#_Toc535408214)

[Come utilizzare la libreria del bottone: 4](#_Toc535408215)

# Manuale d’uso delle librerie

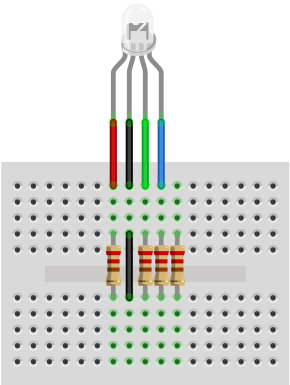
L’utilizzo di queste librerie è raccomandato a delle persone che si stanno avvicinando allo sviluppo sulla piattaforma Arduino e hanno bisogno di una guida nei primi passi della programmazione. Al loro interno ci sono anche tre codici di esempio di utilizzo della libreria per ogni attuatore elencato:

1. Led RGB
2. Push Button
3. Led
4. Potenziometro
5. Piezo buzzer

## Led RGB:

### Come utilizzare il led RGB

Per l’utilizzo di un led RGB bisogna avere come minimo a disposizione 3 pin liberi, e un pin che deve essere collegato alla massa.   
Ai pin che controllano il colore del led bisogna attaccarci 3 resistenza da 150 Ohm.

  
  
Nell’immagine a sinistra si può vedere come bisogna attaccare il Led RGB. Al primo pin partendo da sinistra bisogna collegare il pin che imposterà il colore rosso al led. Nel secondo bisogna attaccare la massa, ovvero quel pin sull’Arduino con scritto GND oppure “-”. Nel terzo bisogna collegare il pin che imposterà il colore verde al led. Nell’ultimo per esclusione si andrà a collegare il pin che imposterà il colore blu al led. Le resistenze che ho utilizzato nello schema sono solo indicative e   
non rispecchiano il vero colore della resistenza.

Un aiuto per migliorare l’esperienza con il led RGB è di attaccarci un giro di scotch intorno così da rendere più omogeneo il colore che si andrà a vedere.



### Come utilizzare la libreria del led RGB:

L’accensione del led RGB può avvenire grazie ad una libreria creata e ideata da me e Paolo Weishaupt.  
Per il giusto funzionamento bisogna fare i seguenti passaggi:

* **Accertarsi che la libreria sia stata inclusa nel progetto**:  
  Facilmente verificabile che all’inizio del programma ci sia scritta una stringa di codice simile a questa:  
  “*#include "LightLed.h*”
* **Verificare che la varibile sia stata creata in modo corretto:**

Il modo più facile per dichiarare una variabile è il seguente:  
“*LightLed\* [nome variabile];*”E nel metodo setup istanziarla

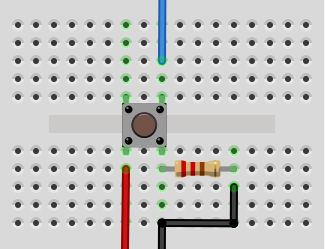
* **Istanziare variabile:**A questo punto bisognerà istanziarla, quindi recarsi nel metodo setup e istanziare la variabile dichiarata in precedenza, nel seguente modo:

*“led = new LightLed([pin colore rosso],[pin colore blu],[pin colore verde]);”*

* **Utilizzare il metodo per l’accensione del led:**Per accendere il led bisognerà quindi chiamare un semplice metodo facilmente utilizzabile, per esempio nel seguente modo:  
  “*led->lightLed(255,0,0);*”  
  Il seguente codice colorerà di rosso il led. I valori possono andare da 0 fino a 255.

## Push button:

### Come utilizzare un bottone:

Nell’immagine a sinistra si può vedere come bisogna attaccare il bottone al circuito. Il pin rosso sta a significare il “+”, quello nero il “-” e quello blu il dato che si andrà a leggere che sarà 0 se il bottone non sarà premuto o 1 quando lo sarà.

### Come utilizzare la libreria del bottone:

La lettura del bottone può avvenire grazie ad una libreria creata e ideata da me e Paolo Weishaupt.  
Per il giusto funzionamento bisogna fare i seguenti passaggi:

* **Accertarsi che la libreria sia stata inclusa nel progetto**:  
  Facilmente verificabile che all’inizio del programma ci sia scritta una stringa di codice simile a questa:  
  “*#include "ButtonState.h"*”
* **Verificare che la variabile sia stata creata in modo corretto:**

Il modo più facile per dichiarare una variabile è il seguente:  
“*ButtonState\* [nome variabile];*”E nel metodo setup istanziarla

* **Istanziare variabile:**A questo punto bisognerà istanziarla, quindi recarsi nel metodo setup e istanziare la variabile dichiarata in precedenza, nel seguente modo:

*“button = new ButtonState([pin del bottone])”;***Utilizzare il metodo per la lettura del bottone:**Per leggere il bottone bisognerà quindi chiamare un semplice metodo facilmente utilizzabile, per esempio nel seguente modo:  
“*button->getStateButton()*  
Il seguente codice ritornerà lo stato del bottone.

## Led:

### Come utilizzare un led:

Nell’immagine a sinistra si può vedere come bisogna attaccare il led al circuito. Il cavo blu sta ad indicare lo stato di scrittura. Il cavo nero invece il “-”. La resistenza bisogna metterla per non rischiare di bruciare il led.

### Come utilizzare la libreria del leb:

L’accensione del led può avvenire grazie ad una libreria creata e ideata da me e Paolo Weishaupt.  
Per il giusto funzionamento bisogna fare i seguenti passaggi:

* **Accertarsi che la libreria sia stata inclusa nel progetto**:  
  Facilmente verificabile che all’inizio del programma ci sia scritta una stringa di codice simile a questa:  
  “*#include "LightSingleLed.h"*
* **Verificare che la variabile sia stata creata in modo corretto:**

Il modo più facile per dichiarare una variabile è il seguente:  
“*LightSingleLed\* [nome variabile];*”E nel metodo setup istanziarla

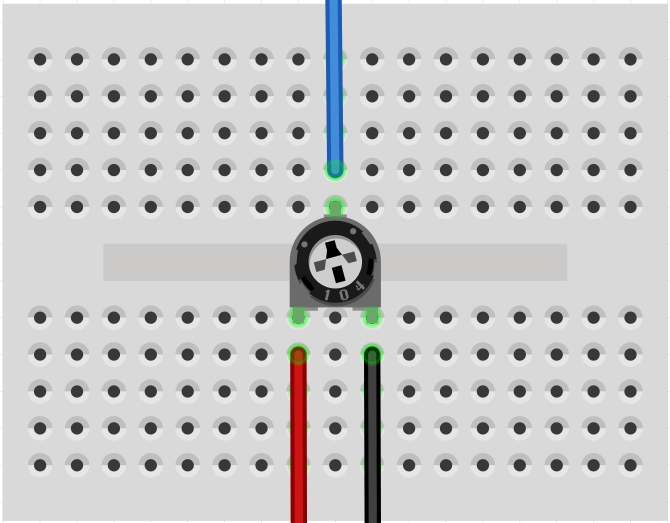
* **Istanziare variabile:**A questo punto bisognerà istanziarla, quindi recarsi nel metodo setup e istanziare la variabile dichiarata in precedenza, nel seguente modo:

*“led = new LightSingleLed([pin del led])”*

**Utilizzare il metodo per l’accensionedel led**Per accendere il led bisognerà quindi chiamare un semplice metodo facilmente utilizzabile, per esempio nel seguente modo:  
“*led->* *lightSingleLed ();*”

## Potenziometro:

### Come utilizzare un potenziometro:



Nell’immagine a sinistra si può vedere come bisogna attaccare il potenziometro al circuito. Il cavo blu sta ad indicare lo stato di lettura. Il cavo nero invece il “-”. Il cavo rosso invece sta a rapprensetare il “+”. La resistenza bisogna metterla per non rischiare di bruciare il led.

### Come utilizzare la libreria del potenziometro:

La lettura dello stato del poteniziometro si può eseguire grazie alla libreria scritta e ideata da me e Paolo Weishaupt.

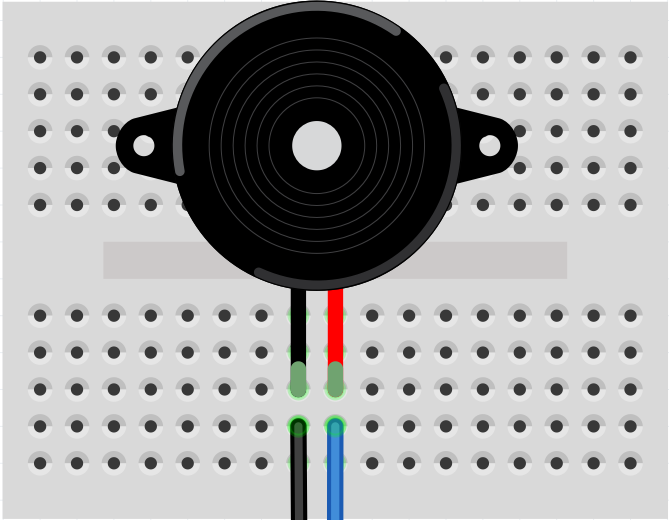
* **Accertarsi che la libreria sia stata inclusa nel progetto**:  
  Facilmente verificabile che all’inizio del programma ci sia scritta una stringa di codice simile a questa:  
  “*#include "Potenziometro.h"*
* **Verificare che la varia****bile sia stata creata in modo corretto:**

Il modo più facile per dichiarare una variabile è il seguente:  
“*Potenziometro\* [nome variabile];*”E nel metodo setup istanziarla

* **Istanziare variabile:**A questo punto bisognerà istanziarla, quindi recarsi nel metodo setup e inizializare la variabile dichiarata in precedenza, nel seguente modo:*“[nome variabile] = new Potenziometro([pin di lettura del potenziometro])”;*
* **Utilizzare il metodo per la lettura del potenziometro**Per leggere lo stato del potenziometro bisognerà quindi chiamare un semplice metodo facilmente utilizzabile, per esempio nel seguente modo:  
  *“potenziometro-> ottieniValorePotenziometro ();”.*Il seguente metodo ritorna un valore di tipo int che va di solito da 0 a 1023, dove 0 è il valore minimo di resistenza che un potenziometro può assumere e 1023 il valore massimo.

## Piezo Buzzer:

### Come utilizzare un potenziometro:



Nell’immagine a sinistra si può vedere come bisogna attaccare il piezo buzzer al circuito. Il cavo blu sta ad indicare lo stato di scrittura. Il cavo nero invece il “-”.

### Come utilizzare la libreria del Piezo Buzzer:

La scrittura dello stato del piezo buzzer si può eseguire grazie alla libreria scritta e ideata da me e Paolo Weishaupt.

* **Accertarsi che la libreria sia stata inclusa nel progetto**:  
  Facilmente verificabile che all’inizio del programma ci sia scritta una stringa di codice simile a questa:  
  “*#include "”Buzzer.h"*
* **Verificare che la varia****bile sia stata creata in modo corretto:**

Il modo più facile per dichiarare una variabile è il seguente:  
 “*Buzzer\* [nome variabile];*”E nel metodo setup istanziarla

* **Istanziare variabile:**A questo punto bisognerà istanziarla, quindi recarsi nel metodo setup e inizializare la variabile dichiarata in precedenza, nel seguente modo:  
  *“[nome variabile] = new Buzzer([pin buzzer])”;*

**Utilizzare il metodo per la lettura del potenziometro**Per far suonare il piezo buzzer bisognerà quindi chiamare un semplice metodo facilmente utilizzabile, per esempio nel seguente modo:  
*“buzzer-> beep ([tempo di suono],[tonalità del suono]);”.*