## Libreria potenziometro:

Bisogna creare due classi, una “Potentiometer.cpp” e una “Potentiometer.h”.

Nella classe “Potentiometer.h” bisogna dichiarare un attributo, il pin del potenziometro (\_pin). Ci sono due metodi, il metodo Potentiometer(int pin), il metodo getValue ().

Nella classe “Button.cpp” invece come prima cosa bisogna includere la classe “Potentiometer.h” e Arduino.h”. Il metodo Potentiometer (int pin) è il costruttore che definisce il pin del potenziometro, il metodo intero getValue () che ritorna il valore del potenziometro.

## Libreria bottone:

Bisogna creare due classi, una “Button.cpp” e una “Button.h”.

Nella classe “Button.h” bisogna dichiarare un attributo, il pin del bottone (\_pin). Ci sono tre metodi, il metodo Button(int pin) , il metodo getButtonValue() e il metodo toogle().

Nella classe “Button.cpp” invece come prima cosa bisogna includere la classe “Button.h” e Arduino.h”. Il metodo Button(int pin) è il costruttore che definisce il pin del bottone, il metodo booleano getButtonValue() che ritorna il valore del bottone e il metodo toogle() che inverte lo stato del bottone.

Libreria LED:

Bisogna creare due classi, una “Led.cpp” e una “Led.h”.

Nella classe “Led.h” bisogna dichiarare un attributo, il pin del Led (\_pin). Ci sono 6 metodi, il metodo Led (int pin), il metodo ledOn(), il metodo ledOff(), il metodo blink(int milliseconds), il metodo setLed(bool stato) e il metodo getState().

Nella classe “Led.cpp” invece come prima cosa bisogna includere la classe “Led.h” e Arduino.h”. Il metodo Led (int pin) è il costruttore che definisce il pin del Led, il metodo getState() che ritorna il valore del Led, il metodo ledOn() e il metodo LedOff() che rispettivamente accendono e spengono il Led, il metodo blink(int milliseconds) che fa lampeggiare il Led con la velocità in base al parametro e setLed(bool stato) che accende o spegne il led in base al valore del parametro.

Libreria buzzer:

Bisogna creare due classi, una “Buzzer.cpp” e una “Buzzer.h”.

Nella classe “Buzzer.h” bisogna dichiarare un attributo, il pin del Buzzer (\_pin). Bisogna dichiarare 5 metodi,

il metodo Buzzer(int pin), il metodo setOnBuzzer(), il metodo setOffBuzzer(), il metodo frequence(int range, int potValue) e il metodo setOnReverseBuzzerFrequenze(int range, int potValue).

Nella classe “Led.cpp” invece come prima cosa bisogna includere la classe “Buzzer.h” e “Arduino.h”.

Il metodo Buzzer (int pin) è il costruttore che definisce il pin del Buzzer, il metodo setOnBuzzer() e setOffBuzzer() che rispettivamente accendono e spengono il Buzzer, il metodo frequence(int range, int potValue), il quale ritorna la frequenza che verrà applicata al buzzer e infine il metodo setOnReverseBuzzerFrequenze(int range, int potValue), il quale setta la frequenza al buzzer all'inverso