Manuale

# Potenziometro

La prima libreria che abbiamo realizzato è per il potenziometro, questa presenta due metodi pubblici. Il primo è “Potenziometro (int pin)”; questo metodo costruttore riceve come parametro il pin a cui è collegato il potenziometro. Fatto ciò esso si occupa di impostare il pin di lettura del valore del potenziometro in base ai pin analogici del Digispark. Il pin 2 viene impostato a 1, il 4 a 2, il 5 a 0 e il 3 rimane 3. Il secondo metodo, chiamato “valoreLetto()” non riceve nessuna parametro e ritorna un int. La prima cosa che fa è impostare la variabile “sensorValue” al valore letto dal pin analogico del potenziometro specificato tramite il costruttore. Successivamente la variabile viene divisa per 4, questo perché il valore del potenzimetro va da 0 a 1023 mentre quello del led va da 0 a 255. Il metodo ritorna poi questa variabile che sarà il valore del led. Quindi nel caso che il potenziometro venga roteato al punto da possedere una valore di 1023, la variabile contenete il valore verrà divisa per 4 e raggiungerà 255, che sarà poi il valore ritornato dal metodo e che imposterà il led alla luminoosità massima.

## Esempio 1 – Fade

Il primo esempio realizzato fa dissolvere la luminosità del led a dipendenza della rotazione del potenziometro. Per cominciare andiamo ad includere la nostra libreria chiamata “Potenziometro.h”. In un secondo momento inizializziamo un oggetto associato alla classe Potenziometro e di nome potenziometro e successivamente dichiariamo una variabile led che conterrà la un valore intero (senza virgola) che indica la porta a cui è collegato il led. Nel setup creiamo una variabile potenziometro che andiamo poi ad istanziare e tramite il metodo pinMode passiamo come parametro la variabile led che contiene appunto il numero del pin che vogliamo impostare come output. Nel metodo loop che contiene una sequenza di istruzioni che viene ripetuta continuamente andiamo a “scrivere” sul led il valore che leggiamo dal potenziometro. Il primo parametro del metodo indica il pin su cui scrviviamo il valore e il seconda indica il valore da scrivere. Noi usiamo il metodo valoreLetto() per sapere il valore del potenziometro.

## Esempio 2 – Blink

Il secondo esempio che abbiamo realizzato fa lampeggiare il led a dipendenza della rotazione del potenziometro. Per cominciare andiamo ad includere la nostra libreria chiamata “Potenziometro.h”. In un secondo momento inizializziamo un oggetto associato alla classe Potenziometro e di nome potenziometro e successivamente dichiariamo una variabile led che conterrà la un valore intero (senza virgola) che indica la porta a cui è collegato il led. Nel setup creiamo una variabile potenziometro che andiamo poi ad istanziare e tramite il metodo pinMode passiamo come parametro la variabile led che contiene appunto il numero del pin che vogliamo impostare come output. Nel metodo loop andiamo inizialmente a far fare un pausa della durata del valore letto dal potenziometro (in millisecondi) moltiplicato per due. Il valore andrà quindi da 0 a 510 che saranno i millisecondi di pausa. Successivamente accendiamo il led tramite il metodo digitalWrite. Come primo parametro passiamo il led da accendere e come seconda parametro la parola chiave “HIGH” che indica appunto che il led è acceso. Adesso svolgiamo come in precedenza una pausa della durata in millisecondi letta dal potenziometro e poi moltiplicata per due. Infine sempre con il metodo digitalWrite andiamo a spegnere il led usando questa volta la parola chiave “LOW” come secondo parametro per indicare che il led viene spento.

## Esempio 3 – FadePro

Il terzo esempio che abbiamo realizzato è una versione alternativa del primo esempio in cui andiamo a far dissolvere la luminosità del led in base alla rotazione del potenziometro. Per cominciare andiamo ad includere la nostra libreria chiamata “Potenziometro.h”. In un secondo momento inizializziamo un oggetto associato alla classe Potenziometro e di nome potenziometro e successivamente dichiariamo una variabile led che conterrà la un valore intero (senza virgola) che indica la porta a cui è collegato il led. Nel setup creiamo una variabile potenziometro che andiamo poi ad istanziare e tramite il metodo pinMode passiamo come parametro la variabile led che contiene appunto il numero del pin che vogliamo impostare come output. Nel metodo loop creiamo una variabile “valore” in cui leggiamo il valore del potenziometro (da 0 a 255) ora tramite i costrutti if ed else andiamo a controllare se il valore del potenziometro è minore di 128 o maggiore. Nel primo caso scriviamo accendiamo il led con un intensità pari al valore del potenziometro moltiplicato per due. Nel secondo caso scriviamo nel led il valore massimo moltiplicato per due, meno il valore letto moltiplicato per 2 (così a meta “corsa” il led comincia a spegnersi).

# Servo Motor

La seconda libreria che abbiamo realizzato è per il servo motor, questa presenta quattro metodi pubblici. Il primo è “ServoMotor(int pin, int velocita)”; questo metodo costruttore riceve come parametro il pin a cui è collegato il potenziometro e si occupa di impostare il pin di lettura del valore del potenziometro in base ai pin analogici del Digispark. Il pin 2 viene impostato a 1, il 4 a 2, il 5 a 0 e il 3 rimane 3. Il secondo parametro che il metodo riceve è “velocita”, indica la velocità di movimento dell’elica del servo motor. Successivamente troviamo il metodo “setVelocita(int velocita)” che serve a modificare la velocità del motore in base al valore del parametro “velocita” e il metodo getPosizione() che ritorna la posizione mappata del motore. L’ultimo metodo, “startServo()”, serve ad avviare il motore del servo motor.