



Arduino

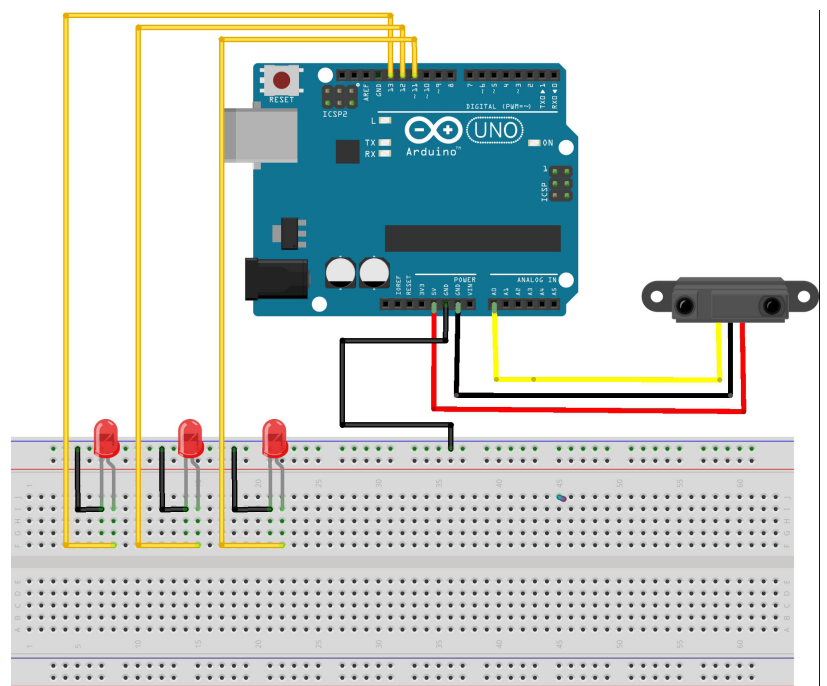
Tutorial #6 - Sensore Sharp + semaforo

materiale utilizzato – Arduino, cavi elettrici, breadboard, sensore Sharp, 3 led.

Per vedere tutti i tutorial: www.marcopucci.it/arduino/

In questo tutorial vediamo un' applicazioni pratica sull'utilizzo del sensore Sharp ([vedi tutorial precedente](#)) e inoltre continueremo la parte informatica riguardante il controllo IF inserendo più controlli sui dati che provengono dal sensore. In questo modo siamo in gradi di gestire più azioni a seconda che un oggetto sia più o meno distante dal sensore. Se non si è in possesso di un sensore Sharp vi invito a seguire lo stesso il tutorial per imparare la parte informatica (è possibile sostituire il sensore Sharp con un semplice potenziometro in modo da simulare il flusso di dati in ingresso ad Arduino, [leggi il primo tutorial](#))

In questo progetto realizziamo un semaforo in grado di avvisarci se, in caso di parcheggio all'interno del garage, stiamo per graffiare la nostra auto contro il muro. In questo caso dobbiamo posizionare il nostro Arduino, il sensore e i tre led (rosso, giallo e verde) ben visibili sul muro di ingresso. Il sensore Sharp rileverà la distanza della macchina all'ingresso del box: se la distanza è maggiore di 30 cm Arduino accenderà il led verde, se la distanza diminuisce a 20 cm accenderemo il led giallo, se la distanza è minore di 10 cm allora avviseremo il conducente accendendo il led rosso. Nelle immagini qui sotto il circuito che dovete realizzare e un piccolo schema di come deve essere installato.





Circuito elettrico

Realizziamo il circuito elettrico:

Collegiamo i tre piedini piccoli dei Led al segnale GND.

I piedini lunghi vanno collegati a tre Pin digitali di Arduino, 9, 10 e 11. In questo modo possiamo comandare l'accensione e lo spegnimento dei tre Led in base al segnale proveniente dal sensore.

Il sensore deve essere collegato al segnale positivo 5V (pin nella parte inferiore di Arduino), al pin negativo GND e all'ingresso di uno dei pin analogici di Arduino, il Pin 0 (in basso a destra).

Informatica

Apriamo il programma di Arduino e scriviamo il nuovo codice:

```
int ledpin_verde= 13;
int ledpin_arancione = 12;
int ledpin_rosso = 11;
int pin0=0;
int lettura_sensore = 0;
```

```
void setup(){
  pinMode(pin0, INPUT);
  pinMode(ledpin_verde, OUTPUT);
  pinMode(ledpin_arancione , OUTPUT);
  pinMode( ledpin_rosso , OUTPUT);
  Serial.begin(9600);
}
```

```
void loop()
{
  Serial.flush();
  delay(100);
  lettura_sensore = analogRead(pin0);
  Serial.println(lettura_sensore);
```

```
if (lettura_sensore<100)
{
  digitalWrite(ledpin_verde, 1);
  digitalWrite(ledpin_arancione, 0);
  digitalWrite(ledpin_rosso, 0);
}
```

```
if ((lettura_sensore>101)&&(lettura_sensore<300))
{
  digitalWrite(ledpin_verde, 0);
  digitalWrite(ledpin_arancione, 1);
  digitalWrite(ledpin_rosso, 0);
}
```

```
if ((lettura_sensore>301)&&(lettura_sensore<600))
{
  digitalWrite(ledpin_verde, 0);
  digitalWrite(ledpin_arancione, 0);
  digitalWrite(ledpin_rosso, 1);
}
```

```
}
```

Dichiariamo i pin che gestiscono i tre led e il pin per ricevere dati dal sensore.

Il pin0 è di INPUT perchè riceve dati dal sensore, gli altri tre sono di OUTPUT perchè devono accendere o spegnere un Led.

lettura_sensore è la variabile in cui arriveranno tutti i dati del sensore.

Se il valore della variabile lettura_sensore è più piccolo di 100 (la macchina è lontana più di 30cm dal sensore allora accendo il Led verde e spengo gli altri due.

Se la macchina si avvicina al sensore allora accendo il Led arancione e spengo gli altri due.

Il codice **&&** indica che per far verificare la condizione IF devono essere vere le due condizioni, ovvero il valore del sensore deve essere sia >101 **E** <300.

Se la macchina è molto vicina al sensore allora accendo il Led rosso e spengo il verde e l'arancione.

Uplodiamo il codice sulla scheda Arduino.



Nel codice che abbiamo scritto è presente una serie di controlli IF posizionati uno sotto l'altro. In questo modo possiamo decidere di effettuare più di un controllo sul flusso di dati proveniente dal sensore.

Il numero di controlli che possiamo inserire è illimitato, dipende dal numero di azioni che vogliamo far compiere ad Arduino.

```
if (lettura_sensore<100)
{
```

AZIONE 1

```
}
```

```
if ((lettura_sensore>101) && (lettura_sensore<300))
{
```

AZIONE 2

```
}
```

```
if ((lettura_sensore>301) && (lettura_sensore<600))
{
```

AZIONE 3

```
}
```

Molto importante è il comando **&&** posto all'interno del controllo IF.

Il simbolo **&&** rappresenta l'operatore logico **AND**. Il controllo IF è vero solamente se la prima condizione è vera (**lettura_sensore>101**) **E** contemporaneamente è vera anche la seconda (**lettura_sensore<300**).

Se uno dei due controlli non si verifica allora l'**AZIONE 2** non verrà eseguita.

Installazione

Una volta uplodato il software e testato il funzionamento del progetto dobbiamo installarlo all'interno del garage. Non possiamo continuare ad alimentare la scheda Arduino con la USB del computer. In questo caso dobbiamo comprare un piccolo alimentatore da 9V o da 12 V e collegarlo ad una presa elettrica (nel prossimo tutorial alimenteremo Arduino con una batteria).



Questo tipo di alimentatori si trovano in tutti i negozi di elettronica e hanno un costo che si aggira intorno ai 10 euro. La presa nera va collegata nell'alimentazione di Arduino (vicino al connettore che utilizziamo per uplodare il codice).

I tre led e il sensore dobbiamo saldarli in una schedina elettrica forata in modo da poterli appendere al muro del garage.