



Arduino - Tutorial #5 - Sensore Sharp

materiale utilizzato – Arduino, cavi elettrici, breadboard, sensore Sharp (o un altro tipo di sensore)

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

In questo tutorial vediamo come utilizzare un sensore di distanza Sharp (se non si è in possesso di questo sensore va bene qualsiasi altro sensore per imparare comunque la parte informatica di controllo dei dati), molto utilizzato in robotica per individuare ostacoli, oppure nelle installazioni interattive per conoscere la posizione di una persona e attivare un evento in base a questa informazione. Esistono diversi modelli dei sensori Sharp e variano in base alla distanza massima che riescono a individuare: 30 cm - 80 cm - 150 cm - il prezzo si aggira intorno ai 15 euro. Può essere acquistato online presso questi due siti:

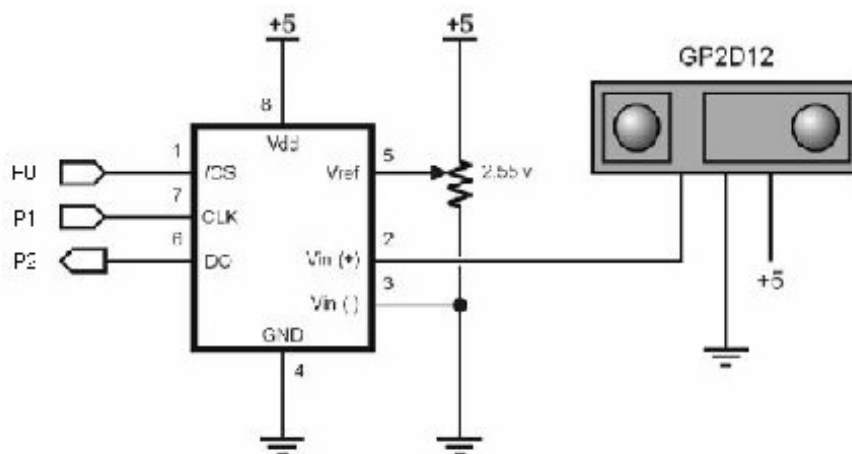
<http://www.robot-italy.com/it/sensors/infrared.html>

http://www.futurashop.it/index.php?route=product/category&path=4024_3189_640

o su ebay: http://www.ebay.it/sch/i.html?_trksid=p2050601.m570.l1313&_nkw=sensori+sharp&_sacat=0&_from=R40

Quando compriamo un sensore, per evitare di bruciarlo immediatamente al primo collegamento, dobbiamo cercare in rete il suo schema elettrico (o datasheet) per verificare i giusti collegamenti con la scheda Arduino.

Qui sotto un'immagine del sensore Sharp (qualsiasi modello). Inseriamo su google 'datasheet Sharp sensor'. Se siamo fortunati, nella ricerca per immagini di google, apparirà subito il nostro schema elettronico. Altrimenti dobbiamo approfondire la ricerca e cercare il pdf o un tutorial con un esempio di collegamento. Non andate a caso perchè un sensore di 17 euro impiega 1 secondo a bruciarsi se invertiamo i collegamenti.



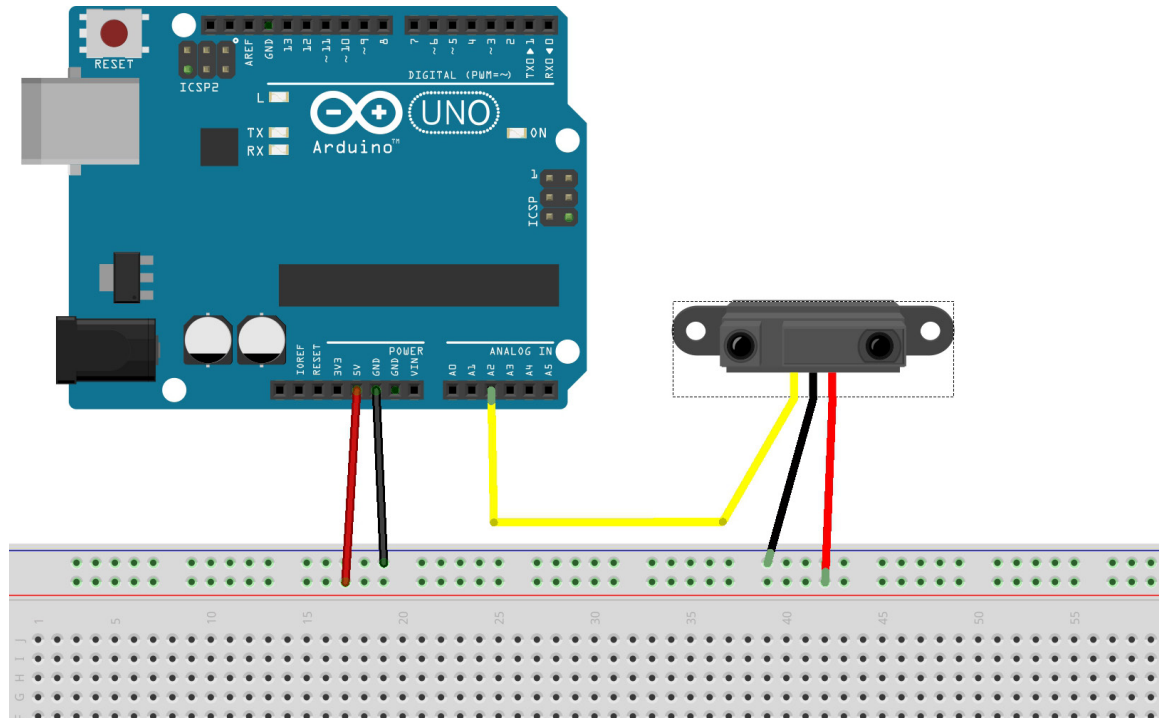
Nell'esempio qui sopra (il primo che risulta dalla ricerca su Google) già possiamo capire i collegamenti che dobbiamo effettuare. Lasciamo stare la parte del circuito di sinistra. A noi serve esattamente capire come collegare i tre piedini del sensore Sharp. In questo modo sappiamo che, se posizioniamo il sensore nella esatta posizione del disegno, il pin di destra va alimentato con 5V (dal pin 5V di Arduino), il pin centrale va collegato al Pin GND di Arduino e quella a sinistra va nel Pin di lettura di Arduino.



Il sensore Sharp ha tre piedini: 2 servono per alimentarlo (la tensione positiva e il segnale di massa GND), il terzo è il pin che invia i dati del sensore ad Arduino che non deve fare altro che leggerlo ([vedi tutorial precedente](#)).

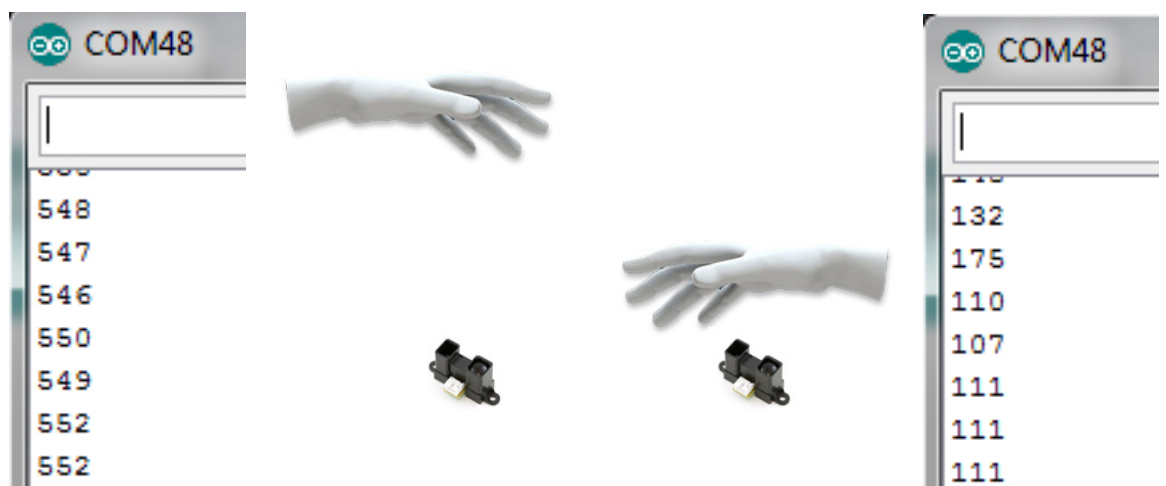
Quasi tutti i sensori analogici devono essere alimentati e poi collegati ad Arduino con un cavetto nel Pin di lettura. La cosa importante è sapere quale dei Pin collegare nel modo giusto.

Una volta effettuati i collegamenti come riportato nello schema qui sotto abbiamo finito.



Apriamo lo sketch del [tutorial precedente](#) (quello che legge i dati dei sensori) per verificare se il sensore funziona e se i suoi collegamenti sono stati fatti nel modo giusto. Uplodiamo il codice e apriamo il Serial Monitor. I dati a cascata (come nella figura qui sotto) dovrebbero cambiare a seconda se la nostra mano è posizionata vicino o lontano al sensore. Se ad esempio stiamo utilizzando il sensore da 30 cm con la mano posizionata molto vicina al sensore il valore sul Serial Monitor sarà di '40'; con la mano posizionata a circa 30 cm il valore sarà di '550' circa.

N.B.: il valore numerico del Serial Monitor non è il valore della distanza in centimetri. È un numero che ci aiuta a capire la distanza dalla nostra posizione al sensore.

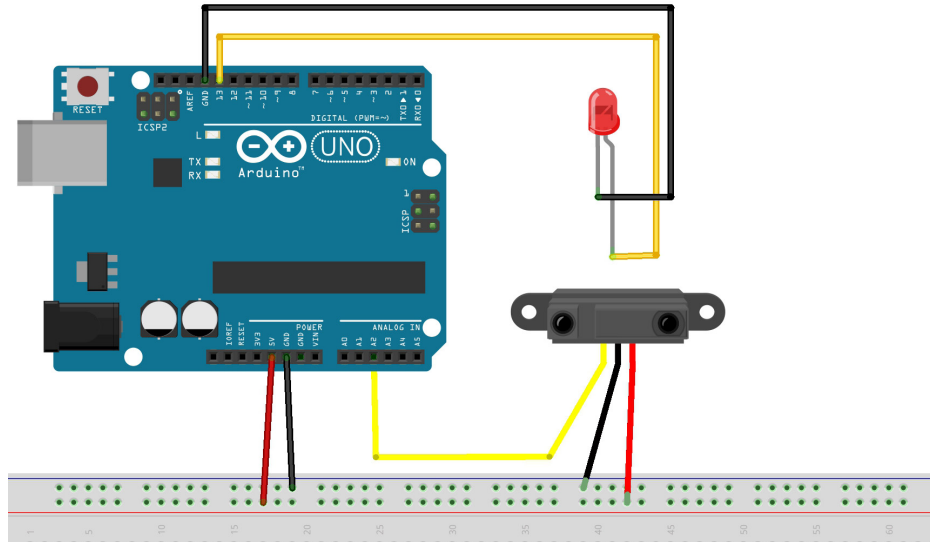




Procediamo ora con la spiegazione di uno dei comandi fondamentali dell'informatica, il controllo **IF**. Ci aiuta a comporre istruzioni come ad esempio: **se** il numero del Serial Monitor è più piccolo di '150' **allora** accendiamo un led, altrimenti lo spegniamo. Scriviamo un nuovo sketch da capo (anche se sarà molto simile a quello precedente consiglio per le prime volte di scrivere il codice sempre a mano per memorizzarlo, evitando il copia-incolla di intere stringhe) e realizziamo il circuito qui sotto.

INFORMATICA: **CONTROLLO IF**

Qui sotto il codice completo. La parte in blu è il controllo IF.



```
int pin2= 2;
int LedPin=13;
int lettura_sensore = 0;

void setup(){
  pinMode(pin2, INPUT);
  pinMode(LedPin, OUTPUT);
  Serial.begin(9600);
}

void loop()
{
  Serial.flush();

  delay(100);
  lettura_sensore = analogRead(pin2);

  if (lettura_sensore>100) {
    digitalWrite (LedPin, 1);
  }
  else {
    digitalWrite (LedPin, 0);
  }

  Serial.println(lettura_sensore);
}
```



Il controllo IF ha bisogno di alcune regole di sintassi che devono essere rispettate.
Il comando if deve essere scritto in minuscolo.

```
if (lettura_sensore>100){  
  
}
```

Questo codice verifica **SE** la variabile **lettura_sensore** è più grande del valore '100' (nel nostro caso significa che la mano è distante dal sensore) allora Arduino compie tutte le operazioni che vengono scritte all'interno delle due parentesi graffe { }.

Nel nostro esempio abbiamo inserito il comando già visto in precedenza **digitalWrite** che accende il led posizionato nel pin 13.

Subito dopo è presente il comando **else { }**. Letteralmente vuol dire ALTRO e indica che se il valore del sensore è maggiore di 100 accendi il led, **ALTRIMENTI** (else) compi le azioni presenti tra le sue parentesi graffe.

In questo esempio il comando è sempre un digitalWrite che però spegne il led.

Il primo esempio con il controllo if è terminato. Nei prossimi tutorial verrà utilizzato quasi sempre come controllo sui dati che arriveranno dai vari sensori.

Nel tutorial n.6 realizzeremo un progetto con il sensore Sharp. Consiglio l'acquisto di un sensore Sharp per il facile utilizzo e le enormi potenzialità di utilizzo.

Più avanti lo utilizzeremo per i tutorial di robotica.