

Usare un pulsante per controllare un Led

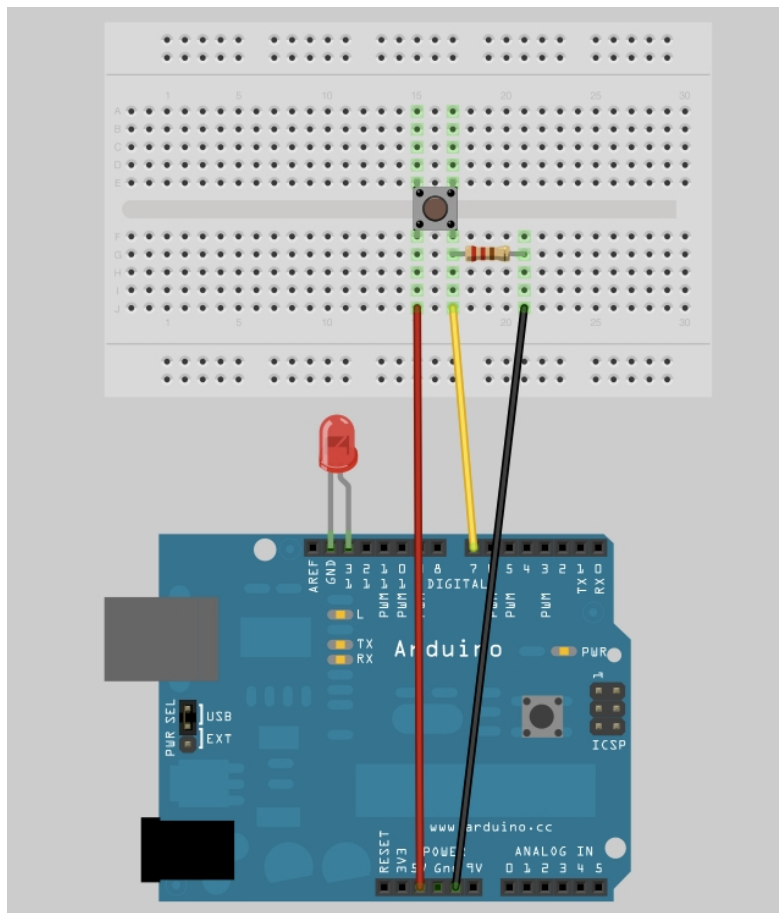
Per controllare lo stato di un interruttore, esiste un'istruzione di Arduino che ora useremo: la funzione "digitalRead()".

"digitalRead()" controlla che al pin specificato tra le parentesi sia applicato un voltaggio e restituisce un valore pari a HIGH o LOW, a seconda di quanto rileva. Grazie a "digitalRead()" possiamo "porre una domanda" ad Arduino e ricevere una risposta che può essere conservata da qualche parte nella memoria e usata per prendere decisioni, subito o più tardi.

Schema di collegamento

In questo caso ci occorrono i seguenti componenti:

- 1) 1 pulsante MicroSwitch
- 2) 1 Resistenza da 10 Kohm
- 3) 1 Led

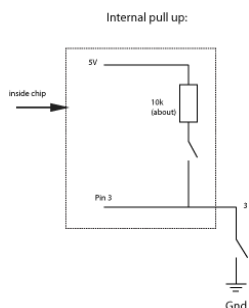
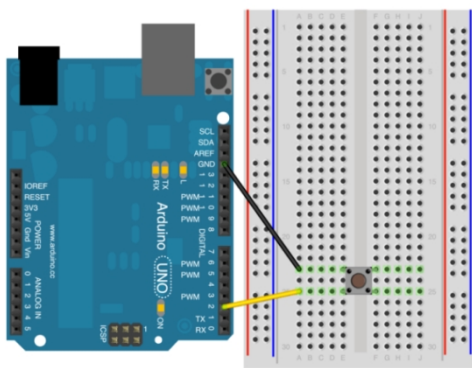


Codice (Sketch) per l'accensione del Led

```
01.#define Led 13 // Definisco il Pin del Led
02.#define Pulsante 7 // Il Pin di Input dove è collegato il pulsante
03.int Valore = 0; // Si userà Valore per conservare lo stato del Pin di Input
05.void setup()
06.{
07.pinMode(Led, OUTPUT); // Segnala ad Arduino che Led è un Output
08.pinMode(Pulsante, INPUT); // E che Pulsante è un Input
09.}
11.void loop(){
12.Valore = digitalRead(Pulsante); // Legge il valore dell'input e lo conserva
13.if (Valore == HIGH) // Controlla che l'Input sia HIGH ( Pulsante Premuto )
14.{
15.digitalWrite(Led, HIGH); // Accende il Led
16.}
17.else
18.{
19.digitalWrite(Led, LOW);
20.}
21.}
```

Dopo la parola chiave IF dovete scrivere una " CONDIZIONE " tra parentesi e se la " risposta ", o il risultato, è vero, viene eseguito il primo blocco di codice; altrimenti, viene eseguito il blocco di codice che segue ELSE. Notate che invece di = ho usato il simbolo ==. Questo viene usato quando si confrontano due entità e restituisce TRUE o FALSE; il primo simbolo invece assegna un valore a una variabile.

Seconda soluzione: ATTENZIONE LA RESISTENZA DI PULL UP NEGA LA LOGICA !!!!



```
pinMode(2, INPUT_PULLUP);
```

collega ad una R interna
di 10 kohm

```
void setup() {
  //start serial connection
  Serial.begin(9600);
  //configure pin2 as an input and
  //enable the internal pull-up resistor

  pinMode(2, INPUT_PULLUP);
  pinMode(13, OUTPUT);
}

void loop() {
  //read the pushbutton value into a variable
  int sensorVal = digitalRead(2);
  //print out the value of the pushbutton
  Serial.println(sensorVal);
  if (sensorVal == HIGH)
    // SE IL PULSANTE NON È PREMUTO
  {
    digitalWrite(13, LOW);
  }
  else {
    digitalWrite(13, HIGH);
  }
}
```