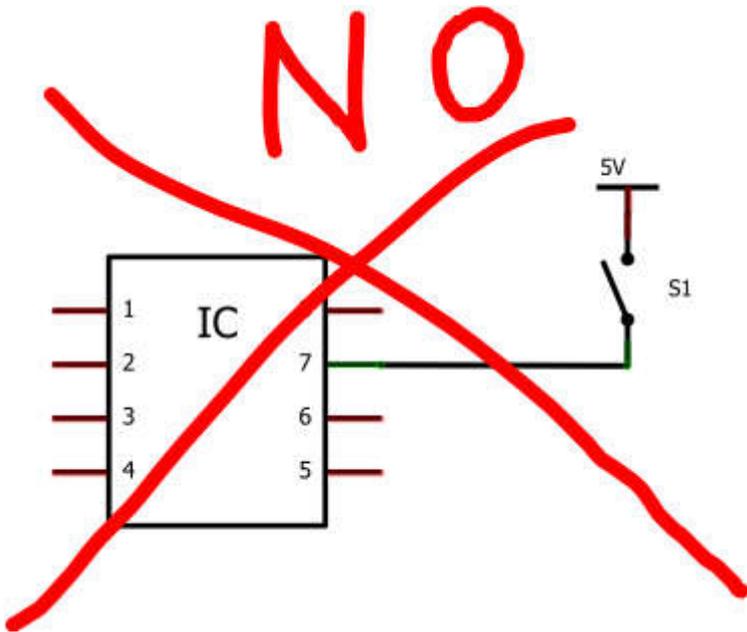
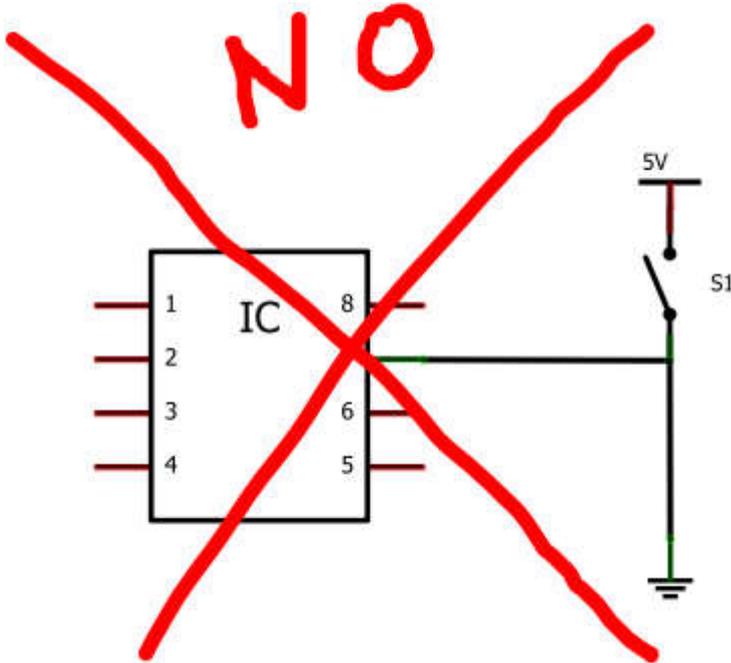


## ARDUINO – RESISTENZE DI PULL-UP & PULL-DOWN



Nello schema sopra riportato con pulsante premuto avremo un valore HIGH sul pin di ingresso, mentre con circuito aperto avremo valori in ingresso oscillanti.

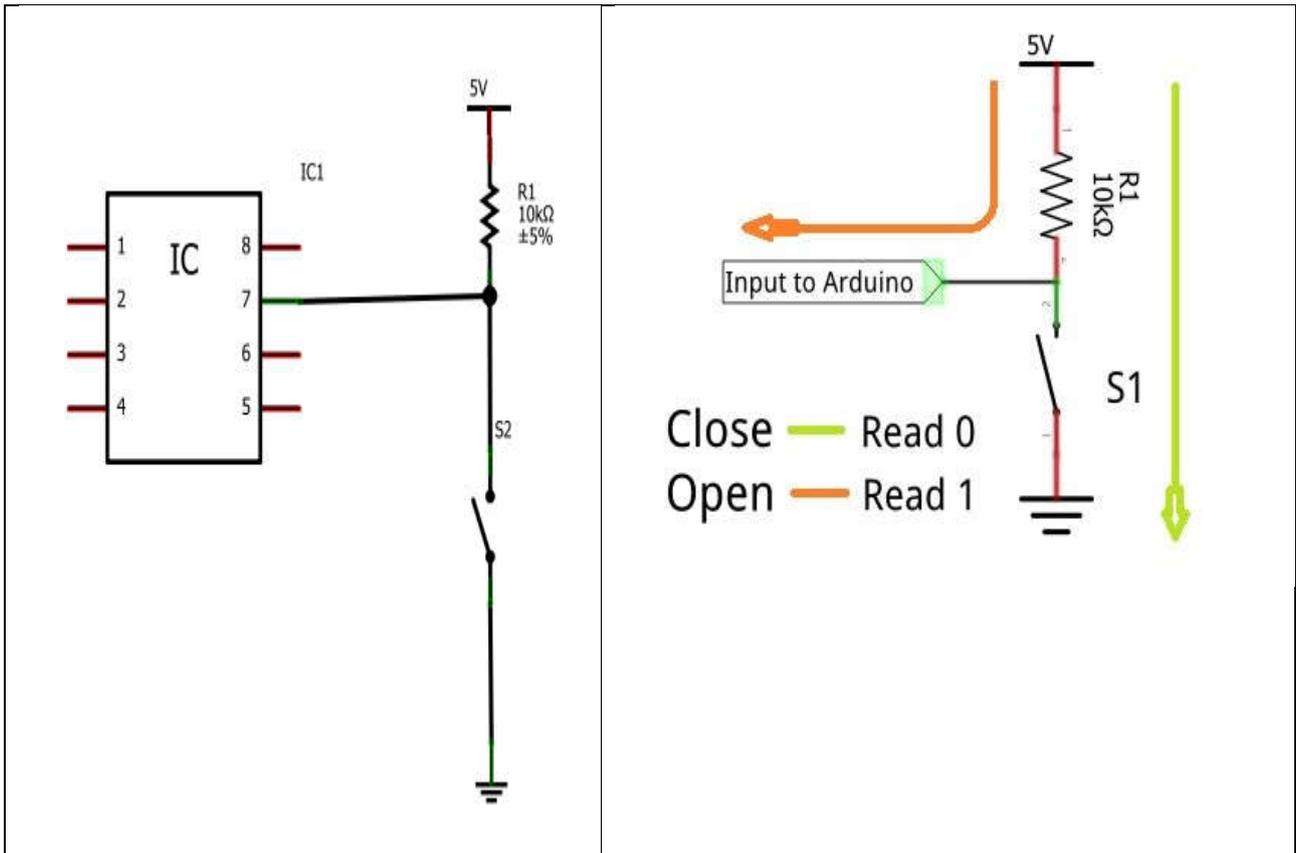
Per questa ragione dobbiamo far sì che il nostro pin sia sempre connesso o a 5V o a 0V. Potremmo ovviare il problema connettendo il circuito aperto a terra:



Ma in questo caso il problema nasce quando chiudiamo il circuito che causa un bel cortocircuito con conseguente danneggiamento del nostro arduino.

### Resistenze di pull up

Per fare questo possiamo utilizzare il seguente schema:

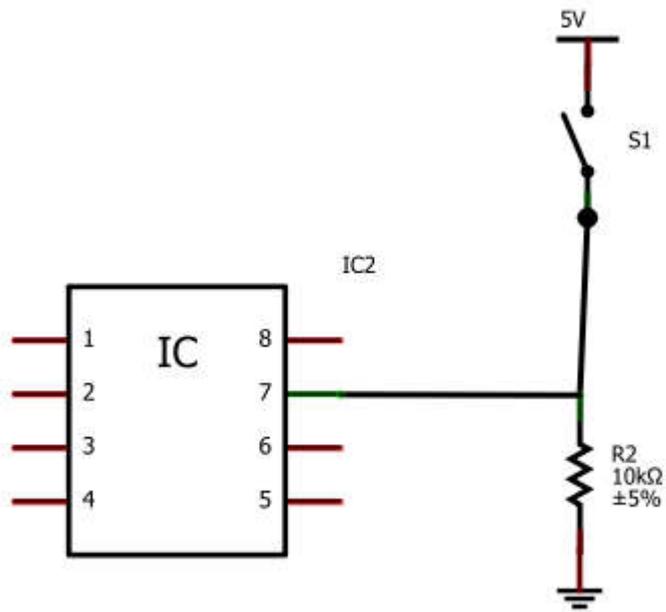


Solitamente per i circuiti di pull up si utilizzano resistenze da 10K, ma se vogliamo possiamo utilizzare resistenze da 47K o superiori.

Utilizzando una resistenza di pull up avremmo uno stato HIGH con il circuito aperto (pulsante non premuto) e LOW con circuito chiuso (pulsante premuto)

#### Resistenze di pull down

In maniera analoga alle resistenze di pull up possiamo utilizzare resistenze di pull down. In questo caso il circuito aperto porterà il pin in stato LOW e il circuito chiuso in stato HIGH.



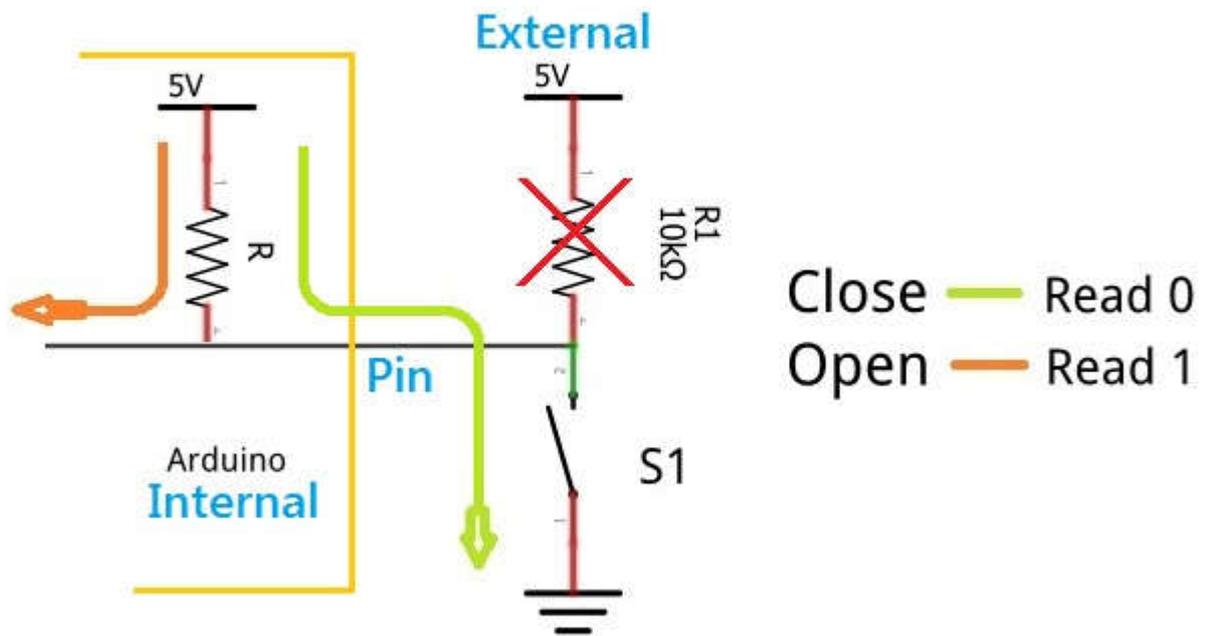
### Resistenza di pull up interna

E' possibile attivare una resistenza di pull up interna all'Arduino con l'istruzione

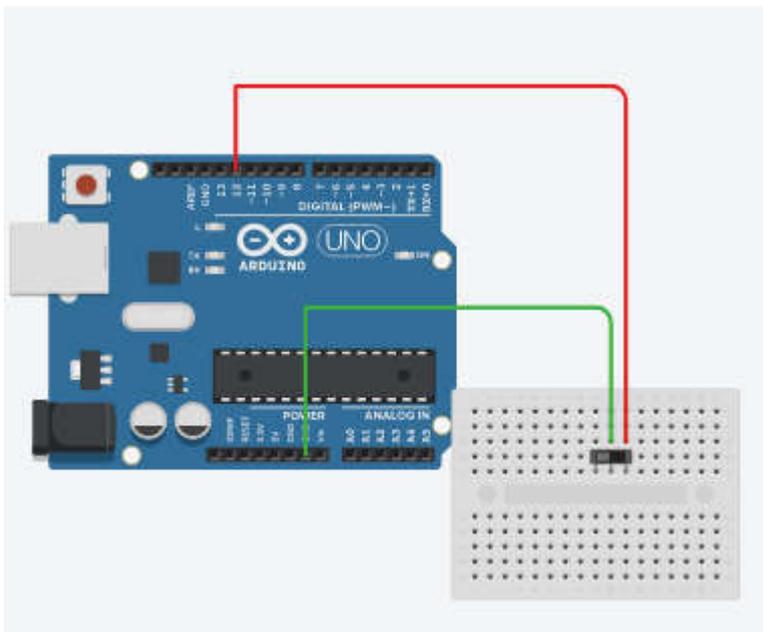
```
pinMode(pin, INPUT);           // set pin to input
digitalWrite(pin, HIGH);       // turn on pullup resistors
```

In questo modo viene attivata sul pin in ingresso una resistenza di pull up pari a 30K.

Rispetto alle resistenze di pullup reali quelle software hanno il limite di avere un valore fisso che in certi casi potrebbe non essere adeguato.



Nello schema seguente vediamo un utilizzo pratico della resistenza interna di pull-up di Arduino.



```
void setup()
{
  pinMode(13, OUTPUT); // set pin 13 as an output, the pin having onboard LED.
  pinMode(12, INPUT_PULLUP); // set pin 12 as an input with internal pull up.
}

void loop()
```

```
{  
  int x = digitalRead(12); // read the pin number 12  
  if( x == 1)  
  {  
    digitalWrite(13, HIGH); // turn the pin 13 HIGH means on (set)  
  }else  
  {  
    digitalWrite(13, LOW); // turn the pin 13 LOW means off (reset)  
  }  
  delay(100); // delay for 100 milli second  
}
```