

05 - lampeggio senza uso dell'istruzione delay

Questo esercizio e' interessante per via del fatto che riesce a temporizzare l'accensione e lo spegnimento di un led senza ricorrere all'istruzione delay. Questa tecnica, basta sulla misurazione del tempo trascorso dall'accensione o dal reset di Arduino, consente di lasciare acceso (o spento) un led per qualche tempo **senza pero' interrompere, nell'attesa, il funzionamento del programma**. Si tratta di una tecnica piuttosto importante poiche' l'istruzione delay blocca il programma e rende quindi Arduino insensibile ai segnali nel frattempo pervenuti dai sensori.

Nota: Questo esercizio e questa nota sono parte di una serie che vede protagonisti arduino ed alcuni dei componenti ad esso collegabili. Per la maggior parte degli esercizi e' anche disponibile un filmato su youtube.

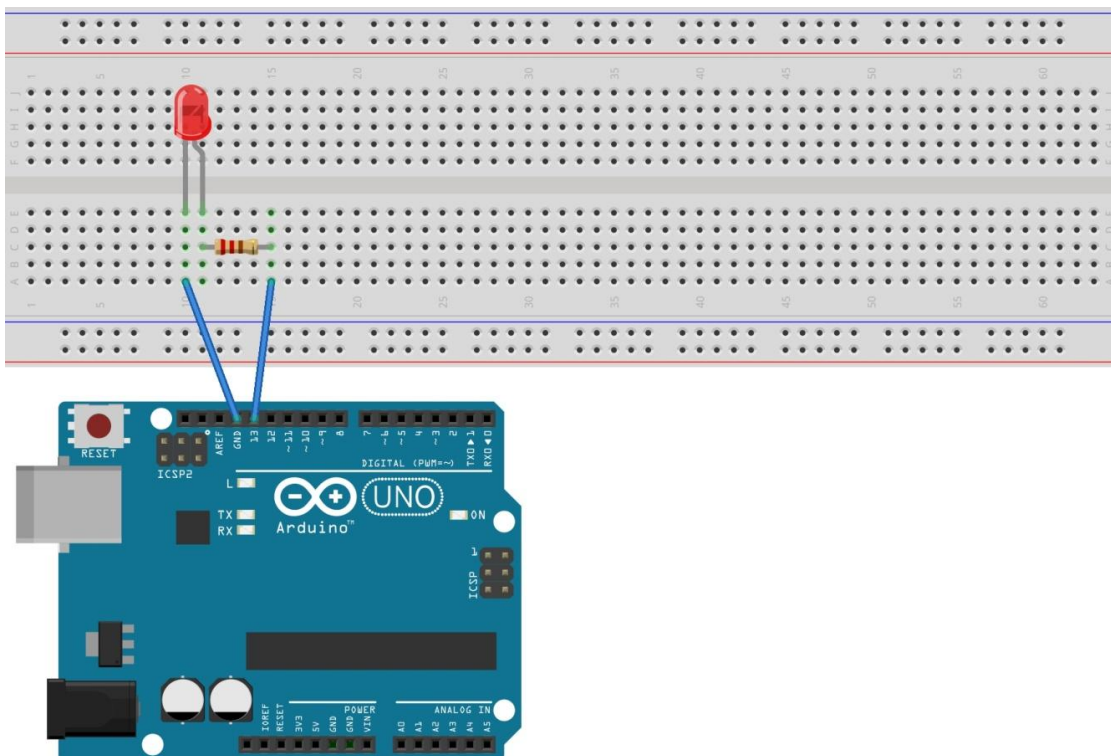
- [Esercizi facenti parte della raccolta](#)
- [Filmati presenti su youtube](#)
- [Informazioni su arduino e sui componenti collegabili \(PDF scaricato nell'area di download\)](#)
- [Breve manuale di programmazione \(PDF scaricato nell'area di download\)](#)

Per eventuali chiarimenti o suggerimenti sul contenuto di questa scheda scrivere a giocarduino@libero.it

Materiali

- 1 breadboard
- 1 resistenza da 220 ohm
- 1 led
- 2 cavi

Schema



Arduino: lampeggio senza uso delay

Programma

/ Attenzione: facendo il copia/incolla dal PDF all'IDE si perde la formattazione del testo. Per rendere piu' facilmente leggibile il programma e' opportuno formattarlo subito dopo il trasferimento nell'IDE, premendo CTRL+T*

Attivare o disattivare un led o un qualunque altro attuatore senza ricorrere all'utilizzo dell'istruzione delay e' piuttosto importante poiche' il delay ferma l'elaborazione e quindi non consente di rilevare l'eventuale variazione di stato di un sensore (esempio un pulsante o anche un tilt) avvenuta nel frattempo. Il circuito e' composto da un led con il negativo associato alla terra (gnd) ed il positivo ad una resistenza da 220 ohm a sua volta associata alla porta 13

Nota: poiche' Arduino gia' dispone di un led associato alla porta 13, il programma puo' girare anche senza alcun componente collegato/*

```
*****definizione delle variabili*****
int statodelled = LOW;      // statodelled e' una variabile, inizialmente LOW (spento) utilizzata
                            // per pilotare lo stato del led
long tempotrascorso = 0;   // tempotrascorso e' la variabile utilizzata per memorizzare il tempo
                            // trascorso dal momento di accensione della scheda. Poiche' il
                            // tempo e' misurato in millisecondi ed i secondi passano
                            // velocemente, e' stata usata una variabile di tipo long, che
                            // impiega circa 52 giorni prima di andare in overflow
long precedente = 0;       // variabile utilizzata per memorizzare il momento della
                            // precedente accensione (o spegnimento) del led
int durata = 1000;        // durata del tempo di accensione o spegnimento (in millisecondi)

***** sezione di setup *****
void setup()
{
  pinMode(13, OUTPUT);    // definisce la porta digitale 13 come porta di output
}

***** sezione di loop *****
void loop()
{
  tempotrascorso = millis(); // inserisce nella variabile tempotrascorso il valore fornito dalla
                             // funzione millis () (tempo trascorso dall'accensione di Arduino)

  if(tempotrascorso - precedente > durata) // condizione di verifica del tempo trascorso
                                           // dall'ultima variazione di stato del led
  {
    precedente = tempotrascorso; // istruzione eseguita per condizione vera: memorizza il
                                  // momento in cui viene variato lo stato del led
    if (statodelled == LOW)      // istruzione eseguita per condizione vera: cambia il
                                  // contenuto della variabile statodelled inserisce HIGH
                                  // se era LOW e viceversa)
      statodelled = HIGH;
    else
      statodelled = LOW;

    digitalWrite(13, statodelled); // istruzione per condizione vera: pone il led nella
                                    // situazione prevista da statodelled
  }
}
```