

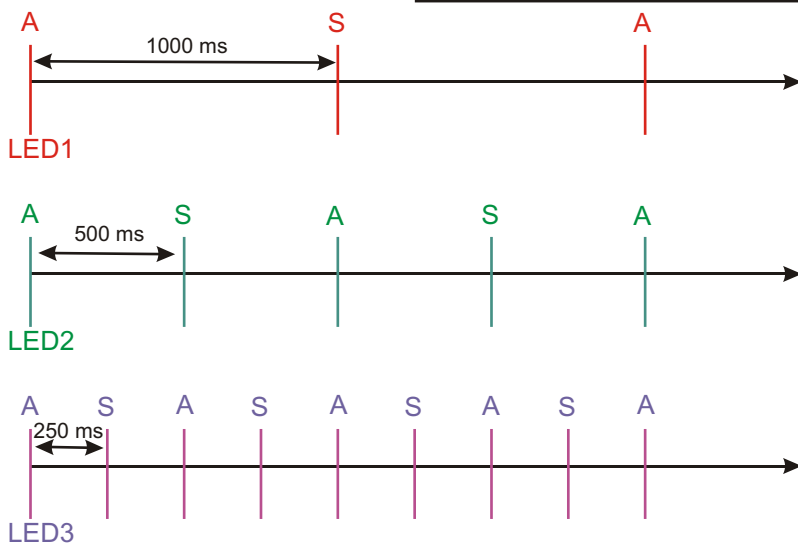
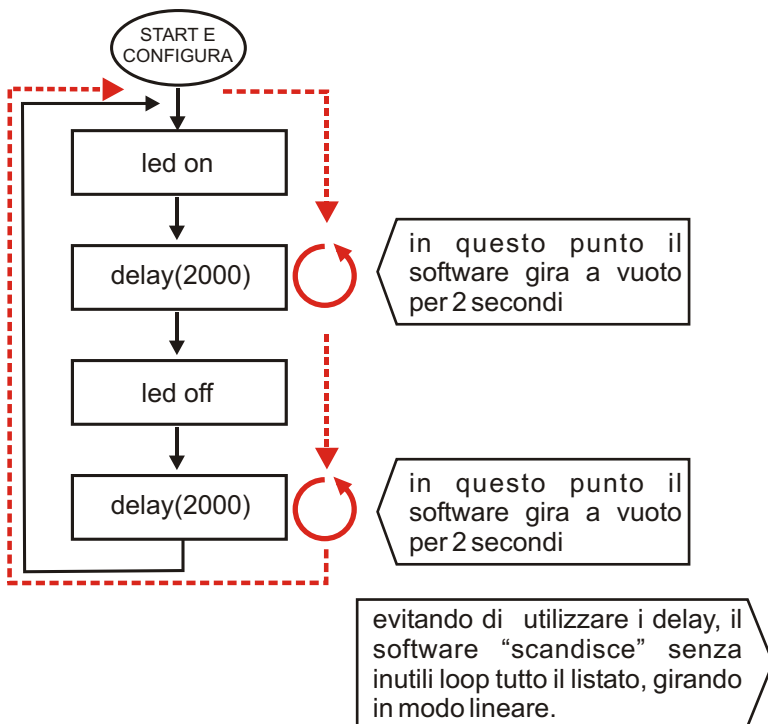
## SCHEDE DI APPROFONDIMENTO

### ACCENSIONE AD INTERMITTENZA DI TRE LED UTILIZZANDO LA FUNZIONE MILLIS

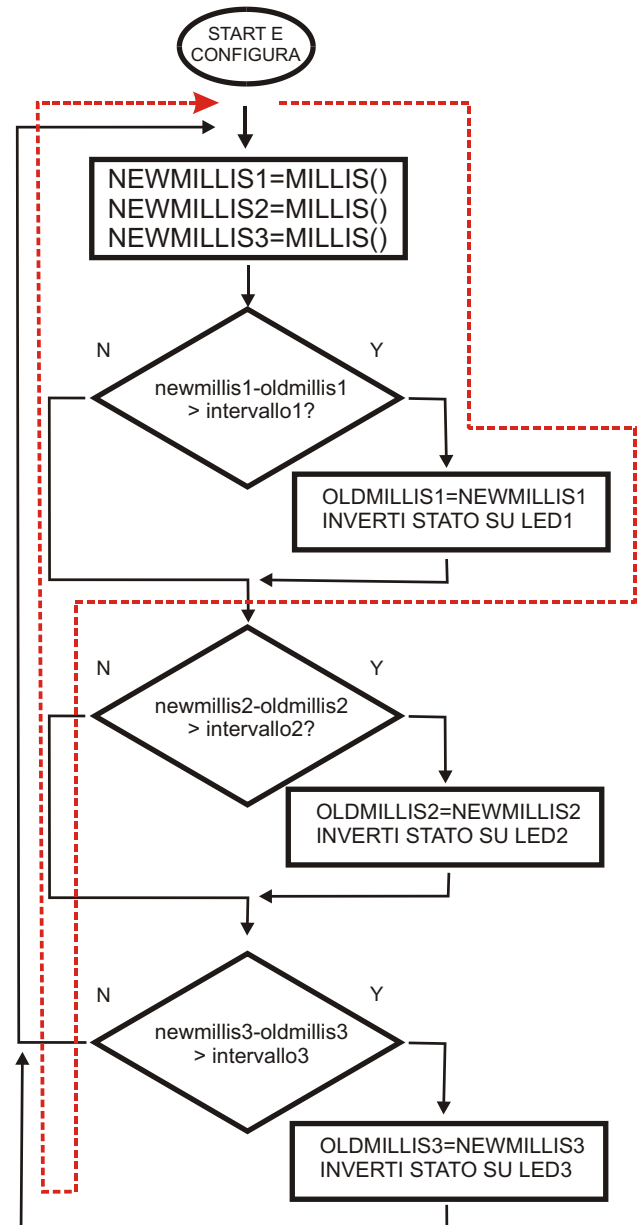
Nel libro “Laborobotica” (pag.91 Vol A), abbiamo studiato la funzione “millis()”, che ci ha permesso di accendere e spegnere un led senza utilizzare i delay. Millis() tiene nota del tempo in millisecondi dall’accensione del controllore fino al momento presente.

L’uso del delay, è spesso una procedura controproducente: infatti mentre il controllore è impegnato a “girare” a vuoto per esaurire il tempo, è in realtà “cieco”; non riesce cioè a far fronte ad eventuali richieste provenienti dai pin collegati a periferiche esterne. Più lunghi sono questi delay e più l’inconveniente diviene fastidioso.

Prendiamo ad esempio il caso in cui dobbiamo far lampeggiare tre led a frequenze differenti; studiando attentamente la temporizzazione dei delay, ci accorgeremo che se un led deve rimanere acceso o spento per lungo tempo, andrà a pregiudicare il funzionamento di tutti gli altri lampeggianti.



Ogni led si accenderà (A) e spegnerà (S) indipendentemente dagli altri due.



Ecco lo sketch per Arduino.

```
1 //millis con tre led
2 int led1=8; //si definiscono i pin
3 int led2=9;
4 int led3=10;
5
6 unsigned long newmillis1;// il nuovo valore di tempo
7 unsigned long newmillis2;
8 unsigned long newmillis3;
9 ;
10 long oldmillis1 = 0; //il vecchio valore di tempo
11 long oldmillis2 = 0;
12 long oldmillis3 = 0;
13
14 long intervallo1 = 1000; //gli intervalli di accensione
15 long intervallo2 = 500;
16 long intervallo3 = 250;
17
18 void setup()
19 {
20   pinMode(led1, OUTPUT); //la configurazione dei pin
21   pinMode(led2, OUTPUT);
22   pinMode(led3, OUTPUT);
23 }
24 void loop()
25 {
26   newmillis1 = millis();//si sincronizza newmillis con millis
27   newmillis2 = millis();
28   newmillis3 = millis();
29
30   if(newmillis1 - oldmillis1 > intervallo1) //se il tempo ha superato intervallo1
31   {
32     oldmillis1 = newmillis1; // assegna il nuovo valore di tempo
33     digitalWrite(led1,!digitalRead(led1)); //cambia lo stato sul led
34   }
35
36   if(newmillis2 - oldmillis2 > intervallo2)
37   {
38     oldmillis2 = newmillis2;
39     digitalWrite(led2,!digitalRead(led2));
40   }
41
42   if(newmillis3 - oldmillis3 > intervallo3)
43   {
44     oldmillis3 = newmillis3;
45     digitalWrite(led3,!digitalRead(led3));
46   }
47 }
```

Modificando queste variabili stabiliremo la frequenza di accensione-spegnimento dei led.