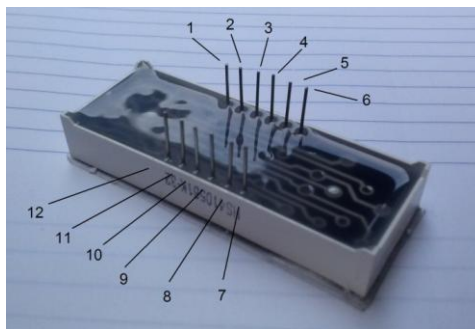


12 - timer da cucina, display a 4 cifre - kitchen timer, four digit display

(some notes at the end of this section)



Il display led da 4 cifre utilizzato in questo esercizio e' il modello HS410561K-32. Questo modello, e' caratterizzato da 4 anodi (uno per ogni cifra) e dai 8 catodi, comuni a tutte le cifre, che pilotano i vari segmenti. Tecnicamente si gestisce una cifra per volta e per effetto della velocita' del microcontrollore e della "latenza" nella luminosit  dei led, le cifre appaiono tutte contemporaneamente luminose. Per comodita' di esposizione i piedini sono stati numerati come in figura, da 1 a12. In pratica, tenendo il display con la parte luminosa rivolta in alto (come nella figura di sinistra), la numerazione parte dal primo piedino in alto a destra e procede in senso antiorario.

Questa la correlazioni tra i piedini e la loro funzione:

- 1 catodo del segmento verticale alto destro
- 2 anodo della terza cifra partendo da sinistra
- 3 anodo della seconda cifra partendo da sinistra
- 4 catodo del segmento verticale alto sinistro
- 5 catodo del segmento orizzontale alto
- 6 anodo della prima cifra da sinistra
- 7 catodo del segmento verticale basso sinistro
- 8 catodo del segmento orizzontale basso
- 9 catodo del punto decimale
- 10 catodo del segmento verticale basso destro
- 11 catodo del segmento orizzontale centrale
- 12 anodo della quarta ed ultima cifra partendo da sinistra

In commercio esistono altri moduli led da quattro cifre i cui piedini potrebbero avere una differente mappatura. Tale mappatura, qualora non fossero reperibili informazioni in proposito, puo' essere facilmente realizzata utilizzando una pila da 1,5 volt, un paio di cavetti, una breadboard e un po' di pazienza.

Nell'esercizio si utilizza il display per realizzare un timer in grado di gestire un periodo massimo di un'ora, settabile mediante un pulsante che fa scorrere velocemente i minuti. L'allarme e' lanciato da un buzzer, che si attiva allo scadere del periodo.

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

Here some notes about this project, translated by google translator



The 4-digit LED display used in this project, is the HS410561K-32 model. This model, is characterized by 4 anodes (one for each digit) and 8 cathodes, common to all the figures, which guide the various light segments. Technically Arduino manages a segment at a time, and due to the speed of microcontroller and "latency" in brightness of each led, digits appear at the same time bright. For convenience in exposure, the legs have been numbered as in the figure, from 1 to 12. In practice, taking the display with the bright part facing upwards (as in the left figure), numbering starts from the first leg at the top right and proceeds counterclockwise.

Here the correlation between pins and functions:

- 1 cathode of right top segment
- 2 anode of third digit from left
- 3 anode of second digit from the left
- 4 cathode of vertical top left segment
- 5 cathode of horizontal top segment
- 6 anode of first digit from the left
- 7 cathode of left lower vertical segment
- 8 cathode of lower horizontal segment
- 9 cathode of decimal point
- 10 cathode of right lower vertical segment
- 11 cathode of central horizontal segment
- 12 anode of fourth and last digit from the left

On market there are other four digit led modules, whose pins may have a different mapping. This mapping, if not found on web, can be easily accomplished using a 1.5 volt battery, a pair of cables, a breadboard and a bit of patience.

In this project you use the four digit display to create a timer that can handle a maximum period of one hour, settable by a button that drives minutes. The alarm is launched by a buzzer, which is activated when time ends

Note: This project and this note is part of a series that sees, as main characters, Arduino and some of connectable components. For most projects there is also a video on youtube.

- [Projects collection](#)
- [Movies on youtube](#)
- [About Arduino and components \(italian; pdf will be downloaded in your download area\)](#)
- [Quick programming guide \(almost english; pdf will be downloaded in your download area\)](#)

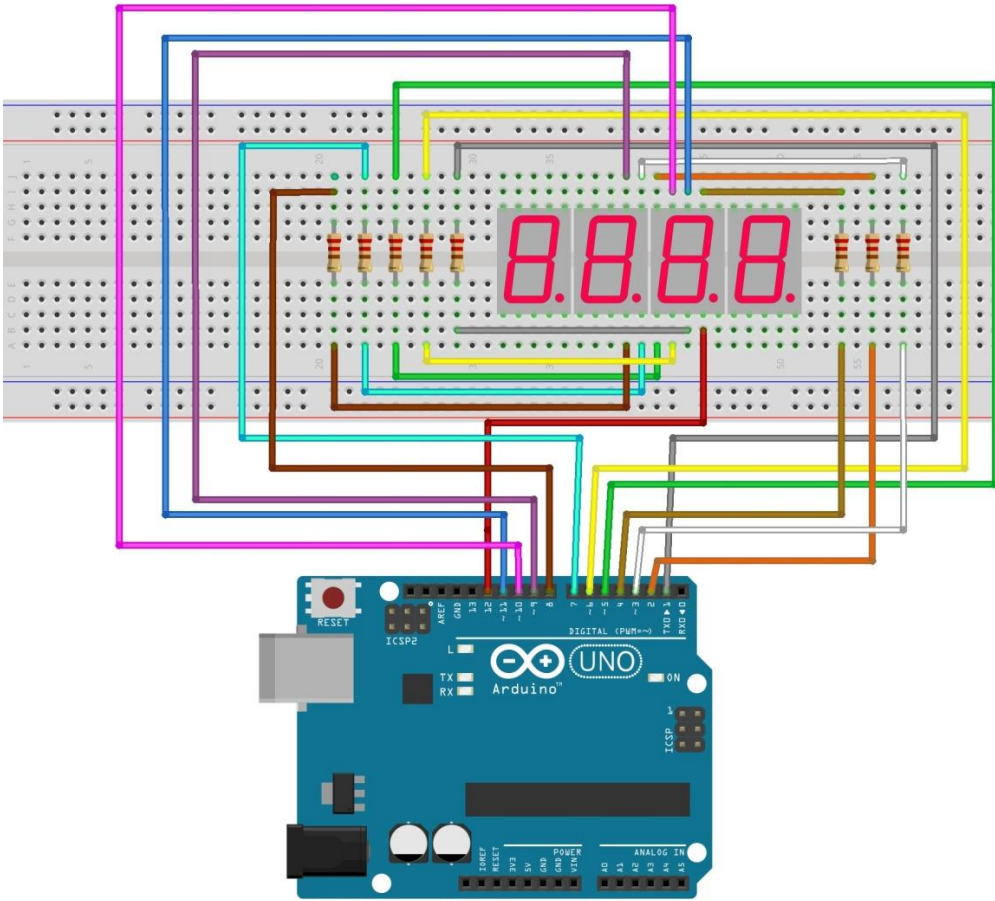
For any questions or suggestions about this note (and on its english translation), please write to giocarduino@libero.it (simple words and short sentences, please)

Materiali

- 1 breadboard
- 1 modulo led a 4 cifre
- 8 resistenze da 220 ohm
- 1 resistenza da 10 k ohm
- 1 buzzer
- 1 pulsante
- Un po' di cavetteria

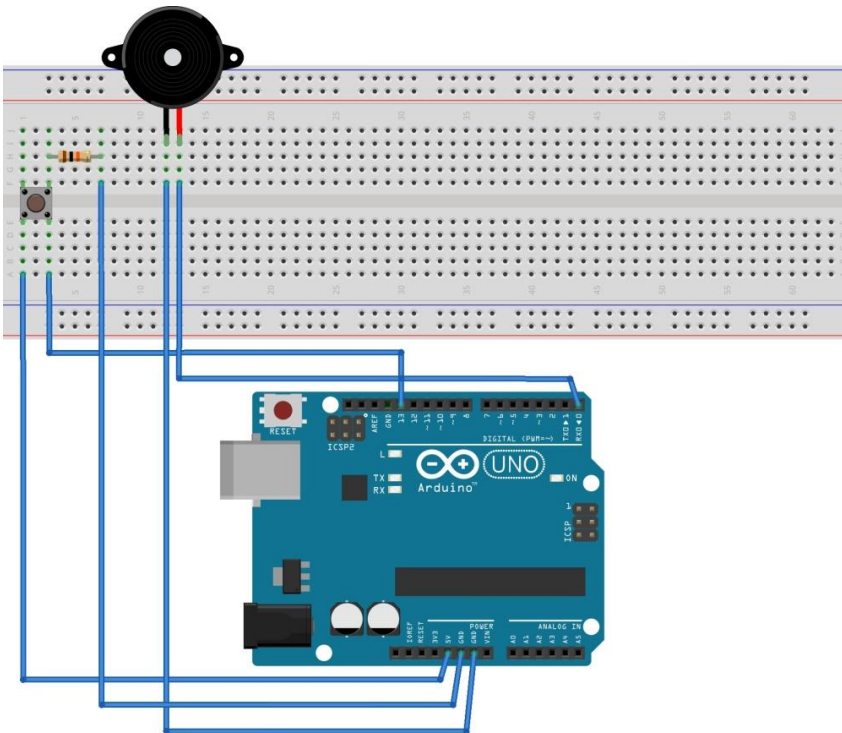
Schema

Arduino: timer da cucina, display a 4 cifre – kitchen timer, four digit display



fritzing

Schema di collegamento del display



fritzing

Schema di collegamento del buzzer e del pulsante

Arduino: timer da cucina, display a 4 cifre – kitchen timer, four digit display

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.
*
* Il display led da 4 cifre utilizzato in questo esercizio e' il modello
* HS410561K-32. Questo modello, e' caratterizzato da 4 anodi (uno per ogni cifra) e dai 8 catodi
* (comuni a tutte le cifre) che pilotano i vari segmenti. Tecnicamente si gestisce una cifra per
* volta e per effetto della velocita' del microcontrollore e della "latenza" nella luminosita' dei
* led, le cifre appaiono tutte contemporaneamente luminose.
*
* Questa la correlazioni tra i piedini e la loro funzione:
* 1 catodo del segmento verticale alto destro
* 2 anodo della terza cifra partendo da sinistra
* 3 anodo della seconda cifra partendo da sinistra
* 4 catodo del segmento verticale alto sinistro
* 5 catodo del segmento orizzontale alto
* 6 anodo della prima cifra da sinistra
* 7 catodo del segmento verticale basso sinistro
* 8 catodo del segmento orizzontale basso
* 9 catodo del punto decimale
* 10 catodo del segmento verticale basso destro
* 11 catodo del segmento orizzontale centrale
* 12 anodo della quarta ed ultima cifra partendo da sinistra
*
* In commercio esistono altri moduli led da quattro cifre i cui piedini hanno una differente
* mappatura. Tale mappatura, qualora non fossero reperibili informazioni in proposito, puo' essere
* facilmente realizzata utilizzando una pila da 1,5 volt, un paio di cavetti, una breadboard e un
* po' pazienza.
*
* In questo esercizio utilizzeremo il modulo per realizzare un timer da cucina in grado di gestire
* un periodo massimo di un'ora, settabile mediante un pulsante che fa scorrere velocemente i minuti.
* L'allarme sonoro e' lanciato da un buzzer, che si attiva allo scadere del periodo.
*
* La mappa di correlazioni tra porte e piedini e' la seguente:
* porta 0 collegata al polo positivo del buzzer
* porta 1 collegata ad una resistenza da 220 ohm a sua volta collegata al piedino numero 11
* porta 2 collegata ad una resistenza da 220 ohm a sua volta collegata al piedino numero 4
* porta 3 collegata ad una resistenza da 220 ohm a sua volta collegata al piedino numero 5
* porta 4 collegata ad una resistenza da 220 ohm a sua volta collegata al piedino numero 1
* porta 5 collegata ad una resistenza da 220 ohm a sua volta collegata al piedino numero 9
* porta 6 collegata ad una resistenza da 220 ohm a sua volta collegata al piedino numero 10
* porta 7 collegata ad una resistenza da 220 ohm a sua volta collegata al piedino numero 8
* porta 8 collegata ad una resistenza da 220 ohm a sua volta collegata al piedino numero 7
* porta 9 collegata al piedino numero 6
* porta 10 collegata al piedino numero 3
* porta 11 collegata al piedino numero 2
* porta 12 collegata al piedino numero 12
* porta 13 collegata ad una resistenza da 10k ohm a sua volta collegata a terra e ad un piedino del
* pulsante, mentre l'altro piedino, quello che chiude il circuito quando viene premuto il pulsante,
* e' collegato a terra.
*
* Poiche' le porte 0 ed 1 di Arduino sono influenzate, anzi disturbate, da segnali che hanno a che
* vedere con la comunicazione seriale, e' opportuno realizzare l'impianto attenendosi ai due schemi
* proposti nel precedente paragrafo.
*
* Attenzione: per consentire il corretto caricamento del programma e' opportuno collegare il buzzer
* alla porta 0 solo dopo che il programma e' stato caricato e usare il timer solo in connessione
* diretta con il pc, in modo che i disturbi indotti sulla porta 1 dal circuito di comunicazione
* E seriale possano transitare (uscire) dalla porta usb.
*
*-----
* Warning: cut&paste from PDF to IDE loses formatting. to restore it press CTRL + T.
* The 4-digit LED display used in this project, is the HS410561K-32 model. This model, is
* characterized by 4 anodes (one for each digit) and 8 cathodes, common to all the figures, which
* guide the various light segments. Technically Arduino manages a segment at a time, and due to the
* speed of microcontroller and "latency" in brightness of each led, digits appear at the same time
* bright.
*
* Here the correlation between pins and functions:
* 1 cathode of right top segment
* 2 anode of third digit from left
* 3 anode of second digit from the left
* 4 cathode of vertical top left segment
* 5 cathode of horizontal top segment
```

Arduino: timer da cucina, display a 4 cifre – kitchen timer, four digit display

```
* 6 anode of first digit from the left
* 7 cathode of left lower vertical segment
* 8 cathode of lower horizontal segment
* 9 cathode of decimal point
* 10 cathode of right lower vertical segment
* 11 cathode of central horizontal segment
* 12 anode of fourth and last digit from the left
*
* On market there are other four digit led modules, whose pins may have a different mapping. This
* mapping, if not finded on web, can be easily accomplished using a 1.5 volt battery, a pair of
* cables, a breadboard and a bit of patience.
*
* In this project you use the four digit display to create a timer that can handle a maximum period
* of one hour, settable by a button that drives minutes. The alarm is launched by a buzzer, which is
* activated when time ends.
*
* The correlations between Arduino pins and display legs, is as follow:
* Pin 0 connected to buzzer positive pole
* Pin 1 connected to a 220 ohm resistor, in turn connected to leg 11
* Pin 2 connected to a 220 ohm resistor, in turn connected to leg 4
* Pin 3 connected to a 220 ohm resistor, in turn connected to leg 5
* Pin 4 connected to a 220 ohm resistor, in turn connected to leg 1
* Pin 5 connected to a 220 ohm resistor, in turn connected to leg 9
* Pin 6 connected to a 220 ohm resistor, in turn connected to leg 10
* Pin 7 connected to a 220 ohm resistor, in turn connected to leg 8
* Pin 8 connected to a 220 ohm resistor, in turn connected to leg 7
* Pin 9 connected to leg 6
* Pin 10 connected to leg 3
* Pin 11 connected to leg 2
* Pin 12 connected to leg 12
* Pin 13 connected to a 10k ohm resistor, in turn connected to ground and to a leg of button,
* while the other leg, the one that closes the circuit when the button is pressed, is connected
* to ground.
*
*
* Since pins 0 and 1 of Arduino are influenced, by serial communications signal, is appropriate to
* implement the system according to the above two schematic.
*
* Attention: to avoid problem, buzzer must be connected to arduino after program load and the timer
* must be used with Arduino connected to pc, via usb cable
*
*-----
*/
int pulsante = LOW;           // variabile di memorizzazione dello stato del pulsante
int i;                       // variabile intera utilizzata nei cicli di for
int j;                       // variabile intera utilizzata nei cicli di for
int z;                       // variabile intera utilizzata nei cicli di for
int h;                       // variabile intera utilizzata nei cicli di for
long tempo=61;              // tempo di countdown, in minuti; viene inizializzato a 61 per forzare
                           // il ciclo di prima volta
long tempoinizio = 0;       // momento di inizio pressione del pulsante (millisecondi trascorsi
                           // dal momento di accensione di arduino)
int decaminuti = 0;         // decine di minuti da visualizzare sul display
int minuti = 0;            // minuti da visualizzare sul display
int decasecondi = 0;       // decine di secondi da visualizzare sul display
int secondi = 0;          // secondi da visualizzare sul display
int numero = 0;           // numero da visualizzate nella cifra che viene di volta in volta
                           trattata
int secondiresidui= 0;     // tempo residuo, in secondi,
int lapse= 0;             // tempo di suono del buzzer, in millisecondi)
long iniziocountdown= 0;  // momento di inizio del countdown, in millisecondi
long iniziobuzzer=0;      // momento di inizio del suono del buzzer, in millisecondi

// Le routine che seguono hanno il compito di visualizzare i numeri e vengono eseguite solo
// nel momento in cui sono richiamate da istruzioni presenti nel percorso attivo del programma

// ***** routine di visualizzazione del numero 1 - number 1 design*****
void uno(void)
{
  digitalWrite(4,LOW);
  digitalWrite(6,LOW);
}

// ***** routine di visualizzazione del numero 2 - number 2 design *****
void due(void)
{
  digitalWrite(3,LOW);
  digitalWrite(4,LOW);
}
```

Arduino: timer da cucina, display a 4 cifre – kitchen timer, four digit display

```
digitalWrite(1,LOW);
digitalWrite(8,LOW);
digitalWrite(7,LOW);
}

// ***** routine di visualizzazione del numero 3 - number 3 design *****
void tre(void)
{
  digitalWrite(3,LOW);
  digitalWrite(4,LOW);
  digitalWrite(1,LOW);
  digitalWrite(6,LOW);
  digitalWrite(7,LOW);
}

// ***** routine di visualizzazione del numero 4 - number 4 design *****
void quattro(void)
{
  digitalWrite(2,LOW);
  digitalWrite(1,LOW);
  digitalWrite(4,LOW);
  digitalWrite(6,LOW);
}

// ***** routine di visualizzazione del numero 5 - number 5 design *****
void cinque(void)
{
  digitalWrite(3,LOW);
  digitalWrite(2,LOW);
  digitalWrite(1,LOW);
  digitalWrite(6,LOW);
  digitalWrite(7,LOW);
}

// ***** routine di visualizzazione del numero 6 - number 6 design *****
void six(void)
{
  digitalWrite(2,LOW);
  digitalWrite(1,LOW);
  digitalWrite(6,LOW);
  digitalWrite(7,LOW);
  digitalWrite(8,LOW);
}

// ***** routine di visualizzazione del numero 7 - number 7 design *****
void sette(void)
{
  digitalWrite(3,LOW);    // disattiva la porta 3 e accende il segmento orizzontale alto
  digitalWrite(4,LOW);    // disattiva la porta 4 e accende il segmento verticale alto destro
  digitalWrite(6,LOW);    // disattiva la porta 6 e accende il segmento verticale basso destro
}

// ***** routine di visualizzazione del numero 8 - number 8 design *****
void otto(void)
{
  digitalWrite(1,LOW);
  digitalWrite(2,LOW);
  digitalWrite(3,LOW);
  digitalWrite(4,LOW);
  digitalWrite(6,LOW);
  digitalWrite(7,LOW);
  digitalWrite(8,LOW);
}

// ***** routine di visualizzazione del numero 9 - number 9 design *****
void nove(void)
{
  digitalWrite(1,LOW);
  digitalWrite(2,LOW);
  digitalWrite(3,LOW);
  digitalWrite(4,LOW);
  digitalWrite(6,LOW);
}

// ***** routine di visualizzazione del numero 0 - number 0 design *****
void zero(void)
{
  digitalWrite(2,LOW);
}
```

Arduino: timer da cucina, display a 4 cifre – kitchen timer, four digit display

```
digitalWrite(3,LOW);
digitalWrite(4,LOW);
digitalWrite(6,LOW);
digitalWrite(7,LOW);
digitalWrite(8,LOW);
}

// **** routine di gestione del display (di esposizione del tempo residuo)- display manager ****
// viene gestita una cifra per volta attivando e disattivando le porte da 9 a 12 alle quali sono
// collegati gli anodi delle quattro cifre del display
// a digit at a time is managed by activating and deactivating the pins 9 to 12 to which are
// connected the anodes of the four digits
//

void esponi (void)
{
  for (z=9; z<=12; z++)          // for di gestione delle porte (da 9 a 12, alle quali sono
                                // collegati i quattro anodi delle quattro cifre)
  {
    for (j=1; j<=8; j++)
      digitalWrite (j, HIGH);    // attiva tutti i catodi, per iniziare con tutti i segmenti spenti
    digitalWrite (z, HIGH);      // attiva l'anodo della cifra da esporre
    switch (z) {

      case 9:                     // se si sta trattando la porta 9 (prima cifra di sinistra)
        numero = decaminuti;     // inserisce le decine di minuti nel numero da esporre
        break;

      case 10:                   // se si sta trattando la porta 10 ( seconda cifra da sinistra)
        numero = minuti;        // inserisce i minuti nel numero esporre
        digitalWrite (5, LOW);   // accende il punto decimale come separatore dei minuti dai secondi
        break;

      case 11:                   // se si sta trattando la porta 11 (terza cifra da sinistra)
        numero = decasecondi;   // inserisce le decine di secondi nella porta da esporre
        break;

      case 12 :                  // se si sta trattando la porta 12 (quarta ed ultima cifra a destra)
        numero = secondi;       // inserisce i secondi nel numero da esporre
        break;
    }
    visualizzanumeri ();        // lancia la routine di visualizzazione del numero nella cifra corrente
    digitalWrite (z,LOW);      // disattiva (spegne) l'anodo della cifra appena visualizzata
  }
}

// *****routine di visualizzazione dei numeri - numbers display *****

// La routine si limita ad attivare i catodi dei vari segmenti
// poiche' l'anodo della cifra corrente e' stato gia' attivato dalla routine chiamante
// activate only cathode

void visualizzanumeri (void)
{
  switch (numero) {

    case 0:
      zero ();
      break;

    case 1:
      uno();
      break;

    case 2:
      due();
      break;

    case 3:
      tre();
      break;

    case 4:
      quattro();
      break;

    case 5:
      cinque();
  }
}
```

Arduino: timer da cucina, display a 4 cifre – kitchen timer, four digit display

```
        break;

    case 6:
        six();
        break;

    case 7:
        sette();
        break;

    case 8:
        otto();
        break;

    case 9:
        nove();
        break;
    }
}

void setup()
{
    for(i=0;i<=12;i++)
        pinMode(i,OUTPUT); // definisce le porte da 0 a 12 come porte di output
    pinMode (13, INPUT); // definisce la porta 13 (il pulsante) come porta di input
}

void loop()
{
    pulsante = 0;
    pulsante = digitalRead (13); // acquisisce lo stato del pulsante
    while (pulsante==1) // settaggio del timer; se il pulsante e' premuto si innesca
        // la routine di settaggio, che aumenta di 1 minuto il tempo
        // di countdown per ogni secondo continuativo di pressione;
        // arrivati a 60 minuti il settaggio ricomincia da 0
    {
        pulsante = digitalRead (13);
        if (tempo >= 60) { // se sono stati superati i 60 minuti, ricomincia il settaggio da 0 minuti
            tempo = 0;
            tempoinizio = millis ();
        }
        else
        {
            tempo = (millis() - tempoinizio)/1000; //calcola il tempo trascorso dalla pressione pulsante
        }
        decaminuti = (tempo / 10); // memorizza nelle decine di minuti il risultato di tempo/10
        minuti = (tempo%10); // memorizza in minuti il resto della divisione per 10 del tempo
        decasecondi = 0;
        secondi = 0;
        esponi (); // visualizza i minuti di settaggio del timer
        secondiresidui = tempo * 60; // memorizza i secondi sino ad ora settati (minuti * 60)
        iniziocountdown = millis(); // memorizza il momento di inizio del countdown, da utilizzare
        // nel caso il pulsante fosse stato rilasciato e dovesse iniziare il countdown
    }

    // *****ciclo di gestione del countdown - countdown manager *****
    tempoinizio = millis();
    if (secondiresidui >=1) { // gestione del tempo residuo e della relativa visualizzazione
        decaminuti = secondiresidui/600;
        minuti = (secondiresidui - (decaminuti*600))/60;
        decasecondi = (secondiresidui - (decaminuti*600)-(minuti*60))/10;
        secondi = (secondiresidui - (decaminuti*600)-(minuti*60) - (decasecondi*10));
        esponi ();
        secondiresidui = ((iniziocountdown + tempo*60000)-millis())/1000; //calcola i secondi residui di
        // countdown
        iniziobuzzer = millis ();
    }
    else
    { // gestione del fine countdown e attivazione del buzzer
        if ((tempo <= 60) && (lapse < 5000))
        {
            digitalWrite (0,HIGH); //attiva il buzzer per 5 secondi (5000 millisecondi)
            decaminuti =0;
            minuti = 0;
            decasecondi = 0;
            secondi = 0;
            esponi ();
            lapse = millis() - iniziobuzzer;
        }
    }
}
```


Arduino: timer da cucina, display a 4 cifre – kitchen timer, four digit display

```
    }  
    else // se e' cessato l'allarme predispone il sistema ad un nuovo settaggio  
    {  
        digitalWrite (0,LOW);  
        tempo = 61;  
        lapse = 0;  
    }  
}  
}
```