Esercizio: Display a 7 segmenti

In questa lezione, esploreremo come utilizzare questi versatili dispositivi per visualizzare numeri e sequenze. Il nostro obiettivo è acquisire una comprensione completa del funzionamento dei display a 7 segmenti, imparare a contare da 0 a 9 e creare librerie personalizzate per semplificare il nostro codice.

Per poter illuminare un segmento del display è necessario che il corrispondente LED venga attraversato da corrente, ovvero l’anodo deve risultare positivo rispetto al catodo. Ovviamente per limitare la quantità di corrente che attraversa ogni LED ad un valore tra i 10 e i 20 mA è indispensabile inserire una resistenza di limitazione. Ad ogni led, quindi, andrà inserita un’opportuna resistenza in serie.

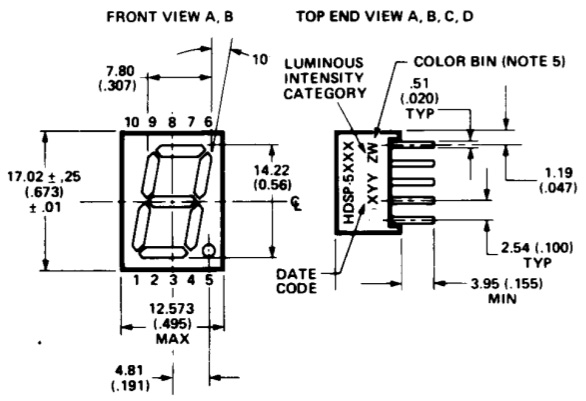
L'obiettivo di questo esempio è contare da 0 a 8 e visualizzare solo i numeri pari sul display a 7 segmenti a catodo comune.

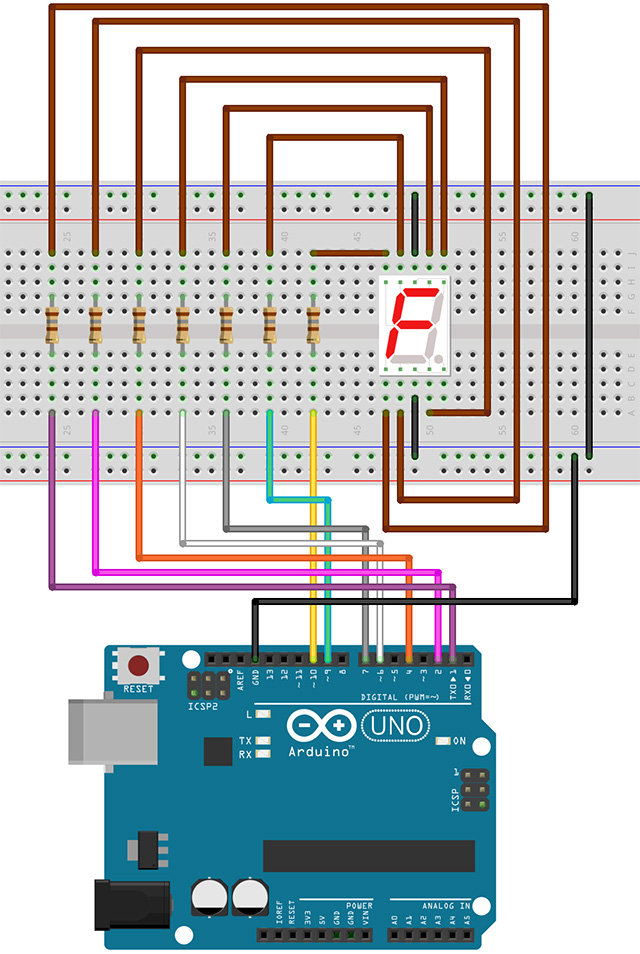
Materiali

Per realizzare tutto questo saranno necessarie:

* 7 resistenze (numero pari ai segmenti del display
* 1 display a 7 segmenti
* opportuni collegamenti da parte delle resistenze ai pin

Per capire come accendere il display si consideri:





// Conteggio numeri pari da 0 a 8 con display 7 segmenti a catodo comune

// definizione delle variabili a cui collegare ogni segmento del display

// la lettera rappresenta il segmento del display il numero il pin

// digitale di Arduino a cui è collegato il corrispondente pin del display

const int a=7;

const int b=6;

const int c=4;

const int d=2;

const int e=1;

const int f=9;

const int g=10;

void AccendiLed(int n)

{

switch(n)

{

case 0:

digitalWrite(a, HIGH);

digitalWrite(b, HIGH);

digitalWrite(c, HIGH);

digitalWrite(d, HIGH);

digitalWrite(e, HIGH);

digitalWrite(f, HIGH);

digitalWrite(g, LOW);

break;

case 1:

digitalWrite(a, LOW);

digitalWrite(b, HIGH);

digitalWrite(c, HIGH);

digitalWrite(d, LOW);

digitalWrite(e, LOW);

digitalWrite(f, LOW);

digitalWrite(g, LOW);

break;

case 2:

digitalWrite(a, HIGH);

digitalWrite(b, HIGH);

digitalWrite(c, LOW);

digitalWrite(d, HIGH);

digitalWrite(e, HIGH);

digitalWrite(f, LOW);

digitalWrite(g, HIGH);

break;

case 3:

digitalWrite(a, HIGH);

digitalWrite(b, HIGH);

digitalWrite(c, HIGH);

digitalWrite(d, HIGH);

digitalWrite(e, LOW);

digitalWrite(f, LOW);

digitalWrite(g, HIGH);

break;

case 4:

digitalWrite(a, LOW);

digitalWrite(b, HIGH);

digitalWrite(c, HIGH);

digitalWrite(d, LOW);

digitalWrite(e, LOW);

digitalWrite(f, HIGH);

digitalWrite(g, HIGH);

break;

case 5:

digitalWrite(a, HIGH);

digitalWrite(b, LOW);

digitalWrite(c, HIGH);

digitalWrite(d, HIGH);

digitalWrite(e, LOW);

digitalWrite(f, HIGH);

digitalWrite(g, HIGH);

break;

case 6:

digitalWrite(a, HIGH);

digitalWrite(b, LOW);

digitalWrite(c, HIGH);

digitalWrite(d, HIGH);

digitalWrite(e, HIGH);

digitalWrite(f, HIGH);

digitalWrite(g, HIGH);

break;

case 7:

digitalWrite(a, HIGH);

digitalWrite(b, HIGH);

digitalWrite(c, HIGH);

digitalWrite(d, LOW);

digitalWrite(e, LOW);

digitalWrite(f, LOW);

digitalWrite(g, LOW);

break;

case 8:

digitalWrite(a, HIGH);

digitalWrite(b, HIGH);

digitalWrite(c, HIGH);

digitalWrite(d, HIGH);

digitalWrite(e, HIGH);

digitalWrite(f, HIGH);

digitalWrite(g, HIGH);

break;

case 9:

digitalWrite(a, HIGH);

digitalWrite(b, HIGH);

digitalWrite(c, HIGH);

digitalWrite(d, HIGH);

digitalWrite(e, LOW);

digitalWrite(f, HIGH);

digitalWrite(g, HIGH);

break;

}

}

void setup()

{

pinMode(a, OUTPUT);

pinMode(b, OUTPUT);

pinMode(c, OUTPUT);

pinMode(d, OUTPUT);

pinMode(e, OUTPUT);

pinMode(f, OUTPUT);

pinMode(g, OUTPUT);

}

void loop ()

{

for (int i=0;i<10;i++)

{

if (i%2==0) { // se il resto della divisione è uguale a 0

// allora viene viene chiamata la funzione AccendiLed()

// che visualizzerà il corrispondente numero pari

AccendiLed(i);

delay (1000);

}

}

}

Immagine che contiene diagramma, testo, Piano, Disegno tecnico

Descrizione generata automaticamente

Immagine che contiene testo, schermata, diagramma, Carattere

Descrizione generata automaticamente

Immagine che contiene testo, schermata, diagramma, Carattere

Descrizione generata automaticamente

Immagine che contiene testo, diagramma, Carattere, schermata

Descrizione generata automaticamente

Immagine che contiene testo, schermata, Carattere, algebra

Descrizione generata automaticamente