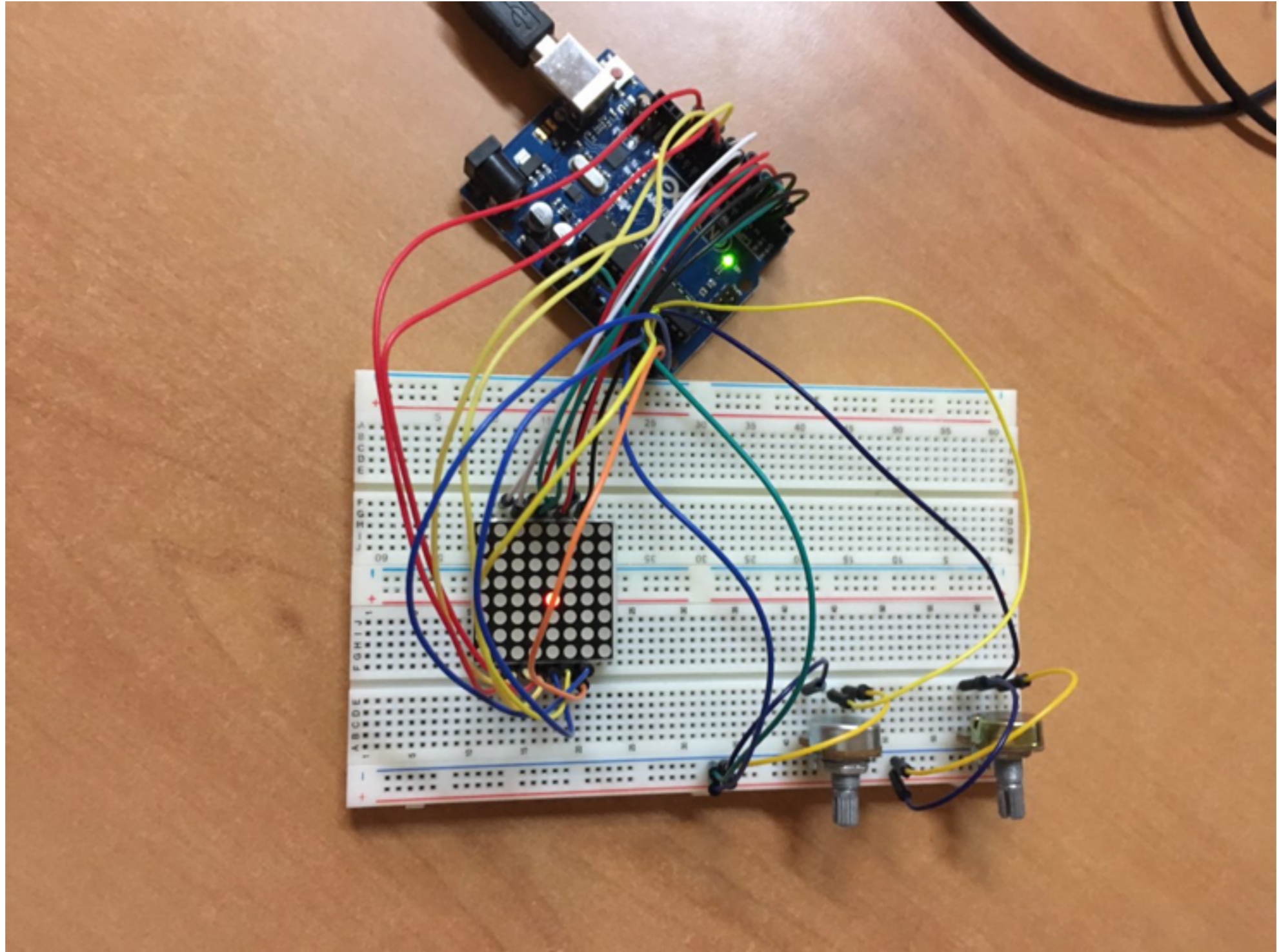


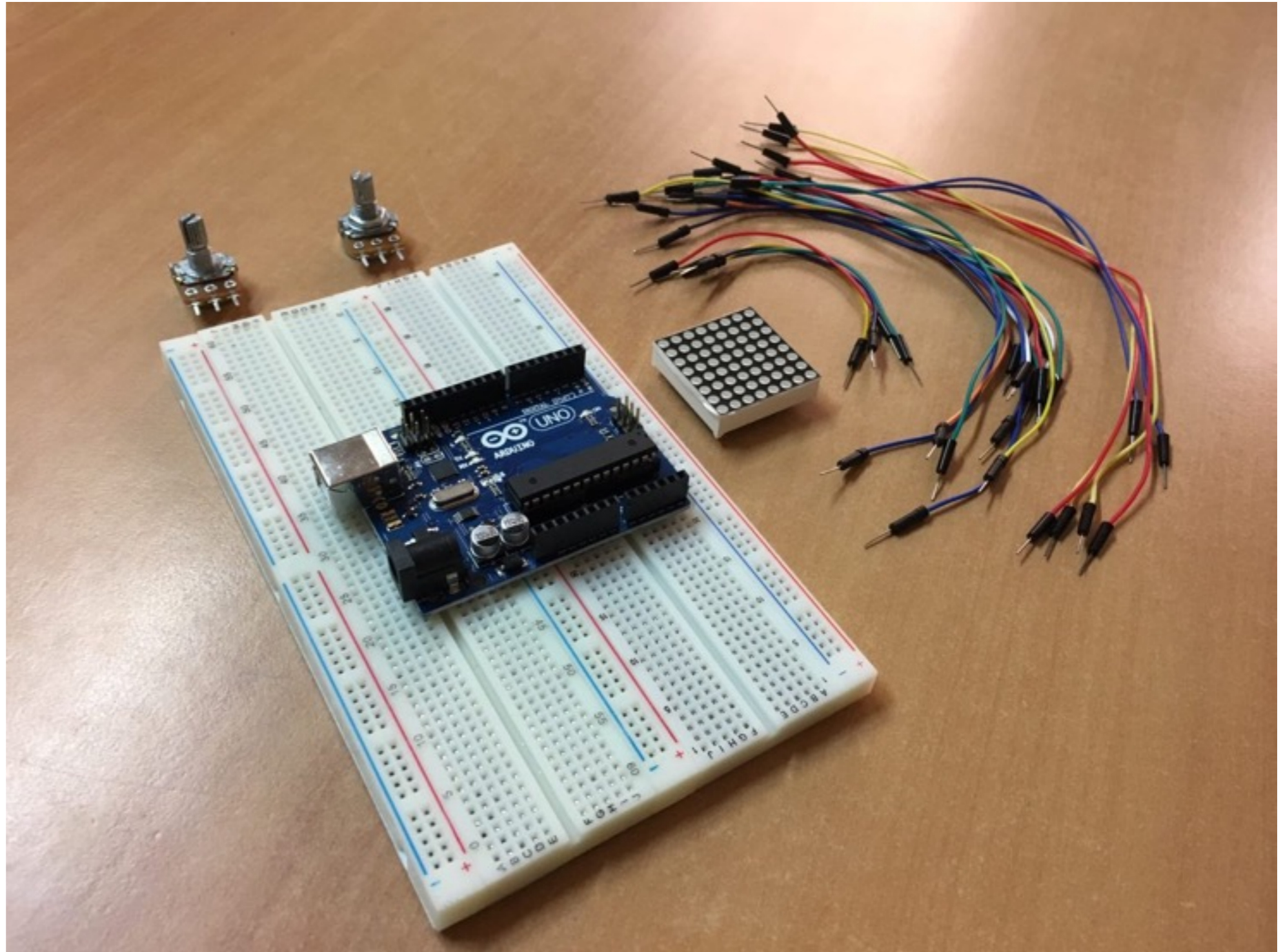
Test matrix 8x8:



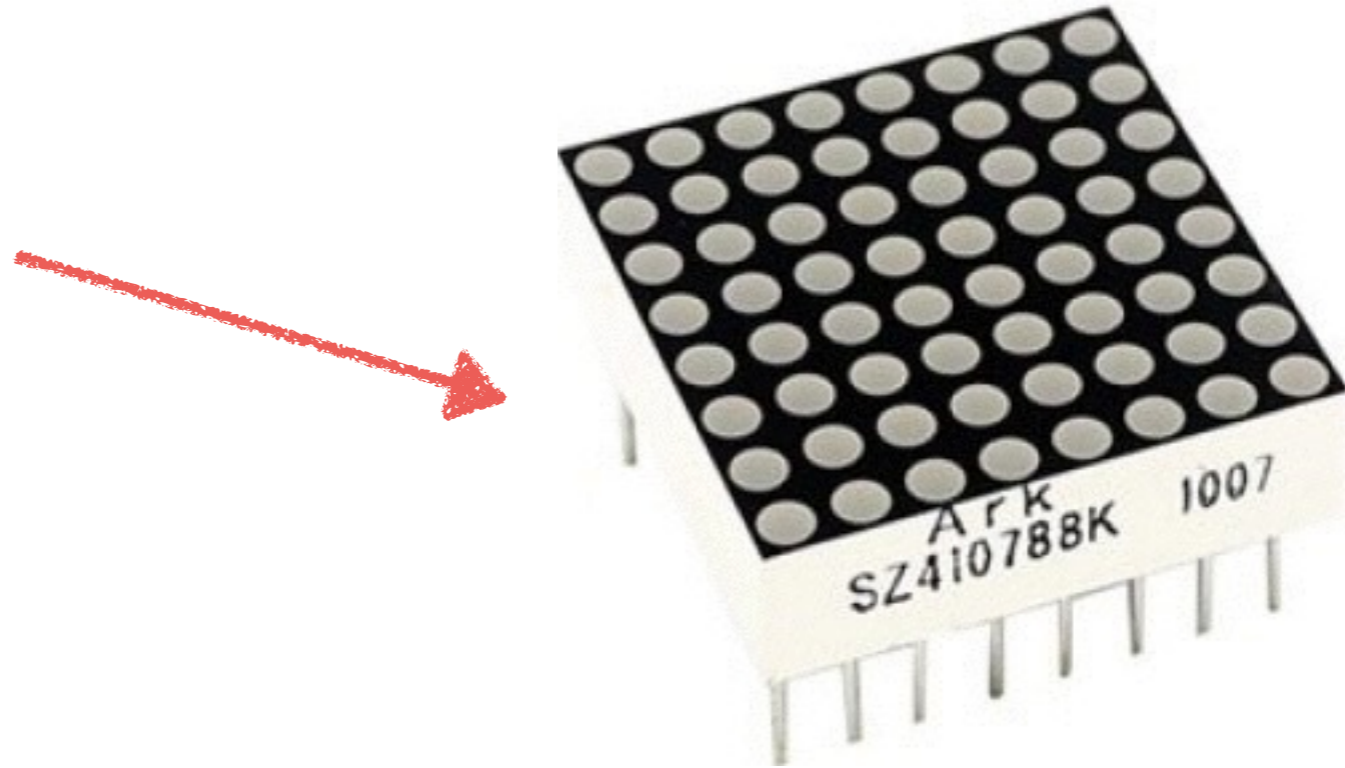
Salvatore Spanu e Roberta Galici

Componenti:

- Arduino or Genuino Board*
- 8 x 8 LED Matrix*
- 2 10k ohm potenziometri*
- collegamenti wires*
- breadboard*



La matrice

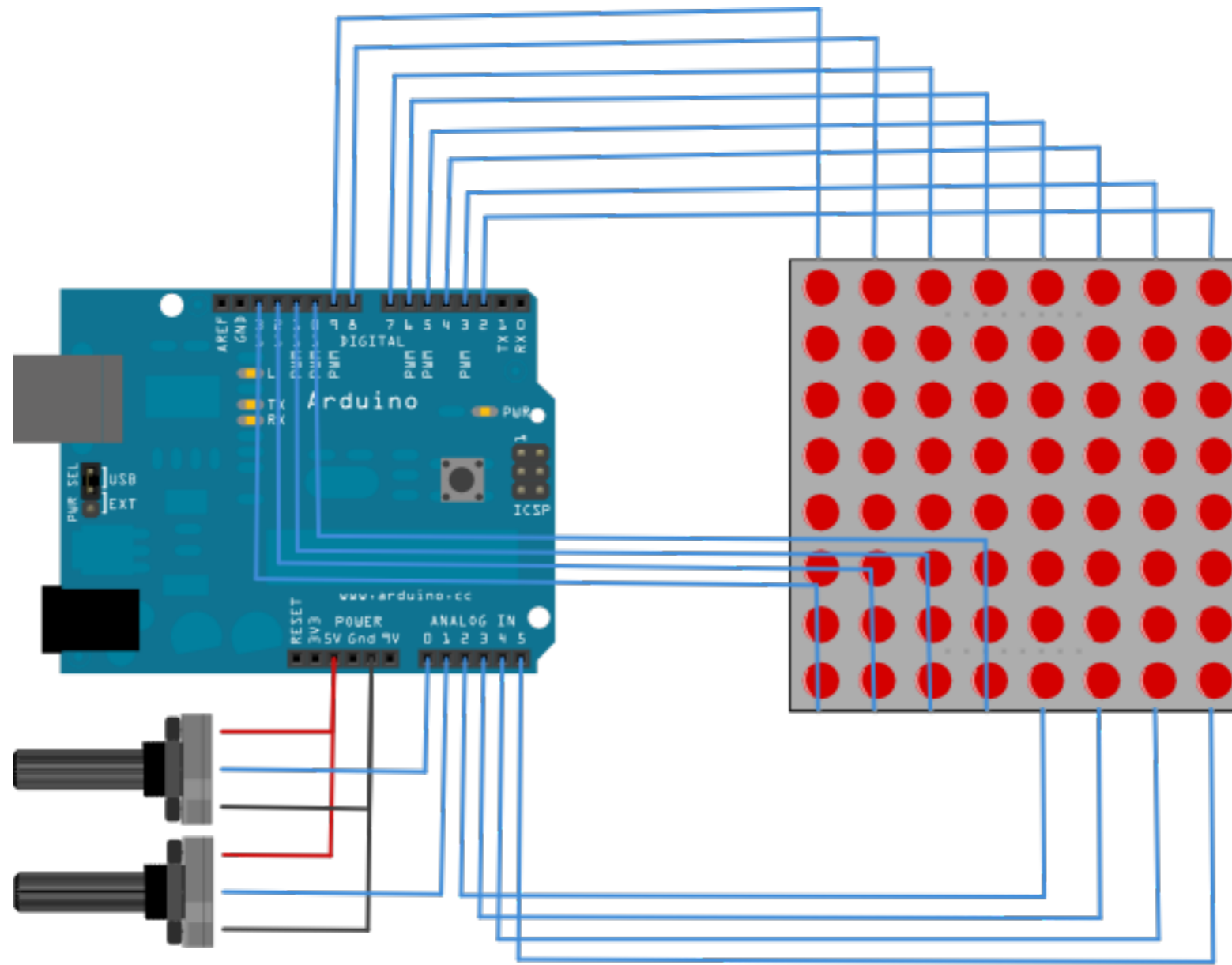


Questo modulo è spesso confezionato come matrice di LED dove gli anodi risultano connessi per colonna mentre i catodi per riga (o viceversa). Un LED si accende quando la riga corrispondente al suo catodo è connessa attraverso una resistenza alla terra (**LOW**) mentre la colonna, corrispondente al suo anodo, è connessa ai +5V. (**HIGH**)

La matrice 8x8 a LED ha 16 piedini. Ogni piedino è associato ad una riga o ad una colonna. Per controllare la matrice 8x8 a LED tramite Arduino occorre quindi collegare sia le righe che le colonne. Il LED si accende quando il piedino corrispondente alla sua colonna è posto ad **HIGH** mentre quello relativo alla sua riga è posto a **LOW**. Se la colonna e la riga non sono posti rispettivamente ad **HIGH** e **LOW** nessuna tensione scorre attraverso il LED e quindi questo non si accende.

Circuito:

I 16 pin (8 su e 8 giù) della matrice 8x8 sono collegati a 16 pin di Arduino attraverso la breadboard. Quattro fra i pin analogici sono usati come input digitali da 16 a 19. L'ordine dei pin è assegnato in due array (all'interno del codice). Due potenziometri, connessi ai pin analogici 0 e 1, controllano il movimento di un piccolo LED nella matrice.



Codice:

```
// 2-dimensional array of column pin numbers:
const int col[8] = {
  6, 11, 10, 3, 17, 4, 8, 9
};

// 2-dimensional array of pixels:
int pixels[8][8];

// cursor position:
int x = 5;
int y = 5;

void setup() {
  // initialize the I/O pins as outputs
  // iterate over the pins:
  for (int thisPin = 0; thisPin < 8; thisPin++) {
    // initialize the output pins:
    pinMode(col[thisPin], OUTPUT);
    pinMode(row[thisPin], OUTPUT);
    // take the col pins (i.e. the cathodes) high to ensure that
    // the LEDs are off:
    digitalWrite(col[thisPin], HIGH);
  }

  // initialize the pixel matrix:
  for (int x = 0; x < 8; x++) {
    for (int y = 0; y < 8; y++) {
      pixels[x][y] = HIGH;
    }
  }
}
```



```
// initialize the pixel matrix:
for (int x = 0; x < 8; x++) {
  for (int y = 0; y < 8; y++) {
    pixels[x][y] = HIGH;
  }
}

void loop() {
  // read input:
  readSensors();

  // draw the screen:
  refreshScreen();
}

void readSensors() {
  // turn off the last position:
  pixels[x][y] = HIGH;
  // read the sensors for X and Y values:
  x = 7 - map(analogRead(A0), 0, 1023, 0, 7);
  y = map(analogRead(A1), 0, 1023, 0, 7);
  // set the new pixel position low so that the LED will turn on
  // in the next screen refresh:
  pixels[x][y] = LOW;
}
```



```
void refreshScreen() {
  // iterate over the rows (anodes):
  for (int thisRow = 0; thisRow < 8; thisRow++) {
    // take the row pin (anode) high:
    digitalWrite(row[thisRow], HIGH);
    // iterate over the cols (cathodes):
    for (int thisCol = 0; thisCol < 8; thisCol++) {
      // get the state of the current pixel;
      int thisPixel = pixels[thisRow][thisCol];
      // when the row is HIGH and the col is LOW,
      // the LED where they meet turns on:
      digitalWrite(col[thisCol], thisPixel);
      // turn the pixel off:
      if (thisPixel == LOW) {
        digitalWrite(col[thisCol], HIGH);
      }
    }
    // take the row pin low to turn off the whole row:
    digitalWrite(row[thisRow], LOW);
  }
}
```

Reperibile qua:



<http://pastebin.com/5dtzpt1W>

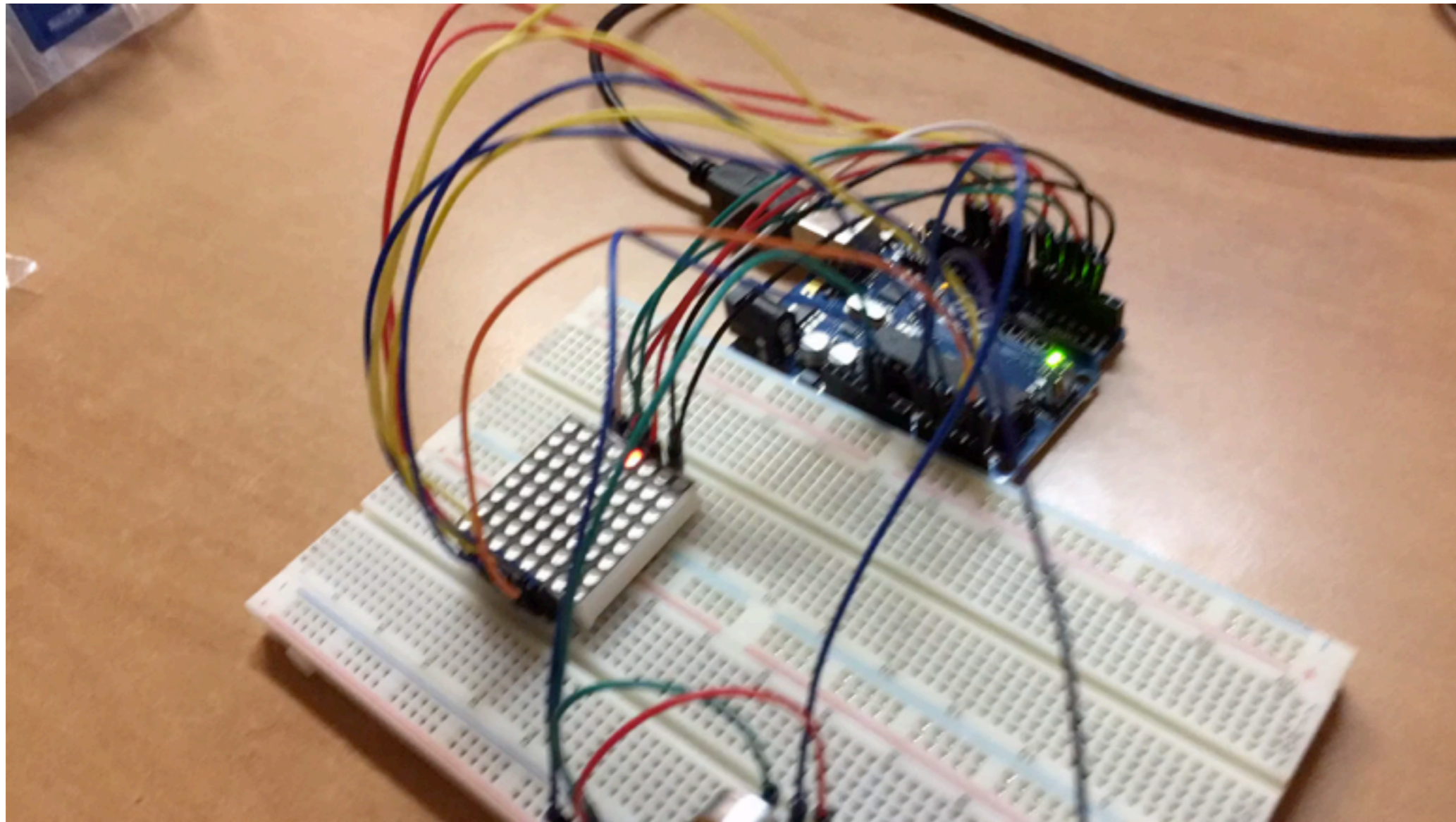
Costi:

Se si acquista lo starter kit si spende intorno ai 30€. Questo è un ottimo sito, i tempi di consegna sono un po' lunghi (circa 1 mese e mezzo/2)

<http://www.aliexpress.com/item/With-Retail-Box-RFID-Starter-Kit-for-Arduino-UNO-R3-Upgraded-version-Learning-Suite-Free-Shipping/32304936825.html?spm=2114.13010608.0.71.PjhXeM>



Piccolo video:



Video completo su <https://youtu.be/VidZ9fVHz98>