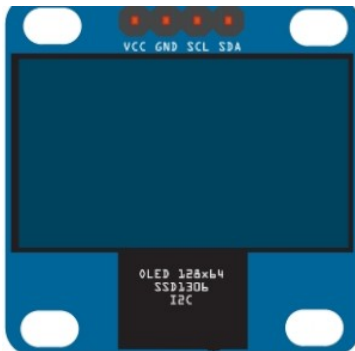
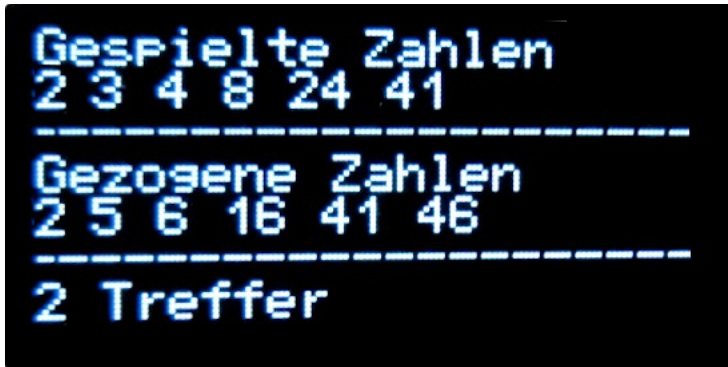


Der Arduino „spielt“ für dich 6 Zahlen. Anschließend werden die Lottozahlen gezogen, aufsteigend sortiert und auf die Anzahl der richtigen Zahlen untersucht.

So sieht es aus:

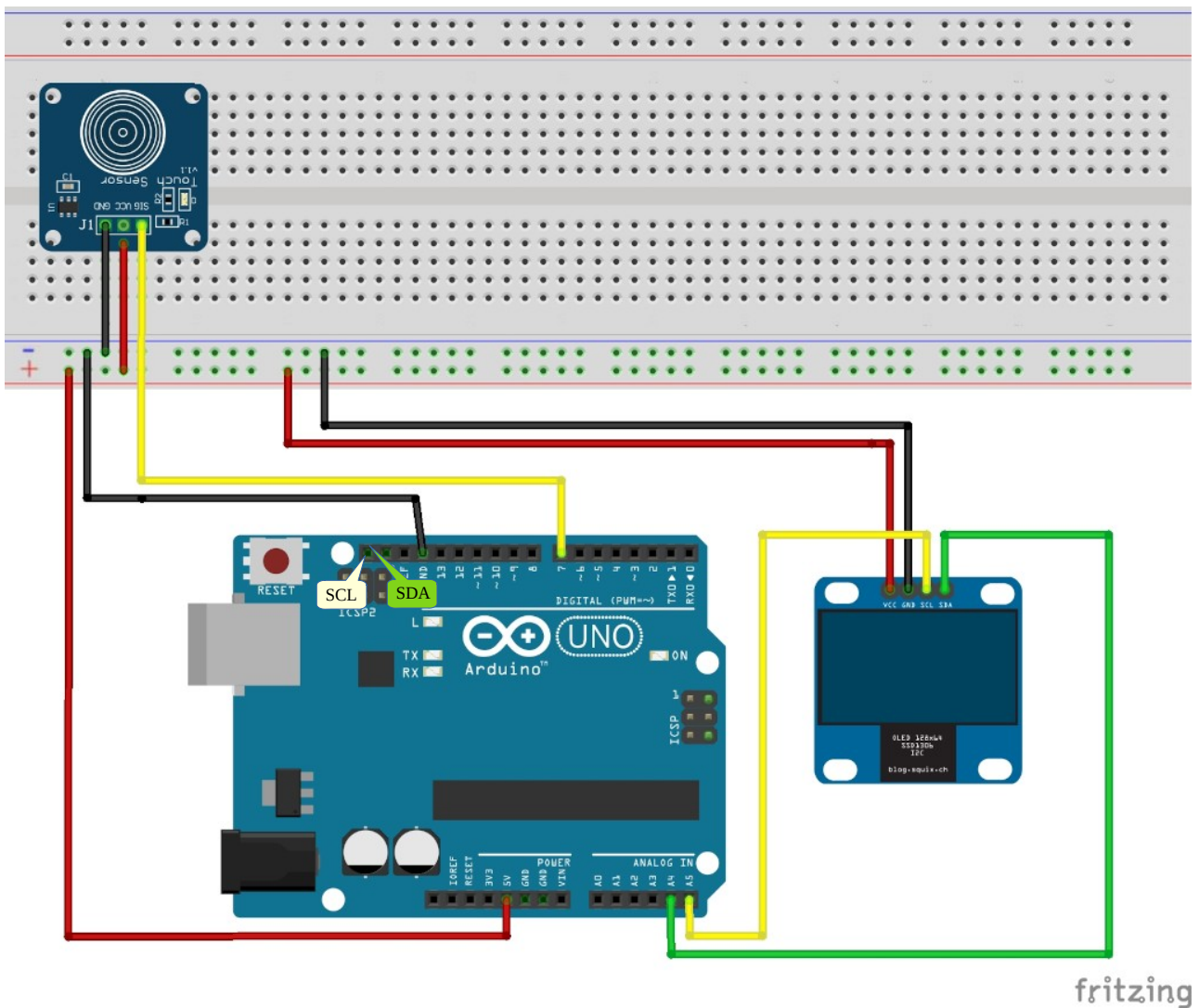


OLED-Displays (Organic Light Emitting Diode) benötigen im Unterschied zu LCDs keine Hintergrundbeleuchtung, sie leuchten selbst. Eine OLED besteht aus zwei Elektroden, von denen mindestens eine transparent sein muss. Im Zwischenraum befinden sich organische Halbleiterschichten aus natürlichen Farbstoffen. Die organischen Schichten leuchten, wenn sie von Gleichstrom durchflossen werden. Basis der Technik ist die Entdeckung der Elektrolumineszenz: ein Festkörper wird durch Anlegen einer elektrischen Spannung dazu angeregt Licht zu erzeugen.

Benötigte Bauteile:

- ➔ Berührungssensor
- ➔ OLED-Display SSD1306
- ➔ Leitungsdrähte

Baue die Schaltung auf.



fritzing

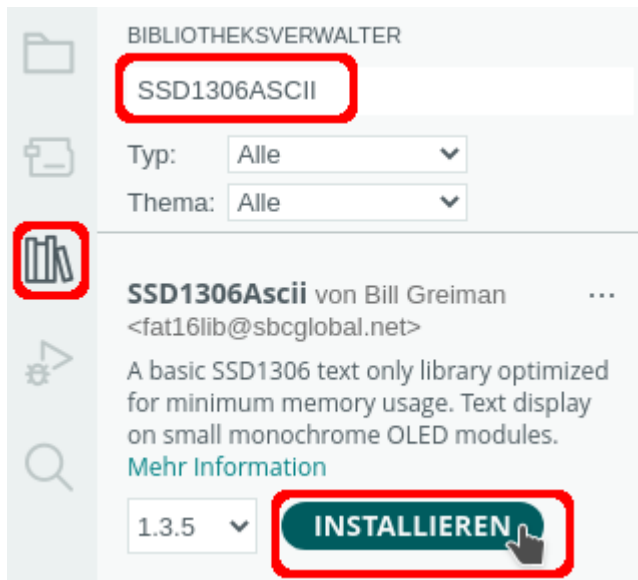
Statt A4 (SDA) und A5 (SCL) kannst du auch die mit SCL und SDA beschrifteten Pins verwenden.



Der Chipsatz des verwendeten Displays ist ein SSD1306 mit I2C.

Die hier verwendete Bibliothek SSD1306ASCII muss mit ...

Sketch → Bibliothek einbinden → Bibliotheken verwalten



... installiert werden.

Binde die benötigten Bibliotheken ein und definiere die Variablen:

```
# include "SSD1306Ascii.h"
# include "SSD1306AsciiWire.h"

// mögliche Adressen I2C: 0x3C+SA0 0x3C or 0x3D
# define I2C_ADDRESS 0x3C

// Name des OLED festlegen
SSD1306AsciiWire oled;

// Minimum/Maximum der Lottozahlen
const byte Minimum = 1;
const byte Maximum = 49;

// Taster an Pin 7
byte TASTER = 7;

// Anzahl der zu ziehenden Zahlen
const byte Anzahl = 6;

// Array für die gezogenen Zahlen
byte LottoZahl[Anzahl];
```

Der setup-Teil:

```
void setup()
{
  Serial.begin(9600);

  pinMode(TASTER, INPUT_PULLUP);

  // Zufallsgenerator starten
  randomSeed(analogRead(0));

  // OLED starten
  Wire.begin();
  Wire.setClock(400000L);
  oled.begin(&Adafruit128x64, I2C_ADDRESS);

  // Schriftart bestimmen
  oled.setFont(font5x7);

  oled.println("Lottozahlen");
}
```



Verwende für Bestimmung der Spielzahlen und die Ziehung der Lottozahlen die Funktion Ziehung:

```
int Ziehung()
{
  for (int i = 0; i < Anzahl; i++)
  {
    /*
     * die Lottozahl gilt solange als bereits gezogen
     * bis in der for-Schleife nachgewiesen wird, dass sie neu ist
     * und BereitsGezogen den Wert false hat
     */
    bool BereitsGezogen = true;
    while (BereitsGezogen)
    {
      // Zahl ziehen
      LottoZahl[i] = random(Minimum, Maximum);
      BereitsGezogen = false;
    }
    /*
     * Zufallszahl mit den bereits gezogenen Zahlen vergleichen
     * i wird in der ersten for-Schleife hochgezählt
     * alle bisherigen Zahlen (ii) werden mit den bisher gezogenen Zahlen
     * (außer der gerade gezogenen) Zahlen verglichen (ii < i)
     * ist sie identisch, erhält BereitsGezogen den Wert true
     * es wird erneut eine Zufallszahl bestimmt
     * wenn die Zahl neu ist, (BereitsGezogen hat noch den Wert false)
     * wird die while-Schleife verlassen und die nächste Zahl
     * gezogen
     */
  }
```

```

        for (int ii = 0; ii < i; ii++)
        {
            if (LottoZahl[i] == LottoZahl[ii]) BereitsGezogen = true;
        }
    }
}
/*
    das Array mit den Zahlen wird an das Hauptprogramm zurückgegeben
    beim ersten Aufruf sind es die gespielten Zahlen
    beim zweiten Aufruf die gezogenen Lottozahlen
*/
return LottoZahl[Anzahl];
}

```

Die Lottozahlen werden mit ZahlenSortieren() aufsteigend sortiert:

```

void ZahlenSortieren(int Zahlen[], int Groesse)
{
    // unsortierten Bereich des Arrays durchlaufen
    for (int i = 0; i < (Groesse - 1); i++)
    {
        // bei jedem Durchlauf wird das jeweils letzte Element weggelassen
        for (int ii = 0; ii < (Groesse - (i + 1)); ii++)
        {
            /*
                wenn die aktuelle Zahl größer als die nachfolgende Zahl ist
                -> aktuelle Zahl temporär speichern
                -> Zahlen vertauschen
                -> temporäre Zahl der nachfolgenden Zahl zuweisen
            */
            if (Zahlen[ii] > Zahlen[ii + 1])
            {
                int zwischengespeicherteZahl = Zahlen[ii];
                Zahlen[ii] = Zahlen[ii + 1];
                Zahlen[ii + 1] = zwischengespeicherteZahl;
            }
        }
    }
}

```

Der loop-Teil. Beachte die Kommentare.

```
void loop()
{
    // Array für die gespielten Zahlen
    byte SpielZahl[6];

    // Strings und Trefferzahl definieren
    String GespielteZahlen = "";
    String GezogeneZahlen = "";
    byte Treffer = 0;
    byte TasterLesen = digitalRead(TASTER);

    if (TasterLesen == LOW)
    {
        delay(200);

        // Ziehung der "angekreuzten" Zahlen
        Ziehung();

        // Zahlen sortieren
        ZahlenSortieren(LottoZahl, 6);

        // String GespielteZahlen aus den Elementen des Arrays LottoZahl
        // zusammensetzen
        for (byte i = 0; i < Anzahl; i++)
        {
            GespielteZahlen = GespielteZahlen + LottoZahl[i] + " ";
            SpielZahl[i] = LottoZahl[i];
        }

        // Ausgabe OLED
        oled.clear();
        oled.println("Gespielte Zahlen");
        oled.println(GespielteZahlen);
        oled.println("-----");

        // Ziehung der Lottozahlen
        Ziehung();

        // Zahlen sortieren
        ZahlenSortieren(LottoZahl, 6);

        // Vergleichen der Elemente der Arrays SpielZahl und LottoZahl
        for (byte i = 0; i < Anzahl; i++)
        {
            for (byte ii = 0; ii < Anzahl; ii++)
            {
                // Übereinstimmung gefunden -> Treffer um 1 erhöhen
                if (SpielZahl[i] == LottoZahl[ii]) Treffer ++;
            }
        }
    }
}
```

```
// String GezogeneZahlen aus den Elementen
// des Arrays LottoZahl zusammensetzen
for (byte i = 0; i < Anzahl; i++)
{
    GezogeneZahlen = GezogeneZahlen + LottoZahl[i] + " ";
}

// Anzeige OLED
oled.println("Gezogene Zahlen");
oled.println(GezogeneZahlen);
oled.println("-----");

if (Treffer == 0) oled.print("keine Treffer");
else
{
    oled.print(Treffer);
    oled.print(" Treffer");
}
}
```

Hartmut Waller (hartmut-waller.info/arduino-blog) Letzte Änderung: 10.05.23