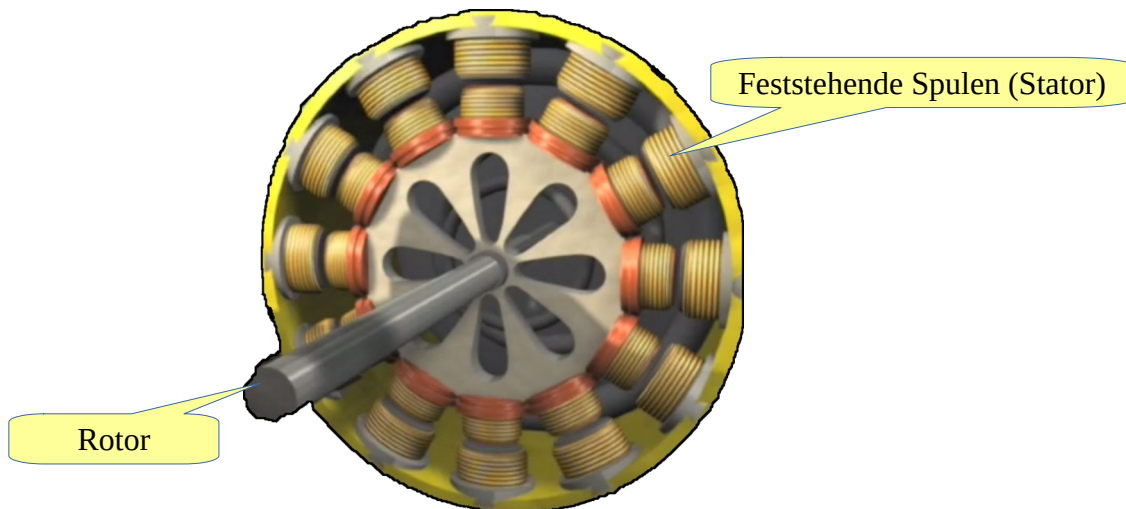


Glücksrad mit einem Schrittmotor

In einem Schrittmotor wird durch das Zusammenspiel von Statoren, feststehenden äußeren



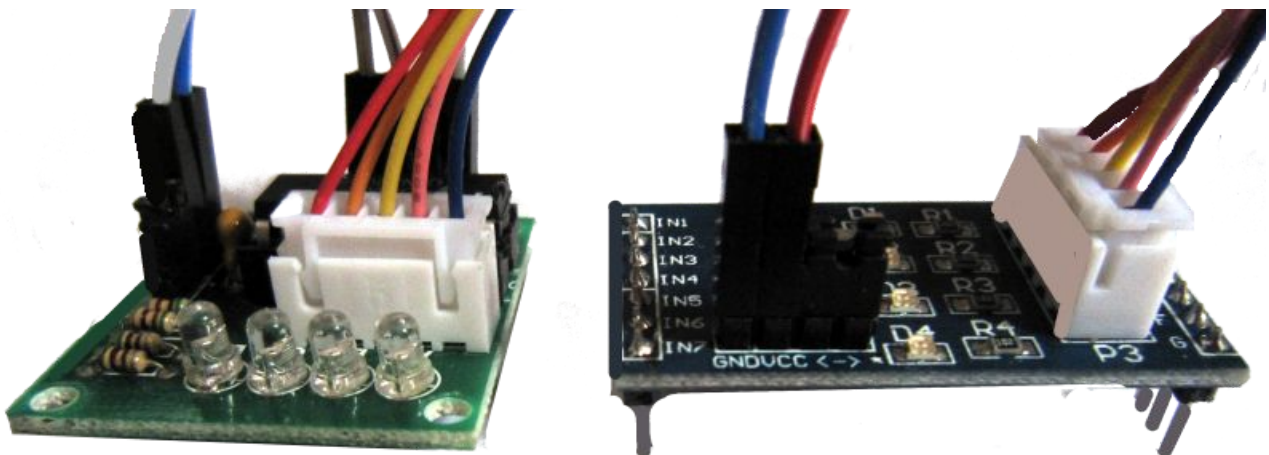
Quelle: Bildschirmschnappschuss aus:
Total phänomenal - Strom aus Strömung

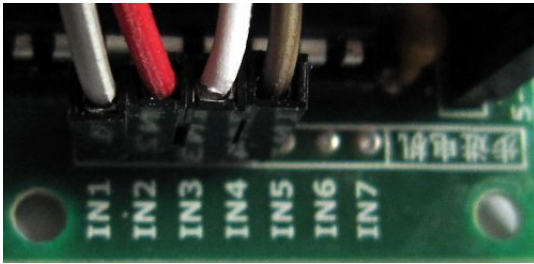
magnetischen Spulen, und einem sich darin drehenden Rotor eine Drehbewegung erzeugt. Die Drehbewegung entsteht also durch gezieltes Ein- und Ausschalten einzelner Spulen.

Für den Schrittmotor wird eine Treiberplatine benötigt. In diesem Fall ist es die ULN2003. Zwei mögliche Ausführungen sind hier abgebildet:

Bei der linken Treiberplatine müssen die Steckverbindungen mit Kabeln zum Arduino hergestellt werden, bei der rechten Ausführung kann der linke Teil direkt in den Arduino gesteckt werden.

Bei beiden Varianten wird der Stecker vom Motor in die Treiberplatine gesteckt.





Pinbelegung
 IN1 → D4
 IN2 → D5
 IN3 → D6
 IN4 → D7

Bei dem Motor handelt es sich um einen Motor mit der Typbezeichnung 28BYJ-48. Im Unterschied zum Servomotor kann der Schrittmotor eine 360°-Drehung ausführen und kann vorwärts und rückwärts laufen.

Ein volle Umdrehung sind bei diesem Motor 2048 Schritte.

Das Glücksrad besteht aus 4 verschiedenfarbigen Feldern. Das Programm ermittelt eine zufällige Anzahl von Schritten zwischen 1 und 2048. Der Motor dreht bis zur angegebenen Position und stellt anhand der zurückgelegten Schritte die Farbe des Feldes fest. Die Farbe wird auf einem LCD angezeigt. Zusätzlich wird die Anzahl der „getroffenen“ Farben gezählt und ebenfalls angezeigt.

So sieht es aus:

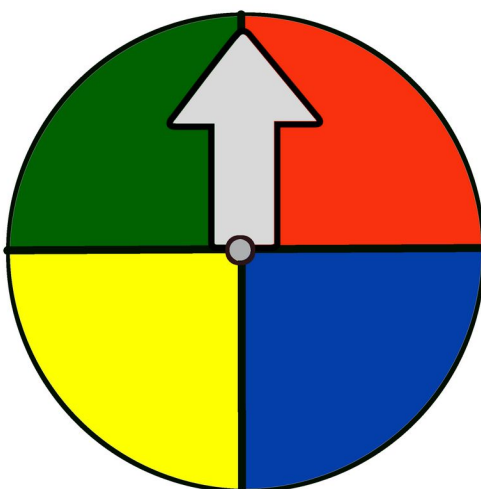


Anzeige LCD



Anzeige Serieller Monitor

Die Vorlage für das Glücksrad und den Pfeil findest du [hier](#).

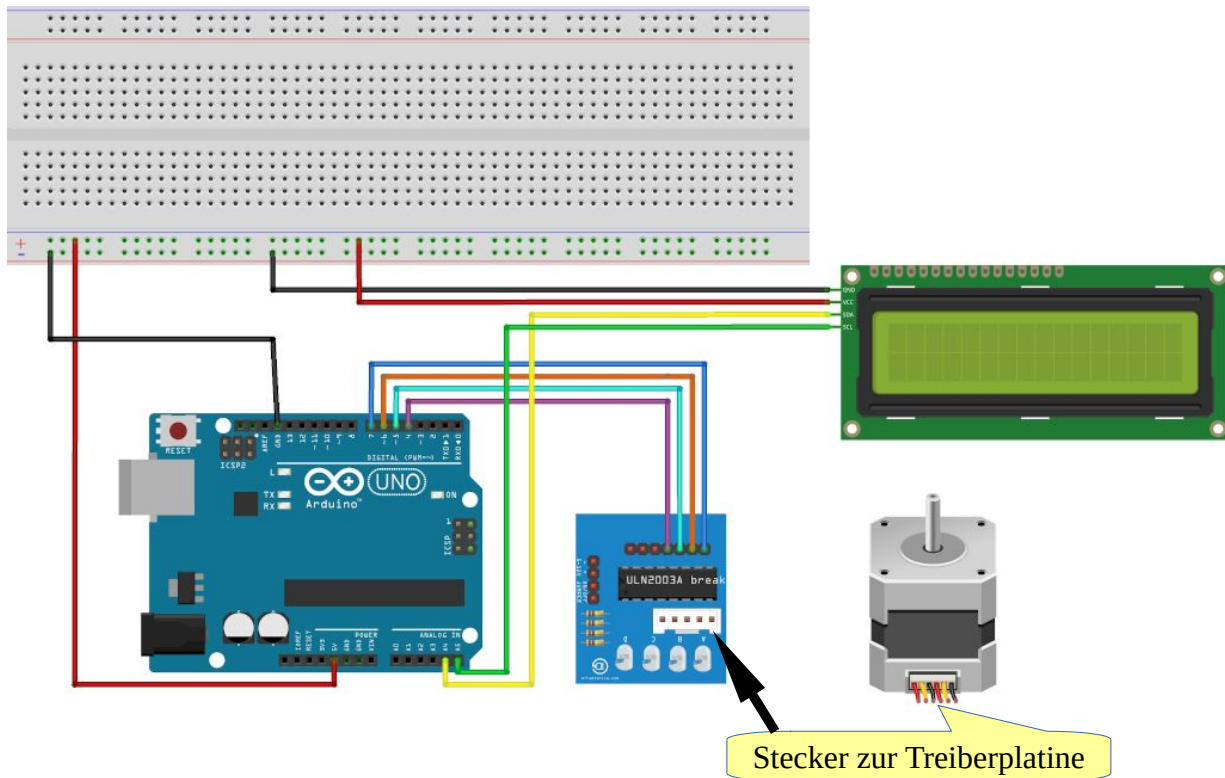


Befestige das Glücksrad und den Pfeil am Schrittmotor. Achte darauf, dass sich das Glücksrad nicht drehen kann, der Pfeil sich aber frei bewegen kann und sich in der richtigen Position befindet.

Benötigte Bauteile:

- LCD 1602
- Schrittmotor mit Treiberplatine
- Leitungsdrähte

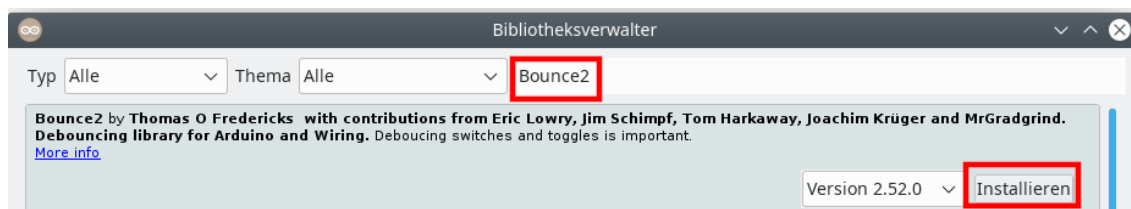
Baue die Schaltung auf.



fritzing

Benötigte Bibliotheken:

Sketch → Bibliothek einbinden → Bibliotheken verwalten



Das „Prellen“ des Tasters soll mit der Bibliothek Bounce2 verhindert werden.
Binde auch die übrigen Bibliotheken ein und definiere die Variablen.
Beachte die Kommentare.

```
# include <Stepper.h>
# include <LiquidCrystal_I2C.h>
# include <Bounce2.h>

LiquidCrystal_I2C lcd(0x27, 20, 4);

// Schritte pro Umdrehung.

int SchritteDrehung = 2048;

/*
  Pinbelegung:
  entweder 4, 6, 5, 7
  oder 5, 7, 4, 6
  diejenige auswählen, bei der die Drehung
  im Uhrzeigersinn ausgeführt wird
*/
Stepper Motor(SchritteDrehung, 5, 7, 4, 6);
int TASTER = 10;

// die Farben
int Rot = 0;
int Blau = 0;
int Gelb = 0;
int Gruen = 0;

// Bibliothek Bounce2
// "Prellverhinderer" für den Taster starten
Bounce Start = Bounce();
```

Der setup-Teil:

```
void setup()
{
  // Instanz des Objekts Bounce für den Taster zuordnen
  // Zeitintervall einstellen
  Start.attach(TASTER);
  Start.interval(20);

  pinMode(TASTER, INPUT_PULLUP);

  // LCD einschalten
  lcd.init();
  lcd.backlight();

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

  // Geschwindigkeit des Motors
  Motor.setSpeed(15);
}
```

Der loop-Teil. Beachte die Kommentare.

```
void loop()
{
  // Taster gedrückt
  if (Start.update())
  {
    if (Start.read() == LOW)
    {
      int Minimum = 1;
      int Maximum = SchritteDrehung;
      int Position = random(Minimum, Maximum);
      Motor.step(Position);
      Serial.println(Position);

      lcd.clear();
      lcd.setCursor(0, 0);
      lcd.print("Position: " + String(Position));
      lcd.setCursor(0, 1);
    }
  }
}
```

```

// Farbe je nach Position bestimmen
if (Position > 0 && Position < 512)
{
    lcd.print("Rot");
    Rot ++;
}

if (Position > 512 && Position < 1024)
{
    lcd.print("Blau");
    Blau ++;
}

if (Position > 1024 && Position < 1536)
{
    lcd.print("Gelb");
    Gelb ++;
}

if (Position > 1536 && Position < 2048)
{
    lcd.print("Gr\365n");
    Gruen ++;
}

lcd.setCursor(0, 2);
lcd.print("Rot: " + String(Rot));
lcd.setCursor(10, 2);
lcd.print("Blau: " + String(Blau));
lcd.setCursor(0, 3);
lcd.print("Gelb: " + String(Gelb));
lcd.setCursor(10, 3);
lcd.print("Gr\365n: " + String(Gruen));

delay(1000);

// Motor an den Ausgangspunkt zurueckfahren
Motor.step(-Position);
}
}
}

```



Glücksrad mit Schrittmotor: Anzeige im Seriellen Monitor