



Ziel des Projekts

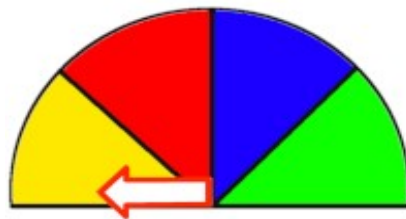


Wenn der Taster gedrückt wird, soll der Servomotor auf eine zufällige Position fahren. Der Zeiger bleibt dann auf einem der farbigen Felder stehen.

Die Hardware



Der 180° Servomotor ist ein spezieller Elektromotor, der eine 180°-Drehung ausführen und der innerhalb dieser Drehung zu einer beliebigen Position gefahren werden kann.

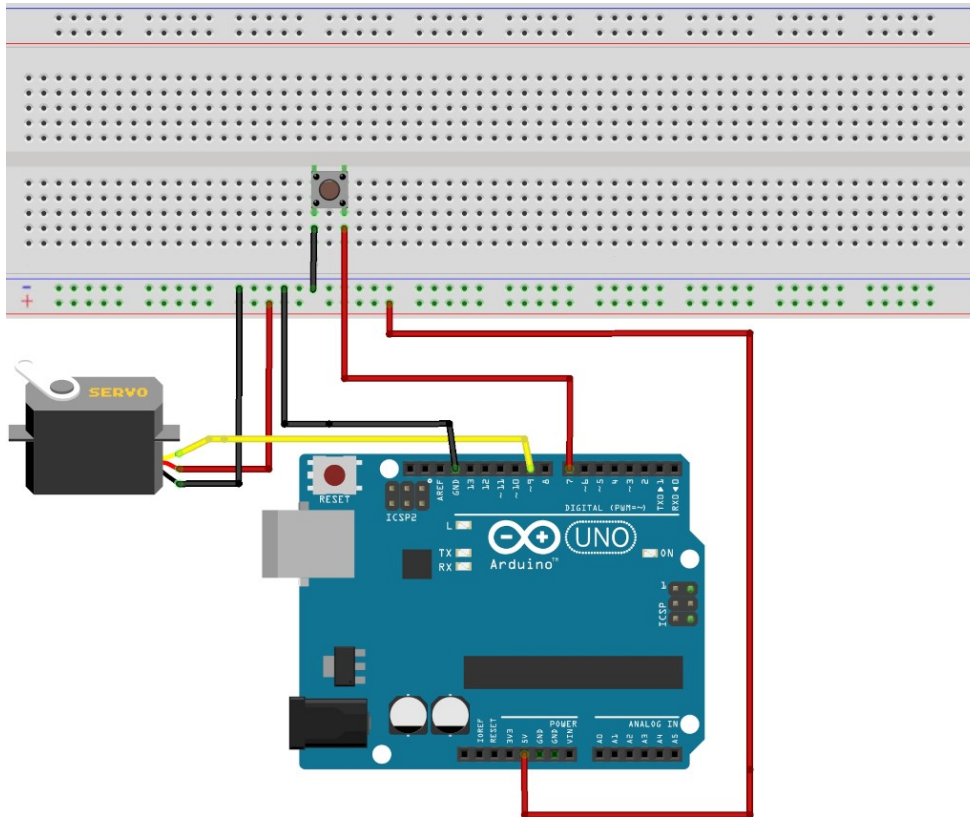


Lege den Motor auf die Seite, befestige das Glücksrad und den Zeiger am Motor.

Benötigte Bauteile

- ➡ 180° Servomotor
- ➡ Taster
- ➡ Leitungsdrähte

Der Schaltplan



fritzing

Das Programm

Binde die Bibliothek `Servo.h` ein und definiere die Variablen.

```
# include <Servo.h>

// Name des Motors festlegen
Servo Motor;

int Position;
int TASTER = 7;
int TasterLesen;
```

Im `setup`-Teil wird der Zufallsgenerator gestartet und der Taster definiert. Der Motor wird auf die 180°-Position (nach links) gefahren. Du musst den Motor und das Glücksrad so positionieren, dass der Zeiger links steht.

```
void setup()
{
  pinMode(TASTER, INPUT_PULLUP);
  pinMode(LAUTSPRECHER, OUTPUT);

  // Pin 9 dem Motor zuordnen
  Motor.attach(9);
```



```
// Zufallsgenerator starten
randomSeed(A0);

// Motor nach links fahren
Motor.write(180);
}
```

Im loop-Teil wird auf Tasterdruck eine Zufallszahl zwischen 5 und 175 bestimmt und der Motor wird zu dieser Position gefahren.

```
void loop()
{
  TasterLesen = digitalRead(TASTER);
  if (TasterLesen == LOW)
  {
    delay(500);
    int Minimum = 5;
    int Maximum = 175;

    // Zufallszahl ermitteln
    int Position = random(Minimum, Maximum);

    // Motor zur Zufallsposition fahren
    Motor.write(Position);
    delay(500);
  }
}
```

Schließe einen Lautsprecher an und ergänze das Programm:
Wenn der Zeiger im blauen Feld stehen bleibt, soll eine Tonfolge wiedergegeben werden.

Ergänze im Kopf des Programms:

```
int LAUTSPRECHER = 6;
```

Füge im loop-Teil hinzu:

```
void loop()
{
  TasterLesen = digitalRead(TASTER);
  if (TasterLesen == LOW)
  {
    delay(500);
    int Minimum = 5;
    int Maximum = 175;
    int Position = random(Minimum, Maximum);
    Motor.write(Position);
    delay(500);
  }
}
```



```
// diesen Teil einfügen
if (Position > 45 && Position < 90)
{
    // Frequenzbereich 1000 bis 2000
    for (int i = 1000; i <= 2000; i += 100)
    {
        tone(LAUTSPRECHER, i);
        delay(100);
        noTone(LAUTSPRECHER);
    }

    for (int i = 2000; i > 1000; i -= 100)
    {
        tone(LAUTSPRECHER, i);
        delay(100);
        noTone(LAUTSPRECHER);
    }
}
// Ende des eingefügten Teils
}
```

Du kannst auch andere Frequenzbereiche verwenden.

Hartmut Waller letzte Änderung: 22.11.25