

Arduino Projekt: Servo ansteuern

Dank der mit der Arduino Entwicklungsumgebung gelieferten Servo Library ist der Anschluss und die Ansteuerung eines [Servomotors \(Wikipedia\)](#) recht einfach.

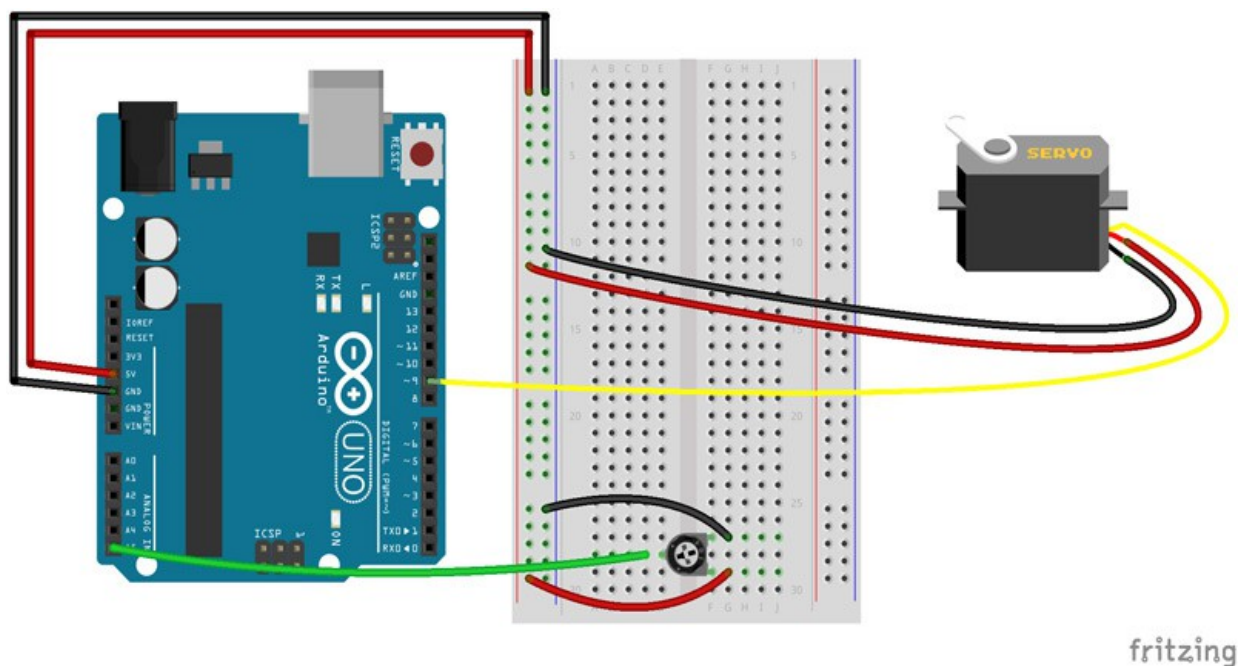
Ein kleiner Standardservo, wie er zum Beispiel im Modellbau verwendet wird, benötigt neben der Stromversorgung nur einen weiteren Pin als Steuerleitung. Für die Stromversorgung reichen die 5V vom Arduino Board aus. Das Steuersignal, welches den gewünschten Einstellwinkel angibt, wird mit Hilfe der [Pulsweitenmodulation \(Wikipedia\)](#) übertragen. Mit den aktuellen Servo-Libraries kann aber ein beliebiges Pin am Arduino gewählt werden.

An die Steuerleitung wird alle 20 ms ein zwischen 1 und 2 ms langer Puls gesendet, wobei eine Pulsdauer von 1 ms für 0 Grad und 2 ms für 180 Grad Einstellwinkel stehen. Ein Puls von 1.5 ms Länge würde den Servo also auf 90 Grad setzen.

Wie ein Servo intern funktioniert, ist hier ganz gut erklärt:

<http://www.electronicplanet.ch/Roboter/Servo/intern/intern.htm>

Der Aufbau der Schaltung:



Im Beispielprojekt koppeln wir die Position des Servos an die Stellung eines Potentiometers.

Dazu wird der Wert des Potentiometers an A5 eingelesen, aus dem Wertebereich von 0-1023 nach 0-180 umgerechnet und als Einstellwinkel an den Servo gesendet. Den komplizierten Teil mit der Umwandlung in ein korrektes PWM-Signal übernimmt dabei glücklicherweise die [Arduino Servo Library](#) für uns.

Und hier der Arduino Sketch dafür:

```
// Servo Library einbinden
#include <Servo.h>

// Neues Servo Objekt erstellen
Servo myservo;
// Wert des Potentiometers hier speichern
int val;

// setup() wird einmal zu Programmbeginn ausgeführt
void setup()
{
  // Servo an Pin 9 koppeln
  myservo.attach(9);
  // Serielle Kommunikation starten
  Serial.begin(9600);
}

// loop() wird endlos wiederholt
void loop()
{
  // Stellung des Potentiometers an Analog-Eingang 5 auslesen
  val = analogRead(5);
  // 10-bit Wert des Analogeingangs (0-1023) in Winkel 0-180 umrechnen
  val = map(val, 0, 1023, 0, 180);

  // Errechneten Winkel zur Kontrolle an den PC übertragen (Seriellen Monitor
  starten!)

  Serial.println(val);
  // Einstellwinkel in Grad an das Servo-Objekt schicken
  myservo.write(val);
  // Kurze Pause, damit der Servo die neue Position anfahren kann
  delay(50);
}
```