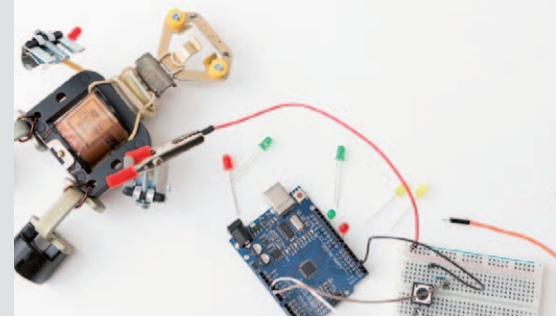


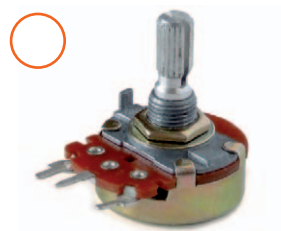
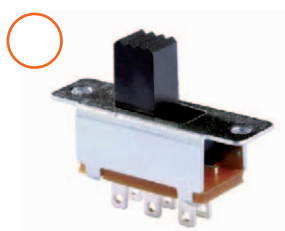
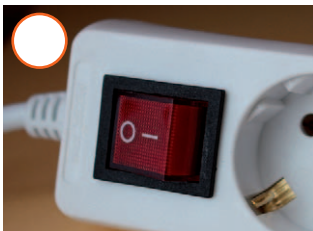
22 Planung und Programmierung der Robotersteuerung

Für die Ansteuerung einer elektrischen Anlage – also auch eines Roboters – wird ein eigenständiger kleiner Computer benötigt. Wir sind von vielen dieser Minicomputer oder Mikrocontroller (MC) umgeben. Hier arbeiten wir beispielhaft mit einem „Arduino“. Um einen MC wie den Arduino nutzen zu können, muss er zuvor programmiert werden. Danach wird er meist mithilfe eines Breadboards verdrahtet und an die benötigten Sensoren und Aktoren angeschlossen.



1. a) Für die Ansteuerung der Servomotoren lassen sich verschiedene Bauteile verwenden. Benenne die abgebildeten Bauteile.

b) Um die Komponenten des Greifarms stufenlos zu bewegen, werden Schaltelemente benötigt, die möglichst viele Wertabstufungen zulassen. Ordne die Bauteile nach der Anzahl der Teilschritte, die sie schalten können und trage die Ziffern 1 bis 5 in die Kreise ein: von 1 = höchste Anzahl der Schaltstufen bis 5 = nur zwei Schaltstufen an/aus möglich (einzelne Ziffern können mehrfach vorkommen!).



c) Wähle das passende Bauteil für die Ansteuerung des Greifarms und begründe.

2. Für die Ansteuerung der Servomotoren wird ein getaktetes Signal verwendet. Dieses kann z. B. von einem Arduino bereitgestellt werden. In einem ersten Schritt muss dazu der Mikrocontroller mit einer passenden Software programmiert werden. Hierfür gibt es eine spezielle Programmiersprache, mit der jeder einzelne Pin angesprochen oder ausgelesen werden kann.

Jedes Programm für einen Mikrocontroller ist immer nach folgendem Muster aufgebaut:

1. Benennung der Variablen,
2. Zuweisung von Funktionsweisen und
3. Programmlogik.

Bearbeite nun die Aufgaben auf der nächsten Seite.

a) Das folgende Programm erläutert die Funktion der drei Teile am Beispiel eines angeschlossenen Servomotors ausführlicher. Analysiere das Programm und beantworte die folgenden Fragen.

```

// Programmierung von einem Servo

#include <Servo.h> // Den Befehl für eine Servosteuerung aktivieren

Servo S1; // Bereitstellen des Servo 1

int potpin1 = A0; // Bereitstellen des analogen Eingang Pin A0 für das Poti 1 Signal
int vall; // Variable zum Lesen des Analogen Signals 1

void setup() {
  S1.attach(8); // Bereitstellen des Anschluss Pin 8 für das Ausgangssignal an den Servo 1
}

void loop() {
  vall = analogRead(potpin1); // Auslesen des Potis 1 (Wert zwischen 0 und 1023)
  vall = map(vall, 0, 1023, 0, 180); // Skalieren der Servodrehbewegung (Wert zwischen 0 und 180)
  S1.write(vall); // Setzen der Servoposition
  delay(40); // Zeit die der Servo 1 benötigt
}

```

Welches Bauteil zur Ansteuerung aus Aufgabe 1 wird verwendet?

An welchem Pin des Arduinos wird das Eingangssignal für dieses Bauteil eingespeist?

An welchem Pin des Arduinos wird das Ausgangssignal für den Servo bereitgestellt?

Um wieviel Grad kann sich der angeschlossene Servo drehen?

Welchen Wert muss das angeschlossene Potenziometer haben?

- b)** Erweitere das Programm auf insgesamt 4 Servomotoren.
- 3.** Das Programm muss nun in den Arduino übertragen werden. Verbinde dazu Arduino und Computer wie folgt mit einem geeigneten USB-Kabel:
- Suche im Programm unter dem Menüpunkt „Werkzeuge“ den Punkt „Boardinformationen holen“.
 - Stelle nun im Programm unter dem Menüpunkt „Werkzeuge“ im Menüpunkt „Board“ das richtige Board ein, im Menüpunkt „Prozessor“ den richtigen Chip deines Arduinos und den zu verwendenden Port.
 - Suche im Programm unter dem Menüpunkt „Sketch“ den Menüpunkt „hochladen“ und übertrage dein Programm. Wenn das Überspielen geklappt hat, blinkt der Arduino und du erhältst eine entsprechende Meldung.