

wird daraus ein kompletter und leistungsfähiger Linux-Rechner. Ein zusätzlicher PC oder Laptop ist dann nicht mehr erforderlich.

Kapitel 2 • Grundelemente der Robotik

Der besondere Reiz, aber auch die Problematik bei der Beschäftigung mit der Robotik liegt darin, dass eine Vielzahl von technischen Disziplinen erforderlich sind, um einen vollständigen Roboter aufzubauen. So sind mehr oder weniger detaillierte Kenntnisse in den folgenden Technologiebereichen notwendig:

- Mikrocontrollertechnik
- Programmierung
- Elektrotechnik und Elektronik
- Sensortechnik
- Mechanik

Die folgenden Kapitel geben jeweils Einführungen in diese Gebiete. Natürlich können die einzelnen Bereiche nicht immer in ihrem vollen Umfang dargelegt werden. Insbesondere für Einsteiger in ein bestimmtes Gebiet kann es daher sinnvoll sein, weitere Literatur zu Rate zu ziehen. Das Literaturverzeichnis am Ende des Buches kann dabei Hilfestellung leisten.

Kapitel 3 • Controller und Prozessoren: Die "Gehirne" der Roboter

Moderne Roboter kommen praktisch nicht mehr ohne einen eigenen Controller oder Prozessor aus. Diese Komponenten ist daher eine der zentralen Einheiten in der Robotik geworden. Der Controller/Prozessor stellt sozusagen das "Gehirn" der Maschine dar. Hier erfolgt die Erfassung von Sensorwerten oder die Ausgabe von Steuerbefehlen für die mechanischen Einheiten des Robots.

Prinzipiell könnten Roboter zwar auch ohne programmierbare Komponenten auskommen. So existieren etwa Konzepte und Bausätze, die Konstruktionen ermöglichen, welche auf rein analoger Basis arbeiten. Eine Hinderniserkennung und -umfahrung kann beispielsweise allein mit optischen Sensoren und Operationsverstärkern umgesetzt werden. Allerdings findet diese Technik auch rasch ihre Grenzen. Komplexere Algorithmen wie etwa kontrolliertes Zurücksetzen oder aber die Erkennung verschiedener Hindernisse lassen sich kaum mehr mit rein analog-elektronischen Schaltungen realisieren.

Andere Aufgaben wie etwa Bild- oder Mustererkennung sind mit rein analogen Mitteln ohnehin kaum lösbar. Spätestens hier muss man auf digitale Komponenten zurückgreifen. Neben der Digitalisierung spielt auch die Programmierbarkeit eine wesentliche Rolle. Fest verdrahtete Systeme können nur noch mit erheblichem Aufwand verändert und verbessert werden. Ein mit Software umgesetzter Algorithmus kann dagegen problemlos kontinuierlich angepasst, modifiziert und optimiert werden. Auf diese Art und Weise können einem Roboter immer wieder neue Lösungswege "beigebracht" werden. Im Rahmen dieses Buches werden vor allem die beiden bereits in der Einleitung angesprochenen Systeme "Arduino" und "Raspberry Pi" eingesetzt, um den Robotern maximale Flexibilität zu verleihen. Die nächsten Kapitel beschäftigen sich daher eingehender mit den beiden beliebten Boards.

3.1 Arduino als programmierbare Steuerzentrale

Das System "Arduino" stellt eine Open-Source-Plattform für den Bau von Mikrocontroller- und Elektronikprojekten dar. Es besteht aus einer Platine mit einem programmierbaren ATmega328 Mikrocontroller und einer Software, der sogenannten IDE (**I**ntegrated **D**evelopment **E**nvironment), die auf einem PC oder Laptop ausgeführt wird. Mit Hilfe der IDE werden Programme, auch als "Sketches" bezeichnet, entwickelt und anschließend auf den Controller hochgeladen.

Die Arduino-Plattform erfreut sich bei Elektroneinsteigern seit über zehn Jahren größter Beliebtheit. Neben einer Vielzahl von Sensoren können auch Aktoren wie Servos oder Motoren mit dem Arduino problemlos angesteuert werden. Dies macht ihn auch für Robotik-Anwendungen zum Mittel der Wahl.

Im Gegensatz zu früheren Mikrocontroller-Systemen benötigt der Arduino keine separate Hardware wie etwa Programmiergeräte oder EEPROM-Brenner, um neuen Code auf den Controller zu laden. Ein einfaches USB-Kabel genügt, um den Arduino zu programmieren. Darüber hinaus verwendet die Arduino IDE eine vereinfachte Version von C++, um das Programmieren zu erleichtern.

Arduinos sind inzwischen in einer nahezu unüberschaubaren Anzahl von Varianten verfügbar. Neben dem klassischen Arduino sind die verschiedensten Größen und Formen erhältlich. Genannt sei hier der Arduino MEGA, der mit einem ATmega1280 oder ATmega2560 bestückt ist. Diese Prozessoren weisen einen erheblich erweiterten Funktionsumfang auf und auch die Anzahl der verfügbaren Pins ist deutlich größer. Wegen der großen Anzahl von I/O Pins haben die Boards auch etwa die doppelte Größe eines klassischen Arduinos.

Wenn man dagegen möglichst kleine und kompakte Geräte aufbauen möchte, kann man auf die Mikro- oder Nano-Versionen zurückgreifen. Diese Boards haben nur noch die Größe einer Briefmarke. Anstelle der Kontaktbuchsen weisen sie lediglich Lötunkte auf. Diese können direkt mit Kabeln verlötet werden. Alternativ sind hier auch Stiftleisten einlötlbar, sodass diese kompakten Boards direkt in ein sogenanntes Breadboard oder aber in eine IC-Fassung eingesetzt werden können.

Als Goldstandard hat sich allerdings der Arduino UNO herauskristallisiert. Wird gemeinhin von einem Arduino gesprochen, so ist meist der "UNO" gemeint (s. Abb.).



Abbildung 3.1: Arduino Uno

Die folgende Tabelle fasst die Technischen Daten des UNOs zusammen:

Mikrocontroller	ATmega328P
Betriebsspannung	5 V
Digitale I/O Pins	14
PWM Kanäle	6
Analoge Eingänge	6
Serielle Ports (USART)	1
I ² C-Ports	1
SPI-Ports	1
Timer	3

Der Vollständigkeit halber hier noch auf die unterschiedlichen Pin-Abstände der Arduino-Buchsen hingewiesen. Bei der Entwicklung der klassischen Arduino-Variante wurden nicht alle Buchsen im gängigen Rastermaß von 1/10 Zoll bzw. 2,54 mm angeordnet. So weisen die beiden oberen Buchsenleisten einen Abstand von nur 1/20 Zoll auf. Damit lassen sich Lochrasterplatinen mit Standardrastermaß nicht direkt über Stiftleisten mit allen Buchsen verbinden. Hier muss man also immer auf Sonderlösungen zurückgreifen.

3.2 Arduino-IDE

Das für die Programmierung des Arduinos erforderliche Programmpaket kann unter

www.arduino.cc/en/Main/Software

kostenlos aus dem Internet geladen werden. In dieser "Arduino-Programmierungsumgebung" werden die Programme erstellt, die der Mikrocontroller des Arduinos später ausführt. Im Arduino-Umfeld sind diese Programme auch unter dem Namen "Sketch" bekannt. Für den Download der Software stehen zwei Optionen zur Verfügung:

1. ein Installationspaket
2. eine ZIP-Datei

Im ersten Fall wird die Programmieroberfläche durch den Aufruf einer Installationsdatei auf dem Rechner installiert. Im zweiten Fall muss man die gepackte Datei herunterladen und in ein eigenes Verzeichnis entpacken.

Ist die Installation abgeschlossen, öffnet man den Softwareordner und startet das Programm mit der Datei `arduino.exe`. Zunächst sind nun zwei wichtige Einstellungen vorzunehmen:

Als erstes muss das richtige Board ausgewählt werden. Im Fall des Arduino UNO wird der Board-Typ "Arduino UNO" verwendet. Die entsprechende Auswahl kann unter

Werkzeuge → Board

getroffen werden.