

1

Einleitung

Mit dem Arduino und dem Raspberry Pi ist es für Bastler leichter denn je, in die Welt der Elektronik einzusteigen, etwa um ein Heimautomatisierungssystem im Eigenbau zu realisieren, mit dem Sie die Beleuchtung und die Heizung über ein WLAN steuern können, oder einfach um Motoren zu regeln.

In diesem Buch zeige ich Ihnen, wie Sie diese beiden beliebten Plattformen einsetzen können, um mit Ihrem Raspberry Pi oder Arduino Bewegungen, Licht und Ton hervorzurufen und zu steuern.

Arduino und Pi

Sowohl der Arduino als auch der Raspberry Pi sind kleine Platinen ungefähr von der Größe einer Kreditkarte, aber im Grunde genommen handelt es sich bei ihnen um sehr unterschiedliche Geräte. Der Arduino ist eine sehr einfache Mikrocontroller-Platine ohne irgendeine Form von Betriebssystem, während es sich bei dem Raspberry Pi um einen Minicomputer mit Linux handelt, an den außerdem externe elektronische Geräte angeschlossen werden können.

Der Raspberry Pi

Wenn Sie noch keine Erfahrungen mit Elektronik haben, aber mit Computern vertraut sind, ist der Raspberry Pi (siehe Abb. 1–1) für Sie das geläufigere Gerät. Er stellt eine äußerst kleine Version eines normalen Linux-Computers dar und verfügt über USB-Anschlüsse für eine Tastatur und eine Maus, einen HDMI-Videoausgang für einen Monitor oder einen Fernseher und einen Audioausgang.

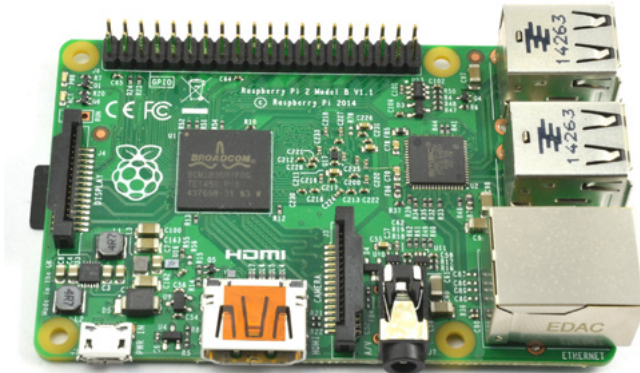


Abb. 1–1 Ein Raspberry Pi 2

Über seinen Ethernetanschluss können Sie den Raspberry Pi auch mit einem Netzwerk verbinden. Sie können aber auch USB-WLAN-Adapter anschließen. Die Stromversorgung erfolgt über eine Micro-USB-Buchse.

Für die Speicherung wird statt eines herkömmlichen Festplattenlaufwerks eine Micro-SD-Karte verwendet. Sie enthält sowohl das Betriebssystem als auch Ihre Dokumente und Programme.

Entwickelt wurde der Raspberry Pi in Großbritannien, hauptsächlich als kostengünstiger Computer, um Schulkindern die Grundlagen der Informatik beizubringen, insbesondere die Programmierung in Python. Der Name Pi soll sogar von *Python* abgeleitet sein.

Zwischen einem normalen Desktop- oder Laptop-Computer mit Linux und dem Raspberry Pi bestehen vor allem folgende Unterschiede:

- Der Pi kostet lediglich um die 40 €. (Ein abgespecktes Modell namens A+ ist für einen noch niedrigeren Preis erhältlich, und das Modell 0 kostet sogar noch weniger.)
- Der Pi hat eine Leistungsaufnahme von weniger als 5 W.
- Der Pi verfügt über eine Doppelreihe von GPIO-Pins (General Purpose Input/Output, also »Allzweck-E/A-Pins«), an die Sie direkt elektronische Geräte anschließen können. (In Abb. 1–1 sehen Sie diese Pins auf der linken Seite der oberen Kante.) Über diese Pins können Sie LEDs, Displays, Motoren und viele weitere Arten von Ausgabegeräten steuern, mit denen wir in diesem Buch noch arbeiten werden.

Außerdem kann der Raspberry Pi über ein WLAN oder ein LAN-Kabel mit dem Internet verbunden werden, was ihn für Projekte im Rahmen des »Internets der Dinge« (siehe Kapitel 16) geeignet macht.

Der in diesem Buch hauptsächlich verwendete Raspberry Pi 2 weist folgende technische Daten auf:

- 900-MHz-Quad-Core-Prozessor ARM v7
- 1 GB DDR2-Speicher
- 100-BaseT-Ethernet
- 4 USB-2.0-Anschlüsse
- HDMI-Videoausgang
- Buchse mit Kameraschnittstelle
- 40-polige GPIO-Stiftleiste (alle Pins werden mit 3,3 V betrieben)

Nach Abfassung der Originalausgabe dieses Buches ist das neue Modell Raspberry Pi 3 auf den Markt gekommen.

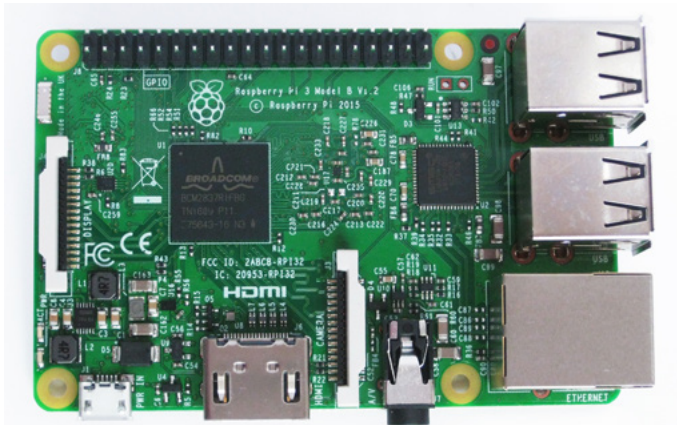


Abb. 1–2 Optisch kaum vom Raspberry Pi 2 zu unterscheiden: das neue Modell 3

Äußerlich sieht der Pi 3 praktisch genauso aus wie das Modell 2, und auch alle vertrauten Anschlüsse sind unverändert vorhanden. Es hat jedoch erhebliche »unsichtbare« Neuerungen gegeben, nämlich bei der Rechenleistung und bei den Anschlüssen. So ist der Raspberry Pi 3 jetzt mit einem 64-Bit-Vierkernprozessor vom Typ ARMv8 mit 1,2 GHz ausgestattet und verfügt über eine 802.11n-WLAN-Verbindung sowie über einen Bluetooth-Anschluss (Bluetooth 4.1 und Bluetooth Low Energy). Der in Kapitel 3 erwähnte WLAN-Adapter ist daher bei diesem Modell nicht mehr nötig.

Wenn der Raspberry Pi für Sie neu ist, können Sie sich mit dem Einführungskurs in Kapitel 3 sehr schnell mit der Hardware und der Programmiersprache Python vertraut machen.

Der Arduino

Es gibt eine breite Palette von unterschiedlichen Arduino-Modellen. In diesem Buch konzentrieren wir uns auf das am weitesten verbreitete, nämlich den Arduino Uno (siehe Abb. 1–3). Er ist noch ein bisschen billiger als der Raspberry Pi – Sie können ihn schon für 25 € bekommen.

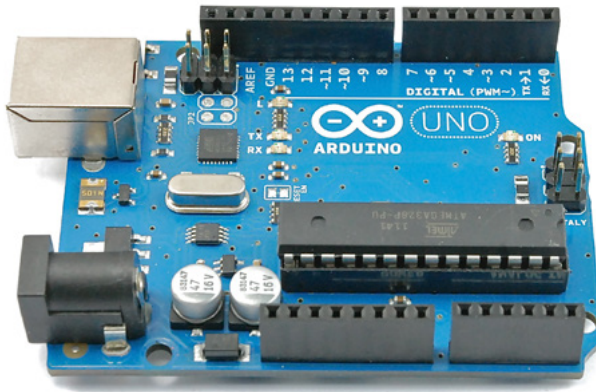


Abb. 1–3 Ein Arduino Uno Revision 3

Wenn Sie die Arbeit mit einem regulären Computer gewohnt sind, werden Ihnen die technischen Daten des Arduino höchst unzureichend vorkommen. Er hat lediglich einen Arbeitsspeicher (unterschiedlicher Art) von 34 KB. Das bedeutet, dass der Raspberry Pi einen etwa 30.000 Mal größeren Arbeitsspeicher aufweist, worin der Flash-Speicher der SD-Karte noch nicht einmal enthalten ist. Des Weiteren beträgt der Prozessortakt des Arduino Uno nur 16 MHz. Es ist auch nicht möglich, eine Tastatur, eine Maus oder einen Monitor an den Arduino anzuschließen oder auf ihm ein Betriebssystem auszuführen.

Vielleicht fragen Sie sich nun, wie dieses kleine Gerät überhaupt irgendetwas Sinnvolles tun kann. Das Geheimnis des Arduino liegt jedoch gerade in seiner Einfachheit. Es gibt kein Betriebssystem, das Sie erst starten müssten, und keine anderen Schnittstellen, die für ein Projekt vielleicht unnötig wären und nur Kosten verursachen und Strom verbrauchen würden.

Der Raspberry Pi ist ein Allzweckcomputer, der Arduino dagegen soll einzig und allein eine Sache gut machen, nämlich das Steuern elektronischer Geräte.

Um einen Arduino zu programmieren, brauchen Sie einen regulären Computer (Sie können dazu auch einen Raspberry Pi verwenden). Auf diesem Computer benötigen Sie eine integrierte Entwicklungsumgebung (Integrated Development

Environment, IDE). Darin schreiben Sie die Programme, die anschließend in den Flash-Speicher des Arduino heruntergeladen werden.

Der Arduino kann immer nur ein Programm auf einmal ausführen. Nachdem er programmiert wurde, merkt er sich das Programm und führt es automatisch aus, sobald er eingeschaltet wird.

Arduinos sind so gestaltet, dass Sie sogenannte *Shields* auf die E/A-Anschlüsse aufstecken können. Diese Shields tragen zusätzliche Hardware, z. B. verschiedene Arten von Anzeigen, Ethernet- oder WLAN-Adapter.

Zur Programmierung eines Arduino verwenden Sie die Programmiersprache C (mehr darüber erfahren Sie in Kapitel 2).

Welches Gerät – Arduino oder Pi?

In diesem Buch wird erklärt, wie Sie elektronische Geräte sowohl an den Arduino als auch an den Raspberry Pi anschließen können. Einer der Gründe dafür besteht darin, dass für einige Projekte der Pi besser geeignet ist, für andere der Arduino. Es gibt noch andere Platinen, die zwischen diesen beiden Extremen liegen. Im Allgemeinen ähneln sie entweder dem Arduino oder dem Raspberry Pi, sodass Ihnen dieses Buch auch dabei helfen kann, mit diesen Alternativen zu arbeiten.

Wenn ich mit einem neuen Projekt beginne, nehme ich im Allgemeinen einen Arduino. Stellt das Projekt aber eine der folgenden Anforderungen, dann ist ein Raspberry Pi wahrscheinlich die bessere Wahl:

- Internet- oder Netzwerkanschluss
- Großer Bildschirm
- Tastatur und Maus
- USB-Peripheriegeräte, z. B. eine Webcam

Mit einigen Kosten und Mühen ist es möglich, einen Arduino mithilfe von Shields so zu erweitern, dass er diese Anforderungen ebenfalls erfüllt. Allerdings ist es schwieriger, die Sachen auf diese Weise zum Laufen zu bekommen, da nichts davon zu den eigentlichen Funktionen des Arduino gehört.

Gute Gründe dafür, den Arduino gegenüber dem Raspberry Pi vorzuziehen, sind die folgenden:

- *Kosten* Ein Arduino Uno ist billiger als ein Raspberry Pi 2.
- *Startzeit* Ein Arduino muss nicht darauf warten, dass das Betriebssystem hochfährt. Es gibt eine kleine Verzögerung von etwa einer Sekunde, in der das Gerät prüft, ob ein neues Programm hochgeladen wird, aber danach ist der Arduino in Betrieb.

- *Zuverlässigkeit* Der Arduino ist ein viel einfacheres und robusteres Gerät als der Raspberry Pi und kommt ohne den Zusatzaufwand eines Betriebssystems aus.
- *Stromverbrauch* Ein Arduino verbraucht nur ein Zehntel so viel Strom wie ein Raspberry Pi. Wenn Sie Ihr Projekt mit Batterie- oder Solarstrom betreiben müssen, ist der Arduino die bessere Wahl.
- *GPIO-Ausgabestrom* Die GPIO-Pins eines Raspberry Pi können nur einen Strom von maximal 16 mA bereitstellen. Die Arduino-Pins dagegen sind für 40 mA ausgelegt. Daher lassen sich manche Dinge (z. B. helle LEDs), die sie auf diese Weise nicht mit einem Raspberry Pi verbinden können, direkt an einen Arduino anschließen.

Sowohl der Arduino als auch der Raspberry Pi sind bestens als Basis für Elektronikprojekte geeignet. Für welches Gerät Sie sich entscheiden, ist in gewissem Maße auch eine Frage des persönlichen Geschmacks.

Beim Anschluss von elektronischen Geräten an den Raspberry Pi müssen Sie unbedingt daran denken, dass er mit 3,3 V betrieben wird – nicht mit 5 V wie der Arduino. Wenn Sie an einen GPIO-Pin des Raspberry Pi 5 V anlegen, können Sie den Pin oder gar den ganzen Pi beschädigen oder zerstören.

Alternativen

Der Arduino und der Raspberry Pi befinden sich an den beiden Enden des Spektrums von Geräten, mit denen sich Geräte steuern lassen. Wie zu erwarten ist, hat der Markt eine große Zahl von anderen Geräten hervorgebracht, die zwischen diesen beiden Polen einzuordnen sind. Manche davon kombinieren die Vorteile der beiden Plattformen.

Neue Geräte kommen ständig auf den Markt. Die Open-Source-Natur des Arduino hat unzählige Varianten möglich gemacht, die auf bestimmte Nischen zugeschnitten sind, z. B. auf die Steuerung von Drohnen oder die Arbeit mit drahtlosen Sensoren.

Abbildung 1–4 zeigt ein Diagramm der am weitesten verbreiteten Geräte auf diesem Gebiet.

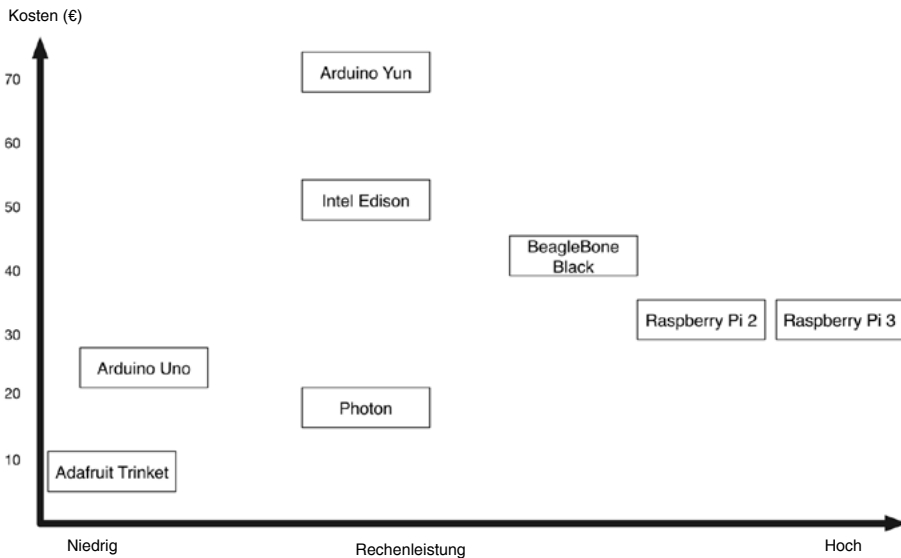


Abb. 1-4 Plattformen für eingebettete Geräte

Unterhalb des Arduino – sowohl was den Preis als auch die Leistung angeht – ist der Adafruit Trinket einzuordnen. Diese interessante Platine verfügt nur über wenige GPIO-Pins, ist ansonsten aber ziemlich gut mit dem Arduino kompatibel. Wenn Sie in einem Projekt nur ein oder zwei Eingänge oder Ausgänge benötigen, ist diese Platine eine Überlegung wert.

Im Mittelfeld liegen Produkte wie der Arduino Yun, der Intel Edison und der Photon. Alle verfügen über WLAN-Fähigkeiten und sind für Projekte im Internet der Dinge (Internet of Things, IoT) gedacht (siehe Kapitel 16). Von diesen Geräten bietet der Photon wahrscheinlich den größten Wert für das Geld. Alle drei Geräte werden in Arduino C programmiert. Was Sie also über die Programmierung des Arduino lernen, können Sie auch für diese Platinen anwenden.

Der BeagleBone Black ist ähnlich konzipiert wie der Raspberry Pi. Auch bei ihm handelt es sich um einen Einplatinencomputer. Was die Rohleistung angeht, fällt die aktuelle Version des BeagleBone Black zwar hinter dem Raspberry Pi zurück, doch dafür bietet er andere Vorteile: Er weist mehr GPIO-Pins auf und verfügt sogar über einige Pins, die als analoge Eingänge dienen können; ein Merkmal, das dem Raspberry Pi 2 fehlt. Den BeagleBone Black können Sie ebenso wie den Raspberry Pi in Python programmieren, aber auch in JavaScript.

Zusammenfassung

In diesem Kapitel haben Sie eine kurze Einführung in den Arduino und den Raspberry Pi erhalten. Wir haben die Vor- und Nachteile der beiden Platinen betrachtet und uns auch einige Alternativen angeschaut. In den nächsten beiden Kapiteln erhalten Sie eine Einführung, wie Sie den Arduino und den Raspberry Pi verwenden und programmieren können.

Wenn Sie bereits Erfahrungen mit dem Arduino und dem Raspberry Pi haben, können Sie auch gleich zu Kapitel 4 vorblättern und einige praktische Projekte damit bauen. Zum Nachschlagen können Sie jederzeit zu Kapitel 2 bzw. 3 zurückblättern.