

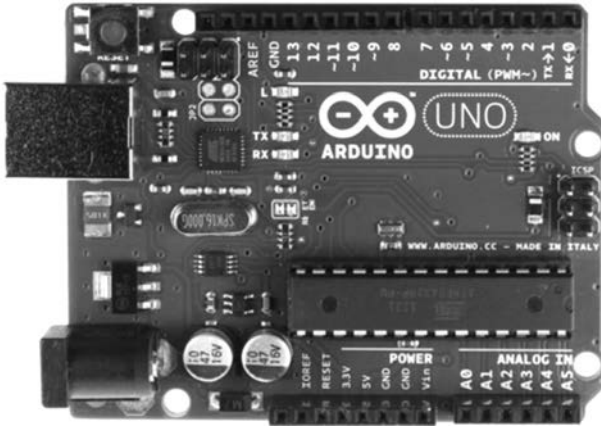
# 1

## Einführung

Haben Sie sich beim Anblick eines technischen Geräts – etwa bei einem ferngesteuerten Bootsmodell, einer Liftsteuerung, einem Verkaufsautomaten oder einem elektronischen Spielzeug – jemals gefragt, wie es funktioniert? Haben Sie jemals den Wunsch verspürt, einen eigenen Roboter, einen elektronischen Signalgeber für eine Modelleisenbahn oder ein System zu bauen, um Wetterdaten aufzuzeichnen und zu analysieren?

Die Arduino-Platine (die Sie in Abb. 1–1 sehen) bietet Ihnen Gelegenheit, einige Geheimnisse der Elektronik auf praktische Weise zu erforschen. Das ursprünglich von Massimo Banzi und David Cuartielles erfundene System Arduino stellt eine günstige Möglichkeit dar, um interaktive Projekte wie ferngesteuerte Roboter, GPS-Ortungssysteme und elektronische Spiele zu konstruieren.

Seit der Einführung im Jahr 2005 ist das Arduino-Projekt exponentiell zu einem blühenden Geschäft gewachsen. Unterstützt wird es durch eine Community von Personen, die den gemeinsamen Wunsch haben, Neues zu erschaffen. Einzelpersonen und Gruppen, von Interessengruppen über Vereine und lokale Hackergemeinschaften bis zu pädagogischen Einrichtungen, zeigen Interesse am Experimentieren mit Arduino.



**Abb. 1-1** Die Arduino-Platine

Um sich eine Vorstellung von der Vielfalt der Arduino-Projekte zu machen, die bisher realisiert wurden, können Sie einfach im Internet danach suchen. Dabei werden Sie verschiedene Gruppen finden, die Einführungskurse und gemeinsame Arbeiten mit gleichgesinnten, kreativen Menschen anbieten.

## Unbegrenzte Möglichkeiten

Schon beim Durchblättern dieses Buches werden Sie feststellen, dass Arduino vielfältige Möglichkeiten bietet – von ganz einfachen Dingen, wie beispielsweise eine Lampe blinken zu lassen, bis zu komplizierten, etwa einer Interaktion mit einem Mobiltelefon.

Sehen Sie sich beispielsweise Philip Lindsays Gerät aus Abbildung 1-2 an. Es kann Textnachrichten von Mobiltelefonen empfangen und sie auf Leuchtbändern darstellen, wie man es in vielen Geschäften sieht. Dieses Gerät enthält eine Arduino-Platine und ein Handy-Shield, um Textnachrichten von einem Mobiltelefon empfangen zu können (ähnlich wie in Projekt 65). Die Nachricht wird dann an ein zwei große, preisgünstige Punktmatrix-Anzeigen geleitet, um sie gut sichtbar darzustellen.



**Abb. 1-2** SMS-Lauftext

Große Anzeigetafeln, die mit einem Arduino verbunden werden können, sind käuflich zu erwerben, weshalb Sie sie nicht selbst basteln müssen. (Mehr dazu erfahren Sie auf <http://www.labradoc.com/i/follower/p/project-sms-text-scroller>.)

Wie wäre es mit einer einzigartigen Form von Heiratsantrag? Tyler Cooper wollte seiner Freundin auf originelle Weise einen Antrag machen, weshalb er ein Gerät konstruierte, das er einen »umgekehrten Geocache« nennt. Es handelt sich dabei um ein kleines Kästchen, das einen Verlobungsring enthält (siehe Abb. 1–3). Wird es in eine bestimmte Gegend getragen (was der interne GPS-Empfänger feststellt), dann entriegelt es sich, um die romantische Nachricht und den Ring freizugeben. Dieses Gerät können Sie mithilfe einer Arduino-Platine, eines GPS-Empfängers und eines LCD-Moduls (wie es in Kapitel 13 verwendet wird) sowie einem kleinen Servomotor leicht nachbauen. Letzterer dient als Riegel, um das Kästchen verschlossen zu halten, bis es sich am richtigen Ort befindet. Der erforderliche Code ist ziemlich einfach und in wenigen Stunden zu schreiben. Am zeitaufwändigsten ist die Auswahl des passenden Kästchens für das System. (Weitere Informationen erhalten Sie unter <http://learn.adafruit.com/reverse-geocache-engagement-box/>.)



**Abb. 1–3** Heiratsantrag via Arduino

Ein weiteres Beispiel: Kurt Schulz wollte den Ladezustand der Batterie in seinem Moped überwachen. Nachdem er aber festgestellt hatte, wie einfach es ist, mit dem Arduino zu arbeiten, entwickelte sich dieses Projekt zu etwas,

was er letztlich »Scooterputer« nannte – ein umfassendes Moped-Überwachungssystem. Der Scooterputer misst nicht nur die Batteriespannung, sondern kann auch Geschwindigkeit, zurückgelegte Strecke, Neigungswinkel, Temperatur, Uhrzeit und Datum, GPS-Position usw. anzeigen. Außerdem enthält er einen Mobiltelefon-Shield, der ferngesteuert werden kann. Damit ist es möglich, das Fahrzeug zu verfolgen, wenn es gestohlen werden sollte, und den Motor ferngesteuert abzuschalten. Das gesamte System wird über einen Touchscreen gesteuert (siehe Abb. 1–4). Jede der Funktionen bildet dabei einen einfachen Baustein, und ein ähnliches Gesamtsystem könnte jedermann innerhalb weniger Wochenenden zusammenstellen. (Mehr darüber auf <http://www.janspace.com/b2evolution/arduino.php/2010/06/26/scooterputer/>.)



**Abb. 1–4** Die Anzeige des Scooterputers (mit freundlicher Genehmigung von Kurt Schulz)

Als Nächstes haben wir John Sarik, der sich gern mit den beliebten Sudoku-Rätseln beschäftigt ... und mit Nixie-Röhren zur Anzeige von Ziffern herumspielt. Diese beiden Antriebsfedern regten ihn dazu an, einen großen 81-stelligen Sudoku-Computer zu bauen. Der Benutzer kann das gesamte 9 x 9-Feld bespielen, wobei der Arduino die Zahlenanzeige steuert und die Einträge auf Gültigkeit prüft. Dies ist zwar schon ein Projekt für Fortgeschrittene, aber durchaus realisierbar, und auch die Elektronik ist nicht sehr

kompliziert. Das Gerät ist ziemlich groß und sieht an der Wand beeindruckend aus, wie Abbildung 1–5 zeigt. (Mehr darüber auf <http://trashbearlabs.wordpress.com/2010/07/09/nixie-sudoku/>.)



**Abb. 1–5** Ein Sudoku aus Nixie-Röhren

Das Team von Oomlout hat mit dem Arduino sogar eine Schreibmaschine gebaut, den TwypeWriter. Dabei ist ein Arduino über einen Ethernet-Shield mit dem Internet verbunden und durchsucht Twitter nach bestimmten Schlüsselwörtern. Wird ein Begriff gefunden, so wird der entsprechende Tweet an eine elektrische Schreibmaschine gesendet, um ihn auszudrucken. Der Arduino ist mit den Tastaturschaltkreisen der Schreibmaschine verbunden, sodass er einen Menschen simulieren kann, der die Tasten betätigt. Das Gerät sehen Sie in Abbildung 1–6. (Mehr darüber auf <http://oomlout.co.uk/blog/twitter-monitoring-typewriter-twypewriter/>.)



**Abb. 1–6** Der TwypeWriter

Dies waren nur einige willkürlich herausgegriffene Beispiele dessen, was mit einem Arduino möglich ist. Eigene Projekte lassen sich ohne große Schwierigkeiten bauen, und nachdem Sie dieses Buch durchgearbeitet haben, ist die Umsetzung für Sie nicht mehr außer Reichweite.

## Die Masse macht's

Die Arduino-Plattform gewinnt von Tag zu Tag mehr Freunde. Wenn Sie besser in Gesellschaft, also in Kursen lernen, dann suchen Sie im Web nach *Cult of Arduino*, um zu erfahren, was andere Personen mit dem Arduino anstellen, und um Arduino-Gruppen kennenzulernen. Die Mitglieder solcher Gruppen werden Sie in die Welt des Arduino als einer Kunstform einführen. Viele arbeiten auch gleichzeitig daran, kleine Arduino-kompatible Platinen zu erfinden. Das Zusammensein mit diesen Gruppen kann viel Spaß machen. Sie lernen dabei interessante Leute kennen und können Ihre Arduino-Kenntnisse weitervermitteln.

## Teile und Zubehör

Wie andere elektronische Geräte ist auch der Arduino bei vielen Einzelhändlern erhältlich, die zudem viele verschiedene andere Produkte und Zubehörteile anbieten. Achten Sie aber darauf, dass Sie einen Original-Arduino kaufen und keine Imitation, da Sie sonst ein fehlerhaftes und wenig leistungsfähiges Produkt erwerben können. Eine minderwertige Platine kann dazu führen, dass Ihr Projekt unter dem Strich mehr kostet. Eine Liste von Arduino-Anbietern finden Sie auf <http://arduino.cc/en/Main/Buy/>.

In der folgenden Liste finden Sie eine aktuelle Auswahl der Händler, die ich für den Kauf von Bauteilen und Zubehör rund um den Arduino empfehle (in alphabetischer Reihenfolge):<sup>1</sup>

- Adafruit Industries (<http://www.adafruit.com/>)
- DigiKey (<http://www.digikey.com/>)
- Jameco Electronics (<http://www.jameco.com/>)
- Little Bird Electronics (<http://www.littlebirdelectronics.com/>)
- Newark (<http://www.newark.com/>)
- nicegear (<http://www.nicegear.co.nz/>)

---

1 Alle URLs in diesem Buch und passende lokale Anbieter und Websites finden Sie unter [www.dpunkt.de/arduino\\_ws](http://www.dpunkt.de/arduino_ws)

- Oomlout (<http://www.oomlout.co.uk/>)
- RadioShack (<http://www.radioshack.com/>)
- RS Components (<http://www.rs-components.com/>)
- SparkFun Electronics (<http://www.sparkfun.com/>)

Wie Sie in diesem Buch noch sehen werden, verwende ich verschiedene Arduino-kompatible Produkte von Freetronics (<http://www.freetronics.com/>). Alle erforderlichen Teile sind jedoch handelsüblich und lassen sich von vielen verschiedenen Einzelhändlern beziehen.

Aber gehen Sie noch nicht los, um einzukaufen! Nehmen Sie sich zunächst die Zeit, die ersten Kapitel zu lesen, um einen Eindruck davon zu bekommen, was sie brauchen, damit Sie kein Geld auf unnötige Dinge verschwenden.

## Erforderliche Software

Ihren Arduino können Sie von praktisch jedem handelsüblichen Computer aus programmieren. An Software benötigen Sie dazu eine *integrierte Entwicklungsumgebung* (Integrated Development Environment, IDE). Sie kann auf folgenden Betriebssystemen ausgeführt werden:

- Mac OS X und höher
- Windows XP 32 oder 64 Bit oder höher
- Linux 32 oder 64 Bit (Ubuntu o. Ä.)

Um die richtige IDE herunterzuladen, blättern Sie zu der Überschrift für das von Ihnen verwendete Betriebssystem vor und folgen den Anweisungen. Sie brauchen auch ein passendes USB-Kabel für den Arduino. Wenn Sie noch nicht über einen Arduino verfügen, können Sie die IDE trotzdem schon herunterladen und ein bisschen damit herumspielen. Die Versionsnummer kann sich rasch ändern, weshalb die in diesem Buch verwendete Version möglicherweise nicht mehr die aktuellste ist. Die Anleitungen aber sollten nach wie vor gelten.

### HINWEIS

*Leider gab es bei Drucklegung dieses Buches Probleme mit der Installation unter Windows 8. Wenn Sie dieses Betriebssystem verwenden, schauen Sie im Arduino-Forum unter <http://arduino.cc/forum/index.php/topic,94651.15.html> nach.*

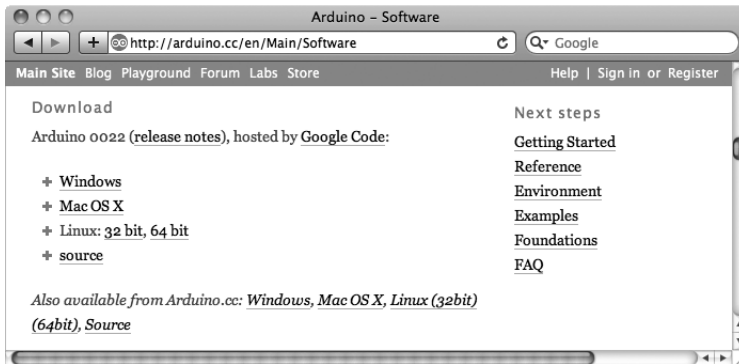
## Mac OS X

In diesem Abschnitt finden Sie Anleitungen zum Herunterladen und Einrichten der Arduino-IDE auf Mac OS X.

### Die IDE installieren

Um die IDE auf dem Mac zu installieren, gehen Sie wie folgt vor:

1. Öffnen Sie in einem Webbrowser wie Safari die Downloadseite für die Software unter <http://arduino.cc/en/Main/Software/> (siehe Abb. 1–7).



**Abb. 1–7** Die Downloadseite für die IDE in Safari

2. Klicken Sie auf den Link *Mac OS X*. Die Datei wird heruntergeladen und erscheint dann im Fenster *Downloads*, das Sie in Abbildung 1–8 sehen.

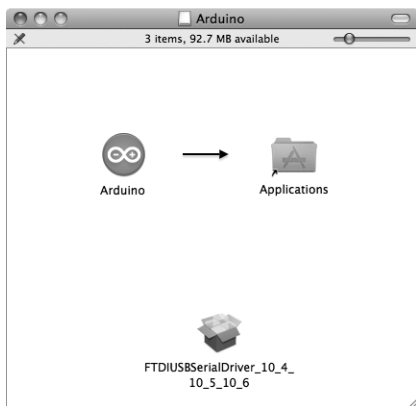


**Abb. 1–8** Die Datei ist komplett heruntergeladen.

3. Wenn der Download abgeschlossen ist, doppelklicken Sie auf die Datei, um die Installation auszulösen. Angezeigt wird das Fenster aus Abbildung 1–9.

### HINWEIS

*Das Paket, das sich hinter dem dritten Symbol verbirgt, müssen Sie nur dann installieren, wenn Sie einen Arduino haben, der älter als das Uno-Modell ist.*



**Abb. 1–9** Der neue Ordner mit der Arduino-IDE

4. Ziehen Sie das Arduino-Symbol über den Ordner *Programme* und lassen Sie die Maustaste los. Während die Datei kopiert wird, ist ein Statusfenster zu sehen.
5. Schließen Sie den Arduino über das USB-Kabel an den Mac an. Nach kurzer Zeit erscheint das Dialogfeld aus Abbildung 1–10.



**Abb. 1–10** Eine neue Arduino-Platine wurde erkannt. In dem Dialogfeld kann anstelle von »Eleven« auch »Uno« angezeigt werden.

6. Klicken Sie auf *Netzwerkeinstellungen* und dann im Feld *Netzwerk* auf *Übernehmen*. Die Statusmeldung »nicht konfiguriert« können Sie ignorieren.

## Die IDE einrichten

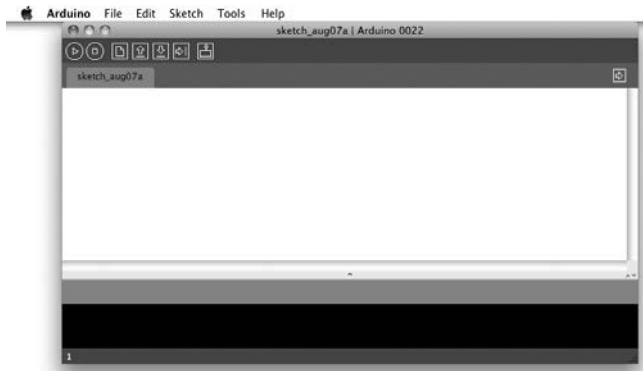
Nachdem Sie die IDE heruntergeladen haben, müssen Sie sie anhand der folgenden Anleitung öffnen und konfigurieren:

1. Öffnen Sie im Finder den Ordner *Programme* (siehe Abb. 1–11) und doppelklicken Sie auf das Arduino-Symbol.



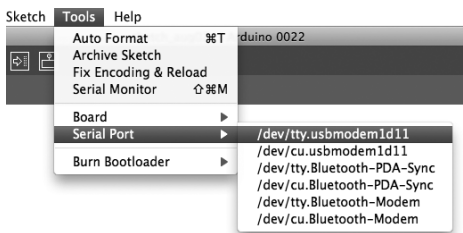
**Abb. 1-11** Der Ordner **Programme**

2. Möglicherweise wird eine Warnmeldung über das Öffnen von Programmen aus dem Web angezeigt. Wenn ja, klicken Sie einfach auf **Öffnen**, um fortzufahren. Anschließend sehen Sie wie in Abbildung 1-12 die IDE.



**Abb. 1-12** Die IDE in Mac OS X

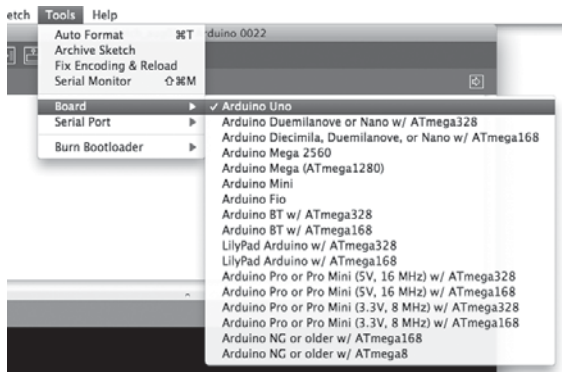
3. Sie sind schon so gut wie fertig. Bevor die Arduino-IDE zur Verfügung steht, müssen Sie nur noch zwei Kleinigkeiten ändern. Als Erstes müssen Sie ihr mitteilen, über welche Art von Anschluss der Arduino mit dem Mac verbunden ist. Wählen Sie **Tools > Serial Port** und dann die Option **/dev/tty.usbmodem1d11** (siehe Abb. 1-13).



**Abb. 1-13** Den USB-Port auswählen

4. Der letzte Schritt besteht darin, der IDE mitzuteilen, was für eine Arduino-Platine Sie angeschlossen haben. Das ist von entscheidender Bedeutung, da es Unterschiede zwischen den Platinen gibt. Wenn Sie beispielsweise

das gebräuchlichste Modell verwenden, den Uno, dann wählen Sie wie in Abbildung 1–14 *Tools > Board > Arduino Uno*. Die Unterschiede zwischen den Platinenmodellen werden in Kapitel 11 ausführlicher erläutert.



**Abb. 1–14** Das Platinenmodell auswählen

Damit sind Hardware und Software jetzt betriebsbereit. Blättern Sie zum Abschnitt »Sicherheit« auf S. 14 vor.

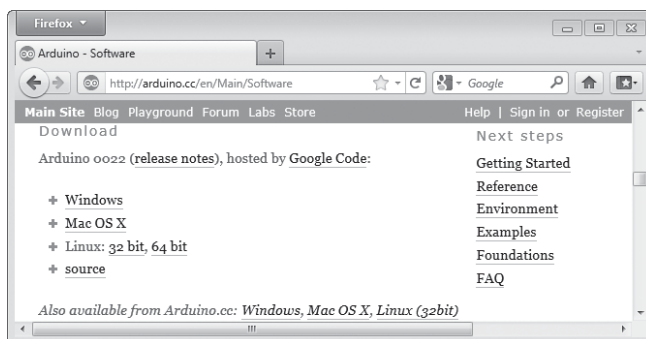
## Windows XP und neuer

In diesem Abschnitt erfahren sie, wie Sie die Arduino-IDE unter Windows herunterladen, wie Sie die Treiber installieren und die IDE einrichten.

### Die IDE installieren

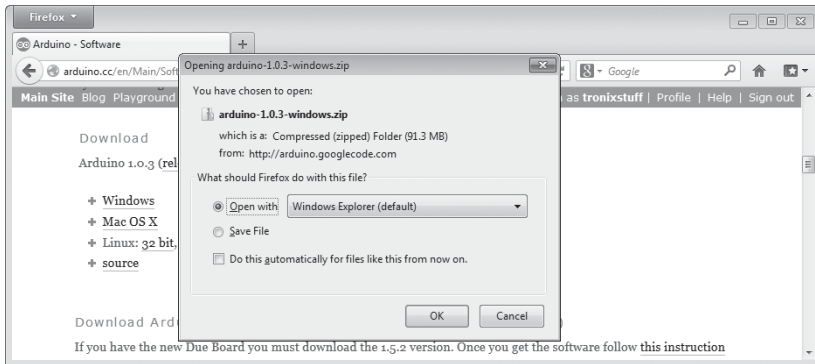
Um die IDE für Windows zu installieren, gehen Sie wie folgt vor:

1. Öffnen Sie in einem Webbrowser wie Firefox die Downloadseite für die Software unter <http://arduino.cc/en/Main/Software/> (siehe Abb. 1–15).

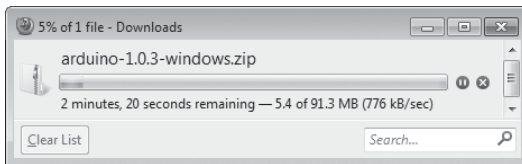


**Abb. 1–15** Die Downloadseite für die IDE in Firefox unter Windows

2. Klicken Sie auf den Link *Windows*. Dadurch erscheint das Dialogfeld aus Abbildung 1–16. Klicken Sie auf *Öffnen mit Windows Explorer* und dann auf OK. Die Datei wird heruntergeladen, wie Sie in Abbildung 1–17 sehen.

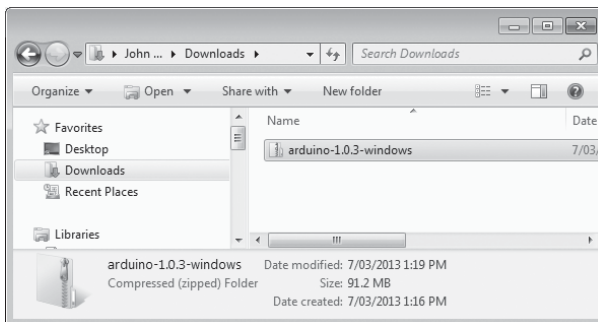


**Abb. 1–16** Die Datei herunterladen



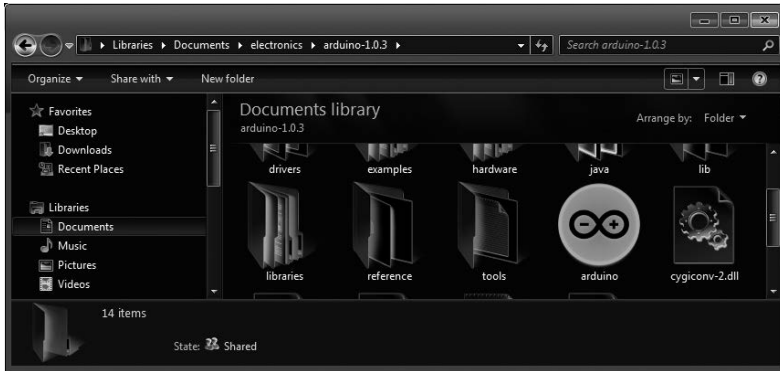
**Abb. 1–17** Firefox zeigt den Fortschritt des Downloads an.

3. Wenn der Download abgeschlossen ist, doppelklicken Sie auf die Datei. Angezeigt wird das Fenster aus Abbildung 1–18.



**Abb. 1–18** Das IDE-Paket

4. Kopieren Sie den Ordner mit dem Namen *arduino-0022* (oder einem ähnlichen Namen) an den Speicherort, an dem Sie Programme ablegen. Öffnen Sie anschließend den Zielordner. Dort sehen Sie jetzt das Programmsymbol *Arduino* wie in Abbildung 1–19. Um in Zukunft leichter darauf zugreifen zu können, sollten Sie eine Verknüpfung auf dem Desktop platzieren.

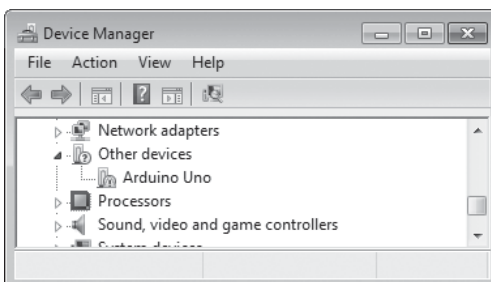


**Abb. 1–19** Das IDE-Paket mit dem Symbol für die Anwendung *Arduino*

## Die Treiber installieren

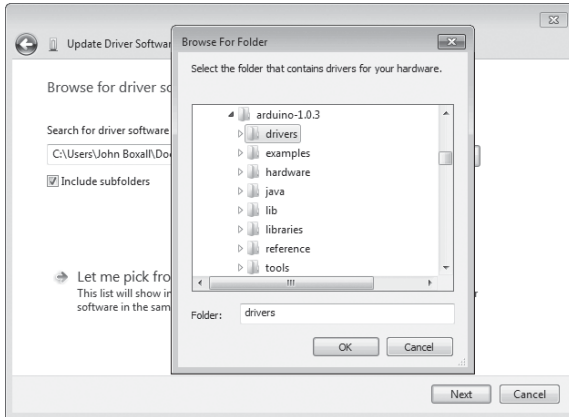
Als Nächstes müssen Sie die Treiber für die USB-Schnittstelle des Arduino installieren.

1. Schließen Sie den Arduino über das USB-Kabel an den PC an. Nach kurzer Zeit wird die Fehlermeldung »Die Gerätetreibersoftware wurde nicht installiert« (oder ähnlich) angezeigt. Schließen Sie einfach das Fenster oder die Sprechblase mit der Meldung.
2. Öffnen Sie die Windows-Systemsteuerung und darin den Geräte-Manager. Scrollen Sie nach unten, bis Sie den Arduino finden (siehe Abb. 1–20).



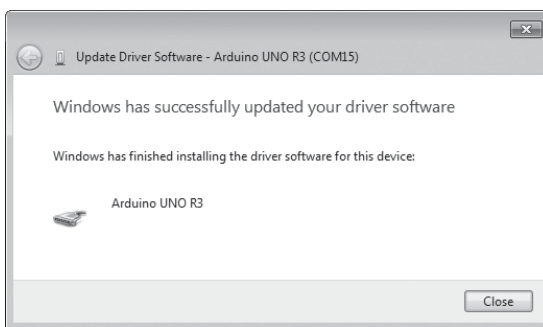
**Abb. 1–20** Der Geräte-Manager

3. Rechtsklicken Sie unter *Andere Geräte* auf *Arduino Uno* und wählen *Treiber aktualisieren*. Aktivieren Sie im nächsten Dialogfeld die Option *Auf dem Computer nach Treibersoftware suchen*. Daraufhin wird das Dialogfeld *Ordner suchen* geöffnet. Klicken Sie auf *Durchsuchen* und wechseln Sie dann zu dem Ordner *driver* in dem Ordner der neu installierten Arduino-Software (siehe Abb. 1–21). Klicken Sie auf OK.



**Abb. 1–21** Den Ordner **drivers** angeben

4. Klicken Sie im nächsten Dialogfeld auf *Weiter*. Möglicherweise zeigt Windows eine Meldung darüber an, dass der Herausgeber der Treibersoftware nicht überprüft werden kann. Klicken Sie auf die Option, die Software trotzdem zu installieren. Nach kurzer Zeit meldet Windows, dass der Treiber installiert ist, und gibt die Nummer des COM-Anschlusses an, mit dem der Arduino verbunden ist (siehe Abb. 1–22).



**Abb. 1–22** Die Treiber wurden erfolgreich aktualisiert.

## Die IDE einrichten

Sie sind fast fertig. Es sind nur noch zwei Kleinigkeiten erforderlich, um die IDE einzurichten.

1. Öffnen Sie die Arduino-IDE und teilen Sie ihr mit, über welche Art von Anschluss der Arduino mit dem PC verbunden ist. Wählen Sie *Tools > Serial Port* und dann die Nummer des COM-Anschlusses, die im Fenster *Treibersoftware aktualisieren* angezeigt wurde.
2. Der letzte Schritt besteht darin, der IDE mitzuteilen, was für eine Arduino-Platine Sie angeschlossen haben. Das ist von entscheidender Bedeutung, da es Unterschiede zwischen den Platinen gibt. Wenn Sie beispielsweise einen Uno haben, dann wählen Sie *Tools > Board > Arduino Uno*. Die Unterschiede zwischen den Platinenmodellen werden in Kapitel 11 ausführlicher erläutert.

Damit ist die Arduino-IDE komplett eingerichtet. Blättern Sie zum Abschnitt »Sicherheit« auf S. 14 vor.

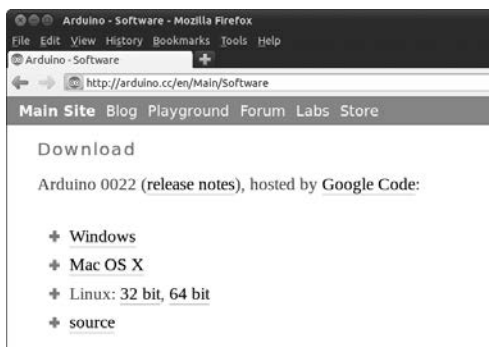
## Ubuntu Linux 9.04 und höher

In diesem Abschnitt erfahren Sie, wie Sie die Arduino-IDE unter Ubuntu Linux herunterladen und einrichten.

### Die IDE installieren

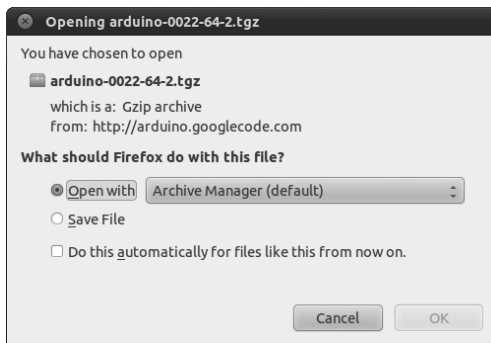
Um die IDE zu installieren, gehen Sie wie folgt vor:

1. Öffnen Sie in einem Webbrowser wie Firefox die Downloadseite für die Software unter <http://arduino.cc/en/Main/Software/> (siehe Abb. 1–23).



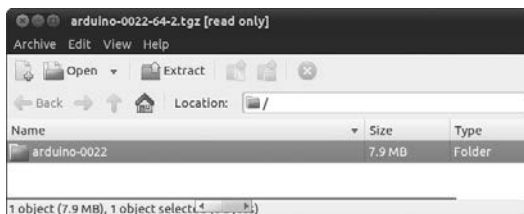
**Abb. 1–23** Die Downloadseite für die IDE in Firefox unter Ubuntu

2. Klicken Sie je nachdem, was für ein System Sie haben, auf den Link *32-bit* oder *64-bit* für Linux. Wenn das Dialogfeld aus Abbildung 1–24 erscheint, wählen Sie *Öffnen mit Archiv-Manager* und klicken dann auf OK.



**Abb. 1–24** Die Datei herunterladen

- Nachdem die Datei heruntergeladen ist, wird sie im Archiv-Manager angezeigt (siehe Abb. 1–25). Kopieren Sie den Ordner mit dem Namen *arduino-0022* (oder einem ähnlichen Namen) in Ihren gewohnten Programm- oder Benutzerordner.

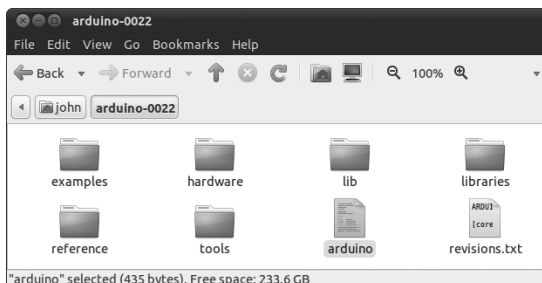


**Abb. 1–25** Das IDE-Paket

## Die IDE einrichten

Als Nächstes müssen Sie die IDE einrichten.

- Schließen Sie den Arduino über das USB-Kabel an den PC an. Öffnen Sie den zuvor kopierten Ordner *arduino-0022* und doppelklicken Sie darin auf die Datei *arduino*, die in Abbildung 1–26 markiert ist.



**Abb. 1–26** Der Ordner der Arduino-IDE mit der Datei **arduino**

2. Wenn das Dialogfeld aus Abbildung 1–27 erscheint, klicken Sie auf *Ausführen*. Dadurch wird die IDE wie in Abbildung 1–28 angezeigt.



**Abb. 1–27** Berechtigungen zum Ausführen der IDE gewähren



**Abb. 1–28** Die IDE unter Linux

3. Teilen Sie der IDE mit, über welche Art von Anschluss der Arduino mit dem Computer verbunden ist. Wählen Sie *Tools > Serial Port* und dann */dev/ttyACMx*, wobei *x* eine einzelne Ziffer ist. (Es sollte nur einen Anschluss mit einem solchen Namen geben.)
4. Geben Sie an, was für einen Arduino Sie angeschlossen haben. Das ist von entscheidender Bedeutung, da es Unterschiede zwischen den Platinen gibt. Wenn Sie beispielsweise einen Uno haben, dann wählen Sie *Tools > Board > Arduino Uno*. Die Unterschiede zwischen den Platinenmodellen werden in Kapitel 11 ausführlicher erläutert.  
Jetzt sind Hardware und Software betriebsbereit.

## Sicherheit

Wie bei jedem anderen Hobby oder Handwerk sind Sie für Ihre eigene Sicherheit und die Ihrer Umgebung verantwortlich. Im weiteren Verlauf dieses Buches beschreibe ich Arbeiten mit einfachen Werkzeugen, batteriebetriebenen elektrischen Geräten, scharfen Messern und manchmal auch Lötkolben. Basteln Sie bei der Arbeit an Ihren Projekten niemals an der

Hausstromversorgung herum, sondern überlassen Sie so etwas einem ausgebildeten Elektriker. Denken Sie daran, dass Kontakt mit dem Netzstrom tödlich sein kann.

## Ausblick

Sie haben einen Weg vor sich, der sehr interessant ist und viel Spaß bereiten kann. Dabei werden Sie Dinge bauen, die Sie niemals für möglich gehalten haben. In diesem Buch finden Sie 65 Arduino-Projekte, von sehr einfachen bis zu ziemlich komplizierten. Alle sind darauf ausgelegt, Ihnen etwas beizubringen und etwas Nützliches zu konstruieren. Fangen wir also an!