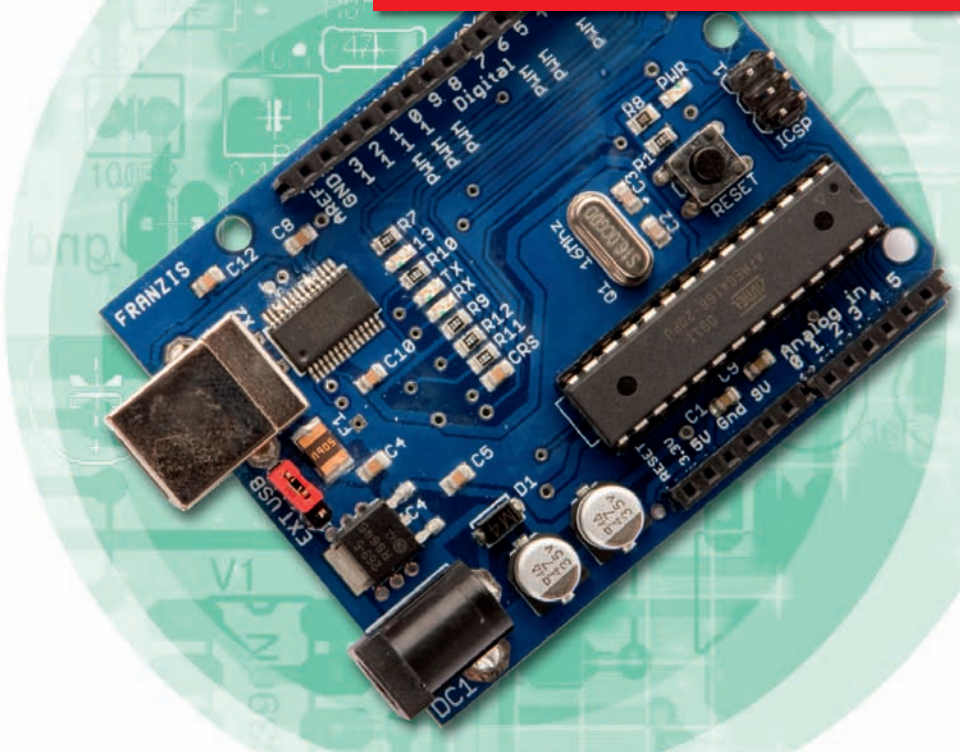
FRANZIS
PC & ELEKTRONIK

Ulli Sommer

- Über 80 praktische Experimente
- Grundlagenkurs zur Programmierung



Arduino

Mikrocontroller-Programmierung mit Arduino/Freeduino

Auf CD-ROM:

- Open-Source-Soft- und Hardware
- Über 80 Quellcodes zu den Experimenten
- Schaltpläne und Datenblätter
- Open-Source-VB.NET-Programme zum Messen und Steuern

Vorwort

Vielen fällt der Einstieg in die Mikrocontroller-Programmierung und die dazugehörige Elektronik schwer. Bei den meisten Mikrocontroller-Systemen muss man sich zuvor durch unzählige und für den Anfänger schwer verständliche Datenblätter wälzen. Die Programmieroberflächen sind meist viel zu kompliziert und mehr für professionelle Programmierer ausgelegt. Somit bleibt manchem der Zugang in die Welt der Mikrocontroller für immer verwehrt.

Arduino ist eine leicht zu verstehende Open-Source-Plattform, basierend auf einem Mikrocontrollerboard und einer Entwicklungsumgebung mit einer API (Programmier-Schnittstelle) für den Mikrocontroller. Für die Interaktion zwischen Mensch und Mikrocontroller können diverse analoge und digitale Sensoren angeschlossen werden, die die Umwelt erfassen und die Daten an den Mikrocontroller weitergeben. Der Mikrocontroller verarbeitet die eingehenden Daten, und durch das Programm entstehen neue Ausgabedaten in analoger oder ditgitaler Form. Hierbei sind der Kreativität des Entwicklers fast keine Grenzen gesetzt.

Die Arduino-Programmieroberfläche unterstützt den Entwickler bei seinen Vorhaben durch ihre vorgefertigten Programme und Funktionsbibliotheken. Das einfache Zusammenspiel aus Hard- und Software bildet die Basis für *Physical Computing*: die Verbindung der realen Welt mit der des Mikrocontrollers, die aus Bits und Bytes besteht. Dieses Buch zeigt Ihnen Schritt für Schritt, wie Sie den leichten Einstieg in diese Welt finden.

Viel Spaß beim Lesen und Experimentieren mit diesem Buch!

Ulli Sommer

CD-ROM zum Buch

Diesem Buch liegt eine CD-ROM bei, die verschiedene Programme, Tools, Datenblätter und Beispiele enthält. Die CD-ROM erleichtert Ihnen das Arbeiten mit diesem Buch. Die hier abgedruckten Beispiele sind auf der CD-ROM enthalten.

Inhalt der CD-ROM

- Arduino-Entwicklungsumgebung (IDE)
- Beispiel-Programmcode zum Lehrgang
- Diverse Tools
- Datenblätter
- Schaltpläne

GPL (General Public License)

Sie können Ihre eigenen Programme mit anderen Anwendern über das Internet austauschen. Die Beispielprogramme stehen unter der Open-Source-Lizenz *GPL* (General Public License). Daher sind Sie berechtigt, die Programme unter den Bedingungen der GPL zu modifizieren, zu veröffentlichen und anderen Anwendern zur Verfügung zu stellen, sofern Sie Ihre Programme dann ebenfalls unter die GPL-Lizenz stellen.

Systemvoraussetzung

Ab Pentium III-PC, Windows 98SE/ME/XP/Vista/Windows 7, Linux, Mac OS, CD-ROM-Laufwerk, Java

Updates und Support

Arduino wird ständig weiterentwickelt. Updates können kostenlos von der Website *www.arduino.cc* heruntergeladen werden (es fallen nur Ihre üblichen Online-Kosten an).

Vorbereitungen

Die vorgestellten Experimente können mit wenigen, meist preiswerten Teilen – aus der Bastelkiste oder extra gekauft – durchgeführt werden. Im Anhang finden Sie eine Liste der Teile und Liefernachweise für den Bezug der Komponenten.

Für die Experimente und Versuche brauchen Sie weder Batterien noch eine zusätzliche Stromversorgung.

Als sinnvolle und hilfreiche Ergänzung kann ein Vielfachmessinstrument (Multimeter) und/oder eine Schnittstelle zum Computer zur Strom- und Spannungsmessung verwendet werden. Damit können zusätzliche Experimente durchgeführt werden und es sind weitere spannende Zusammenhänge erfahrbar. Außerdem ist es nützlich, eine handelsübliche Akkuzelle der Größe AA (Mignon) oder AAA (Micro) für einige Experimente der Ladetechnik zur Verfügung zu haben.

Das Buch vermittelt die wichtigsten Grundlagen der Mikrocontrollertechnik. Außerdem werden beispielhafte praktische Anwendungen vorgestellt, mit deren Hilfe es möglich wird, eigene Schaltungen und Erfindungen rund um die Mikrocontrollertechnik zu entwickeln.

Sie können Ihr Equipment auch um eine Sortimentsbox ergänzen. Darin werden alle Einzelteile griffbereit und übersichtlich aufbewahrt.

Inhaltsverzeichnis

2 Programmierung der Mikrocontroller

Mit der zunehmenden Integration von Halbleiterbauteilen wie Mikroprozessoren hielten Mikrocontroller immer stärker Einzug in die Anwendungsgebiete der Mess-, Steuer- und Regelungstechnik. Aber auch im Hobbybereich wurden die Mikrocontroller immer beliebter. Das liegt zum einen daran, dass heute komplexe, meist analoge Schaltungen durch einfachere digitale Mikrocontroller-Schaltungen ersetzt werden. Ein anderer ausschlaggebender Punkt ist das unschlagbare Preis-Leistungs-Verhältnis von Mikrocontrollern.

2.1 Was ist ein Programm?

Ein Programm ist die Beschreibung eines Informationsverarbeitungsprozesses. Im Lauf eines solchen Prozesses wird aus einer Menge von variablen oder konstanten Eingangswerten eine Menge von Ausgangswerten berechnet. Die Ausgangswerte sind entweder selbst Ziel der Informationsgewinnung oder dienen mittelbar zur Reaktion auf die Eingangswerte. Neben den eigentlichen Berechnungen kann ein Programm Anweisungen zum Zugriff auf die Hardware des Computers oder zur Steuerung des Programmflusses enthalten. Ein Programm besteht aus mehreren Zeilen sogenannten Quelltextes. Dabei enthält jede Zeile eine oder mehrere Rechen- oder Steueranweisungen. Neben diesen Anweisungen selbst bestimmt ihre Reihenfolge wesentlich die eingangs beschriebene Informationsverarbeitung. Die Ausführung der den Anweisungen entsprechenden Operationen durch den Steuercomputer erfolgt sequenziell, also nacheinander. Eine Folge von Programmanweisungen mit einem bestimmten Ziel nennt man auch *Algorithmus*.

2.2 Programmierung in C

C oder auch *ANSI-C* ist eine einfach zu erlernende Programmiersprache. C ist eine imperative Programmiersprache, die der Informatiker Dennis Ritchie in den frühen 70er-Jahren an den Bell Laboratories für das Betriebssystem Unix entwickelte. Seitdem ist sie weltweit stark verbreitet. Die Anwendungsbereiche von C sind sehr verschieden. Es wird z. B. zur System- und Anwendungsprogrammierung eingesetzt. Die grundlegenden Programme aller Unix-Systeme und die System-

kerne vieler Betriebssysteme sind in C programmiert. Zahlreiche Sprachen wie C++, Objective-C, C#, Java, PHP oder Perl orientieren sich an der Syntax und anderen Eigenschaften von C. Es ist also mehr als lohnenswert, sich mit dieser Programmiersprache zu beschäftigen, da man später auch leicht auf andere Mikrocontrollersysteme umsteigen kann. Für fast alle Mikrocontroller existiert ein freier C-Compiler, den die Hersteller zum Download anbieten. Das C von Arduino ist jedoch um einiges einfacher gehalten als die professionellen C-Compiler und nimmt sehr viel Arbeit ab. Vor allem um die komplizierten Hardware-Routinen muss man sich bei Arduino nicht kümmern, da sie bereits als feste Befehle in der Entwicklungsumgebung integriert sind.

3 Eine kleine Übersicht über die ARDUINO-Mikrocontroller-Familie

Die Arduino-Hardware verwendet ausschließlich gängige, allgemein verfügbare Bauteile. Daher ist es leicht, die Funktionsweise zu verstehen und die Schaltung an eigene Wünsche anzupassen oder Erweiterungen vorzunehmen. Den Kern bildet ein ATmega-Controller aus Atmels weitverbreiteter 8-Bit-AVR-Familie. Hinzu kommen Schaltungsteile zur Stromversorgung und eine serielle Schnittstelle. Letztere ist bei den jüngeren Arduino-Versionen als USB-Interface ausgelegt. Über diesen Anschluss erfolgt der Download der Anwenderprogramme und bei Bedarf auch die Kommunikation zwischen PC und Arduino während der Programmausführung.

Weil Arduino-Boards so einfach und universell ausgelegt sind, werden sie häufig auch schlicht als *I/O-Board* bezeichnet. Arduino stellt dem Anwender 14 digitale Ein- oder Ausgänge zur Verfügung, davon sind sechs als Analogausgang (8 Bit PWM) zu verwenden. Weitere sechs Eingänge können analoge Signale erfassen (10 Bit ADC). Bei Bedarf stehen SPI und I²C als weitere Schnittstellen zur (seriellen) Kommunikation zur Verfügung.

Es gibt Arduino-Boards in mehreren Varianten. Die Originale stammen vom Hersteller Smart Projects aus Italien. Es gibt mittlerweile auch zahllose Klone und Nachbauten von anderen Anbietern, schließlich handelt es sich um *Open Hardware*. Ein wichtiger Unterstützer des Arduino-Projekts ist Sparkfun aus Boulder, Colorado. Die Kooperation mit dem US-Partner hat eine Reihe optimierter Arduino-Boards hervorgebracht, die den Zusatz »Pro« im Namen führen. Außerdem ist mit LilyPad ein wichtiger Ableger entstanden, der das Thema *Wearable Computing* aufgreift.

Die meisten Anwender setzen auf das von Smart Projects gefertigte, handtellergroße Arduino Duemilanove (Duemilanove = 2009), das den ATmega-Controller in DIP-Bauform auf einem Sockel trägt. Es unterscheidet sich nur unwesentlich vom überaus erfolgreichen Vorgänger Arduino Diecimilanove, dessen Namensgebung auf die ersten 10.000 verkauften Boards zurückgeht. Auf den Boards ist ein FTDI-Chip aufgelötet, der die USB-Schnittstelle bereitstellt.

Das neue Arduino Mega Board verwendet einen leistungsstärkeren Mikrocontroller (Atmega1280) und bietet mehr Speicher, I/O-Pins und Funktionen auf einer deutlich erweiterten Platinenfläche.

Wesentlich kleiner ist Arduino Mini, ein Board im DIP24-Format. Das ganze Modul lässt sich auf einen 24-poligen DIL-Sockel stecken. Die Version Arduino Pro Mini von Sparkfun ist nahezu identisch, wird aber ohne »Beinchen« (seitliche Stifte) geliefert. Diese Module benötigen zum Programmieren einen USB-Adapter, der an der Schmalseite der Module angesteckt werden kann.

Das LilyPad-Board von Leah Buechley (in Zusammenarbeit mit Sparkfun) ist auch Arduino-kompatibel und verfolgt einen ganz eigenen Zweck. LilyPad und Zubehör sind dafür ausgelegt, in Kleidung eingenäht zu werden, um dort eine möglichst enge Symbiose von Technik und Künstler zu realisieren. Die charakteristische runde Form des LilyPad-Arduinos erregt ebenso Aufmerksamkeit wie die Farbgebung und die kreisförmige Anordnung der Kontakte. Zum Einsatz kommt hier die Low-Power-Version (3,3 V) des ATmega168. Zahlreiche kleine Peripherieplatinen (Sensoren, LEDs, Taster ...) ergänzen LilyPad zu einem ganzen System unter dem Motto »Elektronik mit der Nähmaschine« .

Über weitere Board-Versionen und Zubehörteile informieren Sie die Arduino-Projektseite (siehe Links) und die Produktseiten von SparkFun Electronics.

3.1 Arduino Mega

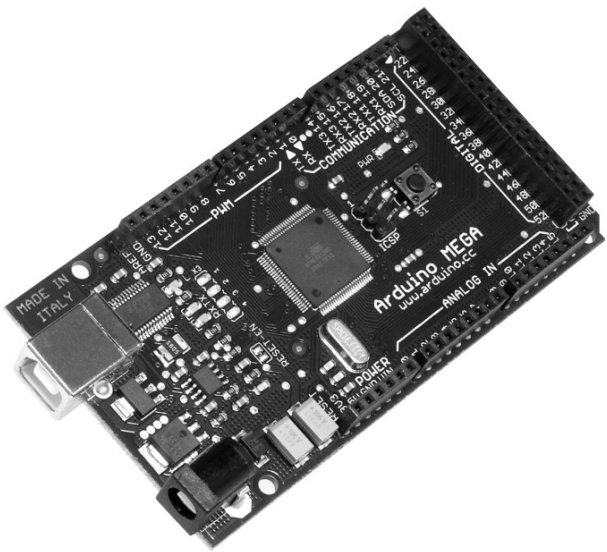


Bild 3.1: Arduino Mega (Quelle: Fa. Elmicro)

Technische Daten:

- ATmega1280 Mikrocontroller
- 128 KB Flash
- 8 KB RAM, 4 KB EEPROM
- 16-MHz-Takt
- 54 digitale I/O-Pins, davon 14 als PWM nutzbar
- 4 Hardware-UARTs
- I²C-Interface, SPI
- 16 analoge Eingänge (10 Bit)
- USB-Interface, Spannungsversorgung, Bootloader etc. wie beim Arduino Duemilanove
- Abmessungen ca. 101 mm x 53 mm x 12 mm

3.2 Arduino Duemilanove

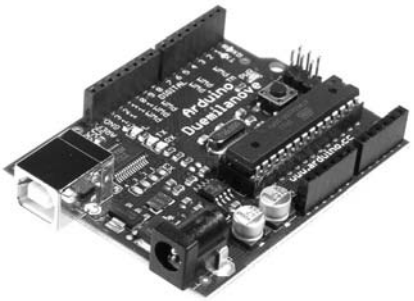


Bild 3.2: Arduino Duemilanove (Quelle: Elmicro)

Technische Daten:

- ATmega328 Mikrocontroller
- 32 KB Flash (davon 2KB für Bootloader)
- 2 KB RAM, 1 KB EEPROM
- 16-MHz-Takt
- 14 digitale I/O-Pins, davon 6 als PWM nutzbar
- sechs analoge Eingänge (10 Bit)
- On-Board-USB-Schnittstelle mit FT232RL von FTDI

- 5 V Betriebsspannung, Speisung über USB oder über Spannungsregler (7 V bis 12 V Eingangsspannung)
- Abmessungen ca. 69 mm x 53 mm x 12 mm
- Bootloader im Lieferzustand bereits installiert, Download ohne Programmieradapter möglich

3.3 Arduino Mini

Bild 3.3: Arduino Mini (Quelle: Elmicro)

Technische Daten:

- ATmega168 Mikrocontroller mit 16-MHz-Quarztakt
- Programmierung über USB-Adapter (ARDUINO/USB, USB-Adapter mit FTDI-Chip)
- 512 Byte EEPROM
- 1 KB SRAM
- 16 KB FLASH (2 KB benötigt der Bootloader für sich)
- Betriebsspannung 5 V
- 14 Digitale I/Os, sechs davon können zur PWM-Erzeugung genutzt werden
- acht analoge 10-Bit-Eingänge
- Versorgungsspannung 7 V bis 9 V

3.4 Arduino Nano

Bild 3.4: Arduino Nano (Quelle: Elmicro)

Technische Daten:

- ATmega328 oder ältere Version 168 mit 16-MHz-Quarztakt
- Programmierung über USB-»On Board Chip«
- Autoreset-Funktion
- 5-V-Technik
- 14 Digitale I/Os, sechs davon können zur PWM-Erzeugung genutzt werden
- acht analoge 10-Bit-Eingänge
- 32 KB oder 16 KB FLASH
- 1 KB SRAM
- 512 oder 1 KByte EEPROM
- Ausgangsstrom pro I/O max. 40 mA
- Versorgungsspannung 6 V bis 20 V
- Abmessungen: 18 mm x 43 mm

3.5 Arduino Pro Mini

Bild 3.5: Arduino Pro Mini (Quelle: Elmicro)

Technische Daten:

- ATmega328 mit 16-MHz-Quarztakt (Genauigkeit 0,5 %)
- Programmierung über USB-Adapter (ARDUINO/USB)
- Autoreset-Funktion
- Diese Version gibt es in 5-V- und 3,3-V-Technik
- Ausgangsstrom max. 150 mA
- Überlastschutz
- Verpolungsschutz
- Versorgungsspannung 5 V bis 12 V
- Power und Status LED bereits »On Board«
- Abmessungen: 18 mm x 33 mm
- Gewicht weniger als 2 g

3.6 Arduino Pro

Bild 3.6: Arduino Pro (Quelle: Elmicro)

Technische Daten:

- ATmega328 und ältere ATmega168 mit 16-MHz-Quarztakt
- Programmierung über USB-Adapter (ARDUINO/USB)
- Diese Version gibt es in 5-V- und 3,3-V-Technik
- 14 Digital-I/O-Pins (sechs davon als PWM nutzbar)
- sechs analoge 10-Bit-Eingänge
- Versorgungsspannung 3,35 V bis 12 V (3,3-V-Version)
- Versorgungsspannung 5 V bis 12 V (5-V-Version)
- Ausgangsstrom pro Digitalport 40 mA
- 32 KB oder 16 KB (ATmega168) FLASH
- 1 KB (ATmega168) oder 2 KB (ATmega328) SRAM
- 512- (ATmega168) oder 1-KB-EEPROM

3.7 LilyPad

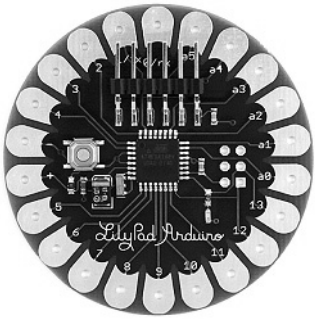


Bild 3.7: LilyPad Arduino (Quelle: Elmicro)

Technische Daten:

- ATmega328V und ältere ATmega168V mit 16-MHz-Quarztakt
- Programmierung über USB Adapter (ARDUINO/USB)
- Spannungsversorgung 2,7 V bis 5,5 V
- 14 Digital-I/O-Pins (sechs davon als PWM nutzbar)
- sechs analoge 10-Bit-Eingänge
- Ausgangsstrom pro Digitalport 40 mA
- 32 KB oder 16 KB (ATmega168) FLASH
- 1 KB (ATmega168) oder 2 KB (ATmega328) SRAM
- 512- (ATmega168) oder 1-KB-EEPROM

3.8 USB-Adapter

Bild 3.8: USB-Adapter mit FTDI-Chip (Quelle: Elmicro)

Diesen Programmieradapter gibt es in 3,3-V- und in 5-V-Ausführung.

Der Adapter wird zum Programmieren der Arduino-Borads ohne USB-Anschluss benötigt. Die Pinbelegung entspricht den Original-Arduino-Spezifikationen. Er kann auch zur Kommunikation (virtuelle serielle Schnittstelle) verwendet werden. Dieses Feature muss man für eigene Entwicklungen einfach haben. Es ermöglicht, einen Sketch auf das Board zu laden, ohne die Reset-Taste zu drücken.

Stichwortverzeichnis

FRANZIS
PC & ELEKTRONIK

Ulli Sommer

Arduino

Arduino ist ein Mikrocontroller-System, das aus einem Mikrocontroller der Firma Atmel und einer Open-Source-Entwicklungsumgebung, die auf einem vereinfachten C-Dialekt basiert, besteht.

Der Mikrocontroller wird über den PC programmiert und kann eigenständig oder in Verbindung mit dem PC agieren. Es können für die Interaktion zwischen Mensch und Mikrocontroller diverse Sensoren angeschlossen werden, die unsere Umwelt erfassen und die Daten an den Mikrocontroller weitergeben. Der Mikrocontroller verarbeitet mit seinem Programm die Daten, und es können Ausgaben getätigt oder z. B. Aktuatoren gesteuert werden. Der Kreativität des Entwicklers sind dabei keine Grenzen gesetzt.

Die Arduino-Programmieroberfläche unterstützt den Entwickler bei seinen Vorhaben durch ihre vorgefertigten Programme und Funktionsbibliotheken. Das einfache Zusammenspiel aus Hard- und Software bildet die Basis für Physical Computing, also die Verbindung der realen Welt mit der Welt des Mikrocontrollers, die aus Bits und Bytes besteht. Dieses Buch bietet Ihnen einen unkomplizierten Einstieg in diese Welten.

Die ersten Kapitel vermitteln Ihnen die Programmierung mit Arduino. Die C-Befehle werden anhand kleiner Beispiele verdeutlicht, Hard- und Software werden detailliert erklärt. Schließlich setzen Sie Ihre neu erworbenen Kenntnisse in Experimenten kreativ und spielerisch in Mess-, Steuer- und Regelanwendungen ein. Nach der Lektüre dieses Buchs werden Sie in der Lage sein, Ihre eigenen Ideen selbstständig umzusetzen.

Aus dem Inhalt:

- Mikrocontroller-Grundlagen
- Mikrocontroller-Programmierung mit Arduino/Freeduino
- Aufbauanleitung zu jedem Experiment
- Von den Grundlagen bis zur eigenen Applikation
- Entwickeln Sie Ihre eigenen Anwendungen und damit praktisch Ihr eigenes Spezial-IC: sei es eine spezielle Alarmanlage, ein Messgerät oder eine Robotersteuerung
- **Über 80 praktische Experimente:**

  Den USB-Brückenchip FT232RL einrichten und anwenden, Ein-/Ausschaltverzögerung, Temperaturschalter, Kapazitätsmessgerät, Schuluhr mit RTC, 6-Kanal-Digitalvoltmeter, Lüftersteuerung, Datenaustausch zwischen VB.NET und Arduino, Sensortaster, Statemaschine, Daten aufzeichnen mit Stamp PLOT, Digitales Speicheroszilloskop, Bewegungsmelder-Alarmanlage, Dämmerungsschalter, romantisches Kerzenlicht, Musikplayer, Datenplotter mit VB.NET, serielle Ein- und Ausgabe, Experimente mit LCD-Displays und vieles mehr

CD-Inhalt:

- Open-Source-Soft- und Hardware
- Über 80 Quellcodes zu den Experimenten
- Schaltpläne und Datenblätter
- Open-Source-VB.NET-Programme zum Messen und Steuern mit Arduino/Freeduino

ISBN 978-3-645-65034-2

Euro **29,95** [D]