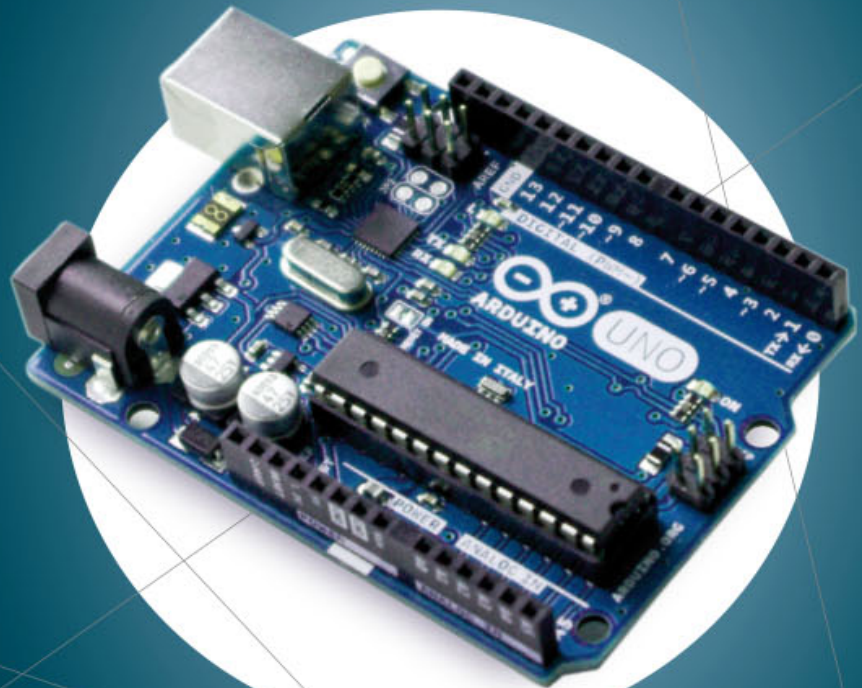


Erik Bartmann

Der
Arduino-
Bestseller in
4. Auflage

Mit
Arduino
die elektronische Welt
entdecken



48 Bastelprojekte
mit dem Arduino


bombini
verlag

Inhalt

Impressum

Grußwort zur vierten Auflage von Teo Swee Ann, CEO von Espressif

Einleitung

Arduino für alle

Der Layoutfehler, der Geschichte machte

Von den Segnungen des Copy&Paste – und seinen Beschränkungen

Wie ich mein Buch aufgebaut habe

Wie ich die Bastelprojekte gestaltet habe

Meine Website mit weiteren Arduino- und Elektronikthemen

Voraussetzungen

Benötigte Bauteile

Verhaltensregeln

Kapitel 1: Arduino: Die Hardware

Das Arduino-Uno-Board

Physikalische Größen

Kapitel 2: Arduino: Die Software

Arduino-IDE oder Arduino Create?

Was befindet sich wo?

Wir testen die Kommunikation zwischen Computer und Arduino

Die Bibliothekenverwaltung

Die Boardverwaltung

Der Sketch-Code in der Entwicklungsumgebung

Troubleshooting

Wie deine Idee in den Mikrocontroller kommt

Kapitel 3: Keine Angst vorm Programmieren - Coding Basics

Was ist ein Programm beziehungsweise ein Sketch?

Was sind Kontrollstrukturen?

Kapitel 4: Das Arduino-Discoveryboard

Das Arduino-Discoveryboard

Bastelprojekt 1: Hallo Welt - das Blinken einer LED

»Hallo Welt« wird geblinkt

Workshop zur blinkenden LED

Bastelprojekt 2: Arduino-Low-Level-Programmierung

Die Zugänge des Mikrocontrollers

Die Programmierung eines Ports

Register und C++-Befehle

Der Pullup-Widerstand

Bastelprojekt 3: Einen Taster sicher abfragen

Die Manipulation interner Pullup-Widerstände

Bastelprojekt 4: Blinken mit Intervallsteuerung

Drücke den Taster – und er reagiert

Bastelprojekt 5: Der störrische Taster

Ich wurde geprellt!

Bastelprojekt 6: Ein Lauflicht

Immer der Reihe nach

Register direkt beeinflussen

Die Bit-Manipulation

Der Lauflicht-Workshop

Bastelprojekt 7: Die Port-Erweiterung

Eine digitale Port-Erweiterung

Ein konventionelles Schieberegister

Ein einfacher Binärzähler

Bastelprojekt 8: Die Port-Erweiterung 2.0

Digitale Porterweiterung 2.0

Der Bit-Manipulations-Workshop

Platinenbau-Workshop

Bastelprojekt 9: Die Erstellung einer Arduino-Bibliothek

Bibliotheken verstehen und nutzen

Die Schieberegister-Library

Wo gibt es viele interessante Libraries?

Bastelprojekt 10: Eine Ampelschaltung

Die Ampelphasen

Eine interaktive Ampelschaltung

Workshop zur Ampelschaltung

Bastelprojekt 11: Der elektronische Würfel

Wie wird ein Würfel simuliert?

Der elektronische Würfel auf Platine – ein Blick in die Zukunft

Workshop zum elektronischen Würfel

Bastelprojekt 12: Der LED-Ring

Acht LEDs in einem Ring

Roulette-Workshop

Bastelprojekt 13: Der Lichtsensor

Ein veränderlicher Widerstand

Wir werden kommunikativ mit Processing

Workshop zum Lichtsensor

Bastelprojekt 14: Der Richtungsdetektor

Vom Arduino zu Processing

Grafikausgabe mit Processing

Workshop zum Richtungsdetektor

Bastelprojekt 15: Die Ansteuerung eines Servos

Was ist ein Servo?

Potentiometer – ein veränderlicher Widerstand

Bastelprojekt 16: Das Tischsonar

Der Ultraschallsensor

Der Ultraschallsensor mit dem Arduino

Der Datenempfang und die Visualisierung mit Processing

Workshop zum Tischsonar

Bastelprojekt 17: Die Siebensegmentanzeige

Die Siebensegmentanzeige genau erklärt

Workshop zur Siebensegmentanzeige

Bastelprojekt 18: Die Siebensegmentanzeige - Teil 2: Mir gehen die Pins aus

Das Problem mit mehreren Ziffern

Zwei Schieberegister

Das Multiplexing

Bastelprojekt 19: Die Temperatur messen mit Thermistoren

Heiß oder kalt oder was?

Bastelprojekt 20: Der Reaktionstester

Wie misst man Reaktionsfähigkeit?

Die Tab-Registerkarte in der Arduino IDE

Workshop zum Reaktionstester

Bastelprojekt 21: Ein Keypad am Arduino

Wie funktioniert ein Keypad?

Bastelprojekt 22: Ein Keypad als Arduino-Shield

Noch ein Keypad?

Ein kleines Zahlenratespiel

Workshop zum Keypad

Bastelprojekt 23: Das LC-Display

Eine alphanumerische Anzeige

Das Zahlenratespiel reloaded

Eigene Zeichen definieren

Ein LC-Display mit mehr Zeilen

Workshop zum LC-Display

Bastelprojekt 24: Die I²C-Kommunikation

Was bedeutet I²C?

Wir programmieren einen EEPROM-Monitor

Workshop zum EEPROM

Bastelprojekt 25: Port-Erweiterung über die I²C-Schnittstelle

Der Port-Expander MCP23017

Bastelprojekt: Beim Port-Expander MCP23017 die Ausgänge ansteuern

Bastelprojekt: Beim Port-Expander MCP23017 die Eingänge abfragen

Bastelprojekt 26: Schritt für Schritt zum Schrittmotor

Noch mehr Bewegung

Ein eigenes Motor-Shield basteln

Einen eigenen Sketch-Code programmieren

Bastelprojekt 27: Der ArduBot und seine Motorsteuerung

Wir werden mobil

Bastelprojekt 28: Der autonome ArduBot

Wir werden autonom

Bastelprojekt 29: Eine Lüftersteuerung

Einen Ventilator sinnvoll steuern

Workshop zum Lüfter

Bastelprojekt 30: Sound und mehr

Hast du Töne

Das Farben-Sequenz-Spiel

Workshop zum Farben-Sequenz-Spiel

Bastelprojekt 31: Data Monitoring

Datenerfassung und Visualisierung

Workshop zur Visualisierung

Bastelprojekt 32: Der Arduino-Talker - Wir programmieren ein Übertragungsprotokoll

Wir sprechen mit dem Arduino

Bastelprojekt 33: Die drahtlose Kommunikation über Bluetooth

Was ist Funkkommunikation?

Das Bluetooth-Modul HC-06

Workshop zu Bluetooth

Bastelprojekt 34: Netzwerkkommunikation

Was ist ein Netzwerk?

Workshop zur Netzwerkkommunikation

Bastelprojekt 35: Das ESP32-Board

Den ESP32 kennenlernen

Blinken mit dem ESP32

Das ESP32-Board D1 R32

Bastelprojekt: Der Temperatur-Logger

Workshop zum Temperatur-Logger

Bastelprojekt 36: Die Digital-Analog-Wandlung

Das R2R-Netzwerk

Bastelprojekt: Wir schalten die Bits

Bastelprojekt mit dem PCF8591P

Analyse des I²C-Datenstroms

Bastelprojekt 37: Arduino mit einer Blocksprache programmieren

S4A – Scratch for Arduino

ArduBlock – Arduino mit Block

Das Open Roberta Lab

Node-RED – Blocksprache fürs IoT

Bastelprojekt 38: Eine Interrupt-Steuerung

Was verstehen wir unter einem Interrupt?

Die Timer des Arduino Uno

Bastelprojekt 39: Das fliegende TFT-Display

Die Ansteuerung eines OLED

Das Kompassmodul CMPS11

Bastelprojekt 40: Build your own Arduino

Wie findet ein Sketch den Weg in den ATmega328?

Die Programmierung mit PlatformIO

Bastelprojekt Bootloader Burner als Mini-Shield

Bastelprojekt 41: Arduino meets Raspberry Pi

Wir erwecken den Arduino unter dem Raspberry Pi zum Leben

Bastelprojekt: Servomotoren mit einem Schieberegler steuern

Bastelprojekt 42: Der ArduTrak

Der ArduTrak und seine Tastatur

Workshop zum ArduTrak

Bastelprojekt 43: Mit Node-RED fürs Internet der Dinge programmieren

Wie Node-RED arbeitet

Bastelprojekt mit dem Temperatur- und Feuchtigkeitssensor DHT11

Bastelprojekt 44: Bluetooth und das Android-Smartphone

Smartphone App mit dem App-Inventor

Bastelprojekt 45: MQTT

M2M-Kommunikation mit MQTT

Bastelprojekt 46: LoRa und LoRaWAN

LoRa-Grundlagen

Zwei LoRa-Nodes kommunizieren miteinander

Das The Things Network (TTN)

Bastelprojekt: Der Temperatursensor DHT11 sendet ins TTN

Workshop zum LoRa-Bastelprojekt

Bastelprojekt 47: Eine Leiterplatte mit KiCad erstellen

KiCad kennenlernen

Günstige PCB-Herstellung

Bastelprojekt 48: Einen MIDI-Controller bauen

Die Digital Audio Workstation

Einige Grundlagen

Workshop zum MIDI-Controller

Mit Arduino die elektronische Welt entdecken

Erik Bartmann



Impressum

Die Informationen in diesem Buch wurden mit größter Sorgfalt erarbeitet. Dennoch können Fehler nicht vollständig ausgeschlossen werden. Verlag, Autoren und Übersetzer übernehmen keine juristische Verantwortung oder irgendeine Haftung für eventuell verbliebene Fehler und deren Folgen.

Alle Warennamen werden ohne Gewährleistung der freien Verwendbarkeit benutzt und sind möglicherweise eingetragene Warenzeichen. Der Verlag richtet sich im wesentlichen nach den Schreibweisen der Hersteller. Das Werk einschließlich aller seiner Teile ist urheberrechtlich geschützt. Alle Rechte vorbehalten einschließlich der Vervielfältigung, Übersetzung, Mikroverfilmung sowie Einspeicherung und Verarbeitung in elektronischen Systemen.

Kommentare und Fragen können Sie gerne an uns richten:

Bombini Verlags GmbH
Kaiserstraße 235
53113 Bonn
E-Mail: service@bombini-verlag.de

Copyright:
© 2021 by Bombini Verlag

Bibliografische Information Der Deutschen
Nationalbibliothek
Die Deutsche Nationalbibliothek verzeichnet diese
Publikation in der Deutschen Nationalbibliografie;

detaillierte bibliografische Daten sind im Internet über <http://dnb.d-nb.de> abrufbar.

Umschlaggestaltung: Michael Oreal, Köln (www.oreal.de)
electronic **pub**lication: III-satz, Husby, www.drei-satz.de

ISBN 978-3-946496-29-8

Grußwort zur vierten Auflage von Teo Swee Ann, CEO von Espressif

Dear Erik,

In 2016, I came across a book of yours, which was about ESP8266, a microcontroller that my company Espressif had developed and released two years earlier. It was the first book ever written about ESP8266. Although I do not speak German and, therefore, I could not read it, I still understood the basic statements you made about ESP8266, as I was going through your text. Your beautiful and numerous illustrations helped me guess the accompanying text. Most importantly, I liked that you promoted the ESP8266 microcontroller as a tool for the Maker movement in as early as 2016. In any case, your assessment back then has been proved correct: Today, in every Maker workshop, no matter where in the world, one can find an ESP8266 or its successor, ESP32, alongside an Arduino and a Raspberry Pi board.

As I learned later, you had also written an extensive book about Arduino. Since its first edition in 2011, this 1,000-page book has been published in three editions already, and has become a reference book about Arduino in German. In this book, not only you do describe the functionality of the Arduino hardware and software, but you also cover the entire Maker catalogue of relevant topics. You explain the basics about electronics in detail and in an easy-to-understand way, you give an introduction to programming, and you show in numerous projects what great things can be developed with the Arduino platform.

In addition, you let the Maker think outside the box, discussing cutting-edge ideas that push the Arduino hobbyist's creativity further. You have included MQTT, Node-RED, and LoRa-related topics, among many others that can help every Maker develop truly awesome and even complicated projects on their own. I am pleased to see that you have also covered ESP32 with its own project in your book.

I wish you much success for the fourth edition of your Arduino book.

Teo Sween Ann



Einleitung

Arduino für alle

Als ich mein erstes Buch über Arduino schrieb, gab es zwei oder drei Händler in Deutschland, über die man das Arduino-Board beziehen konnte, und ungefähr 800 Videos wurden bei YouTube aufgelistet, wenn man das Suchwort Arduino eingab. Für Elektronikprofis war dieser Mikrocontroller kaum ein Thema und wenn, dann belächelten sie eher seine Leistungsfähigkeit.

Die Arduino-Entwickler, die sich 2005 in der italienischen Kunst-Uni Ivrea ans Werk machten, einen leicht zu programmierenden Mikrocontroller zu entwerfen, hatten nicht den Elektronikprofi im Visier, sondern den Kunststudenten mit wenig Programmier- und Hardwarekenntnissen. Ihr Fokus lag dabei auf dem schnellen Realisieren von Ideen, dem Prototyping: eine künstlerische Idee, die der Student ohne Unterstützung von Ingenieuren und Programmierern selbst realisieren konnte.

Der Layoutfehler, der Geschichte machte

Die Arduino-Macher, allesamt keine Hardwarespezialisten, stellten den ersten Arduino-Mikrocontroller, den Arduino Uno, unter eine freie Hardwarelizenz, wobei sie die Erfahrungen der Open-Source-Bewegung auf die Hardwarewelt übertrugen. Sie legten die Schaltpläne offen, so dass jedermann frei darin war, das Board nach- oder

umzubauen. Und die Arduino-Programmierumgebung – oder auch IDE genannt (Integrated Development Environment) – stellten sie ebenfalls unter eine Open-Source-Lizenz, jeder konnte die Software frei nutzen.

Bald schon wurde das Arduino-Board nicht nur von Studenten der italienischen Uni genutzt, sondern entwickelte sich zu einer praktischen Prototyping-Plattform, erst für Elektronik-Nerds, dann für Hobbyisten auf der ganzen Welt. Wesentlich beigetragen zur erfolgreichen Verbreitung dieser Plattform hat das Internet, das es ermöglichte, Schaltungen und dazugehörige Softwareprogramme leicht zu verbreiten – und Dank der freien Lizenz – auch nutzbar für alle zu machen. Ein riesiger weltweiter Pool von Ideen entstand, entwickelt von Hobbyisten, die stolz einer weltweiten Community ihre Projekte vorstellten. Projektideen wurden von irgendjemandem auf der Welt übernommen, vielleicht sogar verbessert und wieder ins Netz gestellt.

In den Folgejahren entwickelten sich der Arduino und seine Entwicklungsumgebung zu einem Quasistandard in der weltweiten Bastlerwelt. Eine neue, leicht zugängliche Plattform für E-Bastler war entstanden. Das blieb auch nicht den großen Hardwareproduzenten verborgen, die registriert hatten, dass zunehmend der Arduino für alle möglichen Steuerungs- und Messaufgaben eingesetzt wurde. Um die Möglichkeiten des Arduino-Boards zu erweitern, wurden Ergänzungsplatinen, die genau auf den Arduino Uno passten, sogenannte Shields, entwickelt.

Spätestens zu dem Zeitpunkt, als einer der größten Produzenten von Hardware seine Entwicklungsabteilung anwies, zukünftig das Layout des Arduino-Boards als Basis für ihre Hardware anzusehen, war klar, dass der Arduino aus der Technikwelt nicht mehr wegzudenken war. Kleine

Anekdote am Rande: Wie schon gesagt, die Arduino-Entwickler waren zu Beginn noch keine Hardwarespezialisten, so dass sie leider einen Layoutfehler in das Arduino-Uno-Board einbauten. Um kompatibel mit dem Quasistandard zu bleiben, übernahm der namhafte weltweite Hersteller brav diesen Layoutfehler. Bis heute.

Von den Segnungen des Copy&Paste - und seinen Beschränkungen

Als ich mein erstes Arduino-Buch plante, fragte ich mich, was ein Bastler vor allem braucht, um mit dem Arduino eigene Projekte zu realisieren. Es war ja schon zu diesem Zeitpunkt möglich, sich über das Internet und über die gerade entstehenden Arduino-Communities und -foren lauffähige Programme zu besorgen, sie auf den eigenen Arduino hochzuladen und schon lief das Projekt auch in heimischer Umgebung, vielleicht noch versehen mit ein paar gezielten Eingriffen in den Code. Musste zusätzliche Hardware verbaut werden, orientierte man sich beim Nachbau an der Schaltung.

Was also sollte ein Buch über den Arduino noch enthalten außer der Beschreibung des Boards und der Entwicklungsumgebung und wie man die Programme vom PC auf den Mikrocontroller bekommt? Was passiert, wenn man eine Schaltung nachgebaut hat und sie dann erweitern möchte? Um einen guten konzeptionellen Ansatz für mein Buch zu finden, fragte ich mich, was mir am meisten dabei geholfen hat, eigene Elektronikprojekte zu bauen. Die Antwort war simpel und ernüchternd: Mir haben meine Grundkenntnisse in Elektrotechnik dabei geholfen, aus Ideen funktionierende Technikprojekte zu machen.

Ich finde es absolut beeindruckend, dass Menschen, die nichts mitbringen außer Bastellust, mit dem Arduino bereits nach einer halben Stunde eine Ampelschaltung auf dem Steckbrett betreiben können, ohne jemals zuvor einen Mikrocontroller in der Hand gehalten oder eine Zeile Code geschrieben zu haben. Es reicht dazu ein YouTube-Video und vielleicht noch eine Schritt-für-Schritt-Beschreibung, schon blinken die LEDs im gewünschten Ampelschaltungsrhythmus auf dem Schreibtisch zu Hause. Vielleicht hat man dann noch Spaß daran, den Rhythmus abzuändern, indem man im Code an der entsprechenden Stelle, die leicht zu finden ist, herumtüftelt – das war es dann aber auch schon. Um mehr daraus zu machen, braucht man ein paar Grundkenntnisse der Elektronik und der Elektrotechnik. Deshalb habe ich dieses Buch geschrieben.

Wie ich mein Buch aufgebaut habe

Zunächst finde ich wichtig, dir möglichst viel über die Hardware und Software von Arduino mitzuteilen, damit du die gesamte Klaviatur kennenlernenst, auf der hier gespielt wird. In **Kapitel 1**, »Arduino: Die Hardware«, stelle ich dir ziemlich detailliert das Arduino-Board mit allen Bauteilen, der Stromversorgung und den Schnittstellen vor. Daneben gibt's eine Auffrischung einiger physikalischer Grundbegriffe, die bei der Mikroelektronik eine Rolle spielen. Kannst du lesen, musst du aber nicht. In **Kapitel 2**, »Arduino: Die Software«, lernst du, das Board zu kontrollieren. Du erfährst, wie der Workflow zwischen PC oder Notebook und dem Arduino-Board abläuft und wie der Arduino-Code aufgebaut ist, um den Mikrocontroller steuern zu können.

In **Kapitel 3**, »Keine Angst vorm Programmieren – Coding Basics«, bekommst du eine geballte Packung Basic-Wissen übers Programmieren präsentiert. Auch dieses Kapitel geht wie die vorherigen Kapitel manchmal sehr in die Tiefe, aber im Zweifelsfall gilt auch hier: Kannst du lesen, musst du jetzt aber nicht. Ich empfehle dir, die ersten Kapitel als eine Art Referenzkapitel anzusehen, auf die du später im Buch – oder sogar später bei deinen eigenen Arduino-Bastelprojekten – zugreifen kannst, wenn du etwas genauer wissen möchtest.

In **Kapitel 4**, »Das Arduino-Discoveryboard«, stelle ich dir ein von mir entwickeltes Entwickler-Board vor. Ich habe auf einer Platine Standardelektronikbauteile verbaut, die bei den Arduino-Bastelprojekten ständig gebraucht werden. Die Bauteile habe ich auf der Platine fest verlötet, so dass ich nur noch den Arduino anschließen muss und die Projekte dann durchführen kann. Der Vorteil? Die lästige Wurschtelei auf dem Breadboard entfällt weitestgehend und der Schaltungsaufbau ist übersichtlicher. Du kannst das Arduino Discoveryboard nachbauen, dabei kannst du dich gleich im Löten üben.

Und dann geht es endlich los mit den Arduino-Bastelprojekten und deiner Bastelleidenschaft! Die Projekte fangen sehr leicht an, aber das ändert sich recht schnell, sie werden von Projekt zu Projekt komplexer. An den Stellen, an denen Grundlagenwissen notwendig ist, streue ich Texte ein, die das Bastelprojekt etwas theoretischer beleuchten. So wächst hoffentlich von Projekt zu Projekt auch dein Elektronikgrundwissen, so dass du schon bald in der Lage sein wirst, meine Projekte abzuändern oder sogar ganz eigene Bastelprojekte zu entwickeln.

Wie ich die Bastelprojekte gestaltet habe

Wie schon erwähnt, bauen die Bastelprojekte aufeinander auf. Was du in einem Projekt gelernt hast, kannst und sollst du auch in den Folgeprojekten anwenden. Benötigst du Grundlagenwissen, dann präsentiere ich es an der Stelle, an der es benötigt wird. Die Projekte haben alle einen ähnlichen Aufbau.

- Codeanalyse: Schritt für Schritt gehe ich den Programmiercode durch und erkläre genau jeden Aspekt des Programms.
- Schaltplan: Der Schaltplan ist die schematische Darstellung des Bastelprojektes, sozusagen der Bauplan.
- Schaltungsaufbau: Wie die Schaltung dann auf einem Breadboard aufgesteckt oder auf einer Platine aufgelötet aussieht, das stelle ich mit Fotos dar.
- Troubleshooting: Die systematische Fehlersuche ist IMHO eine der besonderen Fertigkeiten, die ich gern jedem angehenden E-Bastler nahebringen möchte. Deshalb habe ich sie zum festen Bestandteil meiner Bastelprojekte gemacht.
- Benötigte Bauteile: Ich gebe dir einen Überblick über die in diesem Bastelprojekt benötigten Bauteile. So kannst du auf einen Blick feststellen, ob dir die Bauteile bereits vorliegen.
- Programmcode: Der für die Bastelprojekte verwendete Programmiercode wird vorgestellt, manchmal in kleinen Häppchen.

Gelegentlich biete ich *Quick-and-Dirty*-Lösungen an, die auf den ersten Blick etwas umständlich erscheinen mögen. Anschließend folgt eine verbesserte Variante, was als Anregung dienen soll, nach weiteren Lösungsmöglichkeiten zu suchen und – im Idealfall – eigene Lösungen zu entwerfen. Wenn das geschieht, habe ich genau das erreicht, was ich beabsichtigt habe. Falls nicht, auch gut. Jeder geht seinen eigenen Weg und kommt irgendwann ans Ziel.

Meine Website mit weiteren Arduino- und Elektronikthemen

An dieser Stelle möchte ich auch auf meine Internetseite

<https://erik-bartmann.de/>



hinweisen, auf der du unter anderem einiges zum Thema Arduino findest. Hier kannst du auch alle Programme herunterladen, die ich in meinen Bastelprojekten verwende. Zu den schönsten Momenten für einen Autor gehört es, wenn er Feedback von Lesern erhält. Ich bin manchmal richtig gerührt, wenn mir ein Leser mitteilt, wie viel Spaß ihm mein Buch beim Lesen und Basteln gemacht hat. Darüber freue ich mich sehr und möchte jeden dazu ermuntern, mir mitzuteilen, wie er mein Buch fand, was ihm besonders gefallen hat und auch, was er für verbesserungswürdig an meinem Buch hält. Meine E-Mail-Adresse lautet:

erik.bartmann@yahoo.de.



Sie ist auf meiner Internetseite auch noch einmal aufgeführt.

Voraussetzungen

Die einzige persönliche Voraussetzung, die du mitbringen solltest, ist Interesse am Basteln und Tüfteln. Du musst kein Elektronik-Nerd sein und auch kein Computerexperte, um die im Buch gezeigten Bastelprojekte nachvollziehen und nachbauen zu können. Da wir sehr moderat beginnen, besteht absolut keine Gefahr, dass irgendjemand auf der Strecke bleibt. Setz dich also nicht unter Druck und mach die Dinge nicht schwieriger, als sie sind. Der Spaß steht immer an erster Stelle.

Benötigte Bauteile

Das Arduino-Board für sich alleine ist zwar ganz nett und wir können uns daran erfreuen, wie klein und schön alles konzipiert wurde, doch auf Dauer reicht das natürlich nicht aus. Wir sollten uns daher im nächsten Schritt ansehen, was wir alles von außen an das Board anschließen können. Falls du noch nie in irgendeiner Weise mit elektronischen Bauteilen (wie Widerständen, Kondensatoren, Transistoren oder Dioden, um nur einige zu nennen) in Berührung gekommen bist, ist das nicht weiter schlimm. Die benötigten Teile erkläre ich dir ausführlich, so dass du nachher weißt, wie sie einzeln und innerhalb der Schaltung reagieren. Zu Beginn eines Arduino-Bastelprojektes stelle ich, wie schon oben kurz erwähnt, eine Liste mit den erforderlichen Teilen zur Verfügung, mit deren Hilfe du dir diese Dinge besorgen kannst. Zentraler Bestandteil ist natürlich immer das Arduino-Board, das ich nicht immer explizit erwähnen werde.

Falls du dich an dieser Stelle fragen solltest, was wohl ein Arduino-Board kosten mag und ob du nach dieser Investition deinen gewohnten Lebensstil fortführen kannst, kann ich nur sagen: *Yep, du kannst!* Das Board kostet um die 25€, und das ist wirklich nicht viel. Ich verwende in allen grundlegenden Kapiteln das *Arduino Uno*-Board. Und auch die anderen Bauteile, die wir in den Bastelprojekten verwenden werden, sind bezahlbar.

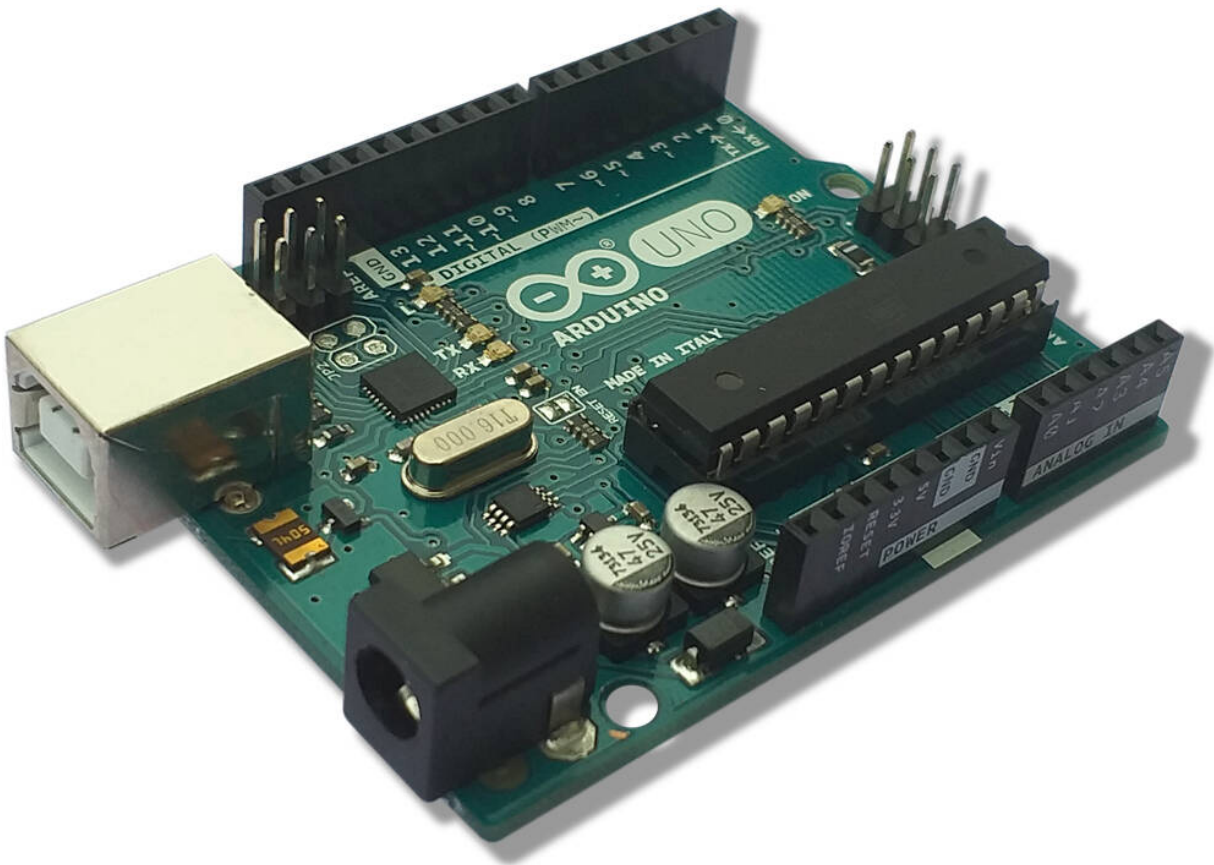


Abb. 1: Der Arduino Uno – das Original

Verhaltensregeln

Wenn du dich so richtig im Brass befindest und voll konzentriert bist auf etwas, was dir unheimlich viel Spaß

macht, treten folgende Effekte auf:

- Verminderte Nahrungsaufnahme, die zu kritischem Gewichts- und besorgniserregendem Realitätsverlust führen kann.
- Unzureichende Flüssigkeitszufuhr bis hin zu Dehydrierung und vermehrter Staubentwicklung in der Umgebung.
- Vernachlässigung sämtlicher hygienischer Maßnahmen wie Waschen, Duschen, Zähneputzen, verbunden mit erhöhtem Auftreten von Ungeziefer.
- Abbruch jeglicher zwischenmenschlicher Beziehungen.

Lass es nicht so weit kommen und öffne ab und zu das Fenster, um zugewanderten Insekten das Verlassen des Zimmers zu ermöglichen und Frischluft und Sonnenlicht hereinzulassen. Um den oben genannten Effekten entgegenzuwirken, kannst du den Wecker stellen, damit du in regelmäßigen Zeitabständen zu einer Unterbrechung deiner Tätigkeit aufgefordert wirst. Ich möchte mich nach der Veröffentlichung dieses Buches nicht mit einer Beschwerdewelle konfrontiert sehen, die von erbosten Partnern oder vernachlässigten Freunden über mich hereinbricht. Sag also nicht, ich hätte dich nicht vor den Risiken gewarnt. Ich wünsche dir nun eine Menge Spaß und viel Erfolg beim Basteln mit deinem Arduino-Board!

BARTMANN

Kapitel 1

Arduino: Die Hardware

Wie in der Einleitung erwähnt, wurde das Arduino-Projekt in Ivrea (Italien) an der dortigen Kunsthochschule entwickelt. In einer Kneipe nahe der Hochschule trafen sich gelegentlich Massimo Banzi und David Cuartielles, die 2005 das erste Arduino-Board entwickelten. Die Kneipe war nach Arduin von Ivrea benannt, der um das Jahr 1000 König von Italien war. *Arduino* ist seitdem die Bezeichnung sowohl für die Software als auch für die Hardware dieses Open-Source-Projektes.

In diesem Kapitel werde ich auf die Arduino-Hardware eingehen. Dabei werde ich auch wichtige Grundbegriffe und -themen erklären, die für die Mikrocontroller-Technik insgesamt von Bedeutung sind. Ich erkläre ausführlich, welche Bauteile zu einem Mikrocontroller gehören. Diese Komponenten befinden sich praktisch in jedem Mikrocontroller-Board, deshalb gehe ich ausführlich darauf ein. Und nebenher frische ich deine Physikkenntnisse ein wenig auf, indem ich einige Grundbegriffe wie Spannung oder Strom behandle.

Ich werde in diesem Kapitel – wie auch im gesamten Buch – mit dem *Arduino Uno*-Board arbeiten. Das Arduino-Uno-Board war das erste Board, das von den Arduino-Machern entwickelt und produziert wurde. Weitere Arduino-Boards kamen dann später hinzu, die technisch etwas besser wurden. Was bedeutet eigentlich besser? Wenn man sich Kennwerte wie Prozessortakt oder verfügbarer Arbeitsspeicher als Entscheidungskriterium für den Kauf eines neuen Arduino-Boards aussucht, dann gibt es sicherlich Boards, die besser geeignet sind, weil sie

schneller arbeiten und größere Programme speichern und verarbeiten können. Doch das ist eben nicht immer besser. Für einen geeigneten Einstieg in die elektronische Bastlerwelt ist das Arduino-Uno-Board eben in meinen Augen die bessere Wahl, weil es sehr robust ist und sehr weite Verbreitung gefunden hat.

Der *Arduino Yún* beispielsweise ist sicherlich ein interessantes Board, das einiges an Erweiterungen wie das Betriebssystem Linux bietet. Dennoch hat sich das Board in der Bastler- und Hobbywelt nicht so richtig durchgesetzt, vermutlich auch, weil der Raspberry Pi bereits auf dem Markt war, als Arduino Yún erschien. Scheller und mehr ist eben nicht zwangsläufig auch besser.

Dennoch möchte ich einige wirklich gute Arduino-Bords anführen:

- Arduino Leonardo
- Arduino Mega 2560
- Arduino Nano

Die genannten Boards unterscheiden sich hinsichtlich ihrer Größe und Anzahl der Buchsen, also der Anschlussmöglichkeiten, um mit der Außenwelt in Verbindung zu treten. Des Weiteren haben sie unterschiedliche Prozessoren, Taktfrequenzen und Speichervolumen. Und dennoch arbeiten sie alle nach demselben Prinzip und können durch die einheitliche Arduino-Entwicklungsumgebung angesprochen und programmiert werden. Je nach Anwendungsgebiet und Erfordernissen ist das eine Arduino-Board vielleicht besser geeignet als das andere. Die einen benötigen ein Board mit vielen I/O-Pins und entscheiden sich beispielsweise für den Arduino Mega oder den Due. Andere wählen den Arduino

Micro oder Nano aus, denn diese sind recht klein und passen wunderbar in kleine Gehäuse. Sie kommen dort zur Anwendung, wo das Platzangebot beschränkt ist.

Das Universalgenie ist in meinen Augen jedoch der *Arduino Uno* und er wird es wohl noch eine lange Zeit bleiben. Er bietet eine ideale Plattform für den Einstieg in die Mikrocontroller-Welt. Für ihn finden sich im Internet auch die meisten Tutorials, Projekte und Diskussionen. Steigen die Ansprüche für deine Projekte, ist es kein Problem, sich ein weiteres Arduino-Modell zuzulegen, denn die Preise sind wirklich moderat. Viele Bastler legen sich im Laufe der Zeit mehrere unterschiedliche Boards zu, um darüber auch mehr und mehr Erfahrungen zu sammeln, was in meinen Augen ein ganz normaler Entwicklungsfortschritt ist.

Über die nachfolgenden Links bekommst du Detailinformationen zu den gezeigten Boards:

- **Arduino Uno:**
<https://www.arduino.cc/en/Main/ArduinoBoardUno>
- **Arduino Mega:**
<https://www.arduino.cc/en/Main/ArduinoBoardMega2560>
- **Arduino Leonardo:**
<https://www.arduino.cc/en/Main/ArduinoBoardLeonardo>
- **Arduino Micro:**
<https://www.arduino.cc/en/Main/ArduinoBoardMicro>
- **Arduino Nano:**
<https://www.arduino.cc/en/Main/ArduinoBoardNano>



Es gibt noch weitere zahlreiche Arduino-Boards und Erweiterungen, die unter den folgenden Adressen zu finden sind: