GANZ EINFACH

Sensoren selbst gebaut mit ESPHome

Smart-Home-Geräte ohne Löten, ohne Cloud und ohne Abo

Für Home Assistant Schritt für Schritt erklärt



Verstehen · Anwenden · Überprüfen

Danilo Sieren

Danilo Sieren

Sensoren selbst gebaut mit ESPHome

Smart-Home-Geräte ohne Löten,
ohne Cloud und ohne Abo
Für Home Assistant Schritt für Schritt erklärt

Erste Auflage 2026
Stand ESPHome 2026.10

Verstehen · Anwenden · Überprüfen

Vorwort

Kennst du das? Du kaufst einen schicken Temperatursensor fürs Kinderzimmer, lädst die App des Herstellers herunter, legst ein Konto an, akzeptierst Datenschutzbestimmungen von zwölf Seiten, und nach zwei Jahren schickt dir der Hersteller eine freundliche Mail: Der Dienst wird eingestellt. Ab nächstem Monat zeigt dein Sensor nichts mehr an. Er funktioniert noch, er misst noch, aber niemand fragt ihn mehr.

Dieses Buch zeigt dir einen anderen Weg. Mit ESPHome baust du deine Sensoren selbst, aus fertigen Modulen, die nur ein paar Euro kosten, und ohne einen einzigen Lötpunkt. Die Geräte sprechen direkt mit deinem Home Assistant, ohne Umweg über eine fremde Cloud, ohne Konto, ohne Abo. Sie laufen auch dann weiter, wenn das Internet ausfällt oder irgendein Unternehmen seine Pläne ändert. Und weil du selbst bestimmst, was sie tun, bekommst du genau den Sensor, den du brauchst, statt eines Kompromisses aus dem Elektronikmarkt.

Klingt nach Bastelkeller und Lötkolben? Ist es nicht. Ich erkläre dir alles so, wie ich es einem Freund am Küchentisch erklären würde: in Ruhe, Schritt für Schritt und ohne vorauszusetzen, dass du schon einmal einen Mikrocontroller in der Hand hattest. Die Bauteile werden nur zusammengesteckt. Die Programmierung erledigt ESPHome für dich; du beschreibst lediglich, was dein Gerät können soll. Seit dem neuen Device Builder, der 2026 erschienen ist, geht das sogar weitgehend per Mausklick.

Durch das ganze Buch begleitet uns die Familie Brandt aus Dortmund mit ihrem Projekt „Das wache Zuhause“. In ihrer Wohnung entstehen nacheinander sechs Geräte: eine CO_2-Lüftungsampel für das Kinderzimmer, eine Klimastation für das Wohnzimmer, ein Anwesenheitsmelder für den Flur, ein Wassermelder für den Keller, ein Leser für den Stromzähler und ein

Bluetooth-Proxy, der die Reichweite von Home Assistant vergrößert. Kapitel 27 fasst alle sechs Geräte mit Einkaufsliste, Verkabelung und vollständiger Konfiguration zusammen.

Dieses Buch knüpft an den Band über Home Assistant aus dieser Reihe an. Du brauchst eine laufende Home-Assistant-Installation; wie du sie einrichtest, steht dort. Programmierkenntnisse brauchst du nicht. Wenn du einmal eine Textdatei bearbeitet hast, kommst du mit den Konfigurationsdateien von ESPHome zurecht. Und wenn nicht, lernst du es in Kapitel 6.

Die redaktionelle Prüfung hat den Stand 1. November 2026. Grundlage ist ESPHome in der Version 2026.10 mit dem Device Builder, der seit Version 2026.6 zum Standard gehört, und Home Assistant in einer aktuellen Version, in der Erweiterungen seit 2026.2 „Apps“ statt „Add-ons“ heißen. Die Konfigurationen in diesem Buch folgen der offiziellen Dokumentation. Anhang F erläutert, welche Angaben wie geprüft wurden und was du vor dem eigenen Nachbau kontrollieren solltest. Die Abbildungen sind bewusst schematisch gehalten: Sie erklären Zusammenhänge und ersetzen nicht das Datenblatt deines konkreten Moduls.

Eine Regel begleitet dich durch das ganze Buch: Erst stecken, dann prüfen, dann einschalten. Und eine zweite, noch wichtigere: Alles in diesem Buch läuft mit Kleinspannung aus einem USB-Netzteil. An 230 Volt Netzspannung bauen wir nicht. Für Geräte, die Lampen oder Steckdosen schalten, stellt dir Kapitel 17 fertige, geprüfte Lösungen vor, die ab Werk mit ESPHome laufen.

So arbeitest du mit diesem Buch

Teil I mit den Kapiteln 1 bis 7 legt die Grundlagen. Du erfährst, was ESPHome ist, welche Mikrocontroller es gibt, welche Bauteile und Kabel du brauchst, wie du die Stromversorgung sicher gestaltest, wie du den Device Builder in Home Assistant einrichtest und wie dein erstes Gerät seine Software bekommt. Lies diesen Teil vollständig, bevor du das erste Projekt baust.

Teil II mit den Kapiteln 8 bis 17 ist das Herz des Buches. Jedes Kapitel stellt einen Sensortyp vor und baut ihn Schritt für Schritt auf: Temperatur und Luftfeuchte, CO_2, Feinstaub, Bewegung und Anwesenheit, Licht und Display, Kontakte und Wassermelder, Temperaturfühler für draußen, den Stromzähler, den Bluetooth-Proxy und schließlich fertige ESPHome-Geräte. Du kannst die Kapitel in beliebiger Reihenfolge lesen, nachdem du Kapitel 8 gemacht hast. Dort lernst du den Ablauf, der sich in allen weiteren Projekten wiederholt.

Teil III mit den Kapiteln 18 bis 23 zeigt dir, wie aus Sensoren ein nützliches System wird: Automationen direkt auf dem Gerät und in Home Assistant, Dashboards, das Energie-Dashboard, Gehäuse, Statusanzeigen und Batteriebetrieb. Teil IV mit den Kapiteln 24 bis 28 behandelt Sicherheit, Wartung und Fehlersuche, fasst das Lernprojekt zusammen und endet mit der Abnahme.

Jedes Kapitel endet mit einer Werkstatt und häufig mit einer Vertiefung. Die Werkstatt ist eine praktische Übung mit einem Ergebnis, das du überprüfen kannst. Die Vertiefung zeigt Grenzen oder Erweiterungen. Die Anhänge enthalten Einkaufslisten, Pinbelegungen, einen Fehlerkatalog, Begriffe, alle Konfigurationen am Stück, den Versionsstand und Vorlagen.

Konfigurationen stehen in Schreibmaschinenschrift in eigenen Blöcken. Achte beim Abtippen genau auf die Einrückungen. ESPHome verwendet das Format YAML, in dem die Anzahl der Leerzeichen am Zeilenanfang über die Bedeutung entscheidet. Verwende immer Leerzeichen, niemals Tabulatoren, und rücke jede Ebene um zwei Leerzeichen ein. Wo ein Block nur einen Ausschnitt zeigt, steht das im Text davor.

> **Tipp:** Leg dir für jedes Gerät eine Karteikarte oder eine Notiz an: Name, Board, angeschlossene Sensoren mit Pins, Standort und Datum. Wenn du in einem Jahr wissen willst, an welchem Pin der Wassermelder hängt, bist du dir selbst dankbar.

Die Quellenkennungen wie [E04] führen zum Quellenverzeichnis am Ende des Buches. Dort stehen die offiziellen Seiten von ESPHome, Home Assistant und den Herstellern. Prüfe bei Preisen, Versionsnummern und Bauteilbezeichnungen vor dem Kauf den aktuellen Stand. Die Preise in diesem Buch sind grobe Richtwerte und schwanken je nach Händler erheblich.

Inhalt

Vorwort 2

So arbeitest du mit diesem Buch 4

Inhalt 6

1 ESPHome verstehen: dein Smart Home ohne Cloud 8

2 Mikrocontroller verstehen 14

3 Die Ausrüstung ohne Löten 19

4 Strom und Sicherheit 25

5 ESPHome in Home Assistant einrichten 30

6 Der Device Builder: das erste Gerät 34

7 Flashen über USB und WLAN 40

8 Erstes Projekt: Temperatur und Luftfeuchte 45

9 Die CO_2-Lüftungsampel 52

10 Feinstaub und Luftqualität 59

11 Bewegung und Anwesenheit 64

12 Licht, Helligkeit und Display 71

13 Türen, Fenster und Wasser 77

14 Temperatur draußen und im Wasser 83

15 Den Stromzähler auslesen 88

16 Der Bluetooth-Proxy 94

17 Fertige ESPHome-Geräte 99

18 Automationen auf dem Gerät 104

19 Dashboards und Benachrichtigungen 110

20 Das Energie-Dashboard 115

21 Gehäuse und Montage 119

22 LEDs und Signale 124
23 Batteriebetrieb und Tiefschlaf 129
24 Sicherheit im Heimnetz 135
25 Updates, Sicherung und Wartung 140
26 Fehler eingrenzen 145
27 Das vollständige wache Zuhause 151
28 Abnahme und Erweiterungsideen 156
Anhang A Einkaufslisten 160
Anhang B Pinbelegungen 165
Anhang C Fehlerkatalog 169
Anhang D Begriffe für die tägliche Arbeit 174
Anhang E Fertige Konfigurationen 179
Anhang F Versionsstand und Grenzen der Überprüfung 201
Anhang G Vorlagen zum Abschreiben 204
Anhang H Projektideen zum Weiterbauen 209
Quellen und offizielle Dokumentation 219
Ausführliches Inhaltsverzeichnis 223
Stichwortverzeichnis 232
Impressum 237

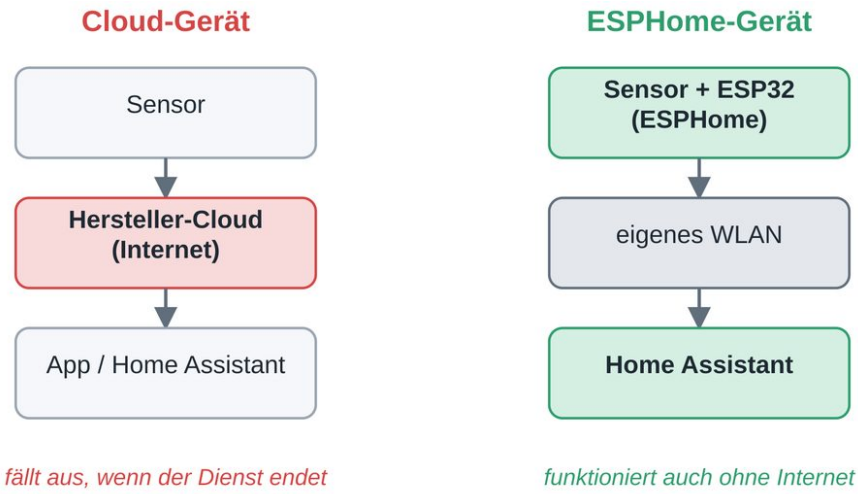
1 ESPHome verstehen: dein Smart Home ohne Cloud

1.1 Das Problem mit den fertigen Geräten

Fertige Smart-Home-Geräte sind bequem. Auspacken, App installieren, fertig. Der Preis für diese Bequemlichkeit zeigt sich oft erst später. Viele Geräte schicken jede Messung zuerst an einen Server des Herstellers, und erst von dort kommt sie zurück in deine Wohnung. Das kostet Zeit, verbraucht Datenvolumen und verrät dem Hersteller, wann du zu Hause bist, wann du lüftest und wie viel Strom du verbrauchst. Und wenn der Hersteller den Dienst einstellt, wird aus deinem Sensor ein hübsches Stück Plastik.

Hinzu kommt die Frage, ob das Gerät überhaupt misst, was du brauchst. Der eine Sensor hat keine Luftfeuchte, der andere zeigt CO_2 nur in der App an, der dritte kann nicht mit Home Assistant reden. Am Ende stehen drei Geräte von drei Herstellern mit drei Apps in der Wohnung. Wer einmal versucht hat, daraus eine gemeinsame Automation zu bauen, weiß, wie mühsam das ist.

ESPHome dreht den Spieß um. Du suchst dir die Sensoren aus, die du brauchst, steckst sie an einen kleinen Mikrocontroller und beschreibst in wenigen Zeilen, was das Gerät tun soll. ESPHome macht daraus eine fertige Software, spielt sie auf den Chip und bindet das Gerät in Home Assistant ein. Kein Konto, keine Cloud, kein Abo. Die Messwerte bleiben in deinem Netz.



Ein Cloud-Gerät hängt am Dienst des Herstellers. Ein ESPHome-Gerät spricht direkt mit Home Assistant.

1.2 Was ESPHome eigentlich ist

ESPHome ist ein freies Projekt, das heute unter dem Dach der Open Home Foundation entwickelt wird, derselben Stiftung, zu der auch Home Assistant gehört. Es besteht aus zwei Teilen. Der erste ist ein Werkzeug auf deinem Rechner oder in Home Assistant, das aus einer Beschreibung eine Firmware erzeugt. Firmware nennt man die Software, die direkt auf einem Mikrocontroller läuft. Der zweite Teil ist diese Firmware selbst, die auf dem Chip arbeitet, Sensoren ausliest und mit Home Assistant spricht. [E01]

Das Besondere ist die Art, wie du die Firmware beschreibst. Statt in einer Programmiersprache Befehle zu schreiben, sagst du ESPHome nur, welche Bauteile angeschlossen sind und wie sie heißen sollen. Für einen Temperatursensor genügen fünf, sechs Zeilen. ESPHome kennt Hunderte von Sensoren, Anzeigen, Leuchten und Schaltern und weiß, wie man sie anspricht. Diese Beschreibung steht in einer Textdatei im Format YAML. Seit 2026 kannst du sie mit dem Device Builder auch zusammenklicken.

Man könnte sagen: ESPHome ist für Mikrocontroller, was ein Baukasten für Möbel ist. Du musst nicht wissen, wie man Holz sägt.

Du musst nur wissen, welches Teil wohin gehört. Den Rest erledigt die Anleitung.

1.3 Warum ESPHome und Home Assistant so gut zusammenpassen

ESPHome-Geräte und Home Assistant sprechen eine gemeinsame Sprache, die native API. Sobald ein neues ESPHome-Gerät in deinem Netz auftaucht, meldet Home Assistant: Neues Gerät gefunden. Ein Klick, und alle Sensoren erscheinen mit ihren Namen, Einheiten und Symbolen. Du musst keine Adressen eintragen, keine Themen in einem Nachrichtensystem anlegen und keine Werte umrechnen. [E30]

Die Verbindung ist verschlüsselt und funktioniert in beide Richtungen. Das Gerät meldet Messwerte, Home Assistant kann Befehle schicken, etwa eine LED einschalten oder ein Relais schalten. Und weil beide Projekte aus derselben Stiftung kommen, werden sie aufeinander abgestimmt weiterentwickelt. Neue Funktionen in Home Assistant, wie der Einrichtungsassistent für Geräte, den Home Assistant 2026.11 anzeigt, unterstützt ESPHome oft schon im selben Monat. [E04]

1.4 Was du mit ESPHome bauen kannst

Die Liste ist lang, und das Schöne ist: Für die meisten Projekte brauchst du nur fertige Module, die du zusammensteckst. Klimasensoren messen Temperatur, Luftfeuchte und Luftdruck. CO_2-Sensoren zeigen, wann gelüftet werden muss. Feinstaubsensoren messen die Partikelbelastung, etwa beim Kochen oder bei Kaminrauch in der Nachbarschaft. Bewegungsmelder und Radarsensoren erkennen, ob jemand im Raum ist, sogar wenn er still auf dem Sofa sitzt. Kontakte melden offene Fenster, Wassermelder einen undichten Schlauch.

Darüber hinaus kann ein ESPHome-Gerät LEDs steuern, kleine Displays beschreiben, Taster auswerten, Infrarotsignale senden,

Stromzähler auslesen und Bluetooth-Geräte in Reichweite an Home Assistant weiterreichen. Es gibt sogar Sprachassistenten auf ESPHome-Basis. In diesem Buch konzentrieren wir uns auf die Projekte, die in fast jedem Haushalt sinnvoll sind und sich ohne Löten umsetzen lassen.

> **Kurz gesagt:** Fast alles, was misst, meldet oder leuchtet, lässt sich mit ESPHome bauen. Was mit 230 Volt arbeitet, kaufen wir fertig.

1.5 Was ESPHome nicht ist

ESPHome ist kein Ersatz für Home Assistant. Ein einzelnes Gerät kann zwar einfache Aufgaben selbst erledigen, etwa eine LED rot schalten, wenn der CO_2-Wert steigt. Aber die Verknüpfung vieler Geräte, Zeitpläne, Dashboards und Benachrichtigungen aufs Handy übernimmt Home Assistant. Wer noch kein Home Assistant betreibt, sollte zuerst dort anfangen; der Band über Home Assistant in dieser Reihe zeigt den Weg.

ESPHome ist auch kein Werkzeug für sicherheitskritische Anwendungen. Ein selbst gebauter Rauchmelder ersetzt keinen geprüften Rauchwarnmelder, ein selbst gebauter Wassermelder keine Versicherung. Unsere Geräte sind wertvolle Helfer, die dich früh auf etwas aufmerksam machen. Für Leib und Leben verlassen wir uns auf zugelassene Geräte.

1.6 Das Lernprojekt: Das wache Zuhause

Familie Brandt wohnt in einer Altbauwohnung in Dortmund mit Kinderzimmer, Wohnzimmer, Bad, langem Flur und einem Kellerraum. Home Assistant läuft bereits auf einem kleinen Rechner im Flurschrank. Bisher steuert es ein paar Lampen und zeigt das Wetter an. Jetzt soll die Wohnung „aufwachen“ und mitbekommen, was in ihr passiert.

Die Familie hat sich sechs Ziele gesetzt. Im Kinderzimmer soll eine Ampel zeigen, wann gelüftet werden muss, weil die Kinder morgens oft über Kopfschmerzen klagen. Im Wohnzimmer soll eine

Klimastation Temperatur, Feuchte und Luftdruck messen und auf einem kleinen Display anzeigen. Im Flur soll das Licht angehen, wenn jemand nach Hause kommt. Im Keller soll ein Wassermelder anschlagen, bevor die Waschmaschine wieder einmal ausläuft. Am Hausanschluss soll der Stromzähler ausgelesen werden. Und ein Bluetooth-Proxy soll dafür sorgen, dass die Thermometer im Bad zuverlässig erreicht werden.

Jedes Ziel wird zu einem Projekt in diesem Buch. Am Ende stehen sechs Geräte, die zusammen weniger kosten als ein einziger hochpreisiger Präsenzmelder aus dem Fachhandel.

1.7 Werkstatt: Deine Wunschliste

Bevor du irgendetwas kaufst, schreib auf, was du wissen oder automatisieren möchtest. Formuliere es als Frage oder als Wunsch, nicht als Bauteil. Also nicht „Ich will einen BME280“, sondern „Ich will wissen, ob es im Schlafzimmer zu feucht ist“. Notiere zu jedem Wunsch den Raum und was passieren soll, wenn ein Wert auffällig wird.

Prüfe dann jede Zeile mit drei Fragen. Gibt es dafür schon ein Gerät in deinem Haushalt, das Home Assistant kennt? Brauchst du den Wert ständig oder nur gelegentlich? Und: Wäre ein Fehler gefährlich? Wenn ja, gehört dieser Wunsch zu einem geprüften Fertiggerät, nicht in dieses Buch.

Bei Familie Brandt standen am Ende neun Wünsche auf der Liste. Drei fielen bei den Fragen heraus: Ein Rauchmelder wird gekauft, für die Gartenbewässerung gibt es bereits ein Fertiggerät, und die Temperatur im Gefrierschrank prüft man lieber mit einem Thermometer mit eigener Warnfunktion. Die übrigen sechs sind unser Lernprojekt.

1.8 Vertiefung: Warum ausgerechnet ESP-Chips?

ESPHome ist nach den Mikrocontrollern der ESP-Familie des chinesischen Herstellers Espressif benannt. Diese Chips haben eine

kleine Revolution ausgelöst, weil sie WLAN direkt eingebaut haben und dabei nur wenige Euro kosten. Vorher brauchte man für einen vernetzten Sensor einen Mikrocontroller und ein separates Funkmodul, was teurer, größer und komplizierter war.

Inzwischen unterstützt ESPHome auch andere Chips, etwa den RP2040 des Raspberry-Pi-Projekts und den nRF52840 von Nordic Semiconductor. Für Einsteiger bleiben die ESP32-Chips aber die beste Wahl: Sie sind überall erhältlich, gut dokumentiert, und für fast jedes Projekt gibt es Beispiele aus der Gemeinschaft. Laut ESPHome laufen übrigens noch rund vierzig Prozent aller Installationen auf dem alten ESP8266. Das zeigt, wie langlebig diese kleinen Chips sind. [E03]

2 Mikrocontroller verstehen

2.1 Ein ganzer Computer auf einem Fingernagel

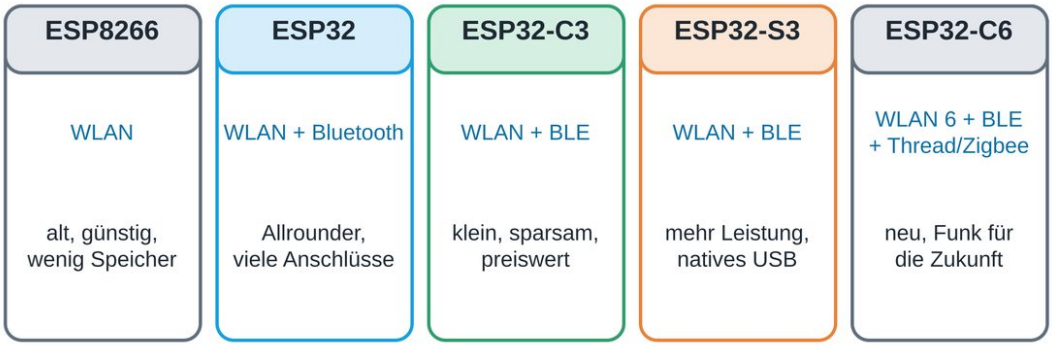
Ein Mikrocontroller ist ein vollständiger kleiner Computer auf einem einzigen Chip: Prozessor, Arbeitsspeicher, Speicher für die Software und Anschlüsse für die Außenwelt. Er ist viel einfacher als dein PC oder dein Smartphone, hat kein Betriebssystem im üblichen Sinn und keinen Bildschirm. Dafür verbraucht er wenig Strom, startet in Sekundenbruchteilen und erledigt eine einzige Aufgabe zuverlässig, Tag und Nacht.

Du kaufst den Chip in der Regel nicht einzeln, sondern auf einer kleinen Platine, einem sogenannten Entwicklungsboard. Darauf sitzen neben dem Chip ein USB-Anschluss für Strom und Programmierung, ein Spannungsregler, eine WLAN-Antenne, zwei kleine Taster und zwei Reihen von Anschlussstiften. An diese Stifte schließt du später deine Sensoren an.

2.2 Die ESP-Familie im Überblick

Für ESPHome kommen hauptsächlich fünf Chips infrage. Der ESP8266 ist der Urahn der Familie. Er hat WLAN, aber kein Bluetooth, wenig Speicher und nur wenige Anschlüsse. Er ist sehr günstig und für einfache Sensoren noch immer brauchbar, für neue Projekte aber nicht mehr die erste Wahl.

Der klassische ESP32 ist der Allrounder. Er hat WLAN und Bluetooth, zwei Prozessorkerne, viele Anschlüsse und genug Speicher für fast alles. Der ESP32-C3 ist kleiner, sparsamer und noch günstiger, mit einem einzelnen Kern und Bluetooth Low Energy. Der ESP32-S3 hat mehr Rechenleistung und einen eingebauten USB-Anschluss, der ohne zusätzlichen Wandlerchip auskommt. Der ESP32-C6 schließlich unterstützt WLAN 6 und zusätzlich die Funkstandards Thread und Zigbee. [E08, E34]



Die wichtigsten Mikrocontroller für ESPHome

2.3 Welches Board für dieses Buch?

Für alle Projekte in diesem Buch verwenden wir das ESP32-DevKitC oder ein baugleiches Board mit dem klassischen ESP32. Es ist günstig, überall erhältlich, hat genug Anschlüsse für mehrere Sensoren gleichzeitig und, ganz wichtig für uns, die Stiftleisten sind bereits angelötet. Achte beim Kauf genau darauf. Manche Händler liefern Boards mit losen Stiftleisten, die erst eingelötet werden müssen. In den Angeboten heißt die richtige Variante häufig „mit Pin Header“, „vorgelötet“ oder „soldered“.

Ein Wort zu den vielen Varianten: Unter Namen wie ESP32 DevKit, ESP32-WROOM-32 oder NodeMCU-32S werden sehr ähnliche Boards verkauft. Sie unterscheiden sich in Breite, Anzahl der Stifte und Beschriftung, nicht aber im Grundsatz. Die Pins, die wir in diesem Buch verwenden, sind auf praktisch allen Varianten vorhanden. Wenn dein Board anders aussieht, vergleiche die Beschriftung neben den Stiften mit der Pinübersicht in Anhang B.

> **Tipp:** Kauf gleich drei oder fünf Boards im Set. Einzeln kostet ein Board oft mehr als die Hälfte eines Fünferpacks, und für das Lernprojekt brauchst du ohnehin mehrere.

2.4 Die Anschlüsse: GPIO, Spannung und Masse

Die Stifte an den Rändern des Boards heißen Pins. Die meisten davon sind GPIOs, General Purpose Input/Output, also Allzweck-Ein- und Ausgänge. Jeder GPIO hat eine Nummer, etwa GPIO21