

Mikrocontroller ESP32

Das umfassende Handbuch

» Hier geht's
direkt
zum Buch

DAS VORWORT

Einleitung

Das *mooresche Gesetz* besagt, dass sich die Komplexität integrierter Schaltkreise mit minimalen Komponentenkosten regelmäßig verdoppelt; die Zeitspanne variiert zwischen 12 und 24 Monaten. Das ist zwar mehr eine Faustregel als ein Naturgesetz, aber wenn Sie sich die Geschichte der Mikroprozessoren anschauen, wird diese These eindrucksvoll belegt. Die Entwicklung verlief rasant von Prozessoren mit ca. 8.000 Transistoren Anfang der 1970er-Jahre bis zu heutigen Bausteinen mit in der Spitze nahezu 10.000.000.000 Transistoren.

Nutznieser dieser Entwicklung ist nicht nur der industrielle Bereich, in dem integrierte Schaltkreise heute nahezu überall Maschinen und Geräte steuern oder zur Verarbeitung von Daten eingesetzt werden. Auch im privaten Consumerbereich sind hochintegrierte Schaltkreise allgegenwärtig und unverzichtbar. Die Miniaturisierung und die Preisentwicklung sind inzwischen so weit fortgeschritten, dass bereits Kleidungsstücke (sogenannte *Wearables*) mit Schaltkreisen ausgestattet werden.

Vielfach genügen industriell gefertigte und angebotene Gegenstände jedoch nicht privaten, ganz spezifischen Anforderungen, wenn Sie etwas Eigenes entwickeln und bauen möchten. Dies betrifft auch den weiten Bereich der Elektrotechnik und Elektronik, in dem Enthusiasten aus purem Interesse für die Sache an sich anspruchsvolle Projekte verwirklichen oder mit den jeweils verfügbaren Bauteilen ganz eigene Lösungen für sehr individuelle Probleme realisieren. Allerdings haben sich im Laufe der Jahre bzw. Jahrzehnte die Projekte, die dafür verwendeten Materialien und die Vorgehensweisen entschieden verändert. Stand vor Jahren noch z. B. der Bau eines Verstärkers oder eines Kurzwellenempfängers im Vordergrund, sind dies heute Dinge, die eher dem Bereich Hausautomation oder Robotik zuzuordnen sind. Dementsprechend haben sich auch die verwendeten Arbeitsmaterialien – wie Röhren, mächtige Transformatoren oder dicke Kondensatoren – verändert und sind heute vielfach zugunsten von Mikrocomputern (z. B. dem *Raspberry Pi* und seinen Derivaten) oder Mikrocontrollern (wie etwa der *Arduino-Plattform* oder dem *ESP32*) in den Hintergrund getreten. In der Folge sind die Anforderungen an das Wissen und die Fähigkeiten der Maker gestiegen, denn ohne ein gewisses Maß an Programmierkenntnissen lässt sich diesen Bauteilen kein Leben einhauchen.

Obwohl sich die beiden Bauteilklassen Mikrocomputer und Mikrocontroller in ihren Anwendungsbereichen häufig überlappen, lässt sich dennoch eine grobe Unterscheidung treffen. Ein *Mikrocomputer* mit einem Mikroprozessor als zentraler Recheneinheit wird häufig mit einem Kleinrechner assoziiert, der – zwar mit deutlich abgespeckter Leistung – die wesentlichen Aufgaben eines normalen PCs abdecken kann.

Der *Mikrocontroller* hingegen ist auf die Bewältigung technischer Anforderungen zugeschnitten. Er tritt im Alltag häufig in Gestalt von sogenannten eingebetteten Systemen in Gebrauchsartikeln wie Staubsaugern, Fernbedienungen, Motorsteuerungen und vielem mehr auf. In Ihrem Haushalt werden Sie wahrscheinlich Hunderte Mikrocontroller finden, wenn Sie genau hinschauen.

Für beide Bauteilklassen ist im Handel eine sehr große Anzahl verschiedener Chips erhältlich, die in einer noch größeren Anzahl an Modulen und Boards verbaut sind. Für sich genommen, ist diese Hardware aber kaum sinnvoll einsetzbar. Richtig interessant wird es erst, wenn diese Bauteile mit anderen Komponenten, z. B. *Sensoren* oder *Schaltern*, verbunden werden und Informationen mittels entsprechender *Protokolle* über das *World Wide Web* austauschen.

Ob nun ein Mikrocontroller oder ein Mikrocomputer für die anstehenden Vorhaben die beste Wahl ist, kann an dieser Stelle nicht beantwortet werden. Der Schwerpunkt dieses Buches ist ein Mikrocontroller, der in einem Modul der ESP32-Baureihe enthalten ist. Bei Aufgabenstellungen, in denen der Mikrocontroller auf »Unterstützung« angewiesen ist (z. B. Datenspeicherung), zeige ich realisierbare Lösungswege auf und erläutere sie.

Der Ansatz des Buches

Dieses Buch ist keine schrittweise Einführung in ein spezielles und abgegrenztes Anwendungsgebiet des ESP32 (z. B. Robotik, Hausautomation oder Internet of Things). Sie werden also kein durchgehendes Projekt finden, das für Sie sehr wahrscheinlich gar keine Relevanz hätte, denn Sie können mit dem ESP32 sehr unterschiedliche Dinge realisieren. Stattdessen möchte ich Ihnen im Stil eines Handbuchs möglichst umfassend verschiedene Aspekte darstellen. Ich möchte auf diese Weise zum Experimentieren anregen und Ihnen den Einstieg auch in komplexere Vorhaben ermöglichen, die dann eingehendere Kenntnisse voraussetzen.

Deshalb erhalten die Themen Entwicklungsumgebungen, Programmierertechnik und die Programmiersprachen C/C++/Arduino IDE einen entsprechenden Stellenwert und Raum. Dabei beschränke ich mich nicht darauf, Beispielprojekte nur in einer Entwicklungsumgebung oder einer Programmiersprache vorzustellen. Im Sinne eines möglichst breiten Ansatzes greife ich Alternativen auf und erläutere diese praxisgerecht. Das soll allerdings nicht zulasten von einfachen und auch komplexeren Anwendungsbeispielen gehen.

Neben den Erklärungen spielen Abbildungen eine tragende Rolle. Getreu dem Motto, dass Bilder mehr als tausend Worte sagen, lässt sich mit ihnen trotz des Platzes, den Bilder nun einmal einnehmen, die zu vermittelnde Botschaft einfach besser darstellen.

Für wen das Buch gedacht ist

Das Buch richtet sich an alle,

- ▶ die sich die Möglichkeiten eines ESP32 zunutze machen möchten,
- ▶ die breit angelegte Informationen rund um den Mikrocontroller ESP32 suchen,
- ▶ die neugierig sind und gerne einmal »über den Tellerrand schauen«,
- ▶ die mit einfachen Mitteln schnell loslegen möchten,
- ▶ die nicht nur einen Einstieg in ein ausgewähltes Anwendungsgebiet von Mikrocontrollern suchen, sondern das gesamte Leistungsspektrum vorgestellt bekommen möchten,
- ▶ die es begrüßen, ausgewählte Techniken der Softwareentwicklung und Programmierung vorgestellt zu bekommen,
- ▶ die gerne eigene Projekte unabhängig von industriellen Produkten entwickeln möchten und
- ▶ die auch gerne neben der komfortablen Arduino-Welt in speziellen Beispielen in die etwas komplexere Entwicklungsumgebung der Firma Espressif eintauchen möchten.

Der Inhalt des Buchs

Dieses Buch soll Ihnen eine umfassende Unterstützung dabei bieten, den Mikrocontroller ESP32 zum Herz einer eigenen Schaltzentrale zu machen oder ihn zumindest für ausgesuchte Projekte einzusetzen (siehe Abbildung 1).

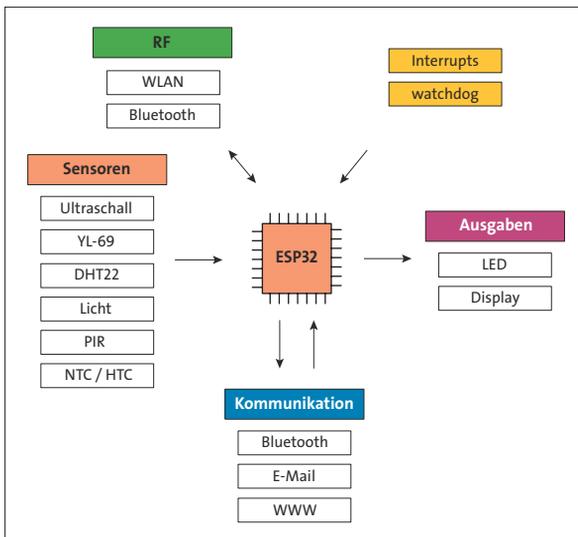


Abbildung 1 Der ESP32 als zentrales Steuerelement

Deshalb sind die Kapitel des Buches in ihrer Reihenfolge zunächst so angeordnet, dass sie Sie von Grund auf in die Welt des ESP32 einführen. Dennoch ist es auch möglich, einzelne Themenbereiche gezielt auszuwählen und für aktuelle Arbeiten heranzuziehen.

► **Kapitel 1: Der Mikrocontroller ESP32**

Der Fokus von Kapitel 1 liegt auf der Hardware ESP32 und der Erstinbetriebnahme an einem PC.

► **Kapitel 2: Die Stromversorgung**

Den Schwerpunkt dieses Kapitels bildet ein Blick auf die verschiedenen Möglichkeiten der Stromversorgung.

► **Kapitel 3: Ihre Werkstatt**

Dieses Kapitel behandelt gebräuchliche Werkzeuge und Softwaretools rund um den Mikrocontroller.

► **Kapitel 4: Entwicklungsumgebungen**

Hier lernen Sie unterschiedliche Möglichkeiten kennen, den Mikrocontroller zu programmieren.

► **Kapitel 5: Programmierung mit C und C++**

Thema dieses Kapitels sind grundlegende Informationen zu den Programmiersprachen C und C++.

► **Kapitel 6: ESP32-Programming inside**

Lernen Sie die Programmierung des ULP-Coprozessors, die Verwaltung von FreeRTOS-Tasks und die Thread-Funktionalitäten kennen.

► **Kapitel 7: Peripherie und Funktionen des ESP32**

Dieses Kapitel befasst sich mit Schnittstellen zu Sensoren und Aktoren.

► **Kapitel 8: Externe Schnittstellen**

In diesem Kapitel geht es um die Kommunikation mit externen Schnittstellen.

► **Kapitel 9: Externe Kommunikations- und Steuerungsmöglichkeiten**

Dieser Teil widmet sich Bluetooth und WLAN sowie den umfangreichen Möglichkeiten der externen Kommunikation und Steuerung, die sich durch diese Techniken eröffnen.

► **Kapitel 10: Projektideen**

In diesem Kapitel stelle ich komplexe und weniger komplexe Projektideen vor und möchte Sie zum Nachbauen anregen.

► Im **Anhang** finden Sie Übersichten zu Sensoren, Fritzing-Parts, verwendete Software und Softwareversionen sowie die RTC-MUX-Pin-Liste.

Hinweis für Schnelleinsteiger

Sie möchten schnell ein erstes Erfolgserlebnis haben? Kein Problem. Schließen Sie Ihr ESP32-Dev-KitC V4 an Ihren PC an, wie es in Abschnitt 1.6 beschrieben wird. Folgen Sie dann der Beschreibung in Abschnitt 4.4 zur Einrichtung der Entwicklungsumgebung und eines Blink-Programms.



Besondere Formate

Verschiedene Objekte werden in einer vom üblichen Text abweichenden Form dargestellt:

- ▶ */Pfad/Zu/Einer/Datei* kennzeichnet einen Dateipfad.
- ▶ Neu verwendete Begriffe, wie z. B. *Entwicklungsumgebung*, werden kursiv dargestellt.
- ▶ Codeelemente, wie z. B. `delay()`, haben ebenfalls eine eigene Schriftart.

Noch eine dringende Bitte vorab

Der ESP32, die Zusatzhardware, die Entwicklungsumgebungen und die Programm-bibliotheken unterliegen einer unglaublich rasanten Entwicklung. Dinge, die gestern noch nicht möglich waren, bereiten heute vielleicht keine Schwierigkeiten mehr oder funktionieren morgen völlig anders. Wenn es Probleme gibt, ärgern Sie sich bitte nicht – und weisen Sie den Verlag entsprechend auf sie hin.

Vielen herzlichen Dank

Udo Brandes

