

FAQ

Espressif ESP32

Antworten auf die häufigsten Fragen

Von Merlin Schumacher

Was ist der ESP32?

! Die ESP-Mikrocontroller-Familie ist für viele Aufgaben besser geeignet als etwa der Raspberry Pi oder Arduinos. Der erfolgreiche ESP8266 hat mit dem ESP32 inzwischen einen leistungsfähigen Nachfolger bekommen.

Einkaufsempfehlung

? Welches Board empfiehlt sich für den Einstieg? Welche Zusatzhardware brauche ich zum Programmieren?

! Das kommt auf Ihre Wünsche an. Die meisten ESP32-Entwicklerboards sind für die ersten Schritte gut geeignet und unterscheiden sich preislich nur minimal. Einige Boards bringen eine Ladeelektronik und einen Anschluss für Li-Ion-Akkus mit. Für ein paar Euro mehr gibt es noch Batteriefächer und kleine Displays dazu. Wie so oft bei derart günstiger Hardware schwankt die Qualität der Produkte. Was es beim ESP nicht gibt, ist eine im Hintergrund stehende Organisation wie etwa die Arduino LLC. Das ESP32-Modul wird nur vom Hersteller Espressif gebaut und vertrieben.

Zusatzhardware brauchen Sie bei den Entwicklerboards nicht. Diese haben einen Micro-USB-Anschluss, über den sie mit dem PC verbunden werden. Wenn Sie jedoch ein ESP32-Modul ohne Zusatzplatine gekauft haben, brauchen Sie einen Programmieradapter. Darauf befindet sich dann der USB-Anschluss.

ESP32 als Arduino-WLAN-Adapter

? Ich verwende Arduinos in Kombination mit ESP8266-Boards für eigene Projekte mit WLAN. Kann ich den ESP32 auch als Verbindungsbrücke verwenden?

! Das ist technisch kein Problem, denn Espressif stellt eine passende AT-

Firmware bereit, mittels derer der Arduino Daten mit dem ESP32 austauschen kann. Effektiver und kostengünstiger ist es vielleicht, die Arduinos durch ESPs zu ersetzen. Das gilt aber nur, wenn die angeschlossene Elektronik mitspielt, denn die ESPs arbeiten mit 3,3 Volt. In der Arduino-Welt sind hingegen 5 Volt üblich. An der Programmierung ändert sich oft nicht viel, aber man gewinnt Flexibilität zu einem günstigeren Preis.

Bestehende ESP8266-Projekte

? Kann ich bestehende ESP8266-Projekte auch auf dem ESP32 nutzen?

! Nutzt ein Projekt nur Arduino-Standardfunktionen ohne irgendwelche Besonderheiten, kann das klappen. Die meisten Projekte verwenden aber externe Bibliotheken, sodass dann Anpassungen für den ESP32 notwendig sind. Schauen Sie daher nach, ob die Autoren der ESP8266-Bibliotheken oder -Projekte Ambitionen haben, Unterstützung für den ESP32 einzubauen.

Unbekanntes OTA-Passwort

? Wenn ich ein Over-the-Air-Update einspielen will, fragt mich die Arduino IDE nach einem Passwort. Ich habe aber keins vergeben.

Mithilfe von Plug-ins kann man sowohl die Fehlermeldungen des ESP entschlüsseln, als auch Dateien auf seinen Flash-Speicher schreiben.

! Lassen Sie das Passwort einfach leer. Die IDE fragt immer danach. Das Update funktioniert trotzdem.

Hardware-Upgrade

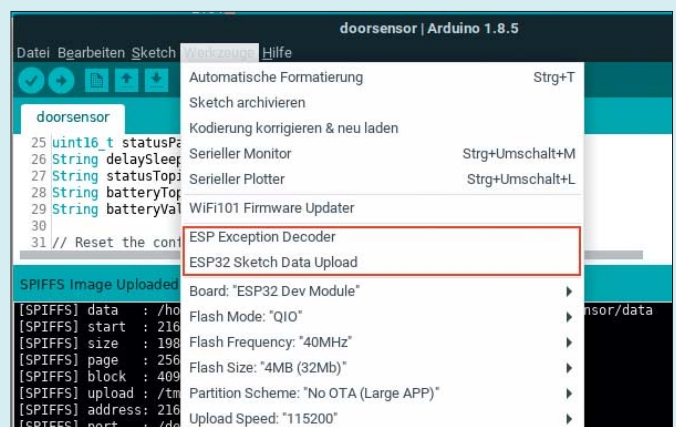
? Lohnt es sich, ESP8266-Geräte auf ESP32 umzurüsten?

! In den meisten Fällen nicht. Wenn der Code eines Projektes droht, die Speicher- oder Leistungsgrenzen des ESP8266 zu sprengen oder eine Erweiterung um Bluetooth sinnvoll erscheint, schon eher. Selbst bei Neuentwicklungen ist ein ESP8266-Board manchmal doch die bessere Wahl, denn es gibt mehr bestehende Software-Projekte und die Boards sind in kleineren Bauformen verfügbar.

WLAN-Reichweite

? Ich will meinen ESP32 im Keller aufstellen. Der WLAN-Empfang ist dort aber nicht besonders gut. Was kann ich tun?

! Der zuverlässigste Weg ist der Kauf eines WLAN-Repeater. Sie können auch ein ESP32-Modul mit einem Anschluss für eine externe Antenne nehmen (ESP32-WROOM-32U). Das ist günstiger als ein WLAN-Repeater. Garantieren kann man leider nichts, denn schlussendlich kommt es auf die Beschaffenheit Ihrer



Räumlichkeiten an. Es gibt auch ESP32-Boards mit Ethernet-Anschluss. Die sind jedoch recht teuer.

Board mit fehlenden Pins

? Das von mir gekaufte ESP32-Entwicklerboard hat zweimal den Pin 23, dafür fehlt der Pin 32. Wieso?

! Einige Boards haben anscheinend Bedruckungsfehler. Um zu überprüfen, welche Nummer für die fraglichen Pins wirklich korrekt ist, schließen Sie eine LED mit Vorwiderstand (100 Ohm) an und lassen Sie den Port blinken. Den dafür notwendigen Code finden Sie in den Beispielen der Arduino IDE unter „Blink“.

ESP zurücksetzen

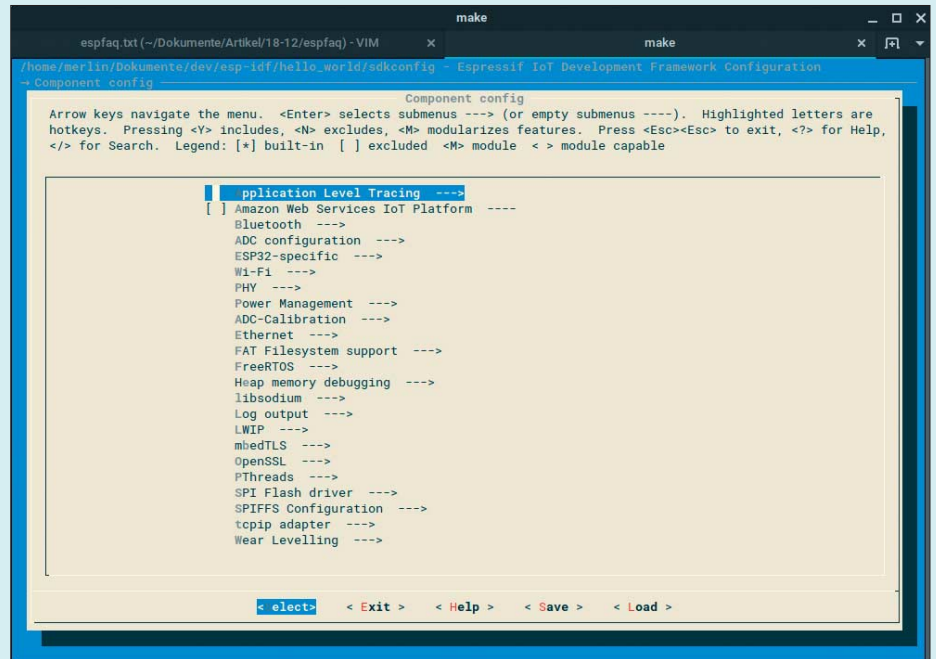
? Ich möchte meinen ESP komplett löschen. Wie geht das?

! Dafür gibt es das Programm Esptool. Es liegt dem ESP-IDF (IoT Development Framework) und damit auch dem Arduino Core for ESP32 bei. Verbinden Sie zunächst den ESP32 mit Ihrem PC. Anschließend wechseln Sie in einer Kommandozeile in das Unterverzeichnis `tools` des ESP-IDF beziehungsweise des Arduino Core. Geben Sie je nach Betriebssystem `esptool erase_flash` (Linux und macOS) oder `esptool.exe erase_flash` (Windows) ein. Falls das Programm den falschen seriellen Port erwischt, geben Sie mit dem Parameter `--port` den richtigen an. Der heißt unter Linux etwa `/dev/ttyUSB1`, unter Windows `COM2` und unter macOS beginnt er mit `/dev/cu`.

Sensoren im Deep-Sleep

? Kann der ESP32 auch im Deep-Sleep Sensoren auswerten?

! Ja, aber nur wenn Sie ihn mit dem IDF programmieren. Nur damit können Sie den ULP-Coprozessor des ESP32 programmieren. Das geht aber nur in Assembler. Daher muss man die Auswertung von Sensoren etwa per I²C auch vollständig in Assembler lösen. In den Quellen des IDF finden Sie zahlreiche Beispiele



Der ESP32 lässt sich dank des ESP-IDF auch ohne Arduino-Umgebung programmieren.

zur Auswertung von analogen, digitalen und I²C-Sensoren im Deep-Sleep.

Dateien flashen

? Kann ich eigene Dateien auf den internen Speicher des ESP32 flashen?

! Ja. Mit dem Arduino-IDE-Plug-in „ESP32 filesystem uploader“. Laden Sie das Plug-in herunter und entpacken Sie es in den Unterordner `tools` im Arduino-Arbeitsverzeichnis. Nach einem Neustart der IDE finden Sie unter Werkzeuge den Punkt ESP32 Sketch Data Upload. Nun erstellen Sie in Ihrem Projekt einen Unterordner namens `data` und legen Sie die Dateien dorthin, die auf den ESP32 geflasht werden sollen. Anschließend klicken Sie auf ESP32 Sketch Data Upload. Das Plug-in erzeugt dann ein Dateisystem-Image, welches mittels des Esptool geflasht wird. Beachten Sie jedoch, dass der Flashvorgang alle vorher bestehenden Dateien überschreibt.

Programmieren ohne Arduino

? Kann ich den ESP32 auch ohne die Arduino-IDE programmieren?

! Ja. Der Arduino Core für den ESP32 basiert auf Espressifs ESP-IDF (Espressif IoT Development Framework). Wenn Sie Erfahrung mit Software-Entwicklung unter Linux haben, dürfte Ihnen die dafür notwendige Arbeit mit Makefiles und GCC vertraut sein. Das IDF bringt auch ein paar Beispielprojekte zur Einarbeitung mit. Die Belohnung für den Verzicht auf den Komfort des Arduino Core sind weniger Platzverbrauch, mehr Möglichkeiten zur Anpassung von Features und tiefergreifende Kontrolle über den ESP32.

Linux für den ESP

? Ich möchte gerne einen Linux-Kernel auf dem ESP ausführen, da mir die Firmware-Programmierung nicht zusagt. Geht das?

! Nein. Der ESP32 ist trotz seiner Leistungsfähigkeit noch immer ein Mikrocontroller. Ihm fehlt es an Flash- und Arbeitsspeicher, außerdem müsste man noch größere Veränderungen am Linux-Kernel vornehmen, damit dieser auf den Prozessoren des ESP32 läuft. Das von Espressif verwendete Betriebssystem FreeRTOS ist eine erheblich bessere Wahl für einen solchen (relativ) simplen Mikrocontroller.

Was ist Basecamp?

Basecamp ist eine von c't entwickelte Arduino-Bibliothek für den ESP32. Sie vereinfacht Aufgaben wie das Herstellen einer WLAN-Verbindung, das Speichern von Konfigurationsdaten und die Verbindung zu einem MQTT-Broker. Dabei greift Basecamp auf weitere Bibliotheken für den ESP32 zurück und vereint diese zu einer Gesamtlösung. Die Konfiguration erfolgt über ein Web-Interface. Inzwischen haben sich zahlreiche Leser auf GitHub an der Entwicklung beteiligt. Den Link und mehr Informationen zum Projekt finden Sie über ct.de/yhug.

Basecamp oder nicht

? Brauche ich Basecamp für eigene ESP-Projekte?

! Nein. Basecamp ist lediglich eine Hilfe, um lästige Standardaufgaben zu delegieren und die Einrichtung zu vereinfachen. Sie können für einfache MQTT-Aufgaben genauso gut eines der Beispiele von Bibliotheken wie PubSubClient oder AsyncMQTTClient anpassen. Ähnlich verhält es sich mit anderen Bibliotheken.

Fehlermeldungen entschlüsseln

? Wie kann ich Fehlermeldungen bei Abstürzen entschlüsseln?

! Auch dafür gibt es ein Plug-in. Der Arduino ESP8266/ESP32 Exception Stack Trace Decoder kann Fehlermeldungen der beiden ESP-Varianten dekodieren. Die Installation erfolgt genauso wie beim ESP32 filesystem uploader. Voraussetzung dafür ist, dass Sie den Quellcode und Binärcode (in Form einer ELF-Datei) des zu debuggenden Programms vorliegen haben. Eine Fehlermeldung eines fremden Projektes, das jemand anderes übersetzt hat, können Sie nicht dekodieren, insbesondere, wenn Sie unterschiedliche Versionen von Bibliotheken verwenden.

Wenn Sie einen Absturz inklusive Backtrace-Meldung im seriellen Monitor

der Arduino-IDE angezeigt bekommen, kopieren Sie die Meldung vollständig und fügen Sie sie in den Exception Stack Trace Decoder ein. Er zeigt Ihnen dann die Stelle im Code, an der Sie nach der Ursache des Fehlers suchen müssen.

Basecamp will nicht kompilieren

? Ich habe Probleme beim Kompilieren von Basecamp. Der Fehler bezieht sich auf Dateien, die mit den Over-the-Air-Updates im Zusammenhang stehen. Was mache ich falsch?

! Anscheinend wird bei einigen Usern eine defekte Version des Arduino Core heruntergeladen. Ersetzen Sie die vom Compiler bemängelten Dateien manuell durch die auf GitHub verfügbaren. Anschließend sollte das Kompilieren wieder klappen.

Basecamp-Master macht Probleme

? Ich habe Basecamp direkt von GitHub heruntergeladen beziehungsweise den Master geklont und dennoch habe ich Probleme beim Kompilieren.

! Bitte verwenden Sie nur die Fassungen, die Sie unter Releases auf Git-

Hub finden. Besser noch: Nutzen Sie den Bibliotheksmanager der Arduino-IDE, um Basecamp zu installieren. Dann bekommen Sie immer eine aktuelle stabile Fassung von Basecamp. Im Master landen oft Features, die noch nicht getestet wurden.

Basecamp-Dokumentation

? Außer den Artikeln in c't habe ich keine Dokumentation zu Basecamp gefunden. Ist da noch etwas in Arbeit?

! Bisher gibt es noch keine ausführliche Dokumentation. Wir arbeiten daran.

MQTT-Broker mit TLS

? Kann Basecamp auch verschlüsselte Verbindungen zum MQTT-Broker aufbauen?

! Leider noch nicht. Die von der Bibliothek AsyncMQTTClient verwendete TCP-Bibliothek beherrscht noch kein TLS mit dem ESP32. Sobald der Entwickler den Support umgesetzt hat, bekommt auch Basecamp Unterstützung dafür. Mit der alternativen MQTT-Bibliothek PubSubClient und dem WifiSecureClient als WLAN-Client-Objekt funktioniert eine verschlüsselte Verbindung zum MQTT-Broker. (m/s@ct.de)

Mit Basecamp wird es einfacher, eigene ESP32-Geräte einzurichten und zu verwalten.

