

Arbeiten mit dem Espressiv Risc V Board **Esp32-C6**

Ich habe mir die Make Sonderausgabe Risc-V SPECIAL gekauft, die einen Risc-V-Microcontroller ESP32-C6 (esp32-c6-devkitc-1) enthält.



Ich wollte versuchen, das Board unter PlatformIO mit dem arduino framework zu programmieren.

Außerdem wollte ich mich darin einarbeiten, das Board mit der Espressif ESP-IDF zu programmieren.

Das Setup für das Programmieren mit PlatformIo Arduino gestaltete sich zunächst schwierig, da in der PlatformIO IDE im August 2025 nur ESP-IDF für das Board unterstützt wurde.

Daher zunächst Hinweise für das ESP-IDF framework:

Zunächst wurde das ESP-IDF framework unter Visual Studio Code installiert, dann auch eigenständig mit dem Installer von Espressif. Mit diesem hat man dann eine speziell konfigurierte Shell für CMD und Powershell zur Verfügung. Wie man mit der eigenständigen Installation umgeht, ist in der Make im Artikel ‚Erste Schritte mit RISC-V‘ beschrieben.

Zum Arbeiten in VS-Code habe ich die Extensions ‚Platformio‘ und ‚PioArduino‘ deaktiviert, ESP-IDF dur ‚Enable‘ aktiviert.

Es fiel nun auf, dass in dem verwendeten Template die Flash Size für mein Board falsch gesetzt auf 2 Mbyte obwohl 4 Mbyte vorhanden waren.

Das kann in der Datei SDKConfig geändert werden, indem die Parameter

```
CONFIG_ESPTOOLPY_FLASHSIZE_4MB=y
```

Und

```
CONFIG_ESPTOOLPY_FLASHSIZE="4MB"
```

gesetzt werden

Nun Hinweise für **PlatformIO** und **arduino** framework:

Bei ersten Versuchen kam zuerst ein Fehler, besagend, dass das Python-Paket intelhex nicht installiert sei.

Dies wurde behoben, in dem das Paket in der PlatformIo spezifischen Python Implementierung wie folgt installiert wurde.

```
C:\Users\<Name>\.platformio\penv\Scripts\python.exe -m pip install intelhex
```

Dann musste noch eine aktuelle Python Version installiert werden und in der Umgebungsvariablen ‚path‘ eine Anpassung erfolgen, damit nicht immer eine alte Python Version verwendet wurde.

Erzeugung eines PlatformIo Arduino Projekts für **,esp32-c6-devkit-1‘**:

Da eine Vorlage für das esp32-C6 board nicht zur Verfügung steht, mussten die folgenden Tricks zur Anwendung kommen;

- 1) Es wird nicht die VS-Code Extension PlatformIo ([PlatformIO](#)), sondern **,PioArduino‘** ([pioarduino](#)) verwendet. Dies ist ein fork von PlatformIo, der speziell auf Esp32 Boards abzielt. PlatformIo sollte inaktiviert werden.
- 2) Unter **,PioArduino‘** Erzeugung eines Projekts für **,esp32-C3-devkit‘**, hierfür steht unter PlatformIo eine Vorlage zur Verfügung. Nachdem das Projekt erzeugt ist und sich kompilieren lässt, werden in der Datei platformio.ini Änderungen vorgenommen.

Vor allem musste (neben Änderung des Namens) platform = espressif32 durch

```
platform = https://github.com/pioarduino/platform-espressif32/releases/download/stable/platform-espressif32.zip
```

ersetzt werden

Vorher: [env:esp32-c3-devkitc-02] platform = espressif32 board = esp32-c3-devkitc-02 framework = arduino	Nachher: [env:esp32-c3-devkitc-1] platform = https://github.com/pioarduino/platform-espressif32/releases/download/stable/platform-espressif32.zip board = esp32-c6-devkitc-1 framework = arduino
---	---

Nachdem man die platformio.ini Datei gespeichert hat, konfiguriert die IDE das Projekt neu und verwendet unter user/.platformio/packages die Ordner **framework-arduinopressif32** und **framework-arduinopressif32-libs**. Will man die Original Platformio Dateien verwenden, kann man die beiden verzeichnisse löschen und das Project neu konfigurieren.

Unter **.vscode c_cpp.properties,json** ist der folgende Pfad für den zu verwendenden Compiler eingetragen.

"compilerPath": "C:/Users/Roland/.platformio/packages/toolchain-riscv32-esp/bin/riscv32-esp-elf-gcc.exe",

Weitere Links:

[mnowak32/platform-espessif32: Espressif 32: development platform for PlatformIO](#)

[ESP32-Code für alle Modelle kompilieren – PlatformIO macht's möglich!](#)