

## Projektstart mit Arduino und ESP8266

### Beschreibung

Für den erfolgreichen Projektstart mit der Arduino-Software und einem Mikrocontroller beim CoderDojo wird hier eine mögliche Vorgehensweise beschrieben um rasch zum gewünschten Erfolg zu kommen. Wir empfehlen bei jedem Projektstart diese Punkte durchzugehen. Siehe dir auch die Schnellstartanleitungen auf der CoderDojo Homepage an.

### Software

Es wird hier eine Liste von benötigten Programmen angeführt. Bei der Installation lasse dir von einem Erwachsenen helfen.

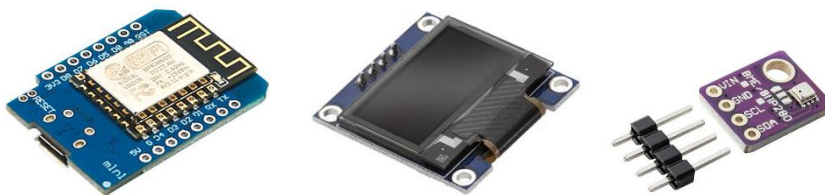
- Arduino: <https://www.arduino.cc/en/software>
- Notepad++: <https://notepad-plus-plus.org/downloads/v7.0/>
- Visual Studio Code und Platform iO: <https://code.visualstudio.com/> (nur für Profis)

Diese Programme können kostenfrei im Internet heruntergeladen und lizenzfrei benutzt werden.

### Hardware

Es wird hier eine Liste von benötigten Bauteilen angeführt. Beim Einkaufen lasse dir von einem Erwachsenen helfen. Siehe im Conrad – Shop nach.

- Grundausrüstung für das Elektronik Basteln (Steckboard, Widerstände, LED, Transistoren, Steck-Draht, ...)
- NodeMCU: ESP8266 Wemos D1 mini
- USB Kabel
- Display: OLED 0,96 Zoll I2C Display 128x64 Pixel
- Sensor: BME280 I2C Temperatur, Luftfeuchtigkeit, Luftdruck

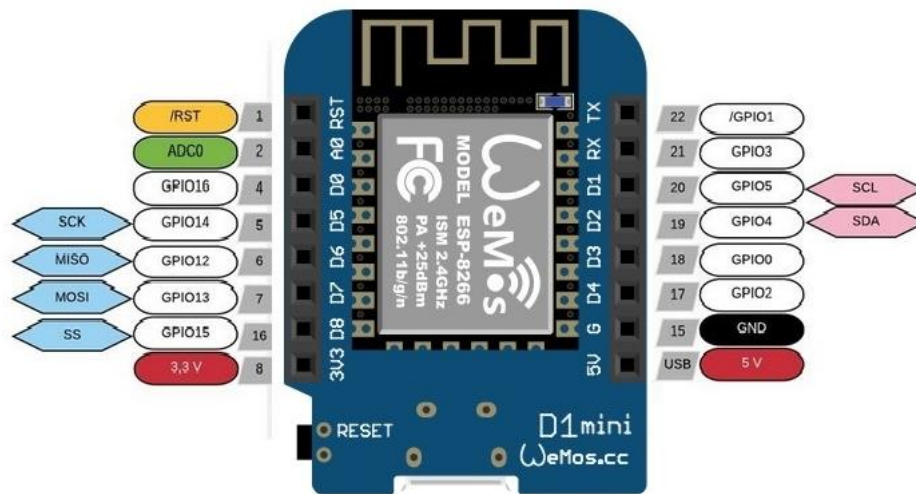


Es wird empfohlen gleich auf Vorrat zu kaufen, da der Preis günstiger wird und wenn etwas kaputt wird nicht gleich das Projekt zu Ende ist.

### Literatur Quellen

- <https://linz.coderdojo.net/uebungsanleitungen/loeten-und-elektronik/esp8266/>
- <https://www.arduino.cc/en/Tutorial/HomePage>
- <https://randomnerdtutorials.com/getting-started-with-esp8266-wifi-transceiver-review/>

## Pin-Belegung D1 mini

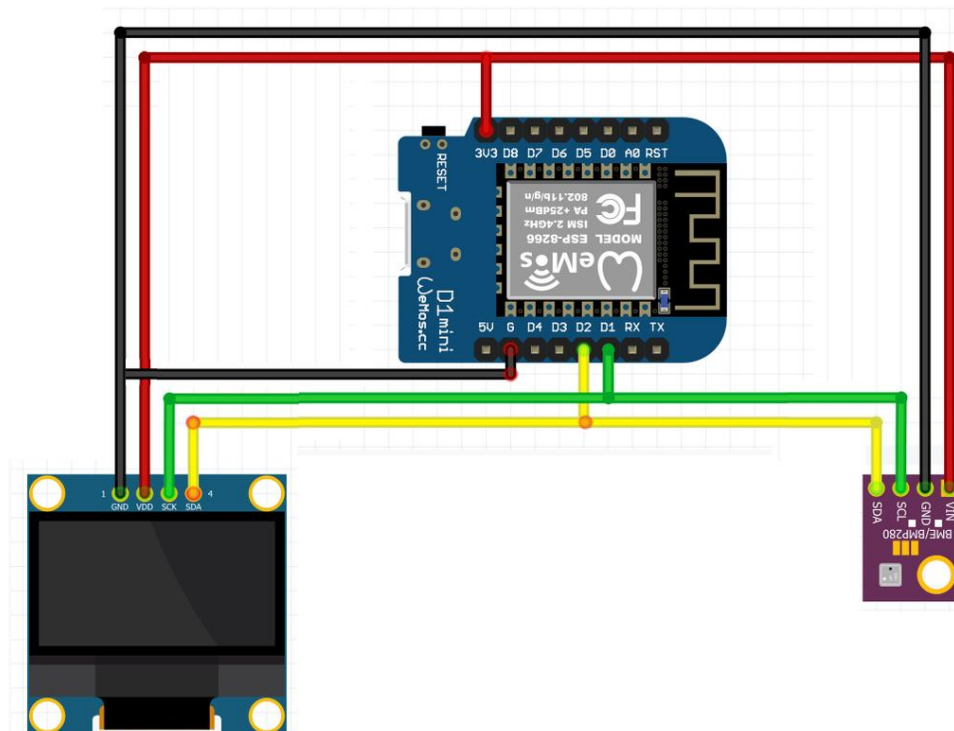


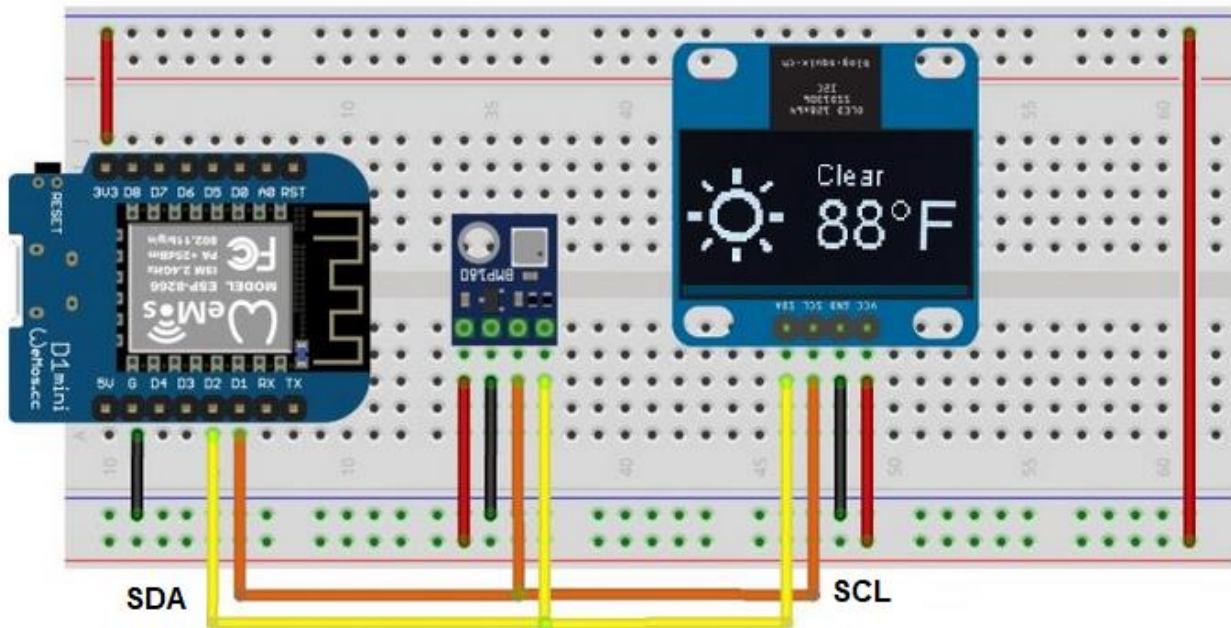
**Achtung: ESP ist nur 3,3V verträglich.**

Der ESP8266 ist nur 3,3V verträglich!! Datenblätter von AZ-Delivery.de herunterladen und beachten.

- Stromversorgung über USB-B Stecker
- 9 digitale I/O Pins (maximal 3,3V!)
- 1 analoger I/O Pins (maximal 3,0V!)
- ESP8266 Variante ESP-12F
- USB-Schnittstelle CH340G

## Schaltung Beispiel





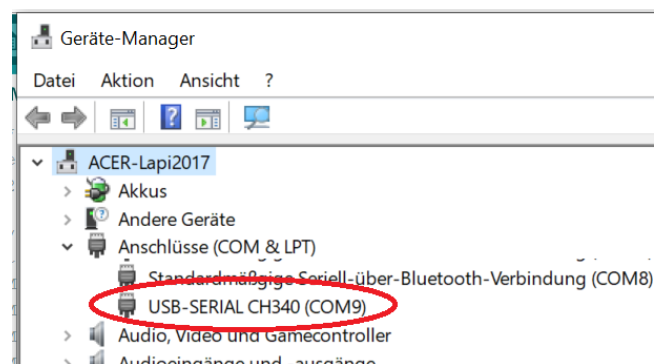
## Test

Nachdem die Schaltung aufgebaut und nochmals auf Richtigkeit geprüft ist, vor **jedem** Projektstart mit der Arduino-Software und einem Mikrocontroller immer einen Systemtest durchführen, um sicherzustellen, dass die Software-Plattform mit der Hardware richtig kommuniziert. Anderenfalls wird eine Fehlersuche sehr schwierig.

- Bei diesem Projekt wird die Vorgehensweise mit dem ESP8266 D1-mini gezeigt.
- Das System wird mit dem Arduino Beispielpogramm „Blink“ getestet.

## Treiberinstallation testen

- Den „Geräte-Manager“ unter Windows öffnen. Mit der rechten Maustaste auf das Windows-Symbol und im Menü auswählen.
- Das ESP-Board an den USB anstecken. Es muss ein Eintrag wie folgender auftauchen. Hier ist alles in Ordnung



- Wenn dieser Eintrag nicht erscheint, dann muss zuerst der USB-SERIAL CH340 Treiber geladen werden.

## Treiberinstallation (wenn notwendig)

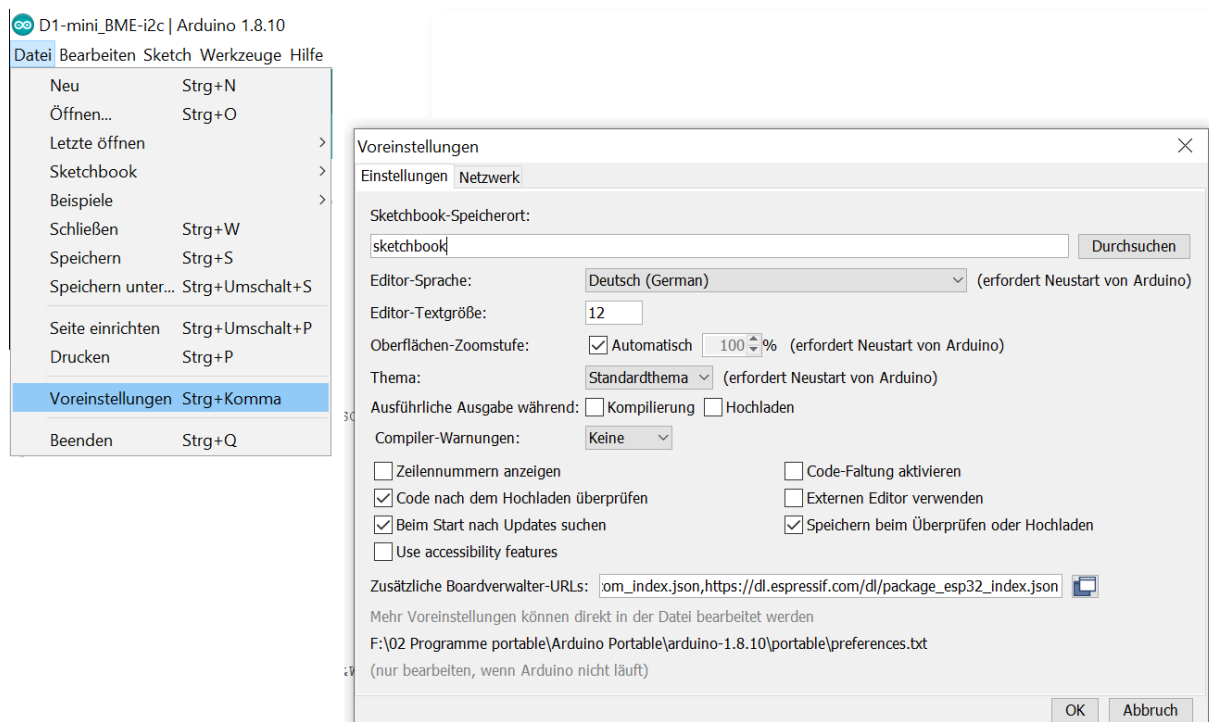
Treiber herunterladen

- Windows: [http://www.wch.cn/download/CH341SER\\_ZIP.html](http://www.wch.cn/download/CH341SER_ZIP.html)
- Mac: [http://www.wch.cn/download/CH341SER\\_MAC\\_ZIP.html](http://www.wch.cn/download/CH341SER_MAC_ZIP.html)

Unter Windows installierst du ihn einfach durch das Ausführen der "SETUP.EXE" im Ordner "CH341SER". Mac-Nutzer folgen am besten den Installationsanweisungen, die dem Treiberpaket beiliegen. Nach dem erneuten Anschließen des UNOs sollte dieser als "USB-SERIAL CH340"-Gerät (Windows) erkannt werden.

## Arduino Software

- Arduino Software starten
- Arduino Voreinstellungen prüfen



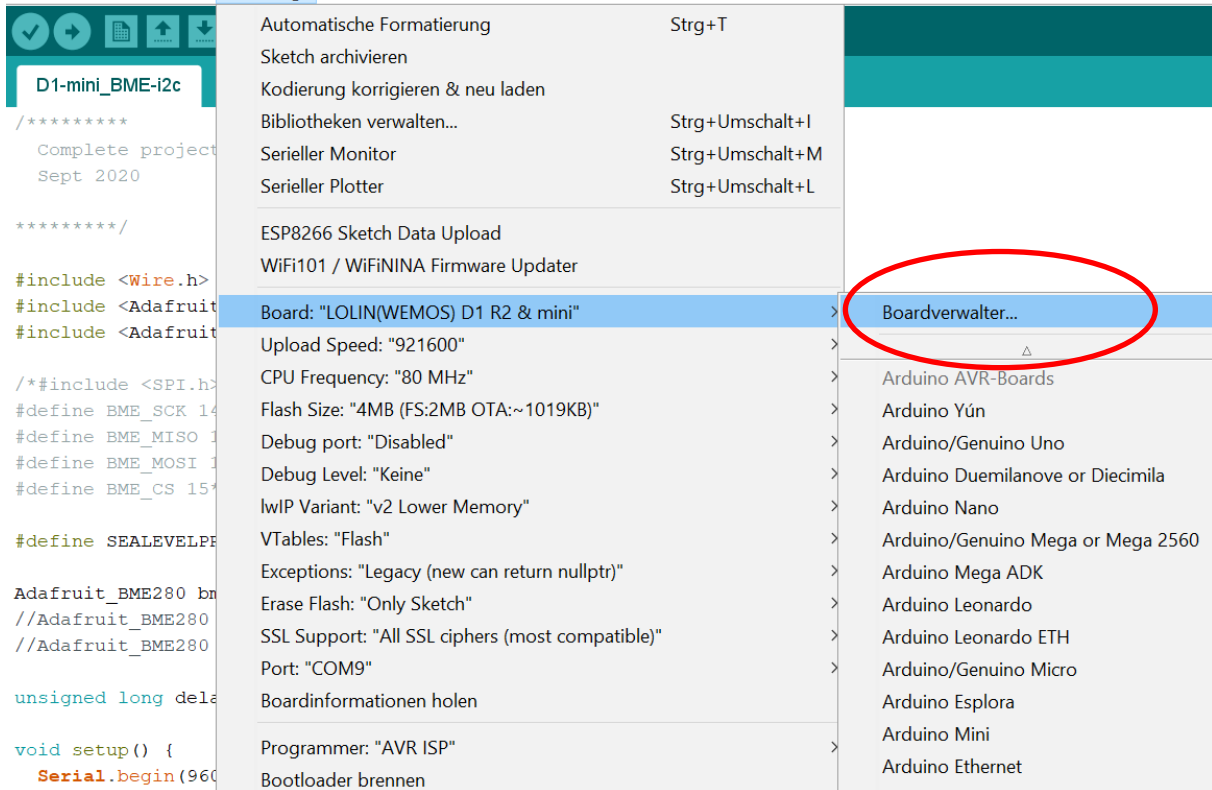
Zusätzliche Boardverwalter URL prüfen oder eintragen (Achtung auf Punkt und Beistrich).

[https://arduino.esp8266.com/stable/package\\_esp8266com\\_index.json](https://arduino.esp8266.com/stable/package_esp8266com_index.json),  
[https://dl.espressif.com/dl/package\\_esp32\\_index.json](https://dl.espressif.com/dl/package_esp32_index.json)

- Arduino „Neustart“ durchführen, sobald hier etwas geändert wurde.
- Einstellung im Menüpunkt „Werkzeuge“ - „Boardverwalter“ prüfen. Das dauert etwas bis der Boardverwalter vollständig geladen ist. Hier ist der Lolin (Wemos) D1 R2 mini bereits eingestellt. Es bleibt immer das zuletzt eingestellte Board ausgewählt.

☞ D1-mini\_BME-i2c | Arduino 1.8.10

Datei Bearbeiten Sketch **Werkzeuge** Hilfe



Automatische Formatierung Strg+T

Sketch archivieren

Kodierung korrigieren & neu laden

Bibliotheken verwalten... Strg+Umschalt+I

Serieller Monitor Strg+Umschalt+M

Serieller Plotter Strg+Umschalt+L

ESP8266 Sketch Data Upload

WiFi101 / WiFiNINA Firmware Updater

**Board: "LOLIN(WEMOS) D1 R2 & mini"**

Upload Speed: "921600"

CPU Frequency: "80 MHz"

Flash Size: "4MB (FS:2MB OTA:~1019KB)"

Debug port: "Disabled"

Debug Level: "Keine"

IWiP Variant: "v2 Lower Memory"

VTables: "Flash"

Exceptions: "Legacy (new can return nullptr)"

Erase Flash: "Only Sketch"

SSL Support: "All SSL ciphers (most compatible)"

Port: "COM9"

Boardinformationen holen

Programmer: "AVR ISP"

Bootloader brennen

**Boardverwalter...**

Arduino AVR-Boards

Arduino Yún

Arduino/Genuino Uno

Arduino Duemilanove or Decimila

Arduino Nano

Arduino/Genuino Mega or Mega 2560

Arduino Mega ADK

Arduino Leonardo

Arduino Leonardo ETH

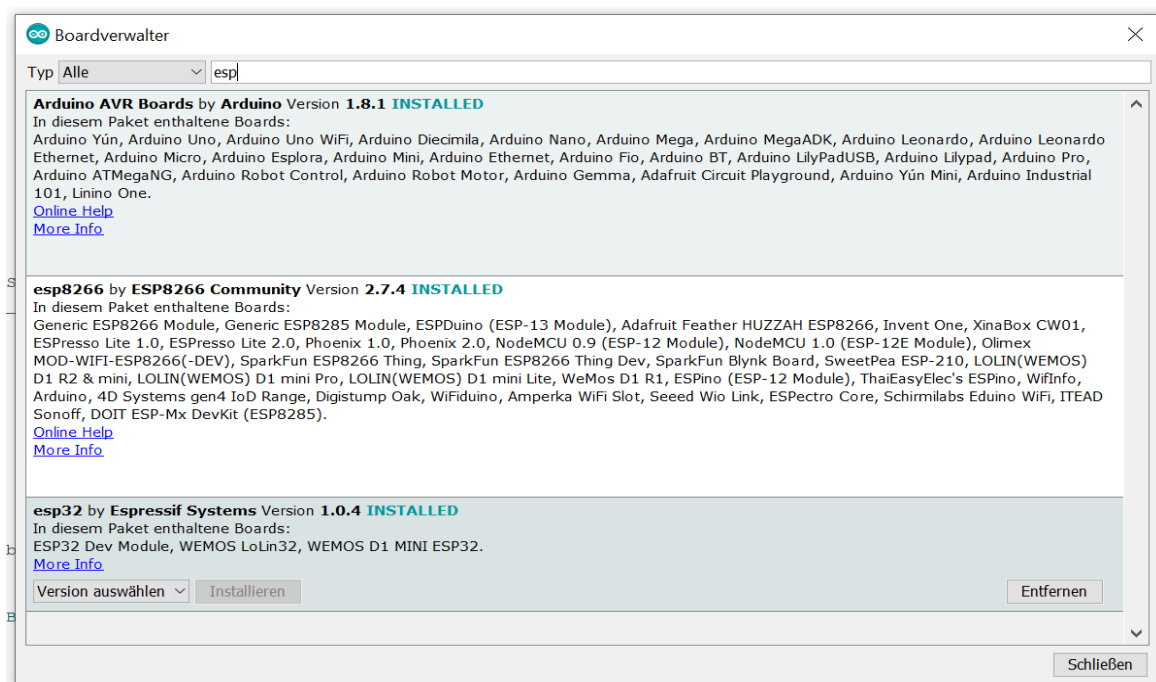
Arduino/Genuino Micro

Arduino Esplora

Arduino Mini

Arduino Ethernet

Zur schnelleren Auswahl kann „esp“ eingegeben werden.



Boardverwalter

Typ: Alle | esp

**Arduino AVR Boards by Arduino Version 1.8.1 INSTALLED**

In diesem Paket enthaltene Boards:

Arduino Yún, Arduino Uno, Arduino Uno WiFi, Arduino Decimila, Arduino Nano, Arduino Mega, Arduino MegaADK, Arduino Leonardo, Arduino Leonardo Ethernet, Arduino Micro, Arduino Esplora, Arduino Mini, Arduino Ethernet, Arduino Fio, Arduino BT, Arduino LilyPadUSB, Arduino LilyPad, Arduino Pro, Arduino ATmegaNG, Arduino Robot Control, Arduino Robot Motor, Arduino Gemma, Adafruit Circuit Playground, Arduino Yún Mini, Arduino Industrial 101, Linino One.

[Online Help](#)  
[More Info](#)

**esp8266 by ESP8266 Community Version 2.7.4 INSTALLED**

In diesem Paket enthaltene Boards:

Generic ESP8266 Module, Generic ESP8285 Module, ESPduino (ESP-13 Module), Adafruit Feather HUZZAH ESP8266, Invent One, Xinabox CW01, ESPRESSO Lite 1.0, ESPRESSO Lite 2.0, Phoenix 1.0, Phoenix 2.0, NodeMCU 0.9 (ESP-12 Module), NodeMCU 1.0 (ESP-12E Module), Olimex MOD-WIFI-ESP8266(-DEV), SparkFun ESP8266 Thing, SparkFun ESP8266 Thing Dev, SparkFun Blynk Board, SweetPea ESP-210, LOLIN(WEMOS) D1 R2 & mini, LOLIN(WEMOS) D1 mini Pro, LOLIN(WEMOS) D1 mini Lite, WeMos D1 R1, ESPino (ESP-12 Module), ThaiEasyElec's ESPino, Wifinfo, Arduino, 4D Systems gen4 IoT Range, Digistump Oak, WiFiduino, Amperka WiFi Slot, Seeed Wio Link, ESpectro Core, Schirmilabs Eduino WiFi, ITEAD Sonoff, DOIT ESP-Mx DevKit (ESP8285).

[Online Help](#)  
[More Info](#)

**esp32 by Espressif Systems Version 1.0.4 INSTALLED**

In diesem Paket enthaltene Boards:

ESP32 Dev Module, WEMOS LoLin32, WEMOS D1 MINI ESP32.

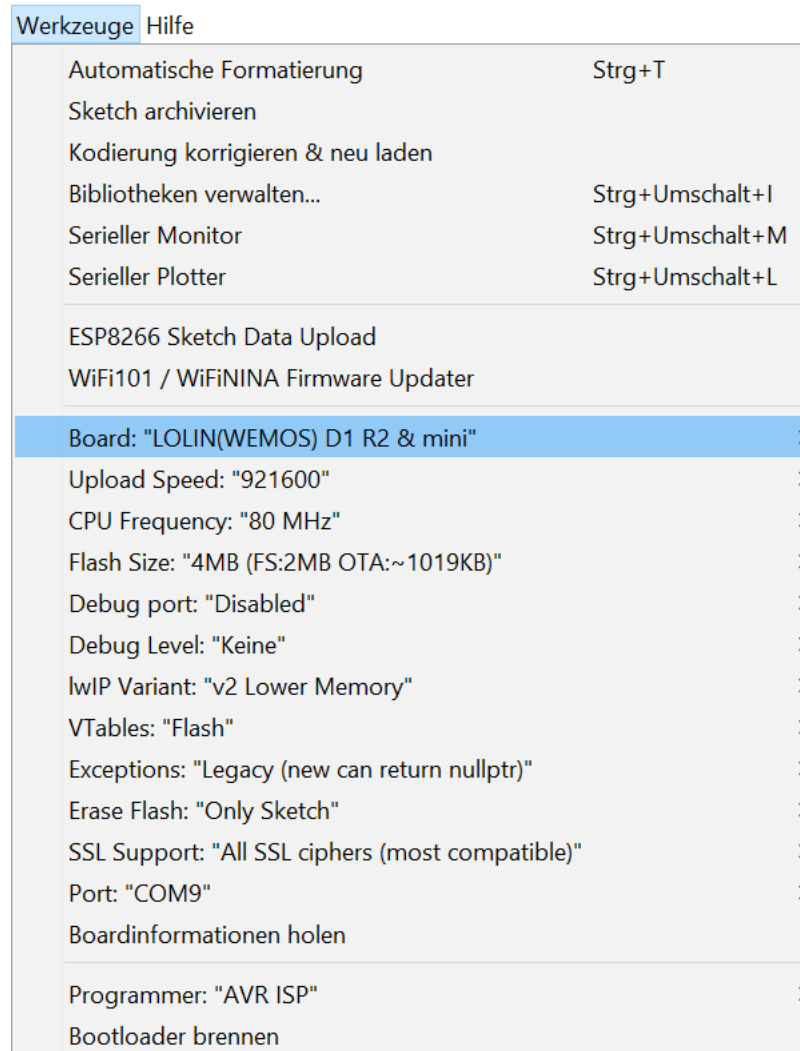
[More Info](#)

Version auswählen | Installieren | Entfernen

Schließen

Hier ist alles ok: esp8266 und esp32 ist installiert. Den Boardverwalter schließen und warten bis Menüpunkt Werkzeuge wieder ausgewählt werden kann.

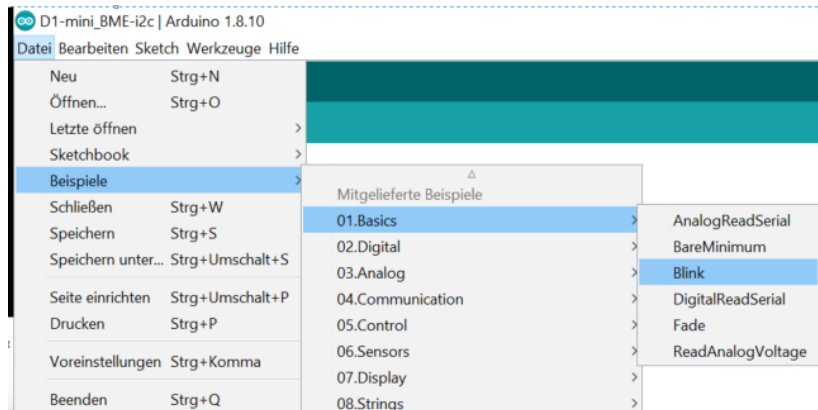
- Einstellung im Menüpunkt „Werkzeuge“ - „Board“ das verwendete Board auswählen.



- Im Arduino Menü „Werkzeuge“- „Port“ auf die entsprechende COMx einstellen.
- Hardware und Arduino Setting sind nun in Ordnung.

## Software-Installation

Das Beispielprogramm „Blink“ laden und auf den Mikrocontroller laden.



Beim Übertragen der Daten blinkt die blaue LED auf dem Board nervös. Sobald das Programm erfolgreich geladen wurde, startet es automatisch und die blaue LED blinkt im Sekundentakt.

Jetzt kann mit dem eigenen Projekt gestartet werden.