



Installation der Arduino IDE





| | |
|---|----|
| 1. Was ist die Arduino IDE | 3 |
| 2. Download der Arduino IDE | 5 |
| 3. Installation der Arduino IDE | 8 |
| 4. Sprach Einstellungen der Arduino IDE | 11 |
| 5. ESP und andere Boards Hinzufügen | 14 |
| 6. Probleme bei der Installation der Boards | 19 |



1. Was ist die Arduino IDE?

Die **Arduino IDE** ist ein Programm, mit dem Mikrocontroller programmiert werden. IDE steht für **Integrated Development Environment**, also eine integrierte Entwicklungsumgebung.

Vereinfacht gesagt ist die Arduino IDE der Ort, an dem:

- Programme geschrieben
- auf Fehler geprüft
- und auf einen Mikrocontroller übertragen werden

Wofür wird die Arduino IDE verwendet?

Mit der Arduino IDE erstellst du sogenannte **Sketches** – das sind kleine Programme, die festlegen, wie sich dein Mikrocontroller verhalten soll.

Beispiele:

- eine LED ein- oder ausschalten
- einen Taster abfragen
- eine Temperatur messen
- auf Ereignisse reagieren

Die Arduino IDE übernimmt dabei die komplette Kommunikation zwischen deinem Computer und dem Mikrocontroller.

Warum ist die Arduino IDE besonders einsteigerfreundlich?

Die Arduino IDE wurde speziell dafür entwickelt, den Einstieg in die Mikrocontroller-Programmierung zu erleichtern:

- kostenlos und plattformübergreifend (Windows, macOS, Linux)
- einfache, übersichtliche Oberfläche
- viele Beispiele sind bereits enthalten
- große Community und umfangreiche Dokumentation

Einsteiger können sich auf das Lernen konzentrieren, ohne sich mit komplizierten Werkzeugen beschäftigen zu müssen.



Welche Programmiersprache wird verwendet?

In der Arduino IDE wird hauptsächlich **C / C++** verwendet, allerdings in einer vereinfachten Form. Viele technische Details werden im Hintergrund automatisch erledigt.

Dadurch ist es möglich, schon mit wenigen Zeilen Code funktionierende Programme zu schreiben.

Wie arbeitet man mit der Arduino IDE?

Der typische Ablauf sieht so aus:

1. Arduino IDE starten
2. Mikrocontroller per USB anschließen
3. Programm schreiben oder ein Beispiel laden
4. Programm kompilieren (übersetzen)
5. Programm auf den Mikrocontroller übertragen

Nach dem Übertragen läuft das Programm selbstständig auf dem Mikrocontroller.

Zusammenfassung

Die Arduino IDE ist eine leicht verständliche Entwicklungsumgebung, mit der Programme für Arduino-Mikrocontroller erstellt und auf diese übertragen werden. Sie bildet die zentrale Schnittstelle zwischen Mensch, Computer und Hardware.



2. Download der Arduino IDE.

Lade die Installationsdatei der Arduino IDE auf der offiziellen Website herunter.

Link:

<https://www.arduino.cc/en/software/>

1. Besuche die Website.

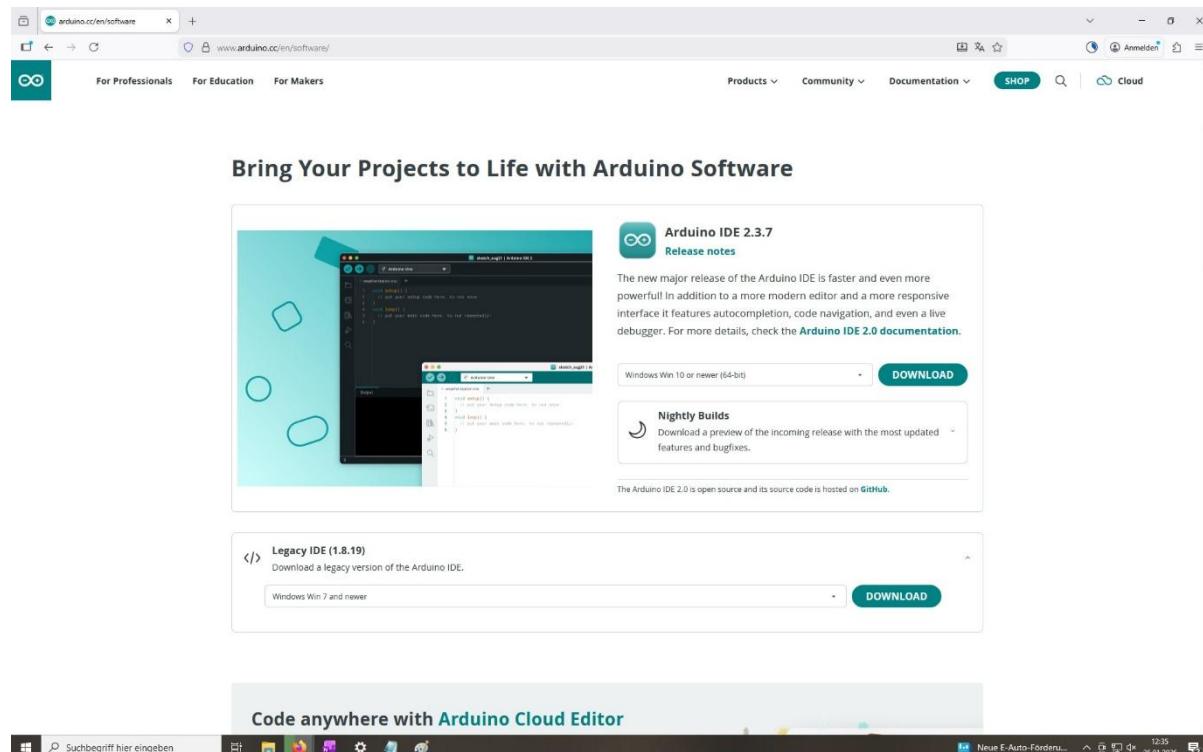


Abbildung 1

2. Wähle dein Betriebssystem.

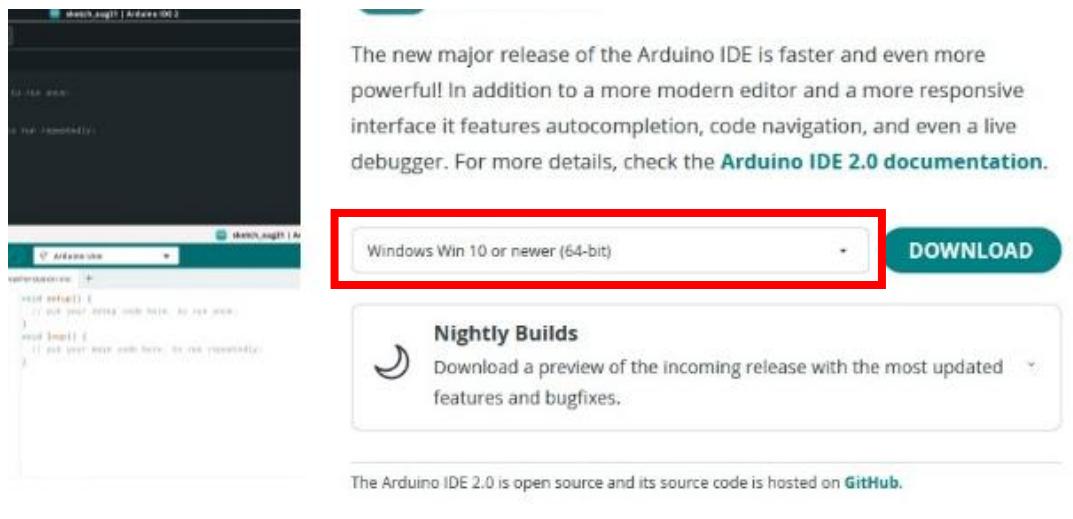


Abbildung 2

3. Klicke den Button „Download“ an.

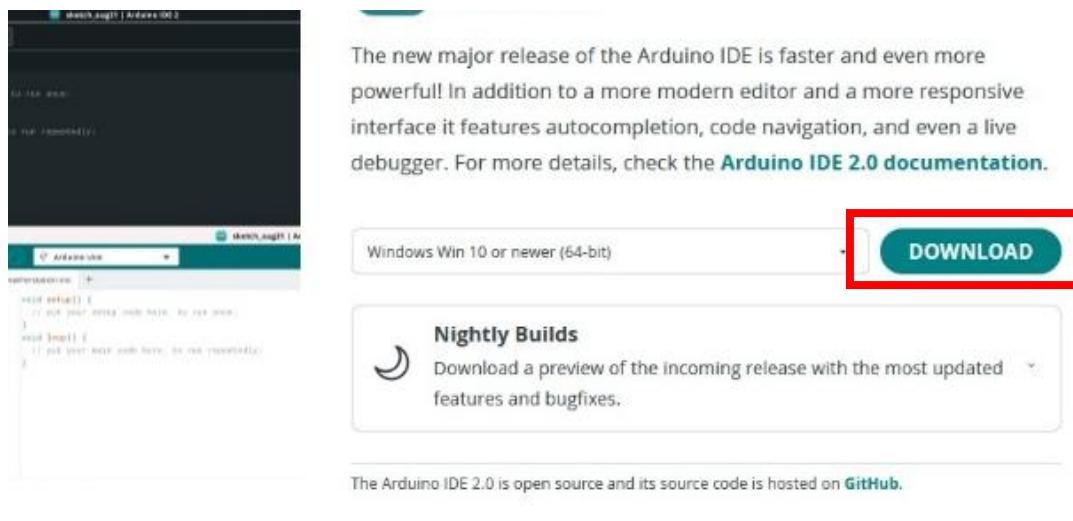


Abbildung 3



4. Der Download läuft.

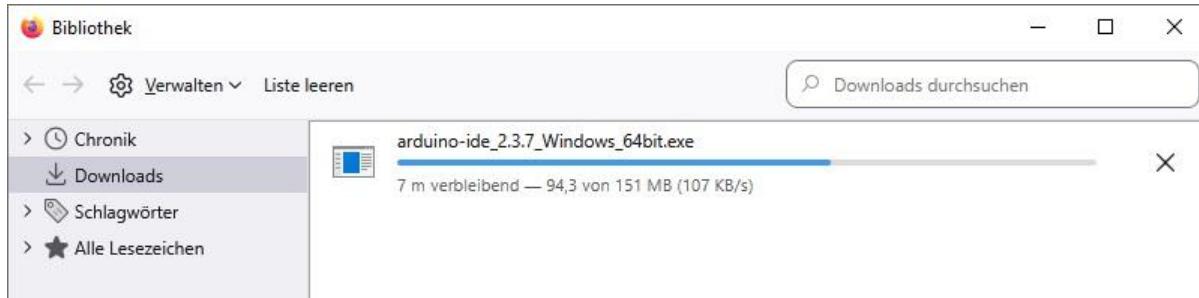


Abbildung 4

5. Wurde die Installationsdatei heruntergeladen, muss die Installation gestartet werden.

3. Installation der Arduino IDE.

1. Klicke auf die Installationsdatei „arduino-ide_xxx.exe“.



Abbildung 5

2. Die Installation wird gestartet, Akzeptiere das Lizenzabkommen durch Klicken auf den Button „Annehmen“.

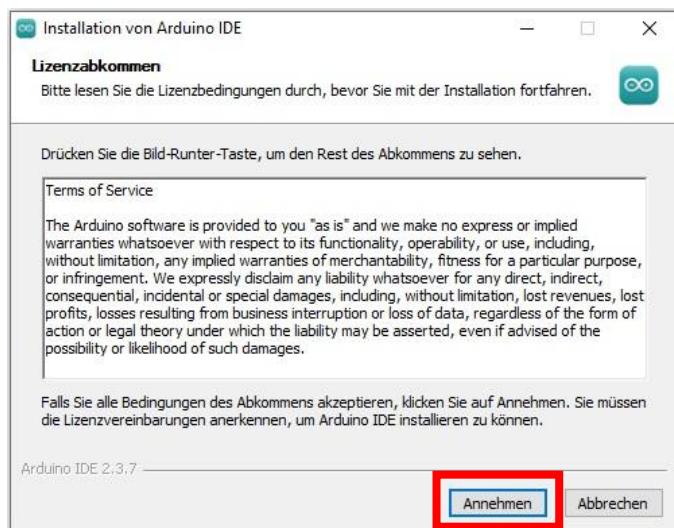


Abbildung 6

3. Installationsoption wählen, „Nur für Mich“ wird empfohlen. Danach Klicke auf den Button „Weiter“.

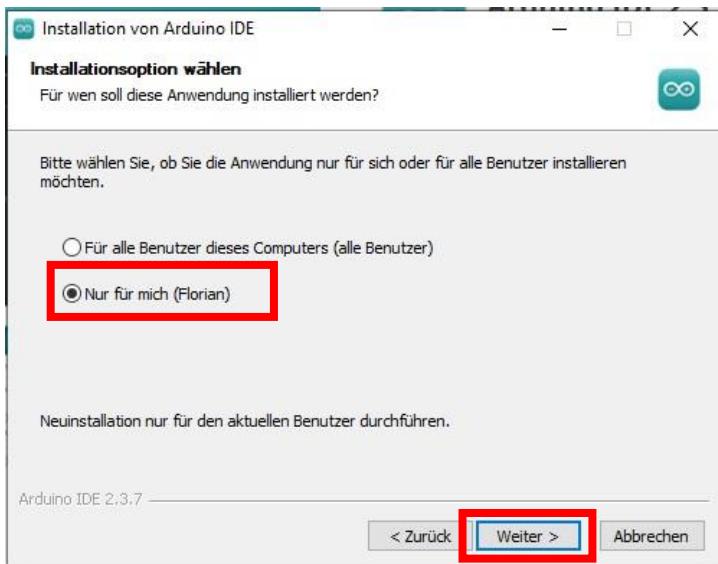


Abbildung 7

4. Installationsverzeichnis auswählen. Es wird empfohlen das vorgegebene Installationsverzeichnis zu behalten. Durch das Klicken auf den Button „Installieren“ wird die Installation ausgeführt.

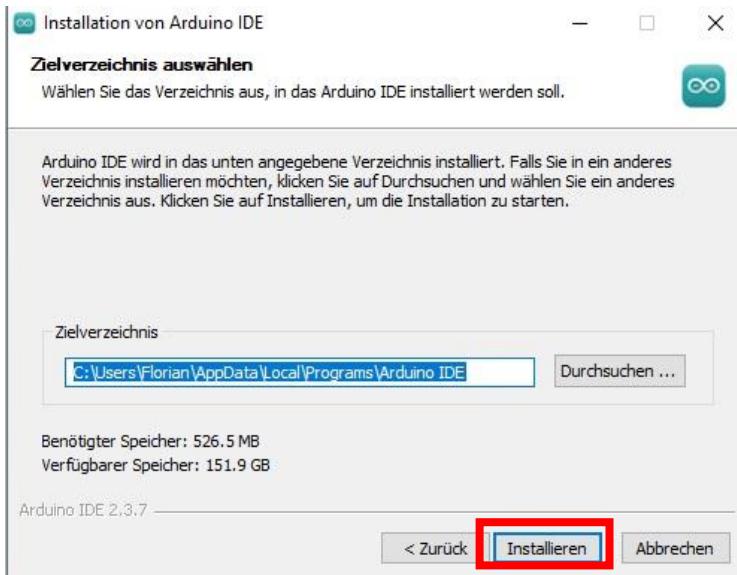


Abbildung 8

5. Die Installation wird ausgeführt. Anhand des grünen Balkens kann der Fortschritt der Installation beobachtet werden.

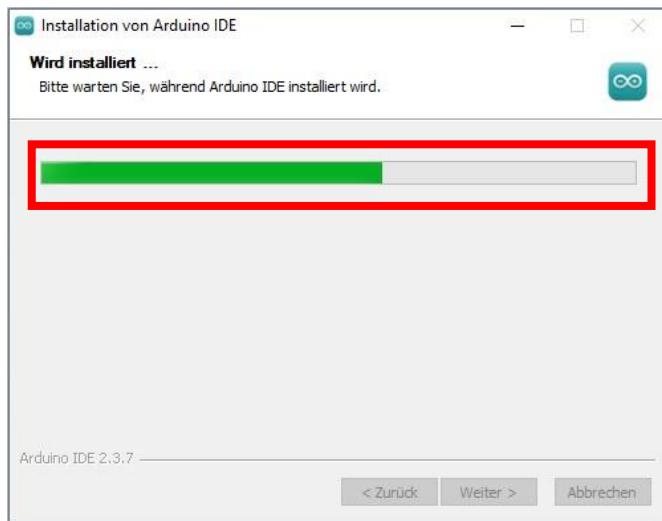


Abbildung 9

6. Durch das Klicken auf den Button „Fertigstellen“ wird die Installation abgeschlossen. Ist die Checkbox „Arduino IDE ausführen“ angewählt, wird nach dem Klicken auf den Button „Fertigstellen“ die Arduino IDE gestartet.



Abbildung 10

4. Sprach Einstellungen der Arduino IDE.

1. Starte die Arduino IDE durch das Klicken auf das Icon „Arduino IDE“

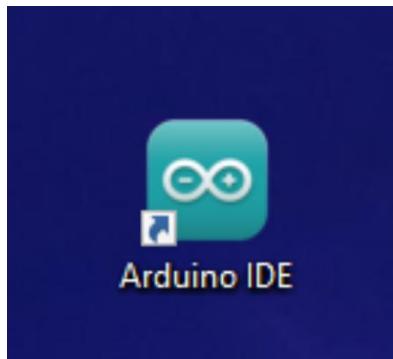


Abbildung 11

2. Die Arduino IDE startet. Nach der Installation kann der erste Start der Arduino IDE bis zu 2 Minuten dauern.

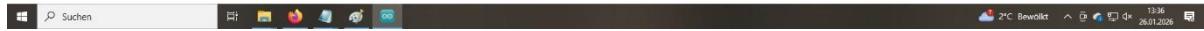


Abbildung 12

3. Öffne in der IDE das Menü „File“.

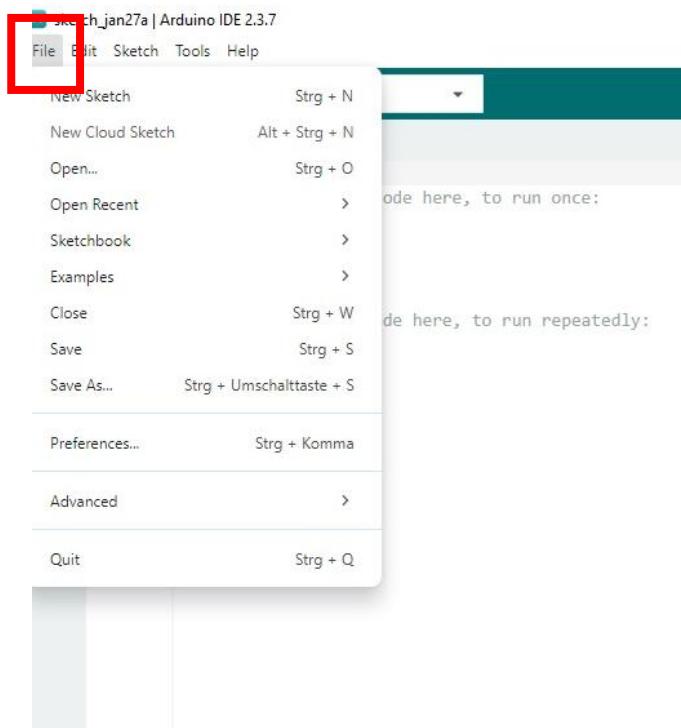


Abbildung 13

4. Klicke auf den Menüpunkt „Preferences“.

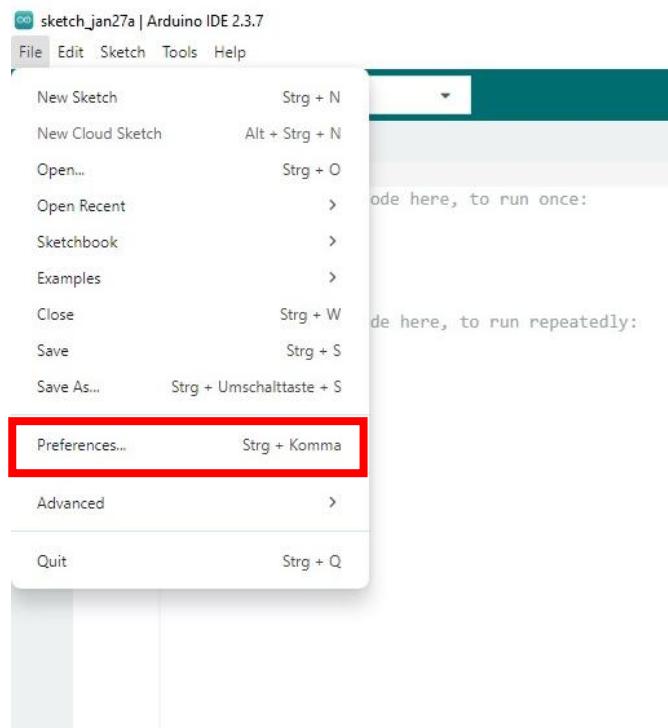


Abbildung 14

5. Wähle im Menüfenster „Preferences“ in dem Drop-down-Menü der Einstellungen „Language“ als Sprache: German und bestätige dies mit klicken auf den Button „OK“

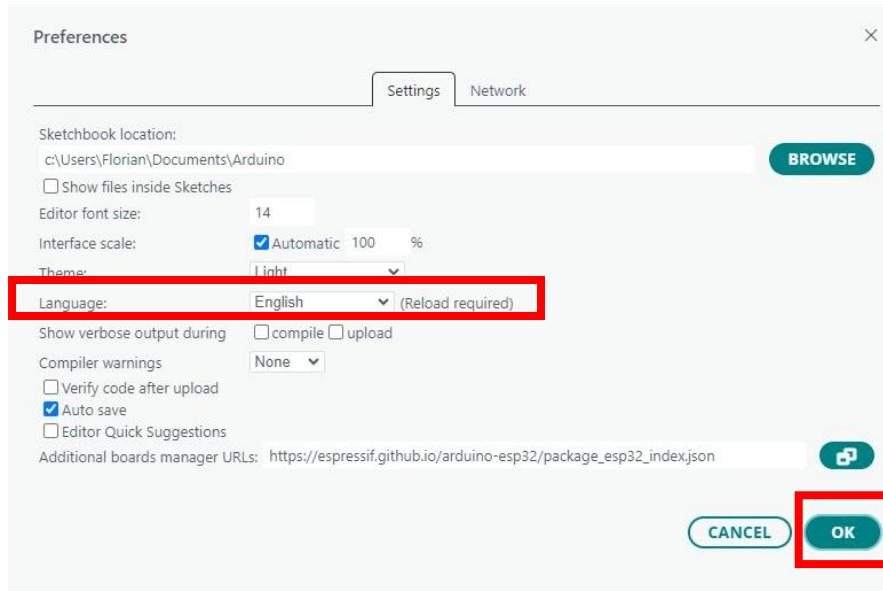


Abbildung 15

6. Die IDE startet neu und ist nun in Deutsch.



Abbildung 16

5. ESP und andere Boards Hinzufügen

1. Starte die Arduino IDE durch das Klicken auf das Icon „Arduino IDE“.

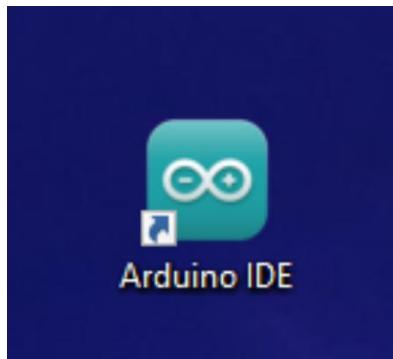


Abbildung 17

2. Die Arduino IDE startet. Nach der Installation kann der erste Start der Arduino IDE bis zu 2 Minuten dauern.

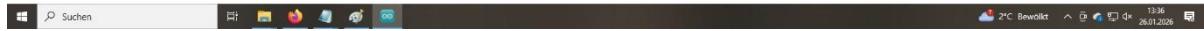


Abbildung 18

3. Öffne in der IDE das Menü „Datei“.



Abbildung 19

4. Klicke auf den Menüpunkt „Einstellungen“.

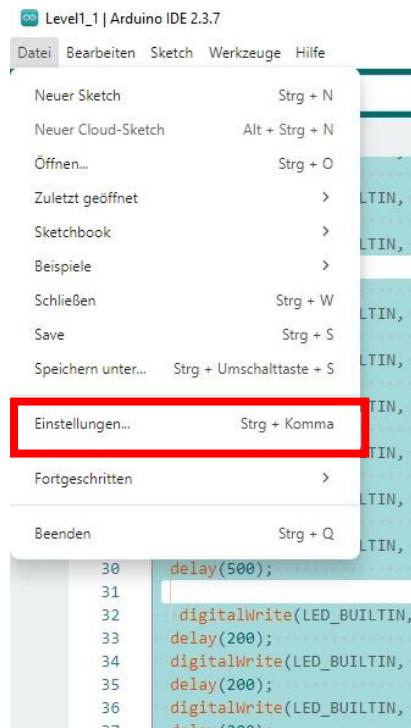


Abbildung 20

5. Wähle im Menüfenster „Einstellungen“ den Button .

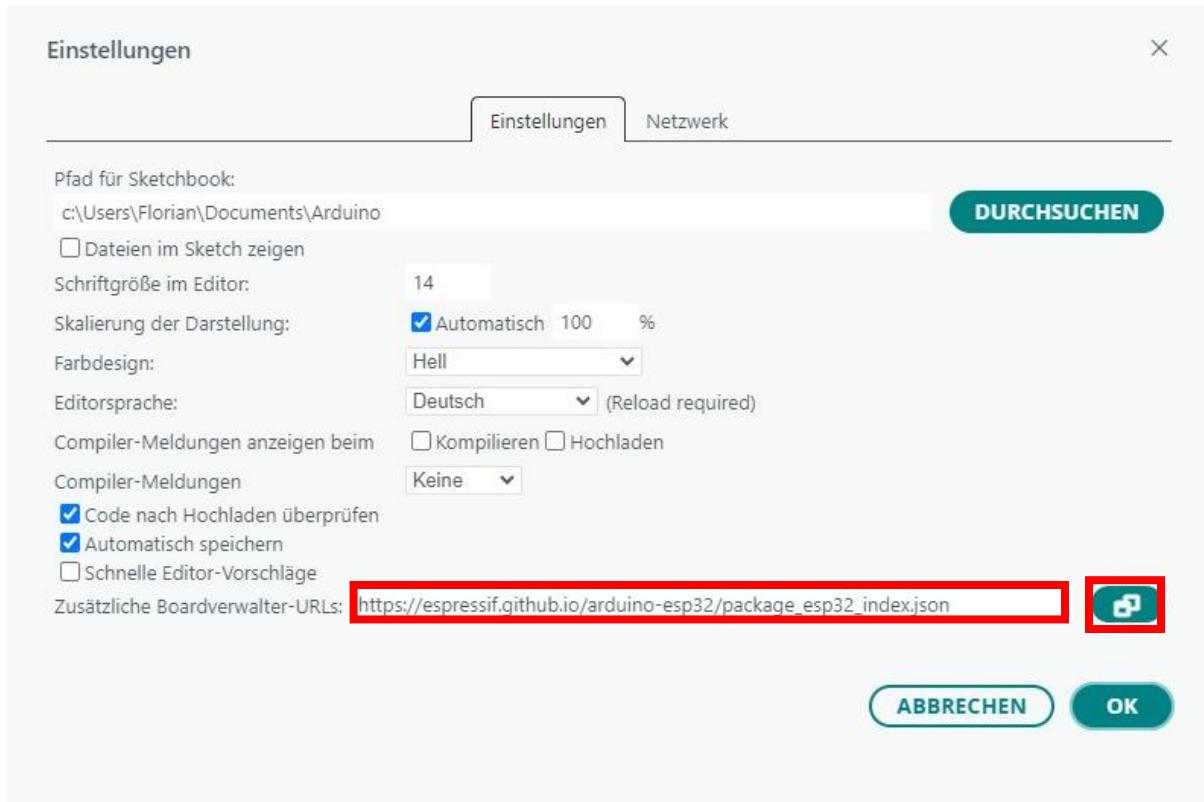


Abbildung 21

6. Das Fenster zum eintragen der Boardverwalter-URLs öffnet sich.

Trage hier die URL ein :

https://espressif.github.io/arduino-esp32/package_esp32_index.json

Und bestätige mit „OK“.

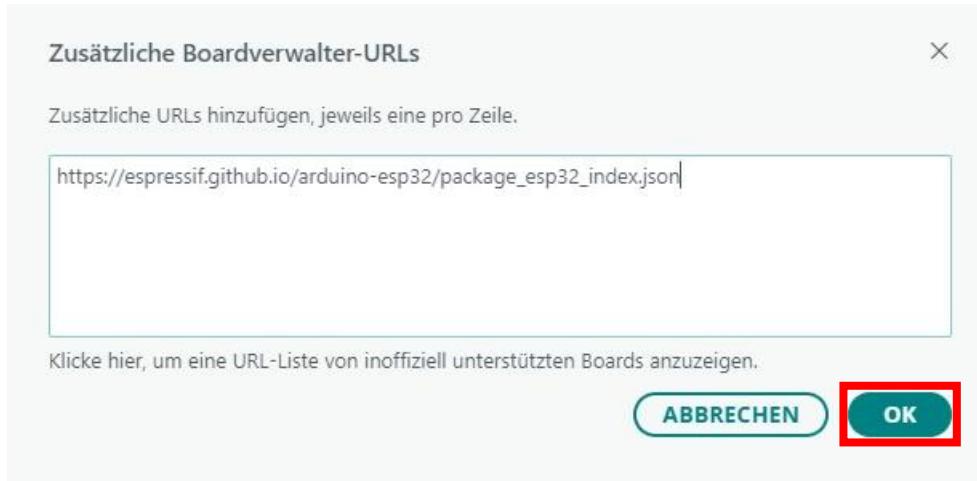


Abbildung 22

7. Öffne die Board-Verwaltung.

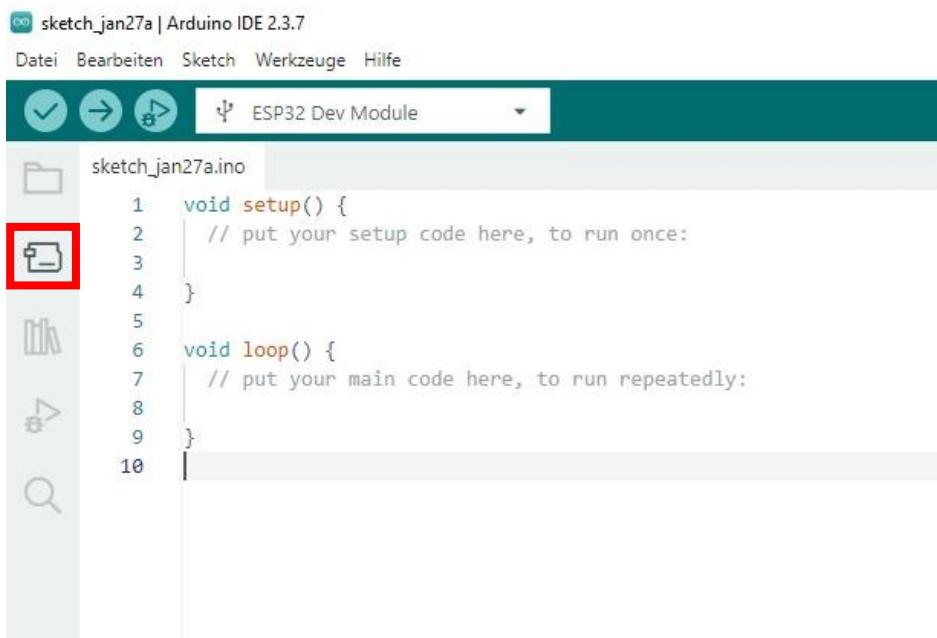


Abbildung 23

8. Trage in die Suchleiste „ESP32“ ein.

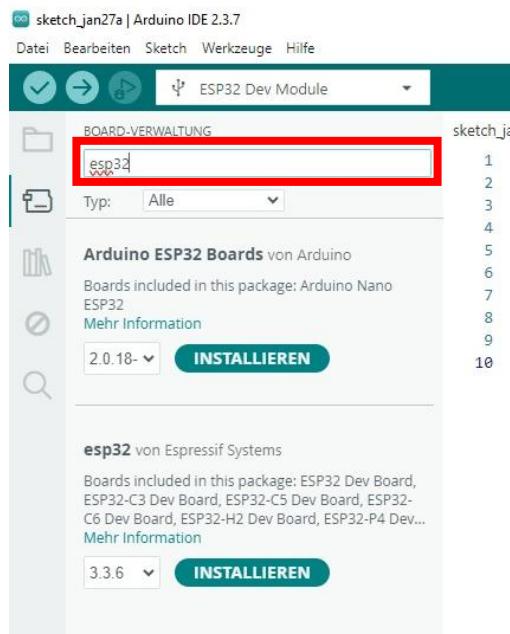


Abbildung 24

9. Klicke auf den Button „Installieren“ um die esp32 Boards zu installieren.

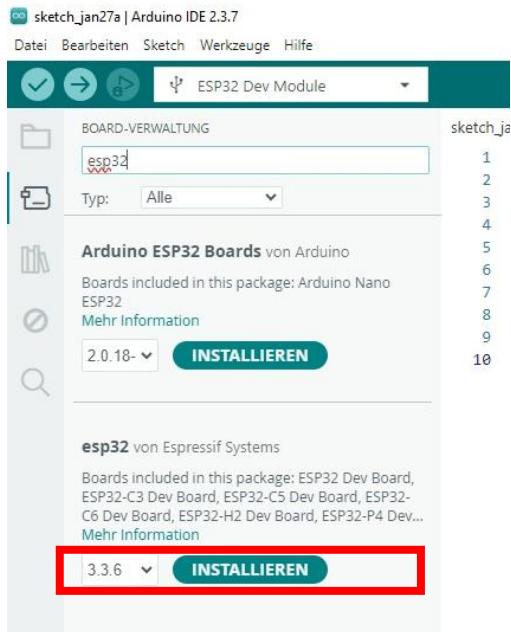
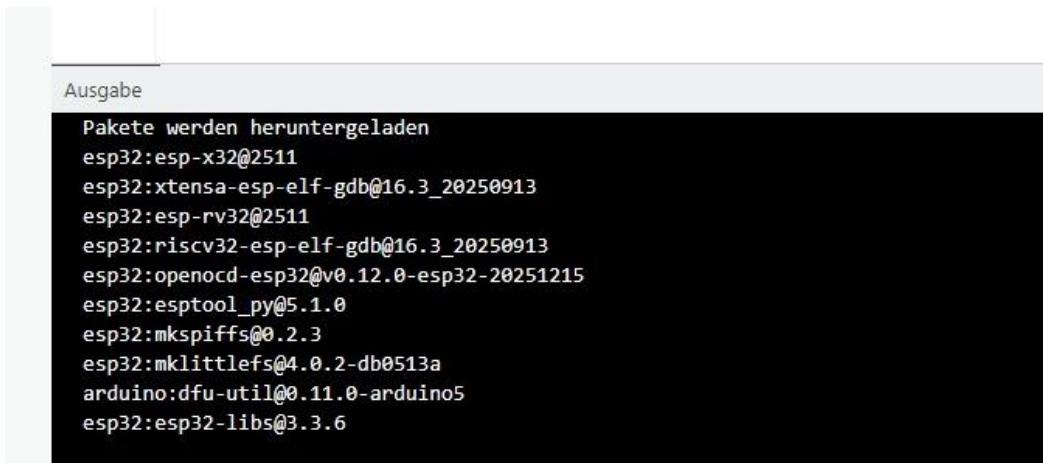


Abbildung 25

10. Der Installationsprozess läuft.



The screenshot shows the Arduino IDE Serial Monitor window with the title "Ausgabe". It displays a list of packages being downloaded:

```
Pakete werden heruntergeladen
esp32:esp-x32@2511
esp32:xtensa-esp-elf-gdb@16.3_20250913
esp32:esp-rv32@2511
esp32:riscv32-esp-elf-gdb@16.3_20250913
esp32:openocd-esp32@v0.12.0-esp32-20251215
esp32:esptool_py@5.1.0
esp32:mkspiffs@0.2.3
esp32:mklittlefs@4.0.2-db0513a
arduino:dfu-util@0.11.0-arduino5
esp32:esp32-libs@3.3.6
```

Abbildung 26

6. Probleme bei der Installation der Boards

Während des Installationsprozesses kann es zu einer Fehlermeldung kommen.

Error: 4 DEADLINE_EXCEEDED.

Es handelt sich hierbei um einen Timeout Fehler.

```
esp32:esp32c6-libs@3.3.6
Failed to install platform: 'esp32:esp32:3.3.6'.
Error: 4 DEADLINE_EXCEEDED: net/http: request canceled (Client.Timeout or context cancellation while reading body)
```

✖ Plattform 'esp32:3.3.6' konnte nicht installiert werden. 4 DEADLINE_EXCEEDED: net/http: request canceled (Client.Timeout or context cancellation while reading body)
 X

Abbildung 27

1. Beenden der Arduino IDE.



Abbildung 28

2. Öffne in deinem Explorer das Verzeichnis:

C:\Users\<DEINNAME>\arduinoIDE\

Dort befindet sich die Datei: „arduino-cli.yaml“

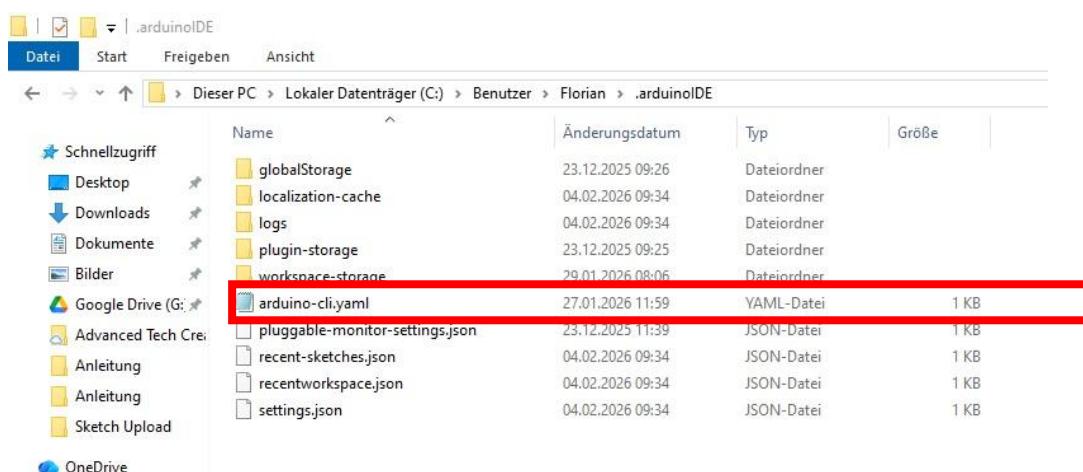


Abbildung 29

3. Öffne die Datei „arduino-cli.yaml“ mit dem Editor.

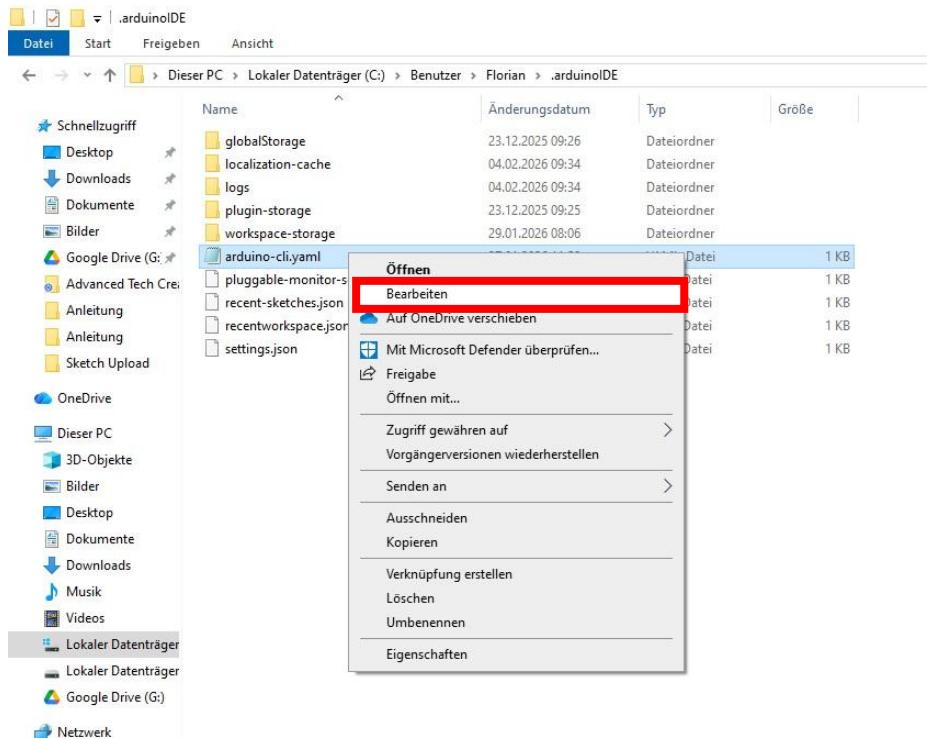


Abbildung 30

4. In der Datei fehlt ein Parameter. Füge am Ende der Datei folgendes Hinzu:

```
,,network:  
connection_timeout: 600s“
```



```
arduino-cli.yaml - Editor
Datei Bearbeiten Format Ansicht Hilfe
board_manager:
  additional_urls:
    - https://espressif.github.io/arduino-esp32/package_esp32_index.json
directories:
  builtin:
    libraries: C:\Users\Florian\AppData\Local\Arduino15\libraries
    data: c:\Users\Florian\AppData\Local\Arduino15
    user: c:\Users\Florian\Documents\Arduino
locale: de

network:
  connection_timeout: 600s
```

Abbildung 31

5. Speichere die Datei.

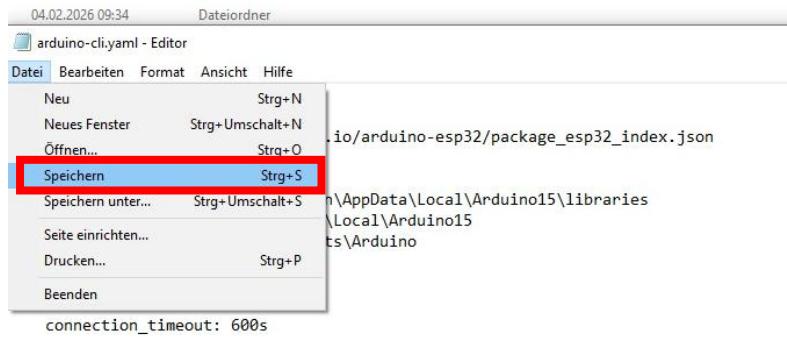


Abbildung 32

6. Alternativ, ersetze die Datei „arduinocli.yaml“ im Verzeichnis „C:\Users\<DEINNAME>\.arduinoIDE“ durch unsere Datei aus dem Verzeichnis: „\ATC\Einsteiger\Level 1\Software\ARDUINO IDE“