

Einen ESP8266 oder ESP32 mit dem Internet verbinden



Stoße die Tür zum Internet of Things weit auf! Wenn du für dein Projekt eine Verbindung zum Internet benötigst, leisten dir die beiden Controller [ESP8266](#)* und [ESP32](#)* gute Dienste. Du benötigst hierfür nur ein WLAN-Netzwerk, die passenden Zugangsdaten und ein paar Zeilen Code.



[AZDelivery NodeMCU Amica Modul V2 ESP8266 ESP-12F WiFi - Node MCU ESP 8266 WiFi Development Board mit CP2102 kompatibel mit Arduino - inklusive Installationsanleitung als E-Book](#)

□ Maße (LxBxH): 48 x 26 x 13 mm

8,49 €



[AZDelivery ESP32 Dev Kit C V4 NodeMCU WLAN WiFi Development Board kompatibel mit Arduino inklusive E-Book! \(Nachfolger Modul von ESP32 Dev Kit C V2\)](#)

□ Leistungsfähiger ESP32 -Microcontroller verbaut! Absolute Neuheit am Markt!

13,99 €

Bevor du loslegen kannst, musst du deine ESP8266 oder ESP32 in deiner Arduino IDE installieren. Wie das geht, erfährst du in diesem [Tutorial auf heise.de](#).

Die passende Bibliothek

Je nachdem, ob du mit einem ESP8266 oder mit einem ESP32 ins Internet möchtest, brauchst du eine passende Bibliothek. Für ersteren ist das die ESP8266WiFi, für den ESP32 die Bibliothek WiFi.h.

Beide Bibliotheken sollten in deiner Arduino IDE bereits

verfügbar sein – sofern du dort dein Board installiert hast.
Du kannst sie wie folgt am Anfang deines Sketchs – noch vor der Setup-Funktion – einbinden:

```
#include <ESP8266WiFi.h>
#include <WiFi.h>
```

Deine Zugangsdaten

Beide Bibliotheken funktionieren auf ähnliche Art und Weise. Bevor sich dein Controller mit deinem WLAN-Netzwerk verbinden kann, benötigt er die passenden Zugangsdaten. Auch diese legst du am Anfang deines Sketchs zum Beispiel in unveränderlichen Konstanten fest:

```
const char* ssid = "Name deines WLAN-Netzwerks";
const char* password = "Dein WLAN-Passwort";
```

Und ab ins Internet!

Jetzt kann es losgehen. Es gibt mehrere Möglichkeiten, die Verbindung einzurichten und im Seriellen Monitor darzustellen. Zentral ist jedoch immer die Funktion **WiFi.begin()** und dass diese bestenfalls im Setup deines Sketchs ausgeführt wird, damit für den Loop alles vorbereitet ist.

So könnte der Verbindungsaufbau aussehen:

```
void setup() {

  WiFi.begin(ssid, password);

  while (WiFi.status() != WL_CONNECTED) {
    delay(1000);
    Serial.println("Ich verbinde mich mit dem Internet...");
  }
}
```

```
Serial.println("Ich bin mit dem Internet verbunden!");  
}
```

Zunächst die Funktion **WiFi.begin()**, der du deine Zugangsdaten als Argumente mitgibst. Der anschließende While Loop wird solange ausgeführt, wie die Verbindung zum Internet noch nicht steht (**WiFi.status() != WL_CONNECTED**) und schreibt jede Sekunde in den Seriellen Monitor, dass die Verbindung aufgebaut wird. Sobald diese steht, erhältst du die Erfolgsmeldung im Seriellen Monitor.

Und das war's, du bist drin! ☐

Spannende Projekte

Jetzt, wo deine Verbindung ins Internet steht, probiere sie doch gleich mal mit diesen Projekten aus:

- [Ein Newsticker per API Call & JSON](#)
- [Erinnerung an deinen Regenschirm](#)
- [Beleuchte deine LEGO ISS im Rhythmus der echten Raumstation](#)