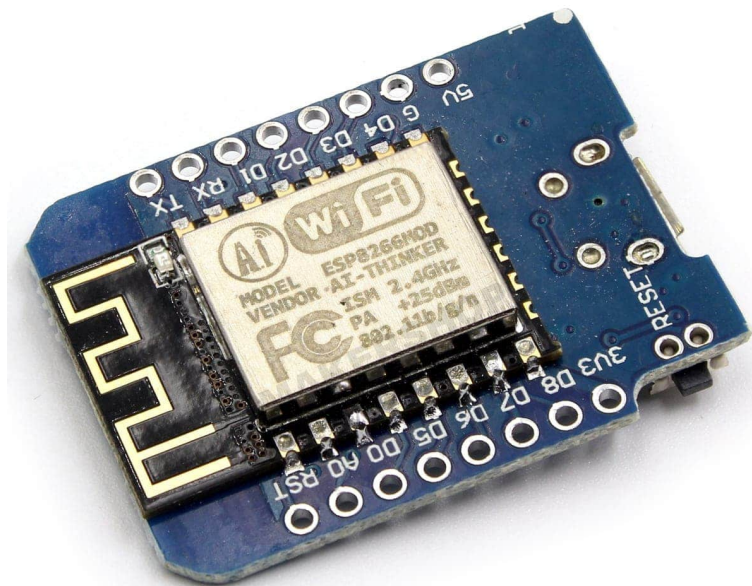


MAKERSHOP TUTORIAL · SKU 509140

D1 mini mit der Arduino IDE starten

Schritt für Schritt vom Boardpaket bis zum ersten WLAN-Scan.



1. Arduino IDE und ESP8266-Paket installieren

Arduino IDE	Aktuelle Arduino IDE 2 herunterladen und installieren.
Boardverwalter-URL	In Datei > Voreinstellungen unter „Zusätzliche Boardverwalter-URLs“ eintragen: <code>https://arduino.esp8266.com/stable/package_esp8266com_index.json</code>
Boardpaket	Werkzeuge > Board > Boardverwalter öffnen, nach „esp8266“ suchen und das Paket „esp8266 by ESP8266 Community“ installieren.
Board wählen	Werkzeuge > Board > esp8266 > „LOLIN(WEMOS) D1 R2 & mini“ auswählen.
Port wählen	Board per USB verbinden und unter Werkzeuge > Port den neu erschienenen seriellen Port wählen.

Kein Port sichtbar?

Anderes USB-Datenkabel und einen direkten USB-Port testen. Je nach Produktionsserie kann ein CH340/CH341-USB-Treiber erforderlich sein. Unter Linux zusätzlich die Zugriffsrechte auf den seriellen Port prüfen.

2. Integrierte LED testen

Die LED des D1 mini liegt üblicherweise an D4 / GPIO2 und arbeitet aktiv LOW. Deshalb schaltet LOW die LED ein und HIGH sie aus.

Blink-Test

```
void setup() {  
  pinMode(LED_BUILTIN, OUTPUT);  
}  
  
void loop() {  
  digitalWrite(LED_BUILTIN, LOW); // LED an  
  delay(500);  
  digitalWrite(LED_BUILTIN, HIGH); // LED aus  
  delay(500);  
}
```

- Sketch mit dem Pfeil-Symbol kompilieren und hochladen.
- Während des Uploads blinkt die serielle Aktivität; danach startet das Board automatisch neu.
- Die integrierte LED sollte nun im Abstand von einer halben Sekunde blinken.

3. WLAN-Netze scannen

Dieser Test verbindet sich mit keinem Netzwerk und benötigt kein Passwort. Er prüft lediglich, ob das 2,4-GHz-WLAN des ESP8266 arbeitet.

WLAN-Scanner

```
#include <ESP8266WiFi.h>

void setup() {
  Serial.begin(115200);
  WiFi.mode(WIFI_STA);
  WiFi.disconnect();
  delay(100);

  Serial.println();
  Serial.println("WLAN-Scan startet ...");

  int count = WiFi.scanNetworks();
  Serial.printf("%d Netze gefunden\n", count);

  for (int i = 0; i < count; i++) {
    Serial.printf("%2d: %-28s %4d dBm Kanal %d\n",
                  i + 1,
                  WiFi.SSID(i).c_str(),
                  WiFi.RSSI(i),
                  WiFi.channel(i));
  }
}

void loop() {
}
```

- Sketch hochladen und Werkzeuge > Serieller Monitor öffnen.
- Im Monitor 115200 Baud einstellen.
- Nach einem Reset erscheinen Anzahl, Name, Signalstärke und Kanal der gefundenen 2,4-GHz-Netze.

Typische Fehler schnell lösen

Upload bleibt bei „Connecting“	Richtigen Boardtyp und Port prüfen. Serielle Monitore schließen und Upload erneut starten.
LED verhält sich umgekehrt	Das ist normal: Die integrierte LED ist aktiv LOW.
Keine WLAN-Netze gefunden	Der ESP8266 unterstützt nur 2,4 GHz. Abstand zum Access Point verringern und Antennenzonen freihalten.
Board startet nicht mit Sensor	Verbindung an D3/GPIO0, D4/GPIO2 und D8/GPIO15 entfernen und Boot-Strapping-Pegel prüfen.
Instabile Neustarts	Kürzeres USB-Kabel oder stabileres Netzteil verwenden; leistungsstarke Verbraucher separat versorgen.

Nächste Schritte

- Sensor über I²C an D1/SCL und D2/SDA anschließen.
- Messwerte über WLAN an einen lokalen Webserver oder MQTT-Broker senden.
- Ein D1-mini-Shield für Relais, Display oder Umweltsensorik verwenden.
- Für batteriebetriebene Projekte Deep Sleep und die Verbindung D0/GPIO16 zu RST prüfen.

[D1 mini ESP8266 bei MAKERSHOP.DE](#)

[Offizielle Installation des ESP8266 Arduino Core](#)

[LOLIN D1 mini Referenz](#)

Hinweis: Boardbezeichnungen und Menüpunkte können sich mit neueren Versionen der Arduino IDE oder des ESP8266-Boardpakets leicht ändern.