

Technische Daten

NodeMCU Board

ESP8266 - 32bit Mikrocontroller mit integrierter Wi-Fi Schnittstelle

Hersteller: Espressif

Spannungsversorgung: 2,5 bis 3,6 V

Stromaufnahme: ca. 25 μ A bis 400 mA, durchschnittlich 80 mA

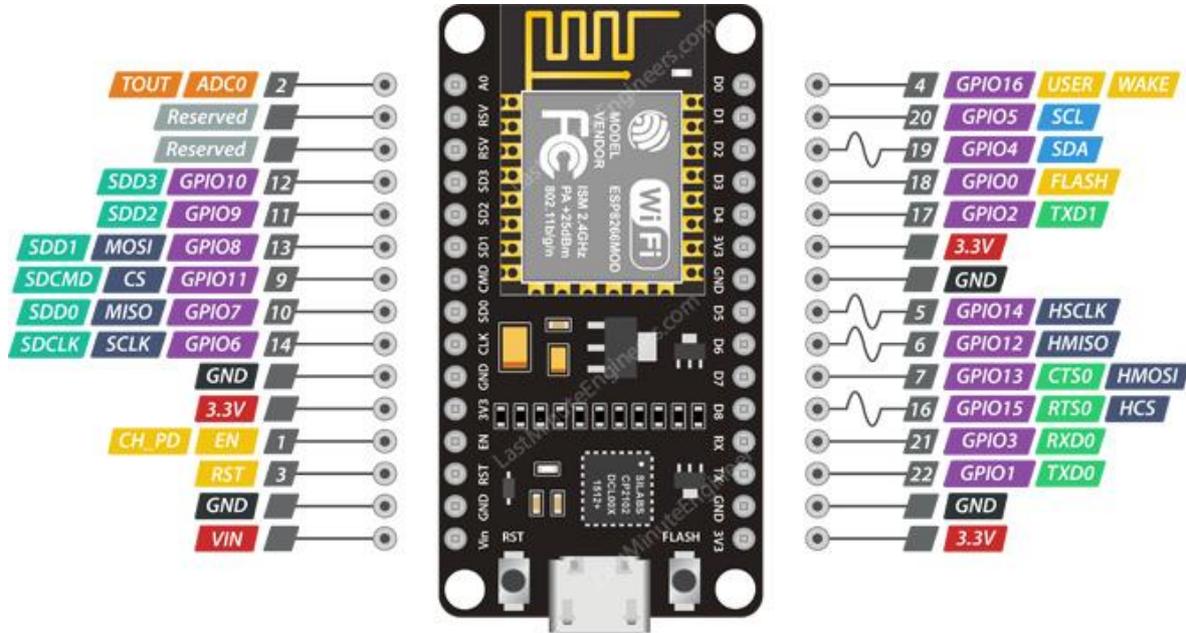
Schnittstellen:

- SPI, I²S, 2-UART (der zweite kann nur Ausgabe)
- 11 oder 13 programmierbare I/O Pins max. 12 mA
- 1 analoger Eingang 0 bis 1 V mit 10 Bit Auflösung
- Alle Eingänge vertragen maximal 3,6 Volt, auch wenn die Versorgungsspannung geringer ist.

Netzwerk:

- Wi-Fi 802.11 b/g/n 2,4 GHz mit WPA/WPA2 PSK
- IPv4, ab SDK 2.0 und Arduino Core 2.5.0 auch IPv6
- UDP und TCP, maximal 5 gleichzeitige Verbindungen
- Broadcasts werden nicht unterstützt
- Durchsatz: 150 bis 300 kByte/s in beide Richtungen
- Latenz: typisch 10 ms
- Soft-AP für bis zu 4 Clients

Pinout



Inhalt der Packung:

1. Microcontroller-Board ESP 8266 NodeMCU



2. OLED-Display 0,96 Zoll I2C



3. Smart-LED RGB



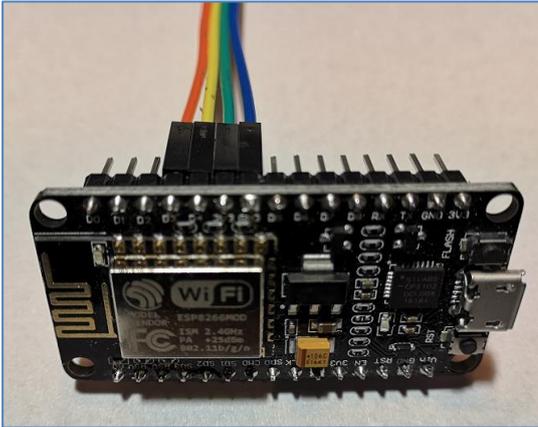
4. 2 Stück Verbindungskabel (3-Adern-Bundle und 4-Adern-Bundle)



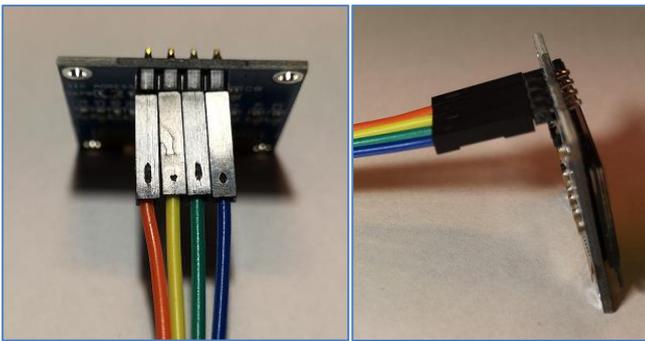
Sie **benötigen** außerdem ein **USB-Kabel mit Micro-USB** Anschluss (üblich zum Laden von Smartphones).

Aufbau des Demo-Setups:

1. Verbinden Sie das 4-Adern-Kabel an die Anschlüsse D3, D4, 3V3 und GND des Microcontrollers.



2. Verbinden Sie das OLED-Displays mit dem 4-Adern-Kabel am Microcontroller (Adern nicht verdrehen, Anschlusspins des Displays oben)



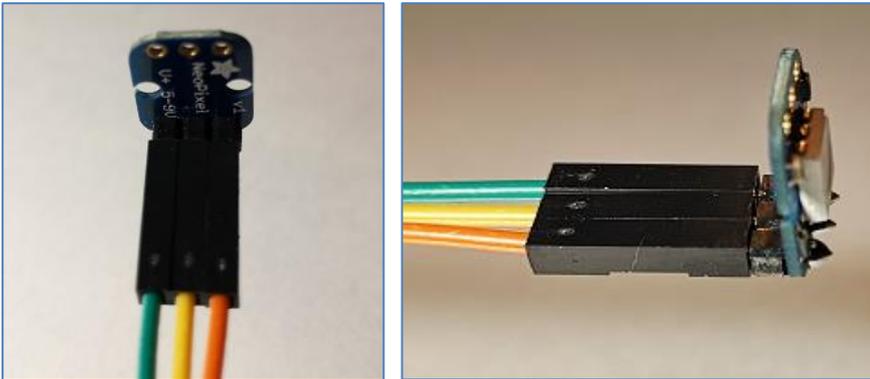
Pins: GND-> GND, 3V3->VCC, D3->SDA, D4->SCL

Bitte prüfen Sie noch einmal, dass nun GND am Microcontroller mit GND am OLED Display verbunden ist sowie 3V3 am Controller mit VCC am Display.

3. Schließen Sie das 3-Adern-Kabel an die Anschlüsse TX, GND und 3V3 an.



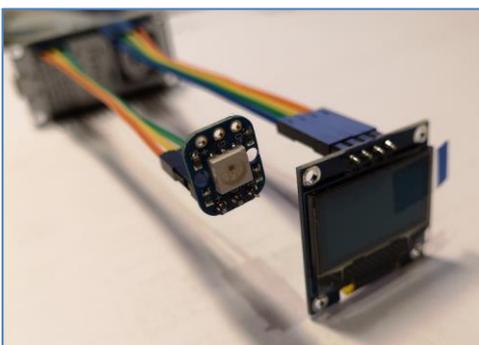
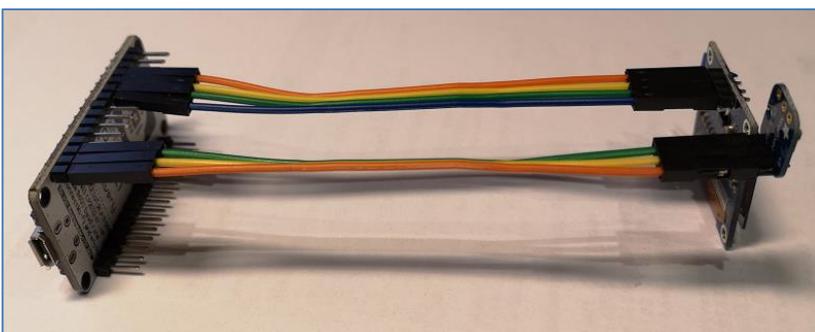
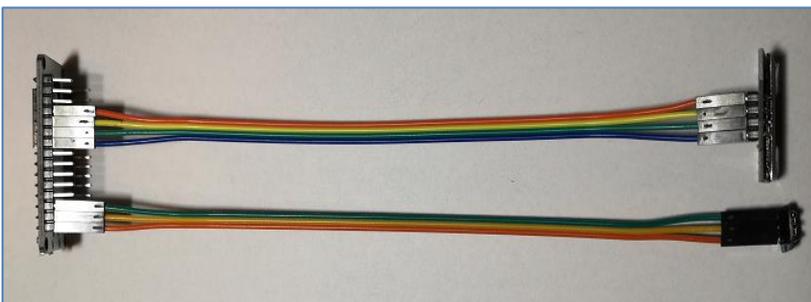
- Nun schließen Sie die Smart-LED ohne Verdrehen der Kabel an das 3-Adern-Kabel an.
(Anschlusspins der LED nach unten halten)

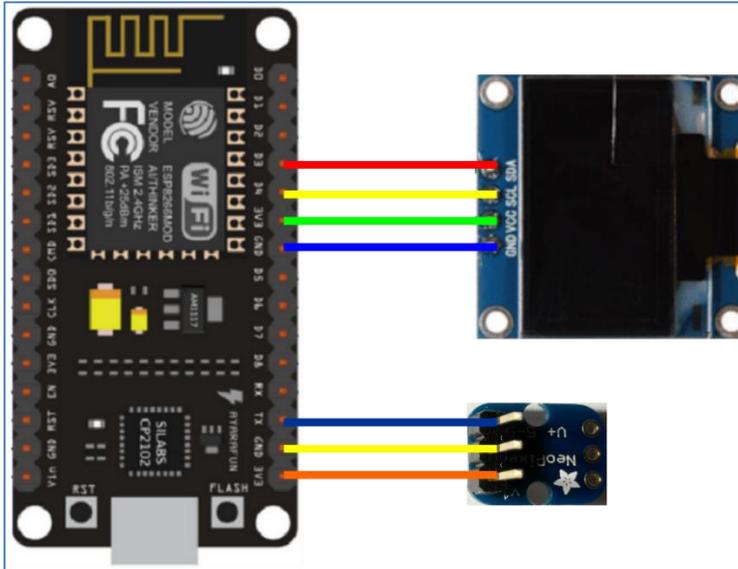


Pins: 3V3->+, GND->G, TX->In

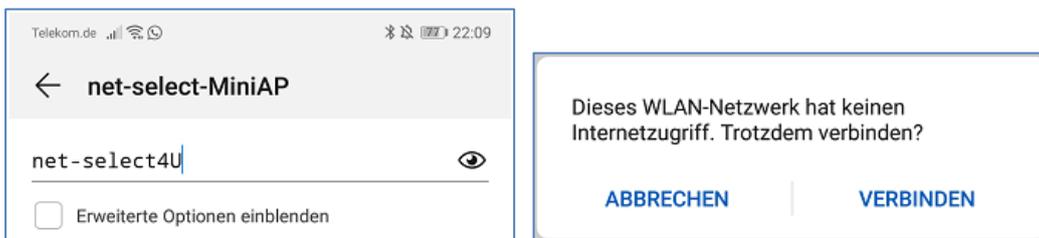
Prüfen Sie, dass der Anschluss GND des Microcontrollers mit dem G Anschluss der Smart-LED, sowie der 3V3 Anschluss mit dem + Anschluss der Smart LED Verbunden ist.

- Ihr Aufbau sollte nun so aussehen:

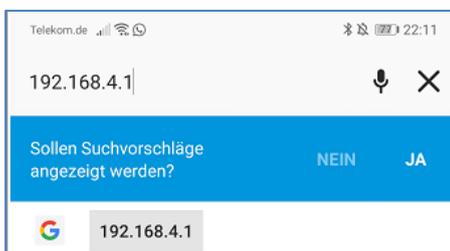




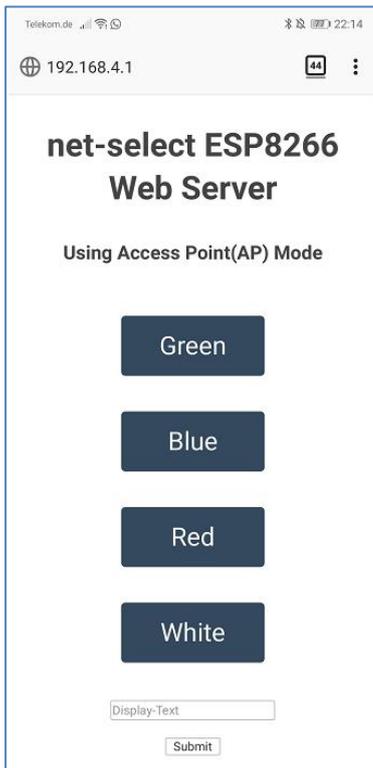
6. Schließen Sie Ihr USB-Kabel an Ihrem PC oder einem USB-Netzteil an und verbinden es mit dem Microcontroller.
Lassen Sie sich von der Anzeige auf dem Display überraschen!
7. Verbinden Sie sich mit Ihrem Smartphone oder Computer mit dem WLAN „net-select-MiniAP“ mit dem Passwort „net-select4U“.
 Eventuell meldet Ihr Gerät, dass das WLAN keinen Internetzugang bietet. Verbinden Sie sich trotzdem mit diesem Netz.



8. Öffnen Sie einen beliebigen Internetbrowser auf Ihrem Gerät, geben Sie in der Adresszeile 192.168.4.1 ein und schließen Sie die Eingabe mit der Enter- oder der Bestätigen-Taste ab.



9. Es sollte sich ein Fenster mit diesem Inhalt öffnen.



10. Sie können nun mit den Buttons die LED-Farbe über die WLAN-Verbindung ändern und über das Textfeld und dem Submit-Button einen Teil des Textes im Display ändern. Bitte nur kurze Texte eingeben - max. 6 Zeichen.

Natürlich können Sie den Microcontroller für viele weitere Projekte nutzen.

Lesen Sie dazu unsere Informationen auf <https://www.net-select.de/iot>.

Weitere Nutzung

Wenn Sie den Microcontroller für eigene Projekte nutzen möchten, gehen Sie wie folgt vor:

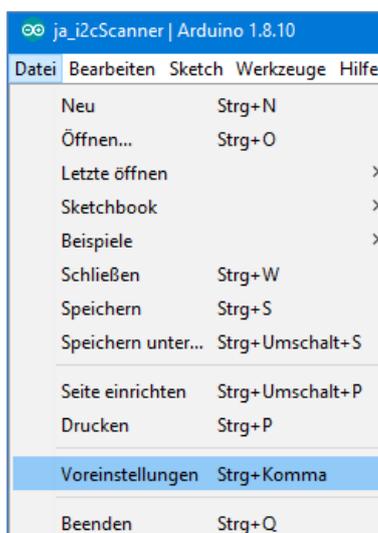
1. Laden Sie sich die Arduino Entwicklungsumgebung (Arduino IDE) von <https://www.arduino.cc/en/Main/Software> herunter. Die Entwicklungsumgebung ist für verschiedene Plattformen verfügbar und kann unter Windows ohne Installation (Download des ZIP Files) verwendet werden.

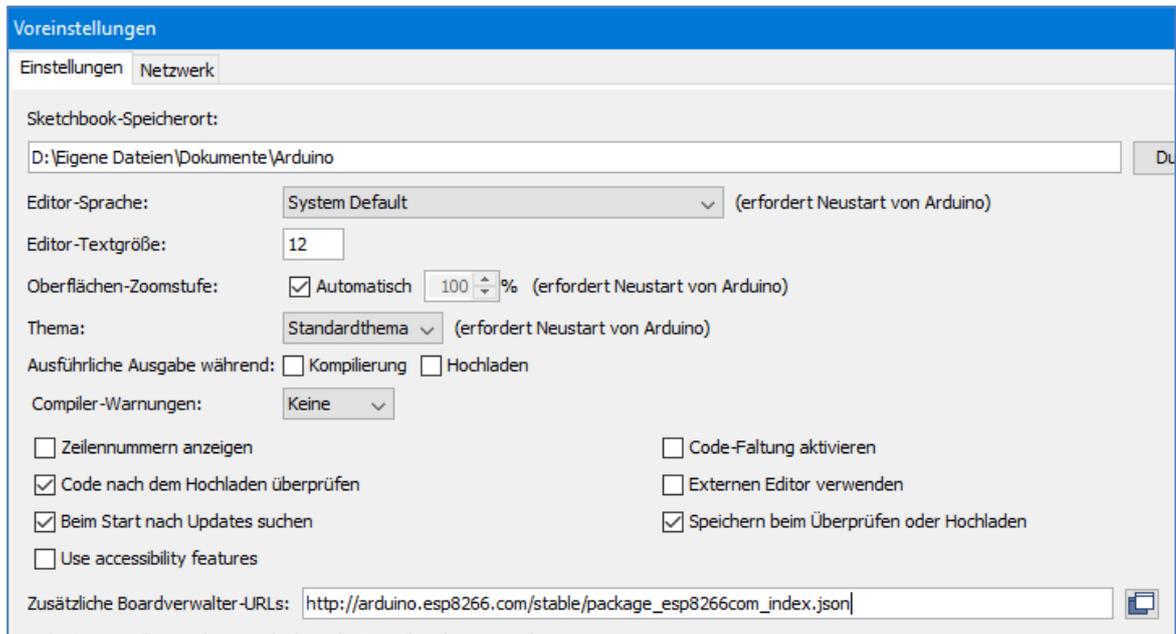
Windows Installer, for Windows XP and up
Windows ZIP file for non admin install

| | | |
|---|---------------------|---------|
| ↑ | .. | |
| 📁 | drivers | |
| 📁 | examples | |
| 📁 | hardware | |
| 📁 | java | |
| 📁 | lib | |
| 📁 | libraries | |
| 📁 | reference | |
| 📁 | tools | |
| 📁 | tools-builder | |
| 🔗 | arduino.exe | 395 KB |
| 🔗 | arduino-builder.exe | 13,9 MB |
| 🔗 | arduino_debug.exe | 392 KB |

Starten Sie die Arduino IDE nach Installation oder nach dem Entpacken.

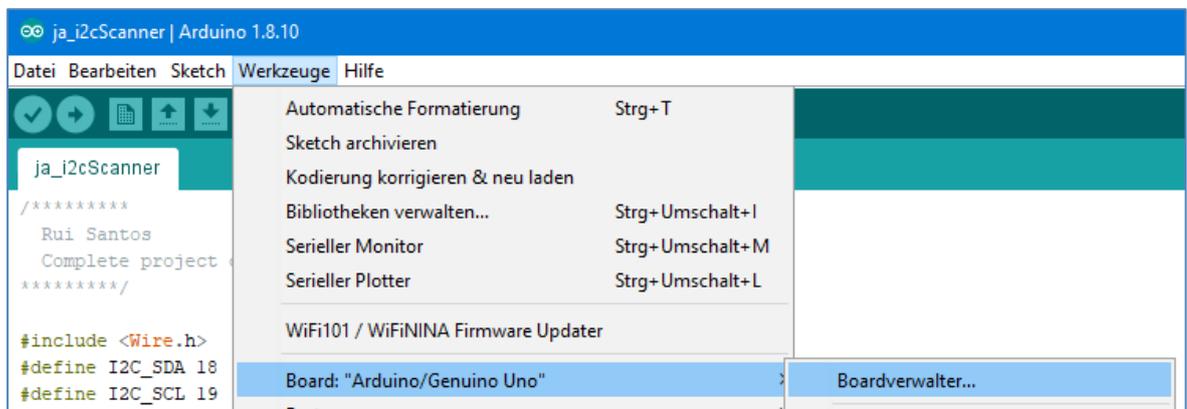
2. Nun muss der Microcontroller ESP8266 zum Arduino IDE zugefügt werden. Dazu werden zuerst die Voreinstellungen bearbeitet.



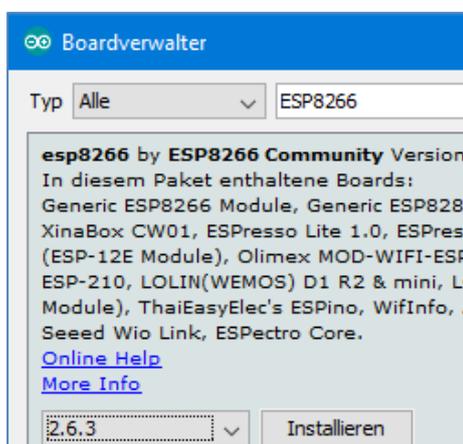


Unter den zusätzlichen Boardverwalter-URLs muss eine URL für die ESP8266 eingetragen werden.
 http://arduino.esp8266.com/stable/package_esp8266com_index.json

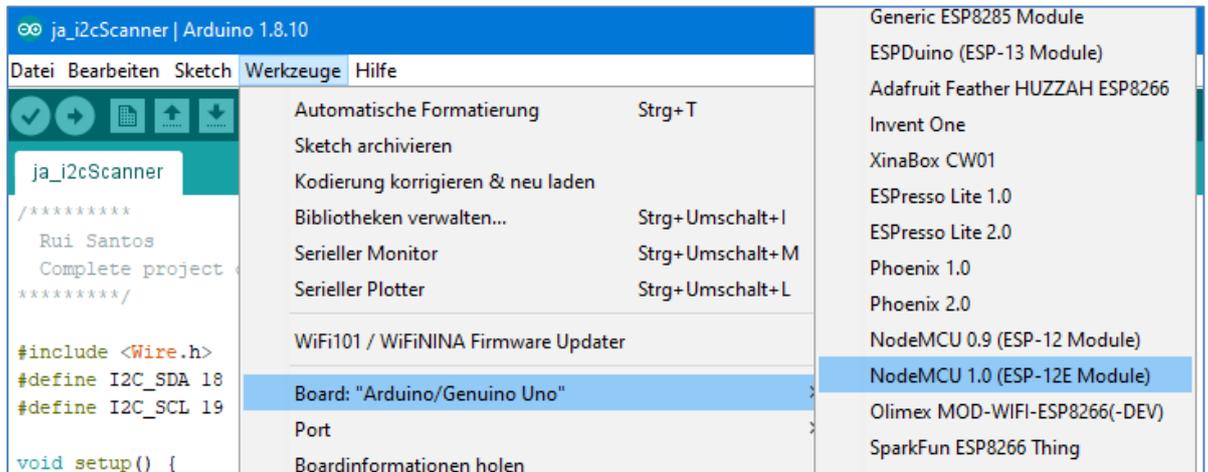
- Öffnen Sie im Menü Werkzeuge – Board den Boardverwalter, um das richtige Board auszuwählen.



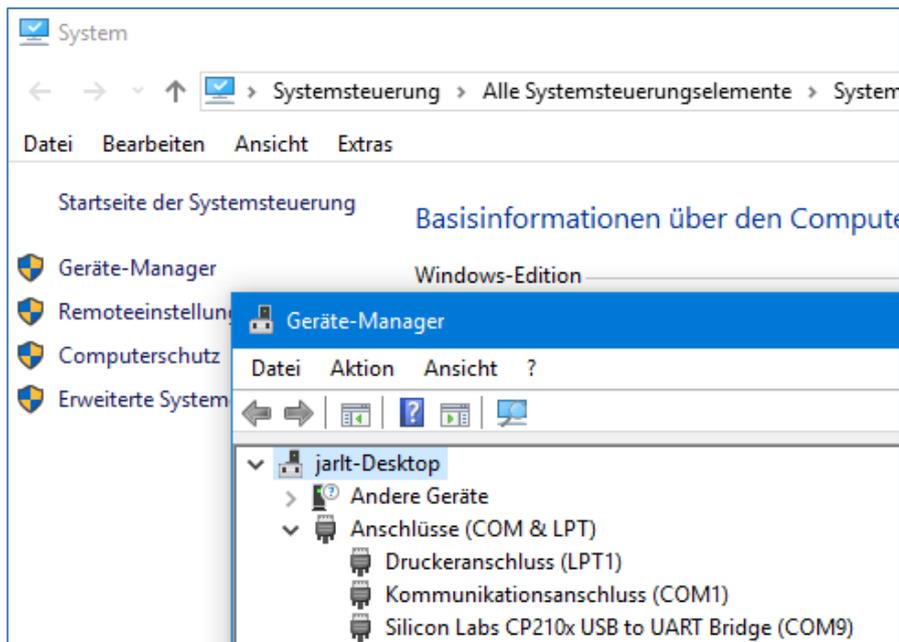
- Im Boardverwalter suchen Sie nach ESP8266 und installieren Sie die ESP 8266 Bibliotheken.



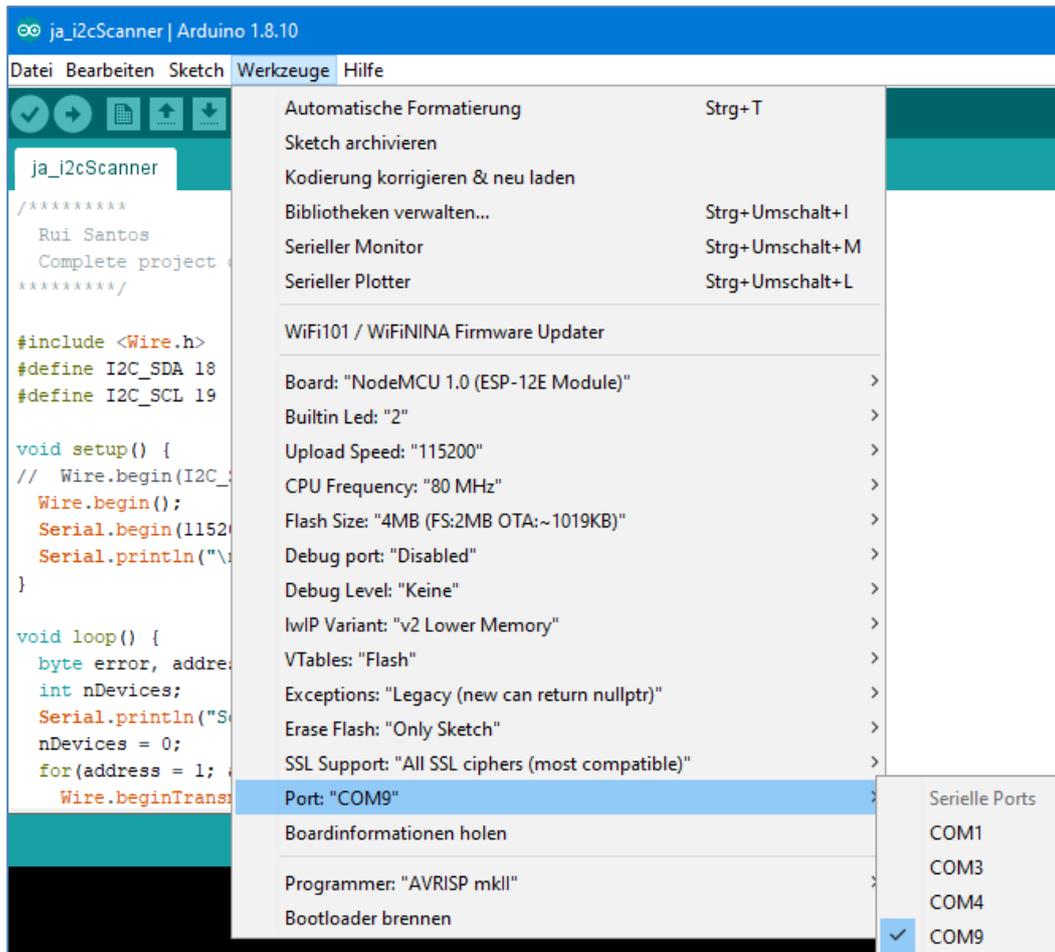
5. Wählen Sie nun im Menü Werkzeuge – Board: das Board „NodeMCU 1.0 (ESP-12E Module)“



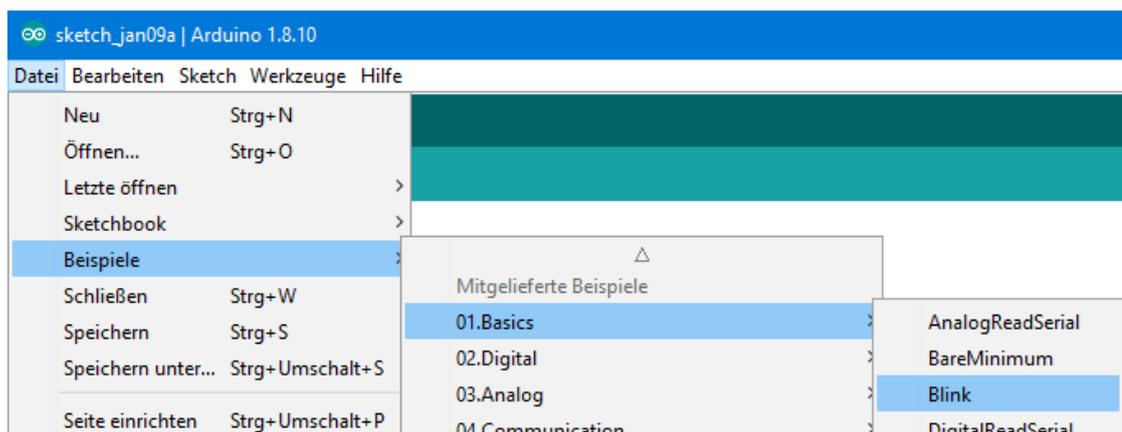
6. Final muss noch der richtige COM Port zur Kommunikation zwischen Computer und Microcontroller gewählt werden.
Den richtigen Port finden Sie nach Verbindung des Microcontrollers über USB mit dem Computer im Gerätemanager von Windows. Hier sollte ein Eintrag „CP210x USB to UART Bridge“ als COM-Anschluss registriert sein.



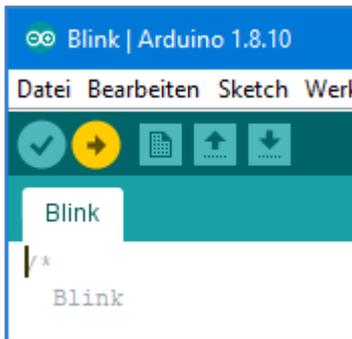
Dieser Port sollte nun unter Werkzeuge – Port konfiguriert werden (in diesem Fall COM9).



- Als einfachstes Beispiel wählen Sie unter Datei – Beispiele - 01.Basics – Blink aus



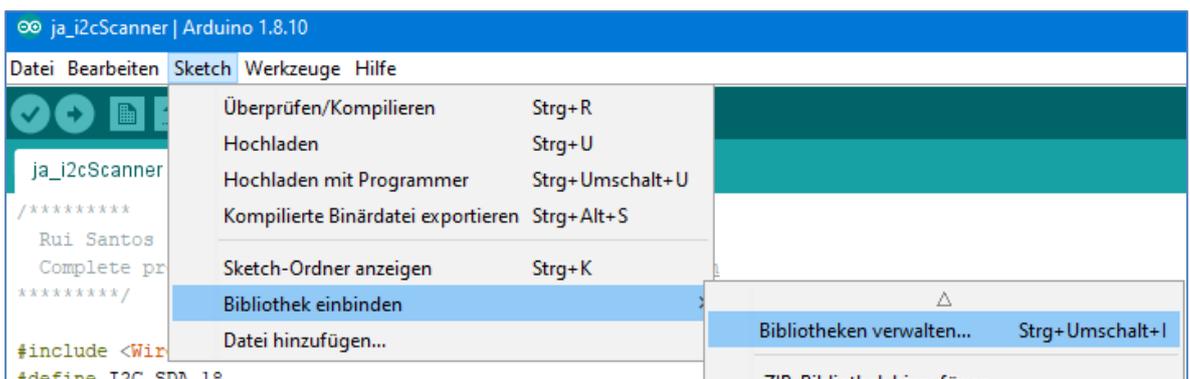
- Starten Sie das Programm mit dem Hochladen Button.



Danach sollte die blaue LED auf dem Microcontroller langsam blinken.

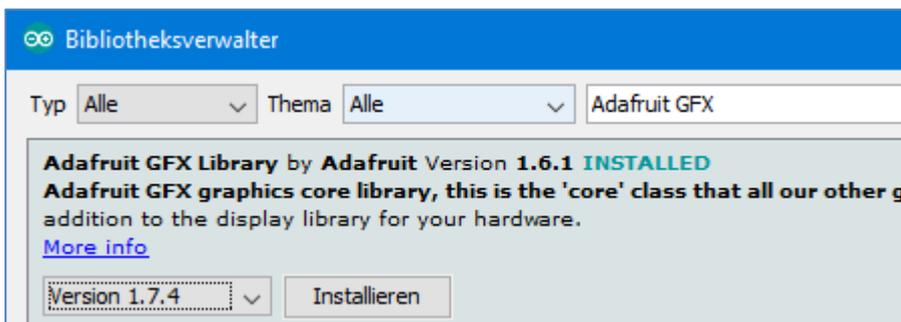
9. Nun können Sie Ihre eigenen Programme entwickeln und testen.
10. Sollten Sie zurück zu dem Demo-Programm der net-select zurückkehren wollen, dann erstellen Sie eine neue Datei und fügen Sie den unten angefügten Programm-Code statt des bestehenden Grundgerüsts ein.

Um das net-select Demo-Programm erfolgreich kompilieren zu können, müssen noch Bibliotheken über Sketch – Bibliothek einbinden zugefügt werden.

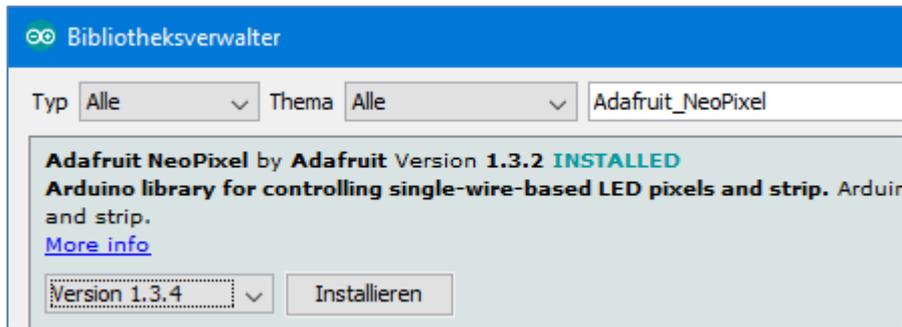


Folgende drei Bibliotheken müssen eingebunden werden:

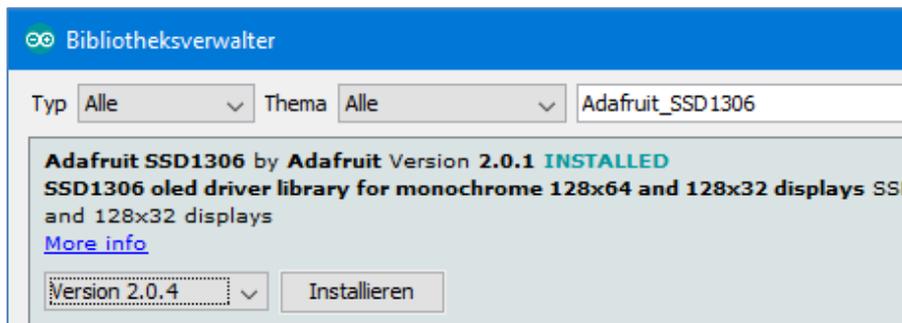
Adafruit_GFX



Adafruit_NeoPixel



Adafruit_SSD1306



Und nun, viel Spaß beim Entwickeln.



Ihr net-select Team