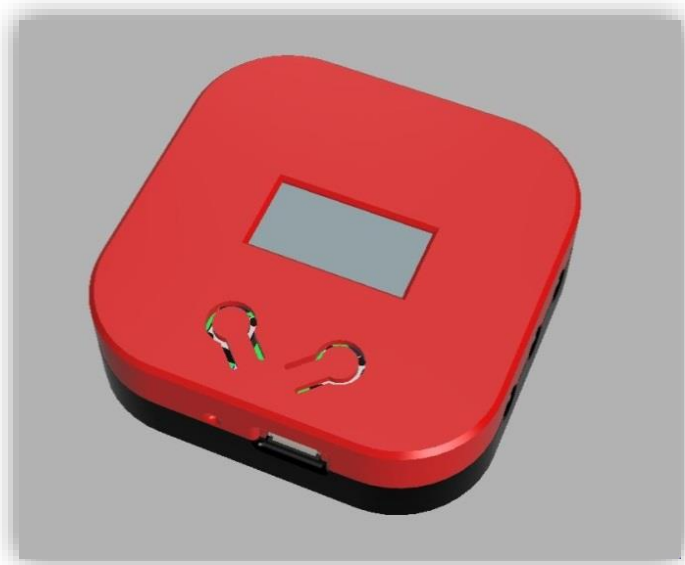


# WLAN THERMO

*nano*



Version 0.0

# Inhaltsverzeichnis

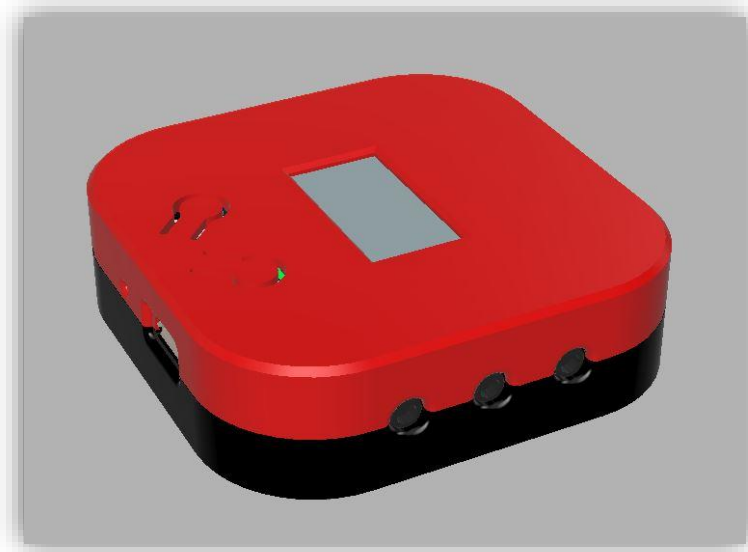
Inhaltsverzeichnis .....	2
1 Einleitung.....	4
1.1 Das Wlanthermo NANO .....	4
1.2 Funktionsübersicht .....	4
1.3 Die Anschlüsse.....	5
1.3.1 Temperaturmesskanäle.....	5
1.3.2 Micro-USB und Lade-LED.....	6
1.3.3 On/Off-Schalter.....	6
1.3.3.1 Batteriebetrieb.....	6
1.3.3.2 Netzbetrieb.....	6
1.3.4 Pitmaster-Buchse .....	7
1.3.5 WLAN Modul .....	7
2 Das OLED Display .....	8
2.1 Bedienungen und Funktionen am Gerät.....	9
2.2 Menüstruktur .....	10
2.2.1 Hauptmenü .....	10
2.2.2 Pitmaster und System Menü .....	11
2.3 Das Hauptdisplay im Detail .....	12
2.4 Die Kanal Einstellungen im Detail .....	14
2.5 Das Pitmaster Menü im Detail .....	15
2.6 Das System Menü im Detail .....	16
3 WEB GUI .....	17
3.1 Bedienungen und Funktionen über WEB GUI .....	17
3.1.1 Hauptseite.....	17
3.1.2 Messkanaleinstellungen .....	18
3.1.3 Wlan Einstellungen .....	19
3.1.4 Systemeinstellungen.....	20
3.1.5 Thinkspeak .....	21
4 Hardware Übersicht .....	22
4.1 Bezugsquellen .....	<b>Fehler! Textmarke nicht definiert.</b>

5	Software Installation .....	23
5.1	Installation des USB Treibers .....	23
5.2	Vorbereitung zum Flashen des FTDI Device mittels FT PROG .....	25
5.3	Arduino IDE.....	31
5.3.1	Arduino IDE installieren: Version > 1.6.11 .....	31
5.3.2	ESP8285 ESP 8266 - Core / Board Unterstützung installieren .....	31
5.3.3	Erweiterung der board.txt .....	33
5.3.4	Installation Arduino Plugin - ESP8266 Sketch Data Upload .....	34
5.3.5	Time - Libary integrieren .....	34
5.3.6	ArduinoJson - Libary einbinden .....	35
5.3.7	Projekt spezifische Librarys integrieren .....	35
5.4	Der Flashvorgang .....	39
5.5	Übertragung der HTML Komponenten .....	42
5.6	Verbindung mit dem Terminalprogramm PUTTY .....	43
5.7	Konfiguration der WLAN-Verbindung über die Serielle Schnittstelle.....	44
5.8	Die Software Schnittstellen.....	45
6	FAQ .....	47

# 1 Einleitung

## 1.1 Das Wlanthermo NANO

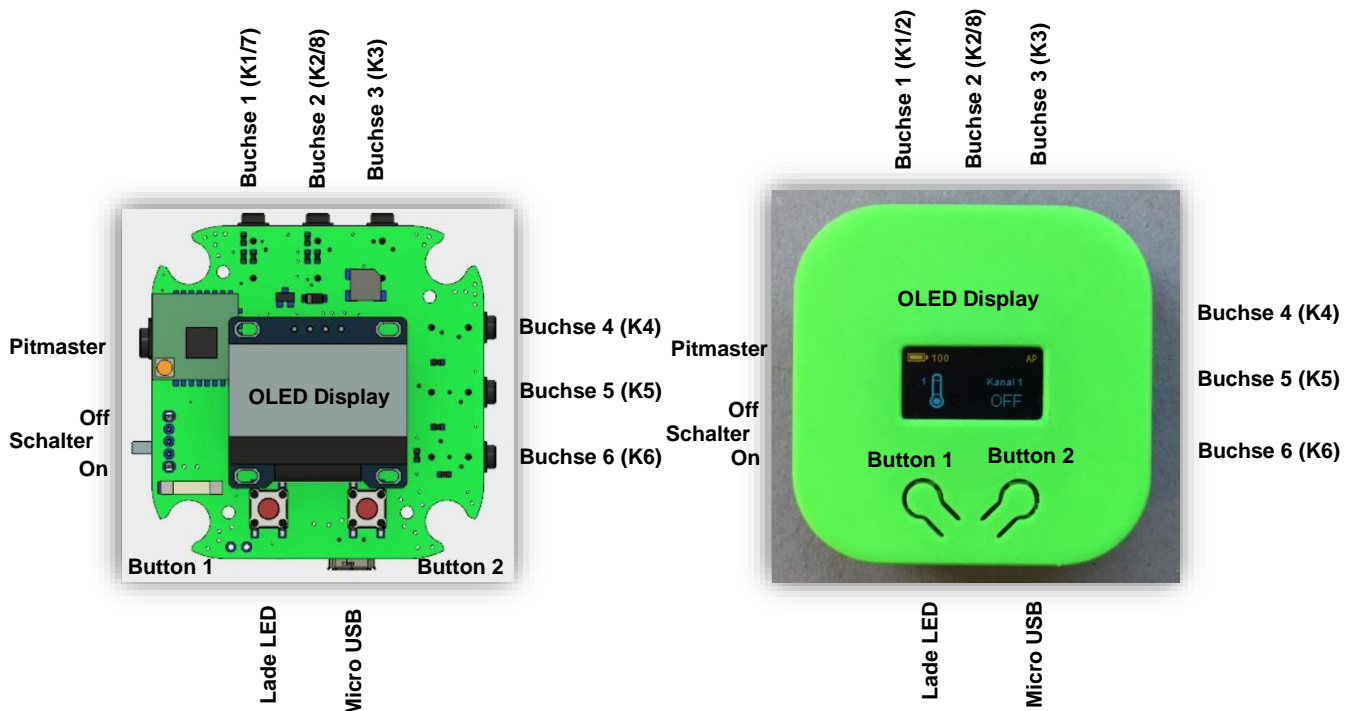
Das neue NANO Wlanthermo wurde auf Basis eines ESP 8285 mit integriertem WLAN Modul aufgebaut. Durch das kompakte Design und dem integrierten Akku ist ein System mit einer hohen Mobilität entstanden.



## 1.2 Funktionsübersicht

- Bis zu 8 Messkanäle können überwacht werden  
(bis zu 6 Einzelfühlern oder 4 Einzelfühlern und 2 Doppelfühlern)
- Integriertes WLAN Modul
- 1 x Pitmaster Anschluss zum direkten Anschluss eines Solid-State-Relais (leistungshungrigere Aktoren wie Lüfter und Servos können über eine zusätzliche Pitmaster-Platine mit eigener Energieversorgung angeschlossen werden (noch nicht released))
- Integrierten 1500 mAh LiPo-Batterie und sehr geringem Stromverbrauch  
Micro USB Buchse zum Laden mit LED Anzeige (energieautonomer Betrieb > 40 h )
- integrierte Ladeelektronik mit Entladeschutz
- OLED Display 2 farbig ( kein Touchdisplay) und 2 Bedientasten
- On/Off-Switch zur kompletten Abkopplung der LiPo-Batterie
- WEB GUI
- Temperatur Alarmgeber

## 1.3 Die Anschlüsse



### 1.3.1 Temperaturmesskanäle

- **Buchse 1** (Kanal 1/7) und **Buchse 2** (Kanal 2/8) können sowohl zum Anschluss von NTC-Einzelfühler (Kanal 1 bzw. 2) als auch zum Anschluss von NTC-Doppelfühler genutzt werden. Bei Doppelfühler wird der zweite Messkanal auf Kanal 7 bzw. 8 abgebildet.

Doppelfühlereinstellungen müssen also für beide Messkanäle getrennt eingestellt werden. Bei Verwendung von Einzelfühlern an Buchse 1 bzw. 2 sind die Kanäle 7 bzw. 8 geblockt und nicht anderweitig nutzbar.

- **Buchse 3, 4, 5, 6** sind zum Anschluss von NTC-Einzelfühler vorbereitet.
- **Buchse 4 und 5** können für die Verwendung von PT1000 umgerüstet werden. Hierzu muss pro Kanal ein 1k Widerstand (R41 / R42) zusätzlich eingelötet werden und der R3 muss durch einen 300 Ohm Widerstand ersetzt werden.

#### Hinweis:

Auf Grund des hohen Energieverbrauchs werden PT100 Fühler von der NANO nicht unterstützt. Der Energieverbrauch würde die Laufzeit zu sehr beeinträchtigen.

### **1.3.2 Micro-USB und Lade-LED**

Die Micro-USB-Buchse dient sowohl zum Flashen und einspielen diverser Einstellungen als auch zum Laden des internen Akkus. Beim Anschluss an einen USB-Host (PC) wird die Batterie nach der Anmeldung mit ca. 280 mA geladen. Ein Ladevorgang wird durch die Lade-LED neben der USB-Buchse sowie im Display im Batteriesymbol angezeigt.

Als Entladeschutz ist eine entsprechende Ladeelektronik integriert.

Beim Anschluss an ein 5V-Netzteil wird die Batterie mit dem maximalen Ladestrom von 500 mA geladen. Die verringerte Laderate am Host ergibt sich aus der maximalen Belastung des USB-Kanals von 500 mA, da neben der Batterieladung auch der ESP mit der USB-Supply betrieben wird, damit die Batterie ohne Last geladen werden kann.

Als Netzteil sollte mindestens eine 1A Version zum Einsatz kommen.

### **1.3.3 On/Off-Schalter**

#### **1.3.3.1 Batteriebetrieb**

Mit diesem Schalter kann die NANO im Batteriebetrieb ein bzw. ausgeschaltet werden. Es erfolgt eine komplette Trennung der Spannungsversorgung, sodass die Batterie im OFF-Zustand nicht entladen wird und die Nano somit komplett ausgeschaltet ist.

Beim Einschalten führt die Nano einen Neustart durch. Der Startvorgang dauert etwa 3 - 4 Sekunden. In dieser Zeit wird versucht eine WLAN-Verbindung aufzubauen. Wird kein bekanntes Netz gefunden, wird ein AP aufgebaut. Die Nano ist bereit, sobald die Temperaturdaten für Kanal 1 angezeigt werden.

#### **1.3.3.2 Netzbetrieb**

Im USB-Netzbetrieb, wenn also die NANO über ein USB Kabel an einem Netzteil oder PC angeschlossen ist, läuft die Nano immer, egal in welcher Stellung der Schalter steht.

Die NANO kann allerdings nur im Zustand ON bedient und zum Messen genutzt werden. Im OFF Zustand wird ein Ladehinweis im Display angezeigt und somit ist die Nano also komplett ausgeschaltet. Bei einem Switch von Off auf On wird wie im Batteriemodus ein Neustart des Systems ausgeführt und das System versucht eine WLAN Verbindung aufzubauen. Flashen kann man das System in beiden Schalterstellungen.

#### 1.3.4 Pitmaster-Buchse

An dem Pitmasterausgang der NANO kann man, vorerst nur ein SSR (Solid-State-Relais) direkt anschließen und betreiben.

Für leistungshungrigere Aktoren wie Lüfter und Servos ist eine zusätzliche Pitmaster-Platine mit eigener Energieversorgung geplant.

#### **!!! Achtung !!!**

Ohne eine Pitmaster-Erweiterung kann/darf kein anderer Aktor wie z.B. ein Lüfter angeschlossen werden. Diese könnte zu einer Zerstörung der NANO führen.

#### 1.3.5 WLAN Modul

Die Nano verfügt über eine eigene integrierte WLAN-Antenne. Die genaue Reichweite, wird der MINI mit dem kleinen USB-Wlan-Stick sehr ähnlich sein.

Sollte eine größere Reichweite notwendig sein, sind entsprechende Anpassungen (externen Antenne...) notwendig

## 2 Das OLED Display



Die Titelleiste (gelber Bereich) dient zur Anzeige von permanent wichtigen Daten wie Batteriezustand, Wlan-Empfang und Pitmasteraktivität...

Im Hauptfeld (blauer Bereich) werden alle weiteren Informationen angezeigt, welche vom User beeinflusst werden können.

Messkanäle die nicht in Verwendung sind, werden mit dem Schlüsselwort OFF gekennzeichnet.

Im Menü "System" kann ein Schnelllauf aktiviert werden (durch ein **F** in der Titelleiste symbolisiert), bei dem nur aktive Kanäle angezeigt werden. Alle OFF-Kanäle werden in diesem Fall nicht angezeigt.

Sobald ein Grenzwert über- bzw. unterschritten wird erscheint im Display ein Warnhinweis im Display mit der Angabe des betroffenen Kanals. Sollten mehrere Kanäle betroffen sein, erscheint nacheinander zu jedem Kanal ein entsprechender Hinweis. Über die Tasten kann der Alarm quittiert werden.



## 2.1 Bedienungen und Funktionen am Gerät

Die NANO verfügt über 2 Buttons über die die komplette Bedienung direkt am Gerät erfolgt.



Die Menü Bedienung folgt generell nach folgenden Regeln:

**B1K** => Button 1 (links) kurz gedrückt (<1s)

**B1L** => Button 1 (links) lange gedrückt (>1s)

**B1D** => Button 1 (links) Doppelklick

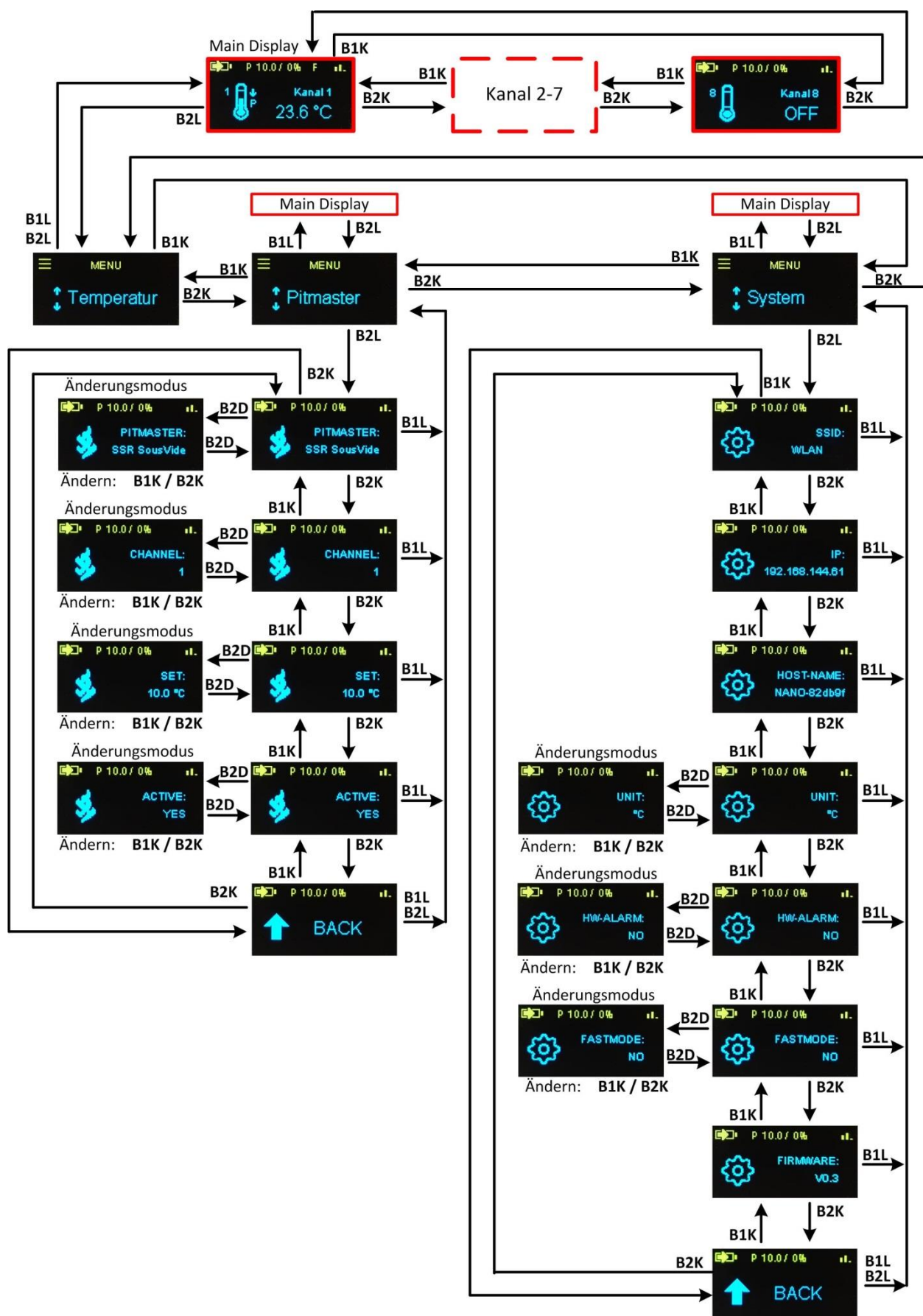
**B2K** => Button 2 (rechts) kurz gedrückt (<1s)

**B2L** => Button 2 (rechts) lange gedrückt (>1s)

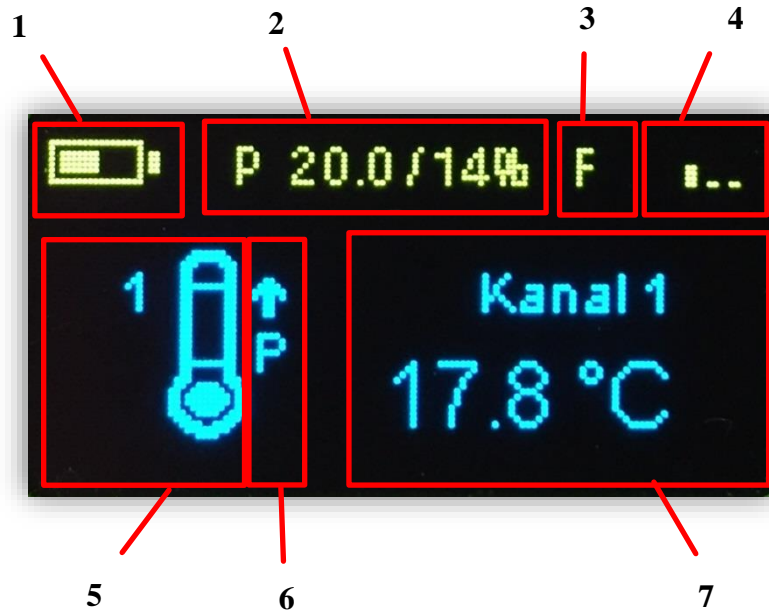
**B2D** => Button 2 (rechts) Doppelklick



## 2.2.2 Pitmaster und System Menü



## 2.3 Das Hauptdisplay im Detail



1. Anzeige für den Ladezustand des eingebauten Akkus.
2. Pitmaster Anzeige :  
P = Pitmaster aktiv  
20.0 = eingestellter Sollwert  
14% = Pitmaster Ausgang
3. Fast Modus Ein / Aus  
Wenn der Fastmodus im Systemmenü aktiviert wird, werden alle Kanäle, die im OFF Modus stehen, ausgeblendet und man kann mit den Tasten nur noch zwischen den aktiven Messkanälen schalten.
4. Anzeige der Signalstärke des WLAN-Signals.
5. Messkanalanzeige mit dem aktuellen Messwert in Grad Celsius oder Fahrenheit
6. Pitmaster: Tendenz des Stellsignals
7. Messkanal mit dem geregelt wird. Im Symbol wird der aktuelle Messwert als Balken dargestellt und skaliert auf die beiden Grenzwerte Temperatur Min und Max

### **Allgemeine Funktionen:**

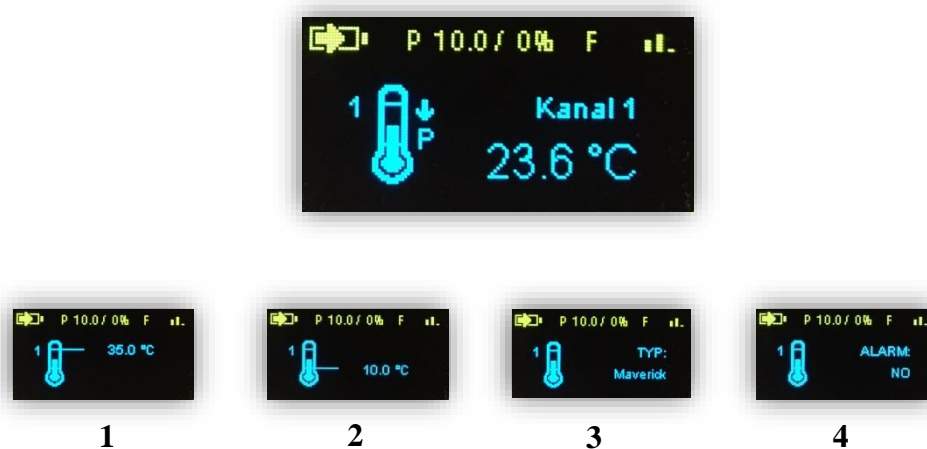
Alarmfunktion am OLED Display:

Sobald ein Grenzwert über- bzw. unterschritten wird und die Alarmfunktion des entsprechenden Kanals aktiv ist, erscheint ein Warnhinweis im Display mit der Angabe des betroffenen Kanals. Der Alarm muss vom Benutzer einmalig quittiert werden.

Erfolgt dann eine neue Unter-/Überschreitung des Grenzwertes muss die Quittierung entsprechen wiederholt werden.

Falls mehrere Kanäle betroffen sind, erscheint nacheinander zu jedem Kanal ein entsprechender Hinweis. So lange ein Kanal außerhalb der Grenzwerte ist, blinkt die aktuelle Temperatur. Falls zusätzlich der Hardwarealarm aktiv ist, ertönt ein Signal (4000Hz Summen mit 4Hz On/Off) bei Unter-/Überschreitung, so lange, bis der Warnhinweis quittiert wurde

## 2.4 Die Kanal Einstellungen im Detail



### 1. Hoch Alarm

Hier kam der Hochalarm für den jeweiligen Messkanal eingestellt werden

### 2. Tief Alarm

Hier kam der Hochalarm für den jeweiligen Messkanal eingestellt werden

### 3. Sensor Typ

Hier kann der Sollwert für den Pitmaster eingestellt werden

### 4. Alarm

Hier kann die Alarmierung des jeweiligen Messkanals ein oder ausgeschaltet werden

## 2.5 Das Pitmaster Menü im Detail



### 1. Pitmaster

In diesem Menüpunkt kann die Pitmasterkonfiguration ausgewählt werden:

- SSR SousVide
- Titan 50 x 50
- Titan 60 x 60

### 2. Kanal für Pitmaster

Hier kann man den Messkanal einstellen mit dem der Pitmaster regeln soll.

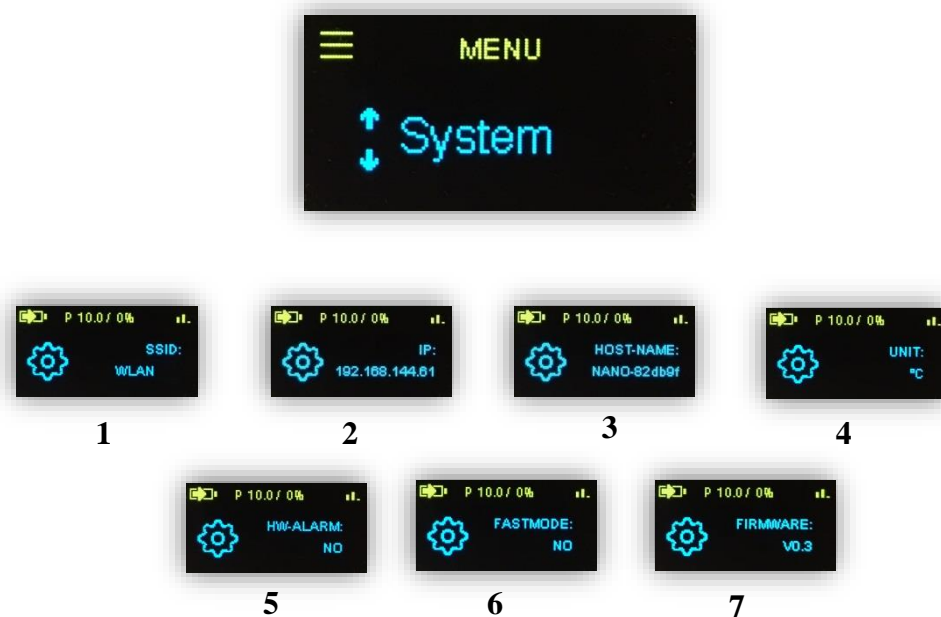
### 3. Sollwert

In diesem Punkt kann der Sollwert für den Pitmaster eingestellt werden

### 4. Pitmaster Ein / Aus

Hier kann der Pitmaster Ein oder Aus Gschaltet werden.

## 2.6 Das System Menü im Detail



### 1. SSIS

Hier kann man die SSID des ablesen mit dem die NANO verbunden ist.

### 2. IP- Adresse

An Stelle dieser findet man die IP Adresse der NANO die vom Netzwerk zugewiesen wurde

### 3. Host-Name

In diesem Punkt ist der Hostname der NANO zu finden.

### 4. Unit

Hier Kann man die Temperatureinheit °C oder °F einstellen

### 5. HW –Alarm

Über diesen Parameter kann der Hardware Alarm ( Summer ) aktiviert oder deaktiviert werden.

### 6. Fastmode

Hier kann der Fastmode auf Ein oder Aus Geschaltet werden

### 7. Firmware

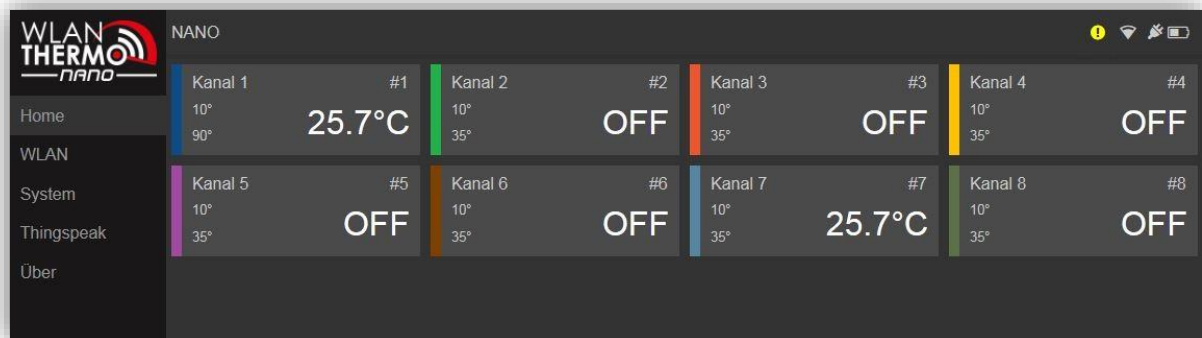
An dieser Stelle ist die aktuelle Firmware Version der NANO zu finden



## 3 WEB GUI

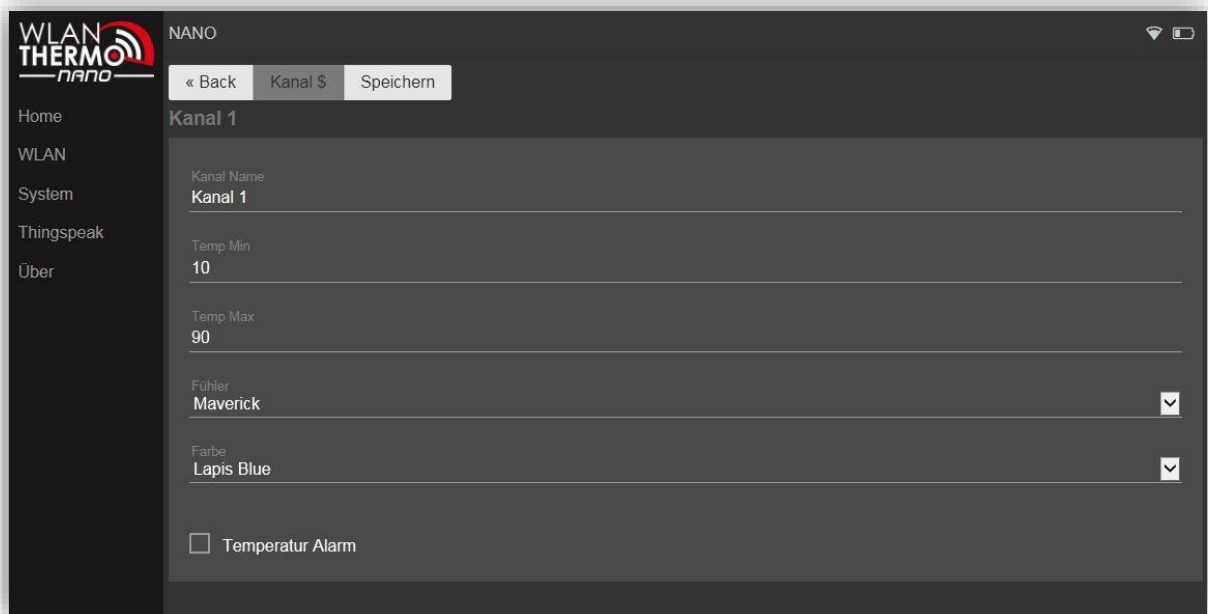
### 3.1 Bedienungen und Funktionen über WEB GUI

#### 3.1.1 Hauptseite



Text

### 3.1.2 Messkanaleinstellungen



WLAN THERMO nano

NANO

« Back Kanal \$ Speichern

Kanal 1

Kanal Name  
Kanal 1

Temp Min  
10

Temp Max  
90

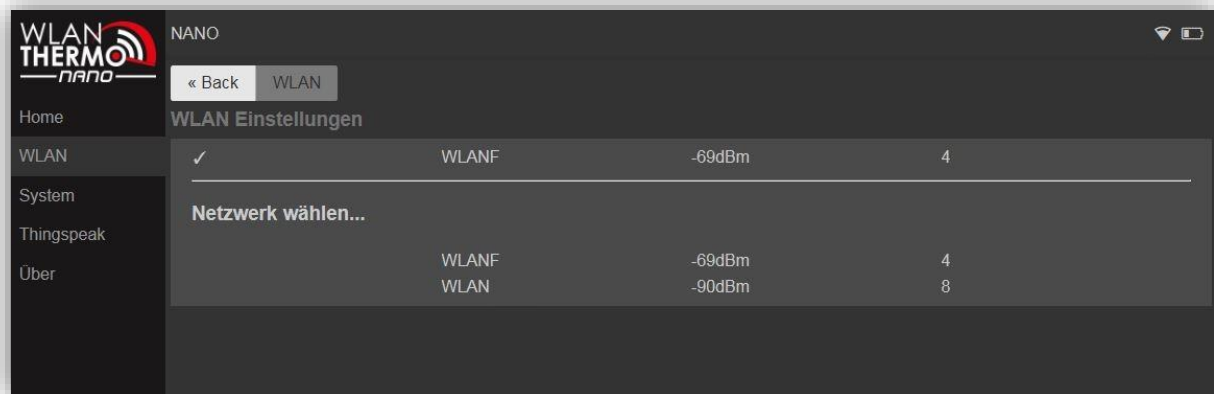
Fühler  
Maverick

Farbe  
Lapis Blue

☐ Temperatur Alarm


Text

### 3.1.3 Wlan Einstellungen



Text

### 3.1.4 Systemeinstellungen



Home
WLAN
System
Thingspeak
Über

NANO

« Back
System
Speichern

System Einstellungen

Hostname  
NANO-82e83f

AP-Name  
NANO-AP

Zeitzone  
MEZ+1

Sprache  
Deutsch

Einheit  
Celsius

Hardwareversion  
V1

☐ Hardwarealarm
☐ Sommerzeit
☐ Schnelllauf
☒ Nach Updates suchen

Text

### 3.1.5 Thinkspeak



The screenshot shows the 'Thinkspeak' configuration page of the WLAN THERMO NANO device. The interface is dark-themed with a sidebar on the left containing links: Home, WLAN, System, Thinkspeak (highlighted), and Über. At the top, there are navigation buttons: « Back, Thinkspeak, and Speichern. The main content area is titled 'Thinkspeak' and contains three input fields: 'Channel Write API Key' with the value 'RD3O2PAUORJCYYSF', 'ThingHTTP API Key' (empty), and 'Channel ID' with the value '288366'. At the bottom, there is a checkbox labeled 'Batterieladezustand anstatt CH8' which is checked.

Text

## 4 Die Hardware

Egal ob man die Nano basierend auf einer bereits bestückten Platine zusammenbaut oder sich an einen kompletten Selbstbau heranwagt, muss man mit Kosten zwischen 70 - 90 € (exklusive Temperaturfühler) rechnen. Eine detaillierte Auflistung ist in der BOM zu finden.

Passende Temperaturfühler könnt ihr euch z.B. hier besorgen. [Link](#)

### 4.1 Zusammenbauvarianten für Selbstbauer:

Der schnelle Weg: Basierend auf einer komplett bestückten Platine (inkl. Magnete und Schrauben) aus dem Verkaufsportal werden lediglich noch die 3D-Druckteile (Boden, Deckel und Batterieschutz) und eine LiPo-Batterie (Modell LiPo 504050 mit 1500 mAh) benötigt. Die Lötarbeiten sowie die benötigten Löterfahrungen sind minimal.

Der harte Weg: Basierend auf einer unbestückten Platine aus dem Verkaufsportal oder einer Platinenbestellung werden neben den 3D-Druckteilen und der LiPo-Batterie noch Bauteile von Mouser (siehe Mouser-Warenkorb) und ein Bauteilkit aus dem Verkaufsportal benötigt. Das Bauteilkit fasst alle weiteren nicht über Mouser beziehbaren Elektronikteile zusammen, um zusätzliche Versandkosten zu sparen. Die Lötarbeiten sind sehr umfangreich und nur für lötfähre Bastler zu empfehlen.

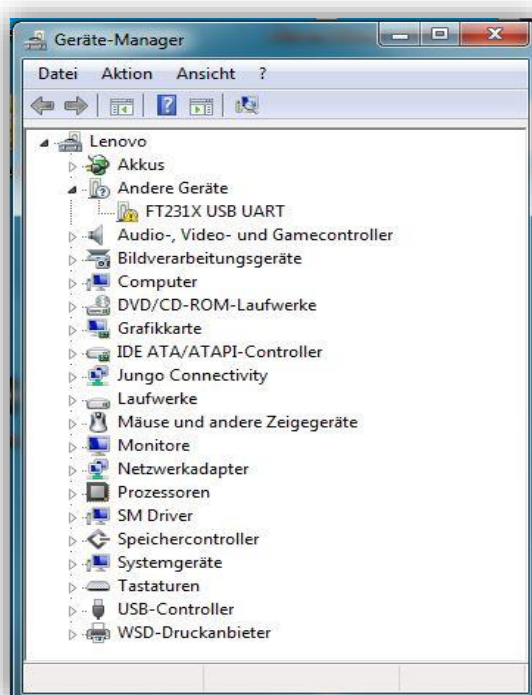
- **Gerber Dateien:** [Link](#)
- **BOM:** [Link](#)
- **Mouser-Warenkorb:** [Link](#)
- **Gehäuse zum selber drucken:** [Link](#)
- **Verkaufsportal WLAN Thermometer Nano** [Link](#)  
**(Platinen und Bauteil-Kit):**

**Wichtiger Hinweis:** Lithium Polymer Batterien, wie sie in der Nano verbaut sind, erfordern stets einen sicheren Umgang. Fehlbehandlung kann zu Explosion, Feuer, Rauchentwicklung und Vergiftungsgefahr führen. Bitte achtet also stets auf einen sachgemäßen Umgang mit eurer Nano.

## 5 Software Installation

### 5.1 Installation des USB Treibers

Beim Verbinden der NANO mit einem PC wird das Gerät erkannt und es muss der USB Treiber installiert werden. Die Erkennung des Device und die automatische Installation kann etwas dauern, also bitte etwas warten bis alles abgeschlossen. Sollte das System den Treiber nicht automatisch finden bzw. installieren und der Gerätemanager ein ähnliches Bild anzeigen wie unten abgebildet, muss die Treiberinstallation manuell erfolgen.



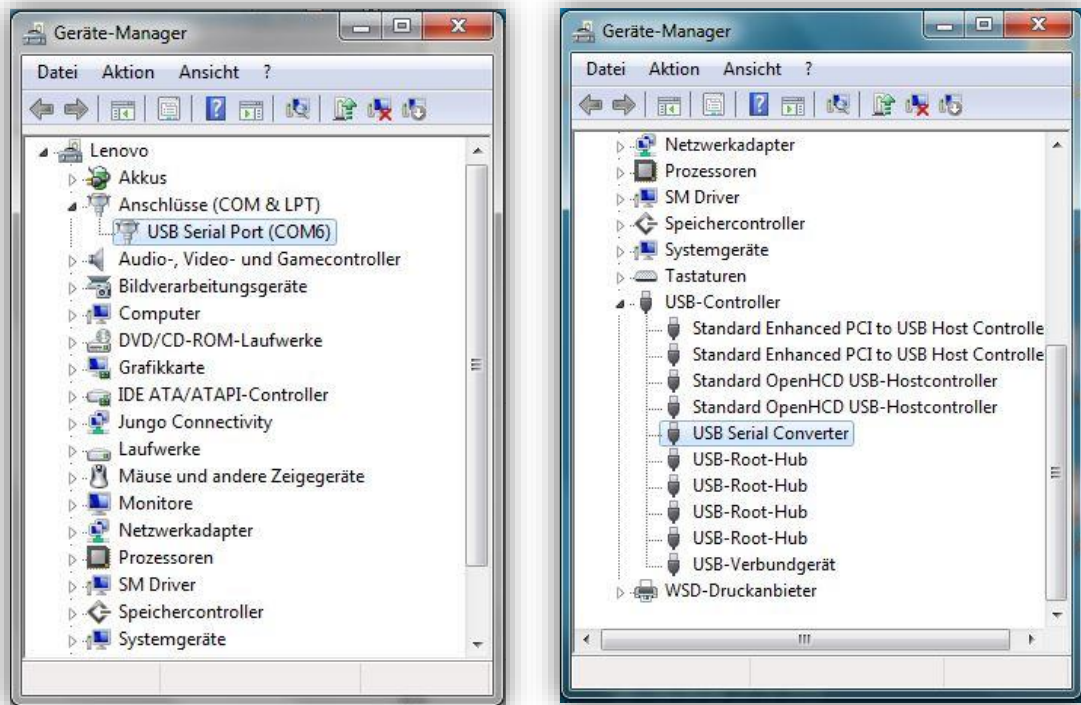
Manuelle Installation des benötigten USB Treibers:

- Die Treiber für das entsprechende Betriebssystem können hier heruntergeladen werden:  
<http://www.ftdichip.com/Drivers/VCP.htm>

Empfohlen wird die „setup executable“ Version zu nutzen.

- Nachdem die Datei heruntergeladen und entpackt wurde, kann durch einen Doppelklick auf die .exe Datei die Installation gestartet werden. Danach bitte den Anweisungen am Bildschirm folgen. Der Vorgang kann einige Zeit dauern.

Wenn die Installation abgeschlossen ist, sollte der Gerätemanager so aussehen:



Vorausgesetzt die NANO ist mit dem PC über ein entsprechendes USB Kabel verbunden und die Treiber sind installiert, findet man im Gerätemanager unter dem Punkt „Anschlüsse (Com & LPT)“ den Virtuellen Seriellen Port der für die weiteren Schritte benötigt wird.

In diesem Fall: COM6




## 5.2 Vorbereitung zum Flashen des FTDI Device mittels FT PROG

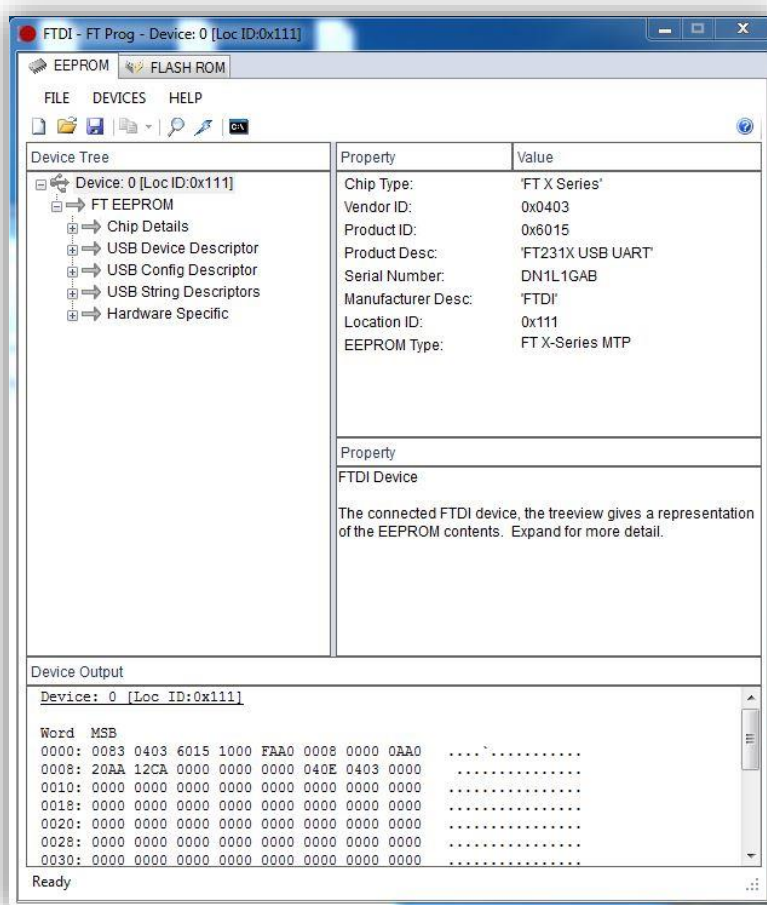
**Hinweis:** Bei fertigen Platinen / Geräten die aus dem Entwicklerkreis kommen ist die Vorbereitung schon erfolgt und muss nicht mehr durchgeführt werden.

Das Programm **FT Prog** kann hier heruntergeladen werden:

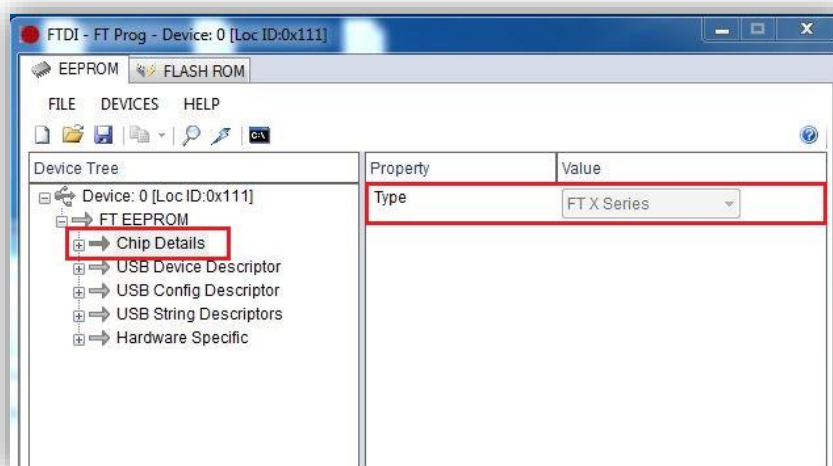
[http://www.ftdichip.com/Support/Utilities.htm#FT\\_PROG](http://www.ftdichip.com/Support/Utilities.htm#FT_PROG)

### Empfohlene Vorgehensweise:

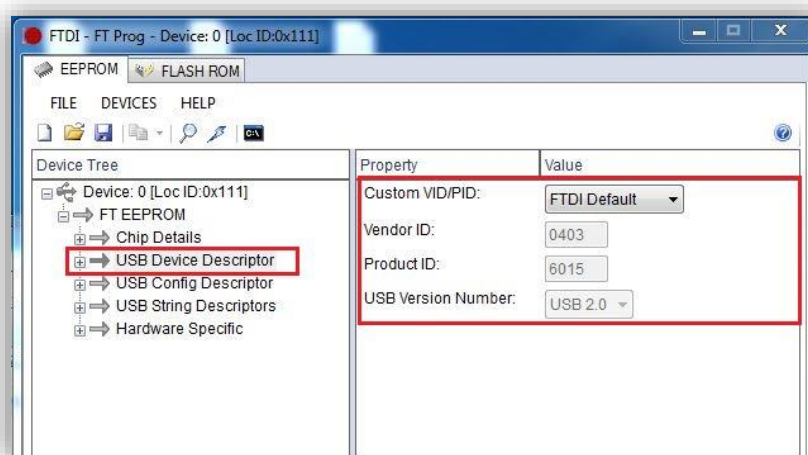
1. Das Programm **FT Prog** öffnen
2. Das USB Device anschließen und danach suchen: 
3. Wenn das Device gefunden wurde sollte es so aussehen:



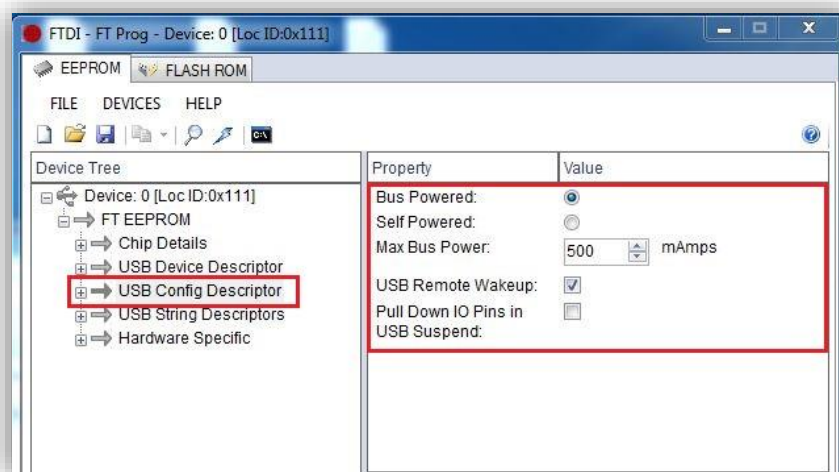
4. Im Bereich *Chip Details* sollte „FT X Series“ stehen:



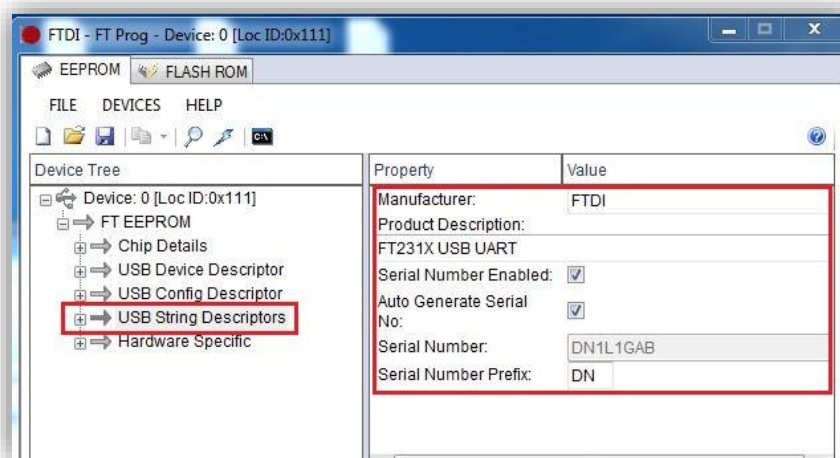
5. Bereich *USB Device Descriptor* bleibt unverändert:



6. Im Bereich *USB Config Descriptor* muss folgendes eingestellt werden:

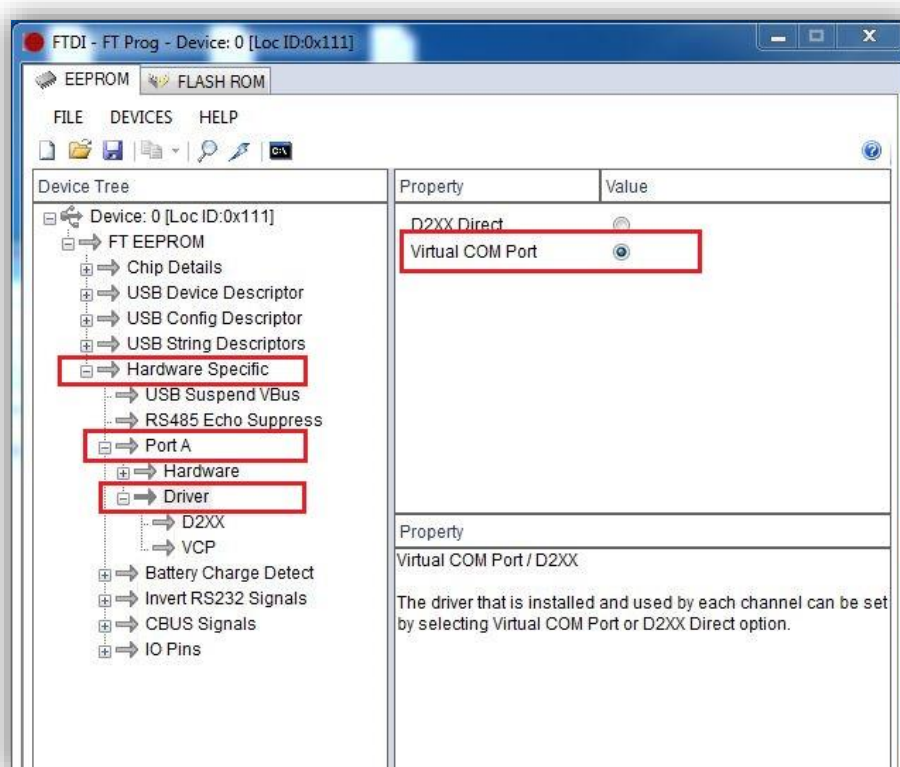


## 7. Der Bereich *USB String Descriptors* bleibt unverändert

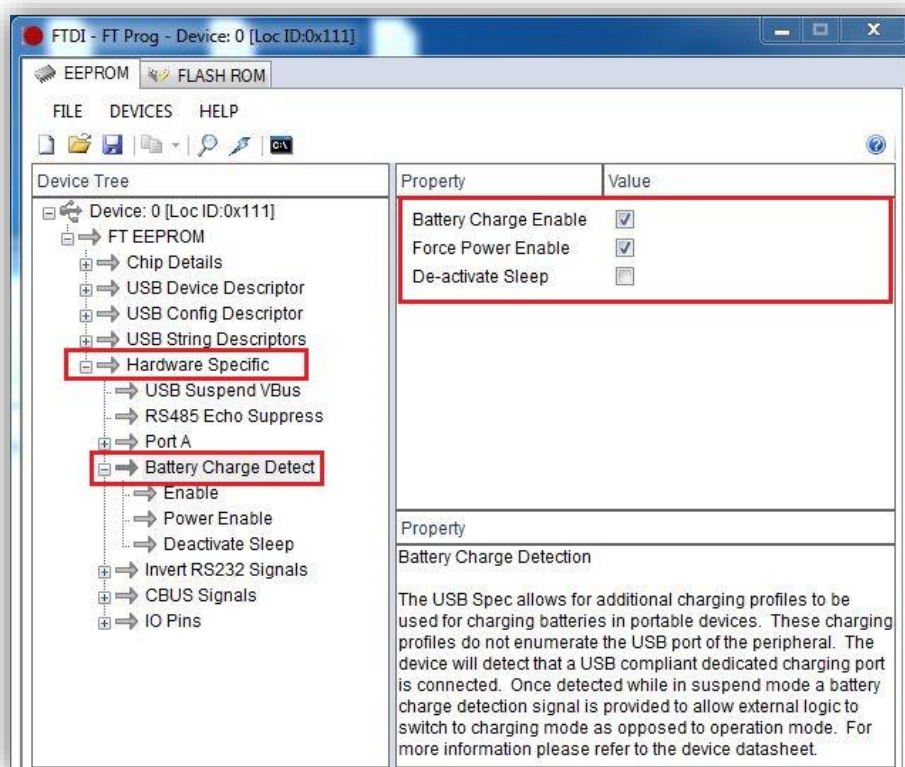


## 8. Bereich *Hardware Specific*:

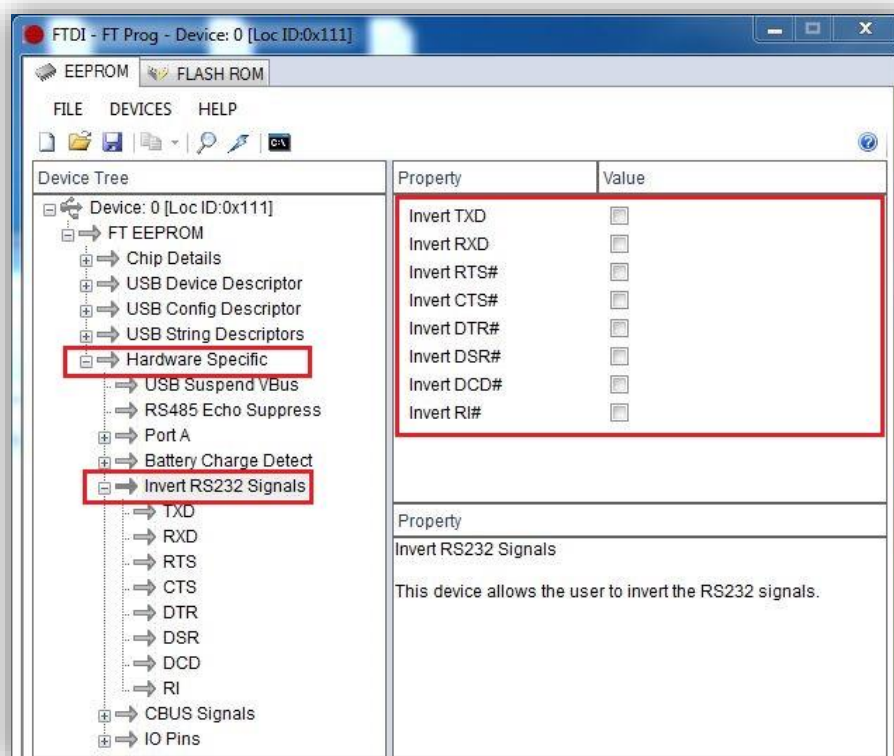
- a. *Port A* => *Driver* sollte „Virtual COM Port“ eingestellt sein:



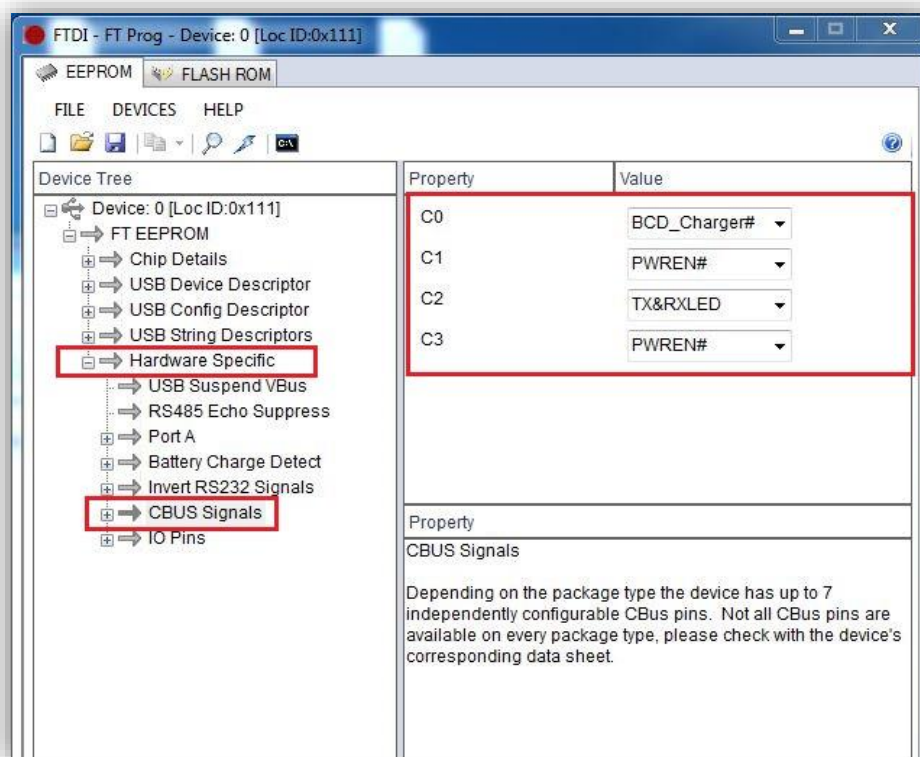
b. Bereich *Battery Charge Detect* folgendes einstellen:



c. Im Bereich *Invert RS232 Signals* darf kein Haken gesetzt sein:



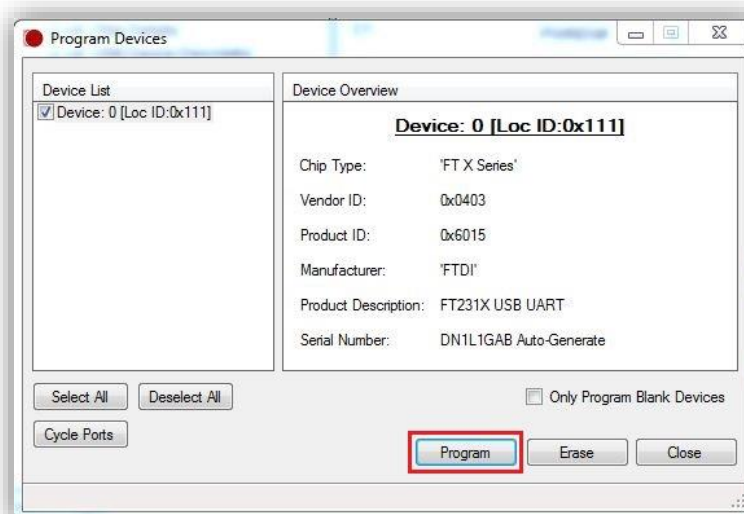
d. Bereich *CBUS Signals* muss folgendes eingestellt werden:



e. Jetzt kann das Device programmiert werden => Klick:



Es öffnet sich ein Fenster => auf „Program“ drücken:



Wenn der Vorgang abgeschlossen ist, sollte das Programm **FT Prog** beendet und das Device vom USB Port getrennt werden.

**!!!Achtung!!!**

Erst nachdem das Device wieder an den USB Port angeschlossen und neu gestartet wurde, sind die Änderungen wirksam.

Da das Device neu konfiguriert wurde kann es sein, dass auch ein neuer USB Treiber installiert werden muss. Dieses kann im Gerätemanager überprüft werden. Des Weiteren kann dem Device nach der neuen Treiber Installation auch ein anderer COM Port zugewiesen worden sein.



## 5.3 Arduino IDE

Im folgenden Abschnitt ist die richtige Installation der Arduion IDE beschrieben und zusätzlich erklärt wie die diverse Komponenten, die zum Flashen der Nano notwendig sind, in die Arduion IDE Umgebung integriert werden können. Die Installation ist am Beispiel eines Windows 7 Systems beschrieben.

### 5.3.1 Arduino IDE installieren: Version > 1.6.11

Hier kann die aktuellste Version heruntergeladen werden:

<https://www.arduino.cc/en/Main/Software>

Es muss die komplette Installation inklusiv der Treiber durchgeführt werden.

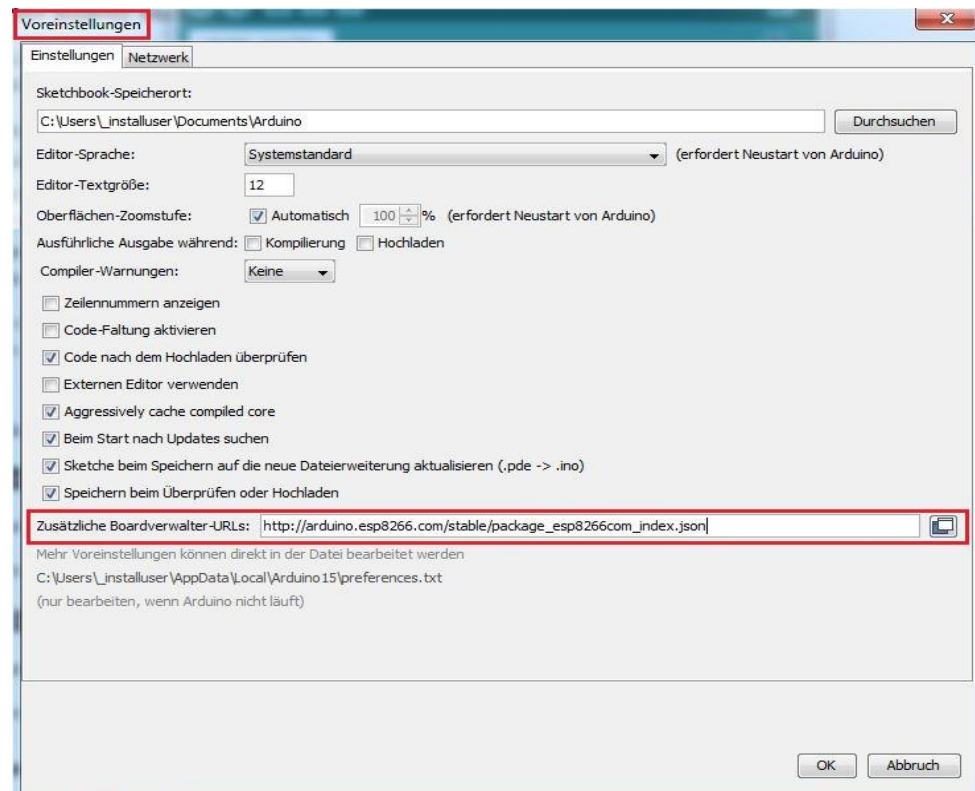
### 5.3.2 ESP8285 ESP 8266 - Core / Board Unterstützung installieren

1. Arduino IDE starten und in dem Menü Pfad :

*Datei => Voreinstellungen => Zusätzliche Boardverwalter-URLs*

folgende Adresse eintragen:

[http://arduino.esp8266.com/stable/package\\_esp8266com\\_index.json](http://arduino.esp8266.com/stable/package_esp8266com_index.json)



## Hinweis:

Die grün gekennzeichneten Pfad Bereiche die im folgenden Text zu finden sind, können bei jeder Installation abweichen!

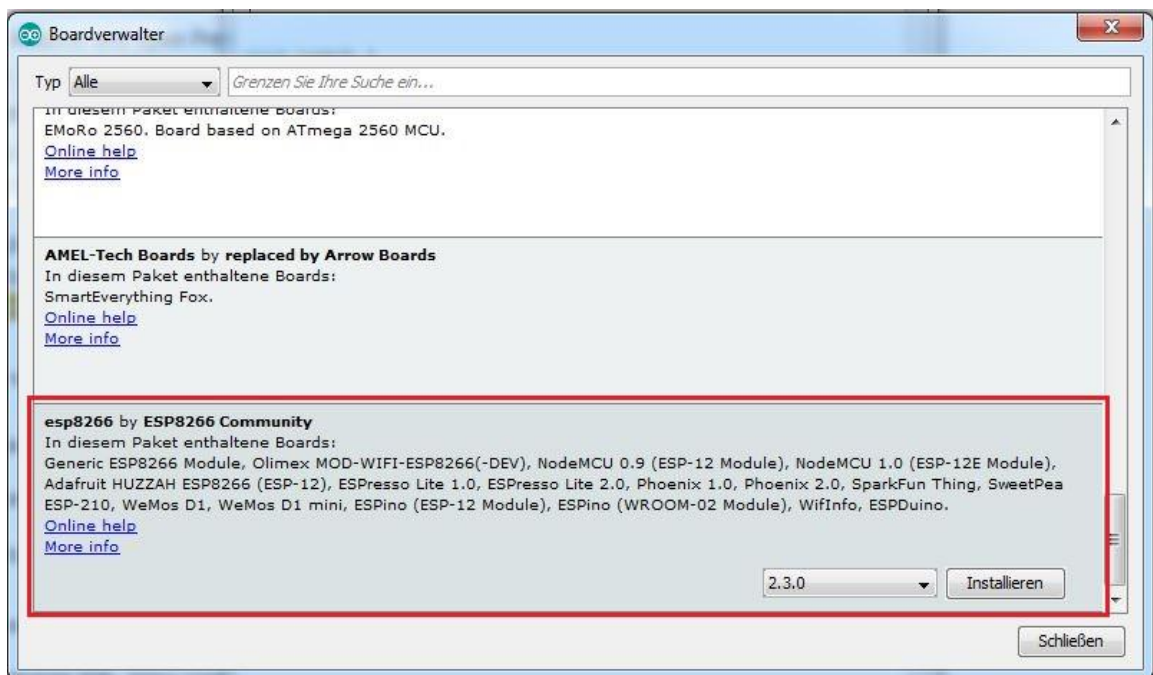
=> Installierter User Name und Sprachversion des Betriebssystems.

## 2. Boardverwalter öffnen:

*Werkzeuge => Board => Boardverwalter*

Nach „esp8266“ suchen bzw. nach unten scrollen bis der nachfolgende Abschnitt zu sehen ist. Nach einem Klick in das Fenster des esp8266 erscheint die Auswahl der zu installierenden Version und der Install Button.

Hinweis: Die Installation kann einige Zeit dauern.

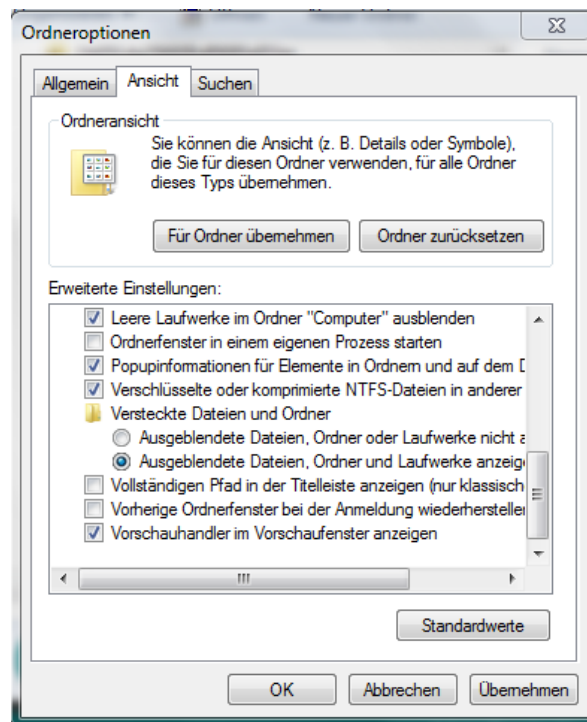




### 5.3.3 Erweiterung der board.txt

Sollte es noch nicht geschehen sein, müssen die ausgeblendeten Dateien und Ordner erstmal sichtbar gemacht werden.

Organisieren => Ordner und Suchoptionen => Ansicht



Die Datei *boards.txt* findet man in folgendem Verzeichnis:

```
C:\Users\_installuser\AppData\Local\Arduino15\packages\esp8266\hardware\esp8266\2.3.0
```

Den Bereich esp8285 suchen und folgende Zeilen einfügen:

```
esp8285.menu.ResetMethod.ck=ck
```

```
esp8285.menu.ResetMethod.ck.upload.resetmethod=ck
```

```
esp8285.menu.ResetMethod.nodemcu=nodemcu
```

```
esp8285.menu.ResetMethod.nodemcu.upload.resetmethod=nodemcu
```

### 5.3.4 Installation Arduino Plugin - ESP8266 Sketch Data Upload

Das Plugin kann hier heruntergeladen werden:

<https://github.com/esp8266/arduino-esp8266fs-plugin/releases/tag/0.3.0>

Hier ist es wichtig dass nicht die Branch sondern die Release Version herunter geladen wird.

Vor der Installation des Plugins muss sichergestellt werden, dass der ESP8266 Core (siehe oben) installiert ist.

Nach dem Download und Entpacken des Files muss der Ordner nach „esp8266fs“ in den folgenden Ordner kopiert werden:

`C:\Program Files (x86)\Arduino\tools\`

Sollte es das erste installierte Tool sein, ist es möglich dass der Unterordner *tools* erstellt werden muss.

Nach dem Kopieren muss Arduino IDE neu gestartet werden.

### 5.3.5 Time - Libary integrieren

Die Library kann hier heruntergeladen werden:

Clone or download ▾

<https://github.com/PaulStoffregen/Time>

Nach dem Entpacken des Files muss der Ordner nach „Time“ umbenannt und in den folgenden Ordner kopiert werden:

`C:\Users\_installuser\Documents\Arduino\libraries`

Alternativ kann man die libraries in diesem Pfad ablegen:

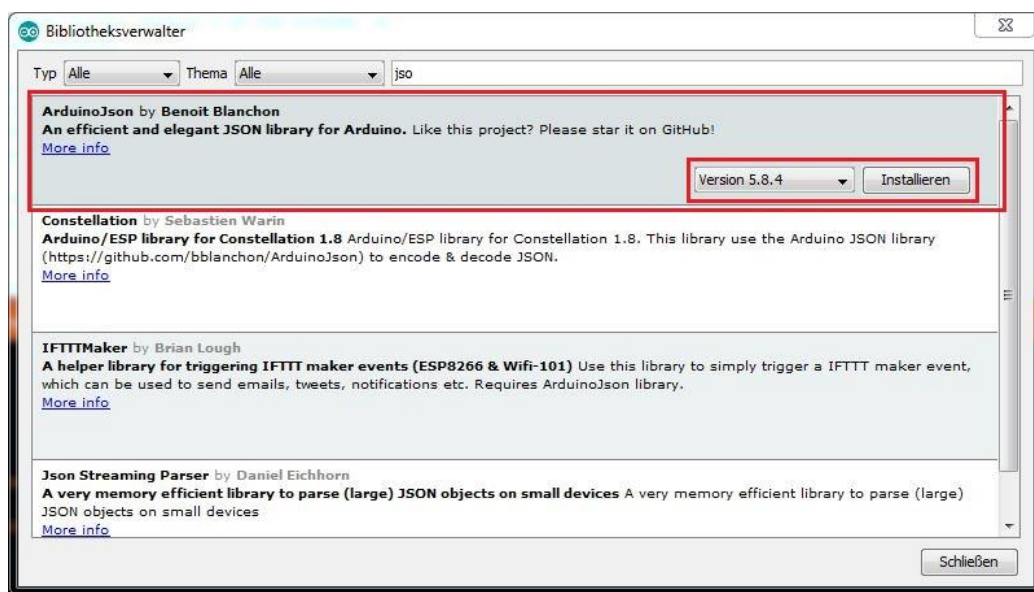
`C:\Program Files (x86)\Arduino\libraries`

### 5.3.6 ArduinoJson - Library einbinden

Bibliothekenverwalter öffnen:

*Sketch => Bibliothek einbinden => Bibliotheken verwalten*

Nach „ArduinoJson“ suchen bzw. nach unten scrollen bis der nachfolgende Abschnitt zu sehen ist. Nach einem Klick in das Fenster der ArduinoJson erscheint die Auswahl der zu installierenden Version und der „Installieren“ Button.



### 5.3.7 Projekt spezifische Librarys integrieren

Das Projekt inklusive der Librarys kann hier heruntergeladen werden:

[https://github.com/WLANThermo-nano/WLANThermo\\_nano\\_Software](https://github.com/WLANThermo-nano/WLANThermo_nano_Software)

Clone or download ▾

Nach dem Download und Entpacken des Files findet man im Ordner „\_includes“ folgende Unterordner :

SSD1306; ESPAsyncTCP; ESPAsyncWebServer; async-mqtt-client

Diese 3 Verzeichnisse müssen in folgenden Ordner kopiert werden:

C:\Users\\_installuser\Documents\Arduino\libraries

Alternativ kann man die libraries in diesem Pfad ablegen:

C:\Program Files (x86)\Arduino\libraries

Nach dem Kopieren muss Arduino IDE neu gestartet werden

**!!!Achtung!!!** Bis auf weiteres müssen noch folgende Anpassungen gemacht werden:

Im folgenden Verzeichnis findet man die Datei „*time.h*“:

```
C:\Users\_installuser\AppData\Local\Arduino15\packages\esp8266  
\tools\xtensa-lx106-elf-gcc\1.20.0-26-gb404fb9-2\xtensa-lx106-  
elf\include
```

Die Datei „*time.h*“ muss einmal kopiert und diese Kopie dann nach „*\_time.h*“ umbenannt werden.

In zweiten Schritt muss man im Verzeichnis 1 oder Verzeichnis 2, je nachdem wo man die library ESPAsyncWebServer installiert hat, die Datei *WebHandlerImpl.h* editieren.

Verzeichniss 1:

```
C:\Program Files (x86)\Arduino\libraries\ESPAsyncWebServer\src
```

oder Verzeichnis 2:

```
C:\Users\_installuser\Documents\Arduino\libraries\ESPAsyncWebS  
erver\src
```

Man öffnet die Datei *WebHandlerImpl.h* mit einem Editor (z.B. Notepad++) und ändert den Eintrag:

von: `#include <time.h>` nach `#include <_time.h>`

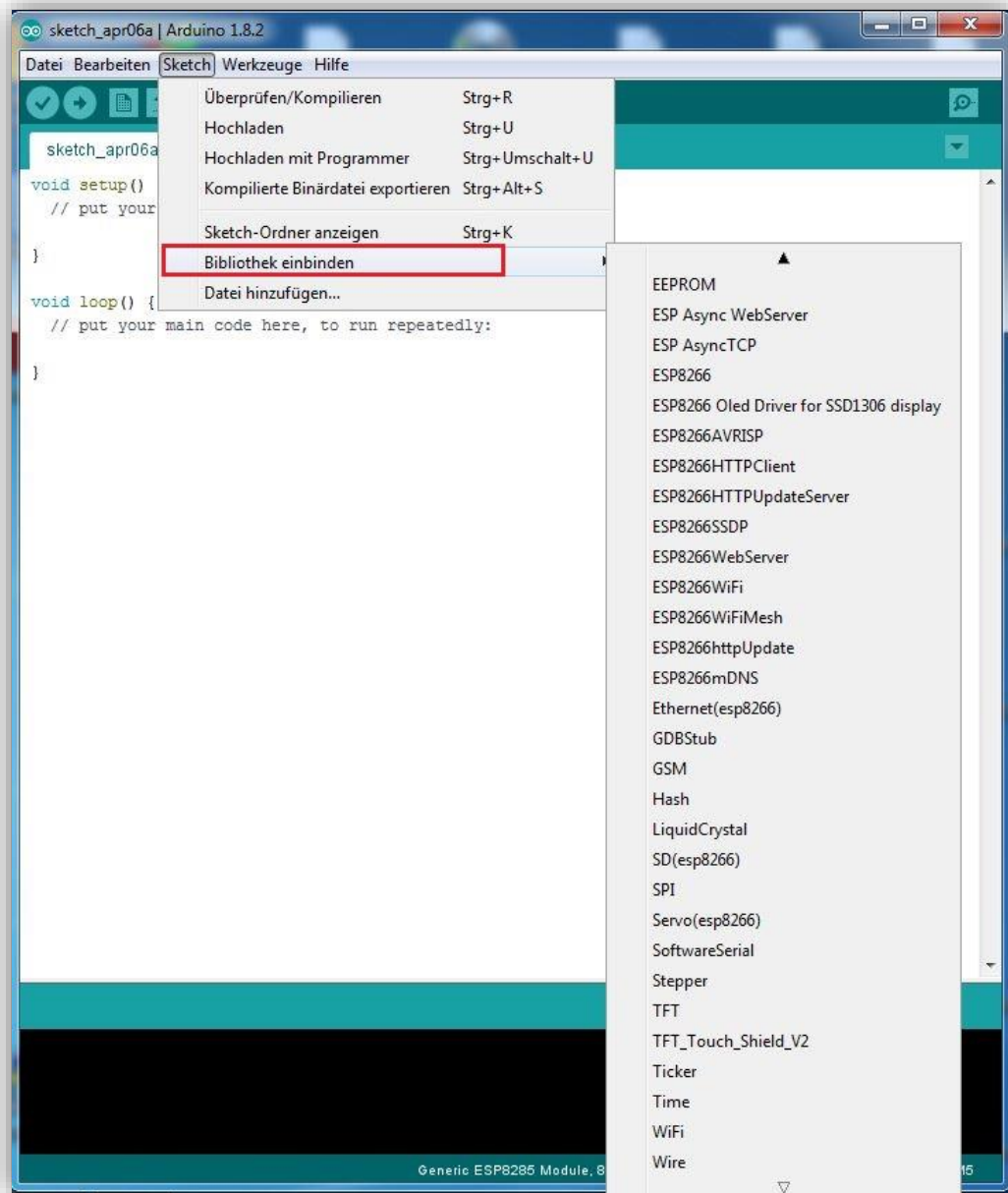
### **Hinweis:**

Da diese Datei meistens nur mit Administrator Rechten editierbar ist, sollte nach dem speichern der Datei sicherheitshalber nochmal geprüft werden ob die Änderung übernommen wurde!!!

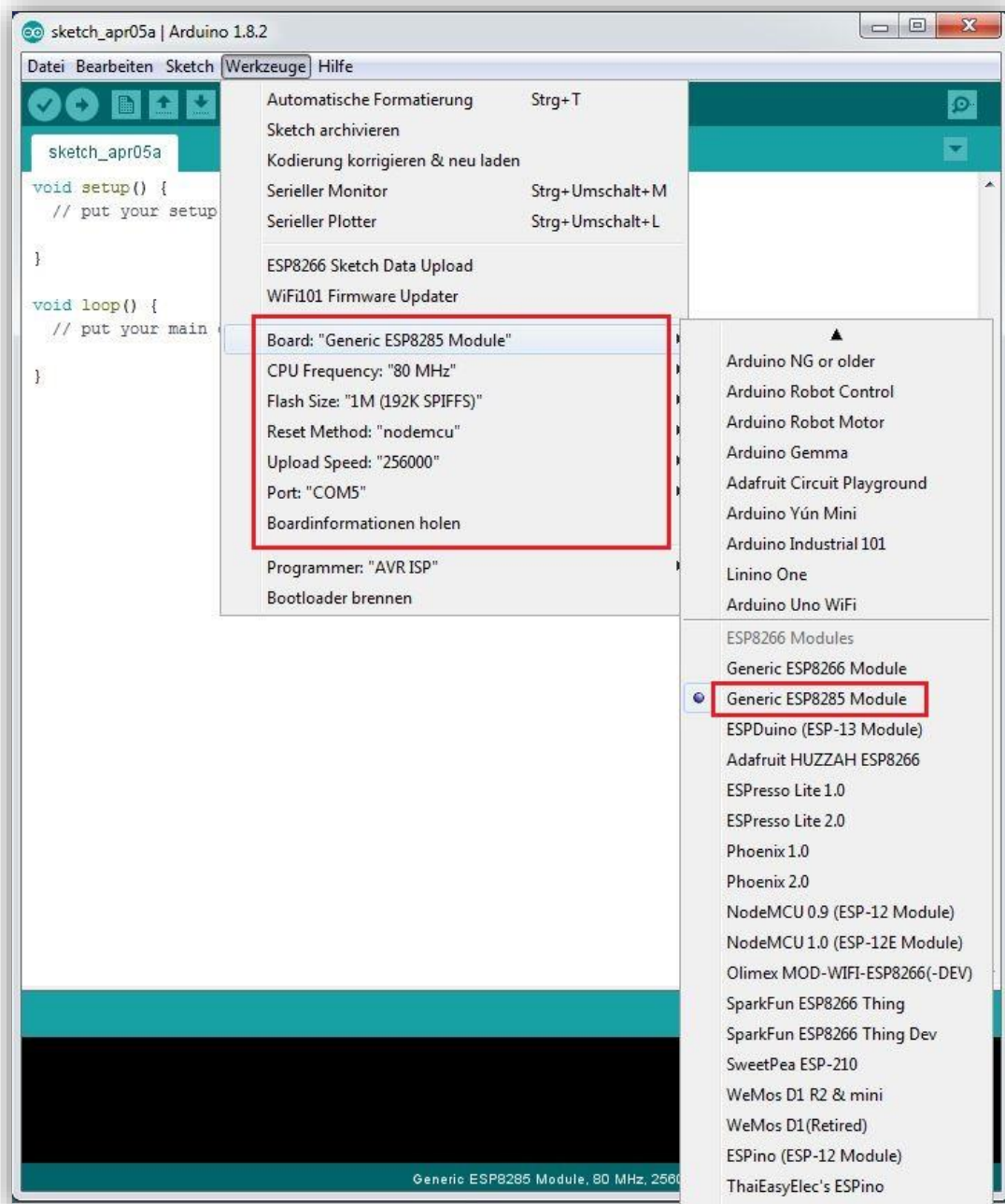
Nachdem die geänderte Datei Abgespeichert wurde muss das Programm Arduino IDE neu gestartet werden.

Wenn die Vorbereitungen erledigt sind, sollten in den diversen Bereichen von Arduino IDE die installierten und konfigurierten Komponenten zu sehen sein:

## 1. Bibliothek:



## 2. Werkzeuge und Board



## 5.4 Der Flashvorgang

Wenn alles für den Flashvorgang vorbereitet ist, kann man damit jetzt weiter machen.

Das aktuelle NANO Projekt kann hier heruntergeladen werden:

Clone or download ▾

[https://github.com/WLANThermo-nano/WLANThermo\\_nano\\_Software](https://github.com/WLANThermo-nano/WLANThermo_nano_Software)

Nach dem Download und Entpacken des Projektes startet man das vorbereitete **Arduino IDE** Programm und öffnet die Datei \*.ino aus dem entpackten Projektverzeichnis.

**!!! Achtung !!!** das Projektverzeichnis muss „Nemesis“ heißen, sollte es nicht der Fall sein, muss es entsprechend vorher umbenannt werden.

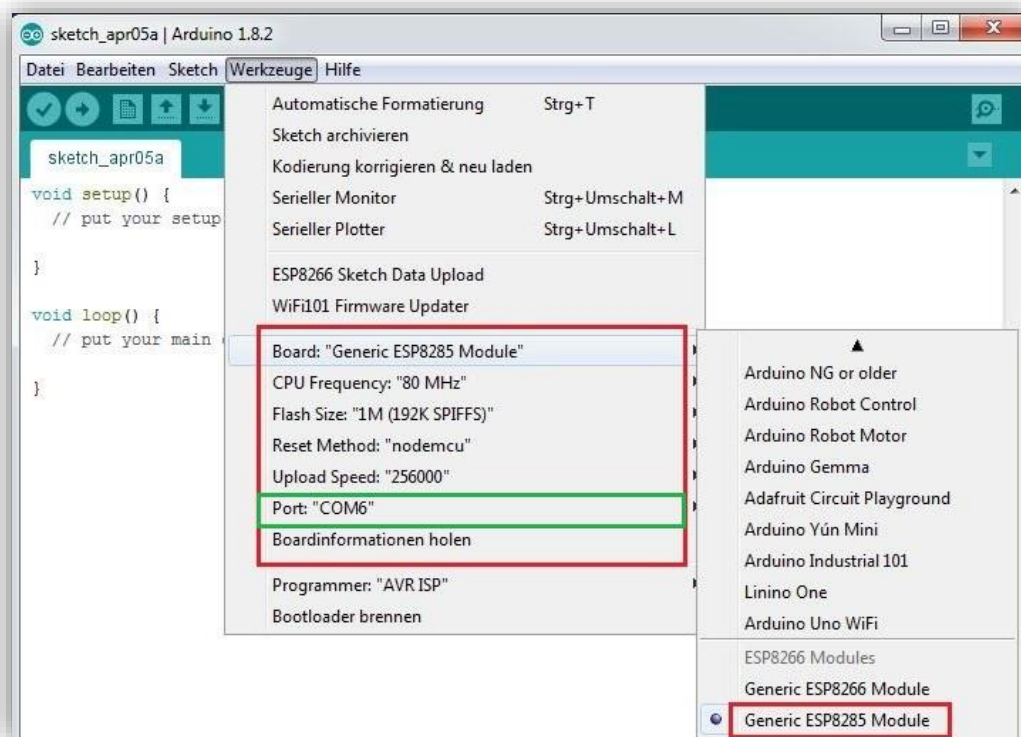
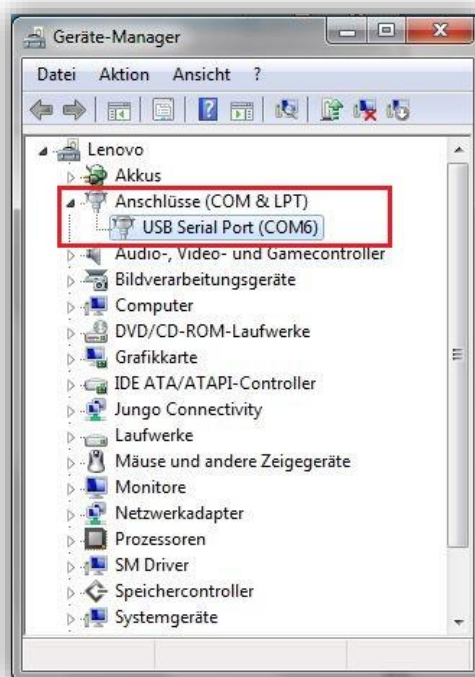
Der Inhalt der „Nemesis“ Verzeichnisses sieht dann so aus:

Name	Typ	Größe
_include	Dateiordner	
data	Dateiordner	
c_bot.h	H-Datei	4 KB
c_frames.h	H-Datei	17 KB
c_fs.h	H-Datei	24 KB
c_icons.h	H-Datei	8 KB
c_init.h	H-Datei	23 KB
c_median.h	H-Datei	3 KB
c_ota.h	H-Datei	5 KB
c_pitmaster.h	H-Datei	5 KB
c_sensor.h	H-Datei	7 KB
c_server.h	H-Datei	12 KB
c_temp.h	H-Datei	5 KB
c_wifi.h	H-Datei	7 KB
LICENSE.md	MD-Datei	35 KB
Nemesis.ino	Arduino file	6 KB
README.md	MD-Datei	1 KB



- Auswahl der COM Schnittstelle:

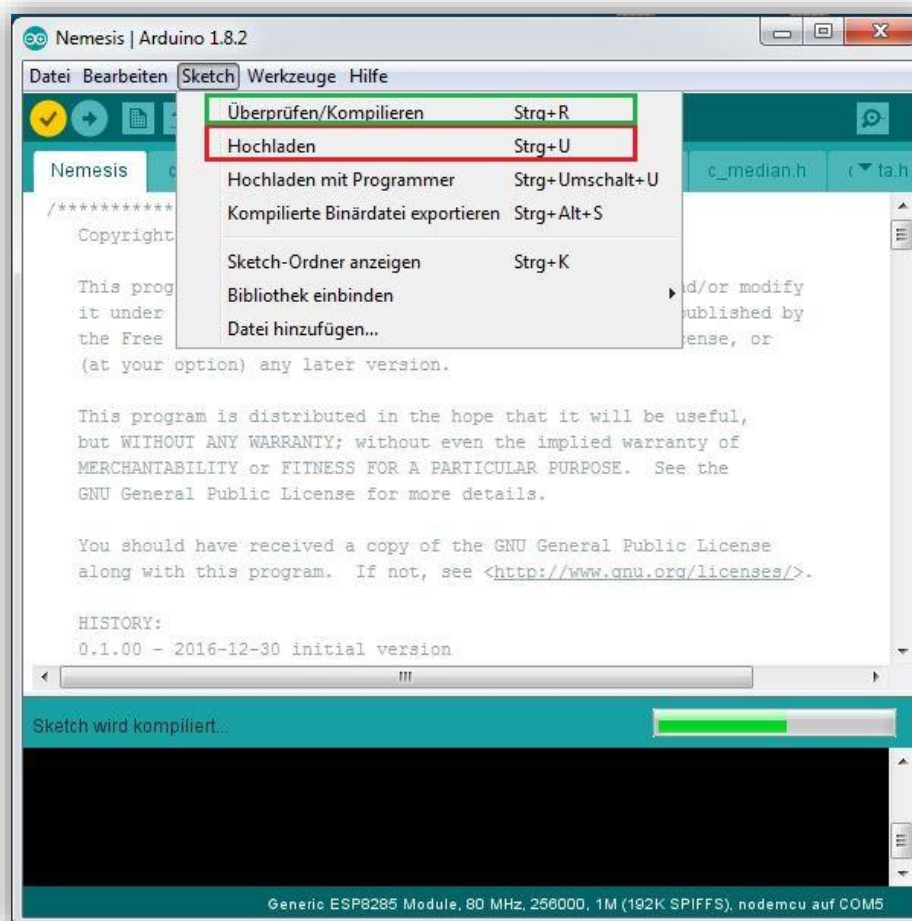
Die zu flashende NANO muss über ein entsprechendes USB Kabel angeschlossen werden. Wenn das Device erkannt wurde, kann im Gerätemanager die entsprechende COM Schnittstelle ausgelesen und in dem entsprechenden Feld ausgewählt werden. (z.B. COM 6)





Wenn man „Überprüfen/Kompilieren“ anwählt kann man, wie der Menüpunkt es schon sagt den Sketch / das Projekt prüfen ob es fehlerfrei kompilierte werden kann.

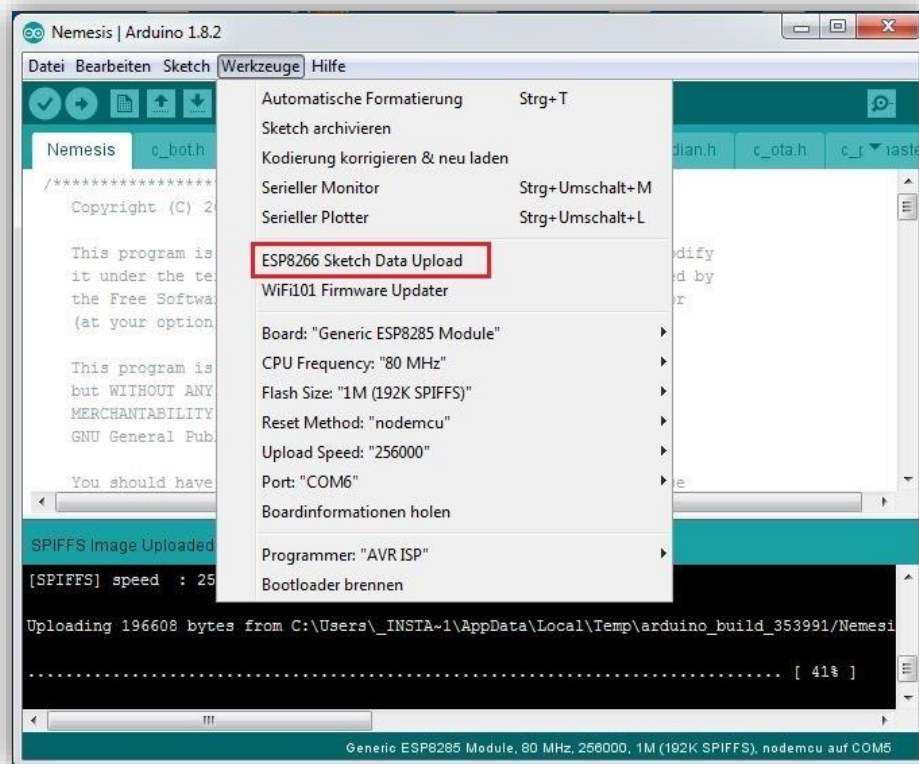
Sollten keine Fehlermeldungen im Terminalfenster erscheinen kann man über den Menüpunkt „Hochladen“ die neue Software auf die NANO flashen.



## 5.5 Übertragung der HTML Komponenten

Die HTML Komponenten müssen in das Filesystem des ESP 8285 Controllers übertragen werden. Dieses geschieht mit Hilfe des im **Arduino IDE** hinzugefügtem Plugin: **ESP8266 Sketch Data Upload**.

Für den Flashvorgang der NANO wurde das Projekt von GitHub heruntergeladen und entpackt. Im Unterverzeichnis „data“ befinden sich die benötigten HTML Komponenten.



Durch das Anwählen des Menüpunktes: *ESP8266 Sketch Data Upload* startet der Upload der HTML Komponenten. Der Fortschritt kann im Terminalfenster beobachtet werden.

## 5.6 Verbindung mit dem Terminalprogramm PUTTY

Um eine serielle Verbindung zur NANO mittels PUTTY aufzubauen wird empfohlen folgende Einstellungen vorzunehmen:

Das Programm **PUTTY** kann hier heruntergeladen werden: <http://www.putty.org/>

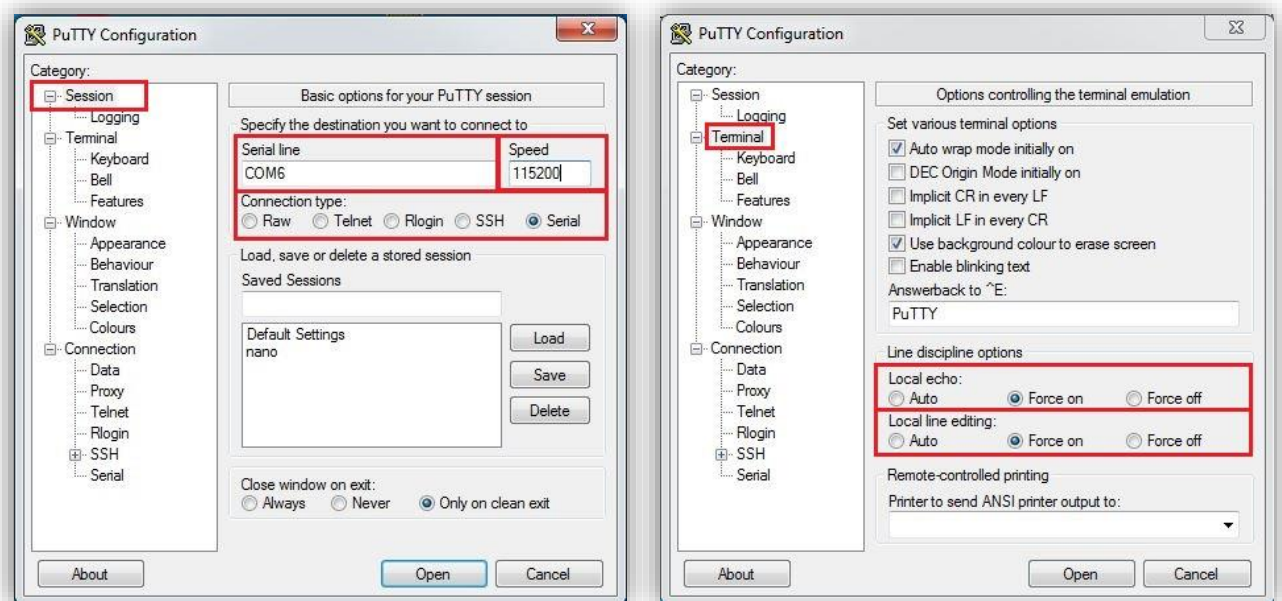
Bereich *Session*:

- *Conection type*: Serial auswählen
- *Serial line*: COM6 – hier wird die COM Schnittstelle die im Gerätemanager ausgelesen wurde eingetragen
- *Speed*: 115200

Bereich *Terminal*:

- *Local echo*: Force on
- *Local line editing*: Force on

Werden diese Optionen im Bereich Terminal angewählt, kann man den Befehl im Terminalfenster eingeben, diesen überprüfen und erst dann mit Hilfe der Enter Taste zur NANO gesendet werden.



Mit den Button „Open“ kann man jetzt die Verbindung mit der NANO aufbauen und es öffnet sich automatisch ein Terminalfenster.

Wenn die serielle Verbindung steht kann man diese testen indem man den Befehl *help* übermittelt. Wenn die NANO antworten ist die Verbindung erfolgreich aufgebaut.

## 5.7 Konfiguration der WLAN-Verbindung über die Serielle Schnittstelle

Die WLAN Konfiguration kann über die Serielle Schnittstelle und einem seriellen Terminalprogramm z.B. PUTTY an die NANO übertragen werden.

Es können bis zu 5 WLAN Konfigurationen in der NANO konfiguriert werden.

Nachdem eine Verbindung zur NANO aufgebaut ist kann man im Terminalfenster folgende Befehle eingeben:

1. Löschen der bisherigen WLAN Daten:

```
{ "command": "setWIFI", "data": [] }
```

2. Übergabe den neuen WLAN Daten:

```
{ "command": "addWIFI", "data": [ "SSID", "passwort" ] }
```

## 5.8 Die Software Schnittstellen

Hier werden die Software Schnittstellen der NANO dokumentiert ...



## **6    FAQ**

... Falls notwendig kann man hier dann die spezielle Fragen die aufkommen sammeln und zusammenfassen