

Diese Anleitung bezieht sich auf die Software Verarbeitung mittels der Arduino Boards „**Nano**“, „**Uno**“ und **Kompatible** !

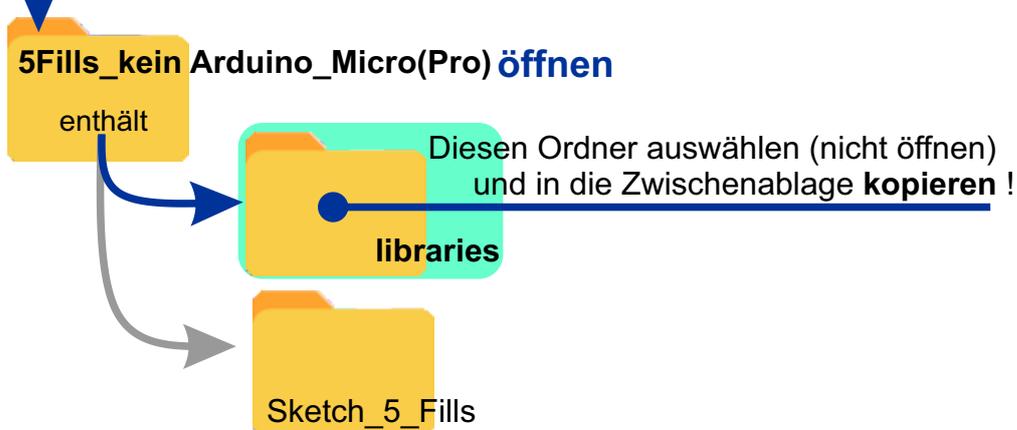


Nicht für „Pro Micro“ oder „Micro (Pro)“

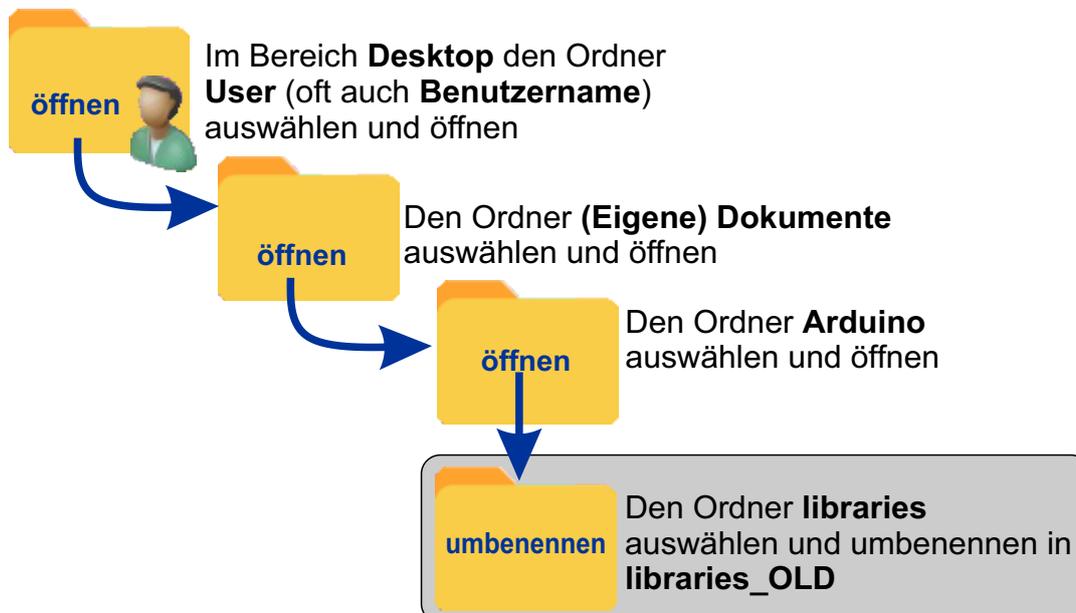
Voraussetzung: Das Programm „**Arduino (IDE)**“ ist geladen und installiert !

Schritt 1

Den entpackten Ordner



Schritt 2



Nun aus der Zwischenablage den zuvor kopierten Ordner „**libraries**“ (aus dem Ordner **5Fills_kein Arduino_Micro(Pro)**) in den geöffneten Ordner **Arduino** einfügen !

Im immer noch geöffneten Ordner **Arduino** befinden sich nun 2 Ordner !



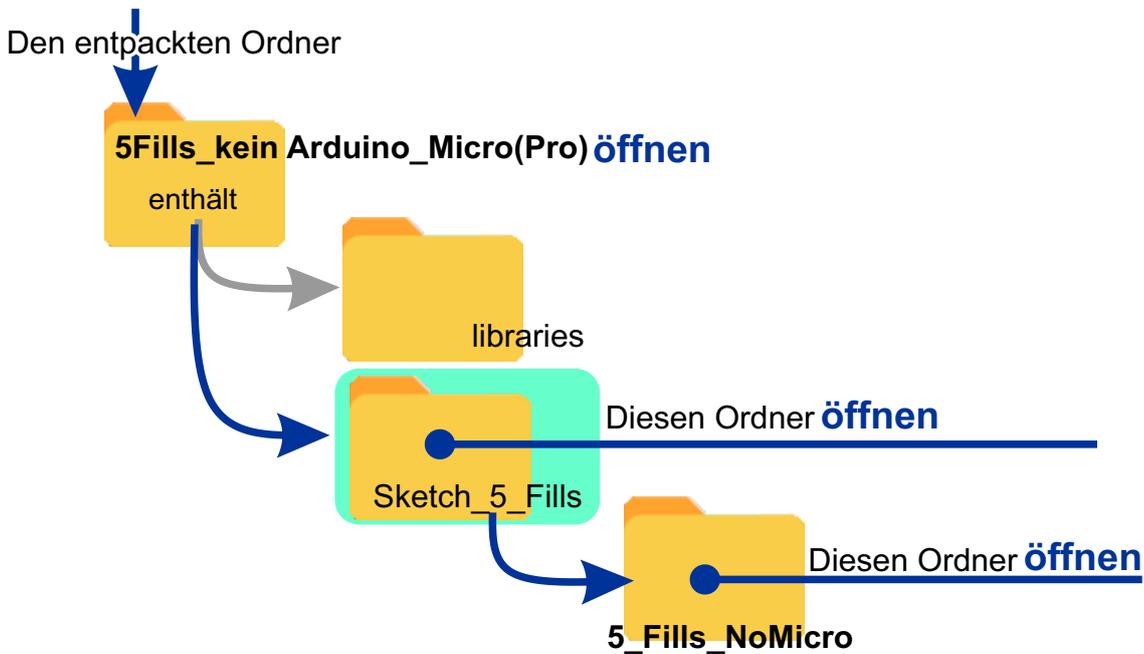
libraries

libraries_OLD

Damit sind die vorbereitenden Arbeiten in diesem Bereich abgeschlossen.

Nun wieder zurück in den Ordner **5Fills_kein Arduino_Micro(Pro)** navigieren !

Schritt 3



Dieser Ordner **5_Fills_NoMicro** enthält den Sketch **5_Fills_NoMicro.ino**

Hinweis: Ein Sketch muss sich immer in einem **gleichnamigen Ordner** befinden !

Schritt 4

Den Sketch **5_Fills_NoMicro.ino** öffnen und „**Werkzeuge**“ auswählen

„Board“ auswählen, auch wenn dort schon etwas (Default) eingetragen ist

Hier nun das Board auswählen, das wir für unseren Fußtreter gekauft haben. Sein Name erscheint dann hinter Board: und auch hier !
(beispielsweise hier ein „Uno“)

```
const byte SWITCH_1 = 2; // Switch_1 (Taster_1)
const byte SWITCH_2 = 3; // Switch_2 (Taster_2)
const byte SWITCH_3 = 4; // Switch_3 (Taster_3)
const byte SWITCH_4 = 5; // Switch_4 (Taster_4)
const byte SWITCH_5 = 6; // Switch_5 (Taster_5)

const byte SEND_1[] = {201, 87}; // Das ist der zum Taster
const byte SEND_2[] = {201, 88}; // sende den Wert 201,
const byte SEND_3[] = {201, 89}; // das heißt 201 (Decima
const byte SEND_4[] = {201, 90}; // und 87 (auch eine De
const byte SEND_5[] = {201, 91}; // dem entsprechend wer
```

Schritt 5

Damit sind die Vorbereitungen zum Hochladen dieses Sketches in das Arduino Board abgeschlossen. Das Board muss nun über eine USB Leitung mit dem PC verbunden werden (weil ja die entsprechenden Daten zum Steuern des Fußreters nur im PC vorliegen).

Dass dabei die entsprechenden USB-Stecker an diesem Kabel vorhanden sein müssen, entbehrt hier wohl jeden Hinweises, aber:

Auf keinen Fall ein reines USB-Ladekabel verwenden, wie man diese mittlerweile für unzählige, Akku betriebene Geräte einsetzt !

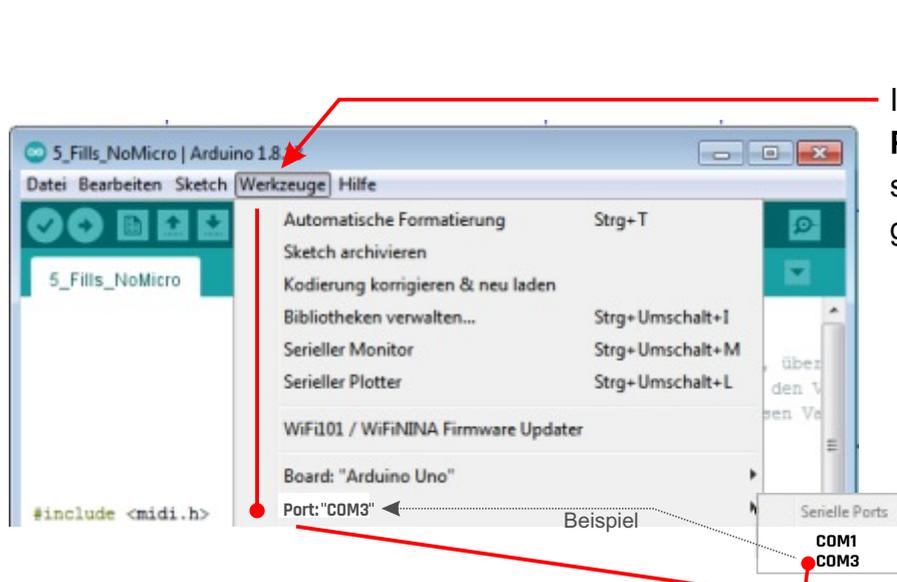
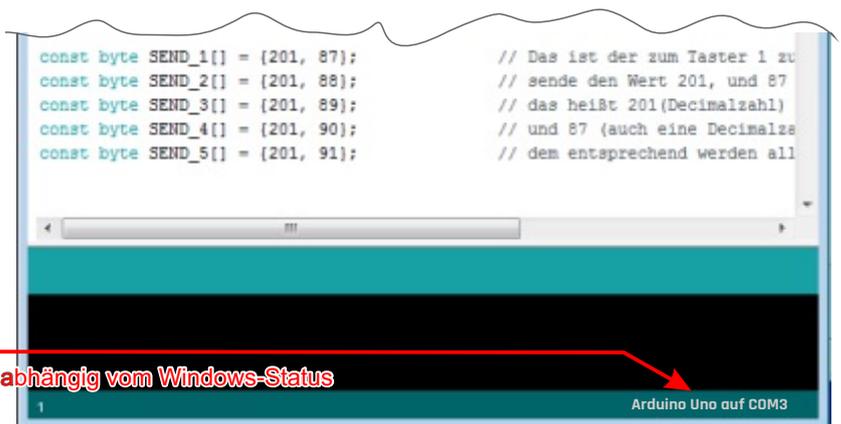
Ich benutze beispielsweise ein USB Kabel, das die Verbindung einer Digital-Kamera mit meinem PC herstellt, um Bilder zu übertragen, also Daten !

Ist dies erfolgreich abgeklärt und somit vorhanden, steht der Datenübertragung vom Sketch (PC) zum Arduino Board nichts mehr im Wege.

Ob wir das Board dabei erst *frisch* aus der Packung entnehmen, oder ob dieses schon hardwaremäßig in unserem Fußtreter verbaut ist, spielt dabei grundsätzlich keine Rolle. Nur sollte das Board in **keinem Fall mit einer externen Versorgungsspannung** verbunden sein.

Sobald das Arduino Board mit dem PC verbunden (und dadurch auch versorgt) ist, sollte es von *Windows* als USB-Gerät erkannt und einem freien Port zugewiesen werden. Hier wird dies dann auch angezeigt.

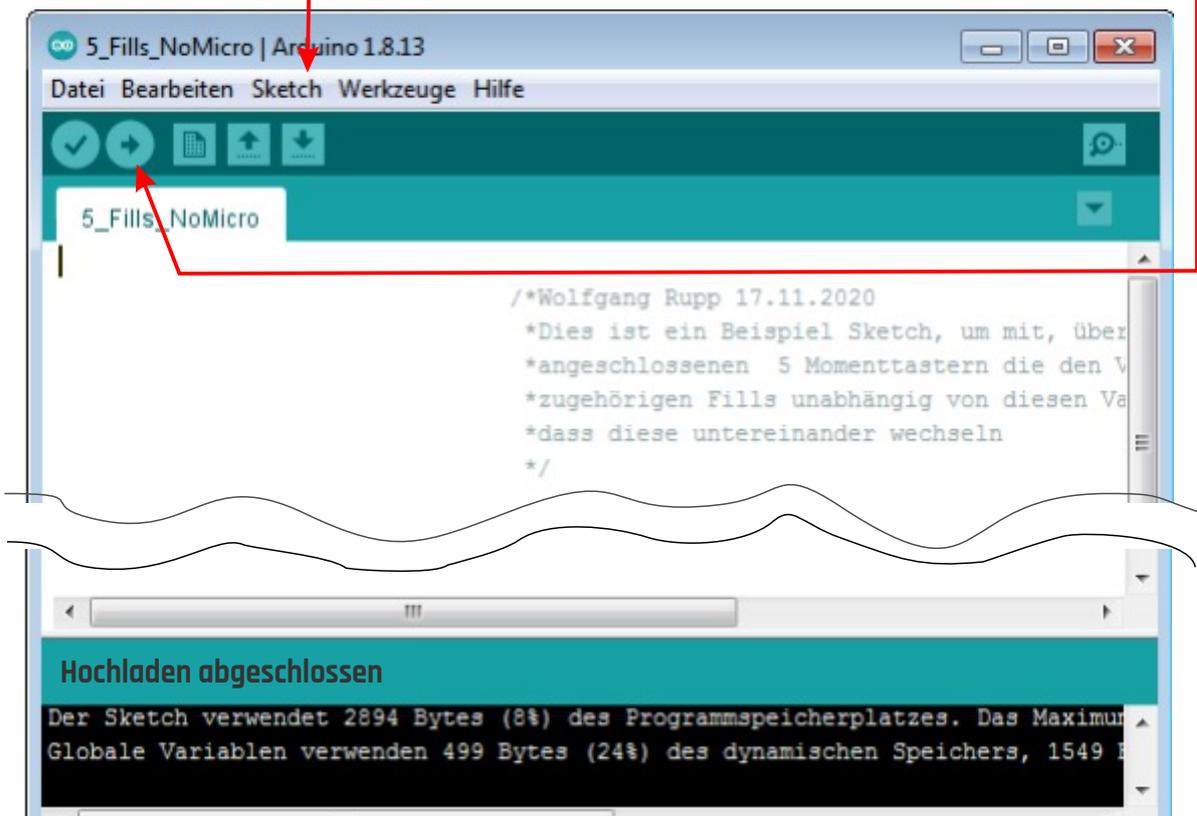
COM3 ist ein Beispiel, ist abhängig vom Windows-Status



Im Menü „**Werkzeuge**“ den Punkt **Port** auswählen und, wenn nicht schon geschehen den vorher vergebenen „**COM**“ auswählen.

Schritt 6

Jetzt entweder im Menü „Sketch“ den Punkt **Hochladen**, oder das dem entsprechendem Symbol anklicken !



Ist bis hierhin alles korrekt erledigt worden, wird der Sketch nun automatisch gespeichert, kompiliert, zum Board geladen und mit „Hochladen abgeschlossen“, abgeschlossen !

Welche „LED Farbenspiele“ auf dem Board dabei während dem angeschlossenen Zustand und dem Hochlade-Prozess angezeigt werden, kann da von Board zu Board abweichen, weshalb hier auch nicht näher darauf eingegangen wird.

Das betreffende **Arduino Board** bildet damit nun das „Betriebssystem“, beziehungsweise ist nun die **eingebaute** Software zu einem **Midi-Controller** für die individuelle **Fill In Anwahl** einiger derzeit aktueller **Korg Pa-Keyboards** !