

## **Arduino Nano: Ping Pong Wall**

Ein echter Hingucker mit 400 LEDs und Ping Pong Bällen als Diffusor.  
Alles zum Nachbau und zur Installation folgt hier:

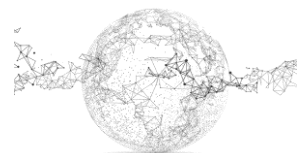
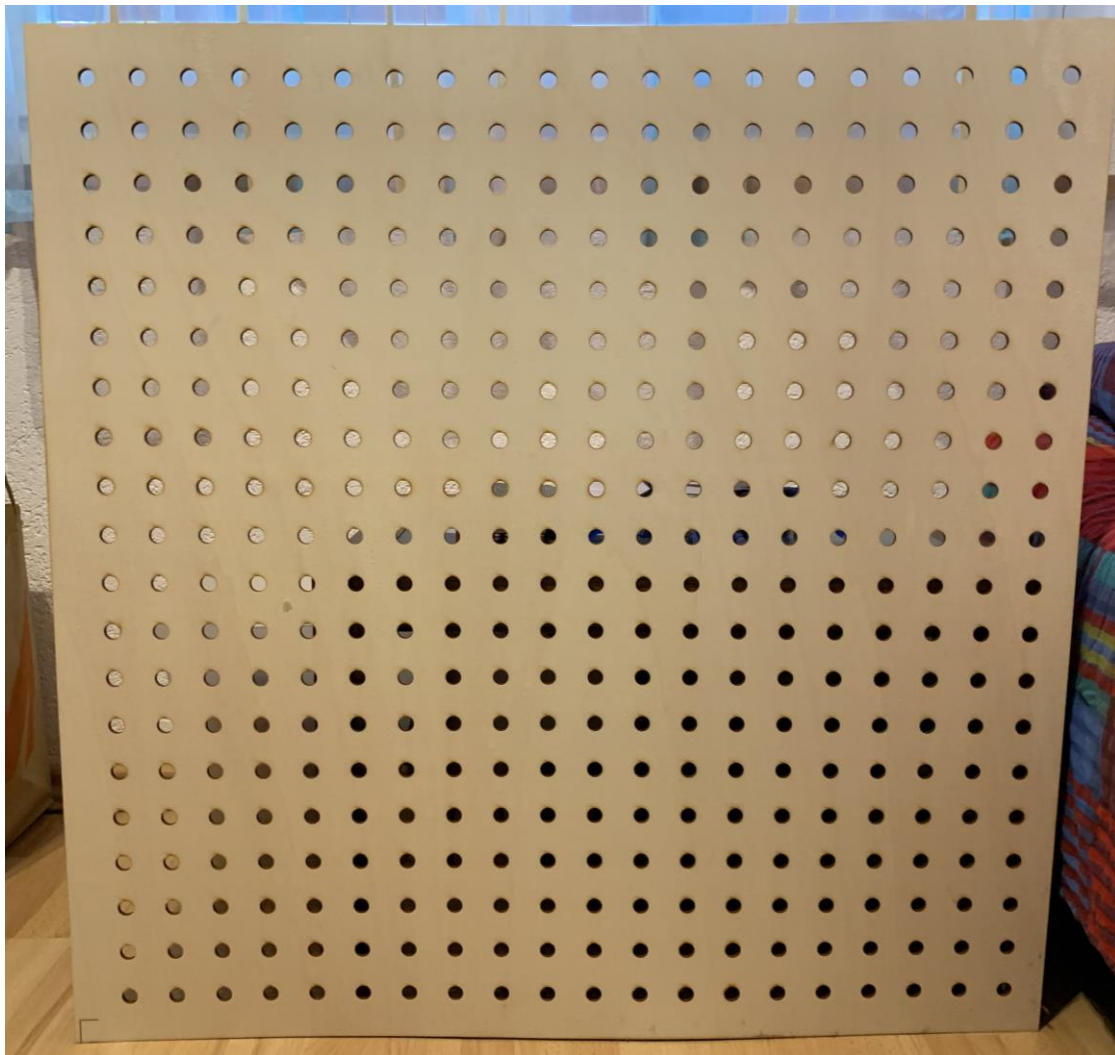
### **Rahmen und Raster:**

Eine 4 mm MDF-Platte wird auf 80x80 cm zugeschnitten und mit 400 (20x20) Löchern versehen. Die Löcher mit Vorteil an einem Laser Cutter schneiden, damit sie passgenau sind für die LEDs. Bohrung von Hand ist auch möglich.

Das Schnittmuster für den Raster liegt bei den Downloads. Geschnitten wurde mit einem Laser-Cutter.

Ein Loch hat einen Durchmesser von 12 mm und der Abstand von Lochmittelpunkt zu Lochmittelpunkt beträgt 38 mm. Mit diesem Abstand werden sich die Ping Pong Bälle schön berühren.

Die Platte auf einen Holzrahmen schrauben (vorbohren!). Dazu Fichtenlatten verwenden:  
13x57 mm



Danach mit beliebiger Farbe sprayen.



Die LED-Ketten werden hintereinander geschlauft und zusammengesteckt (8x 50 LEDs).

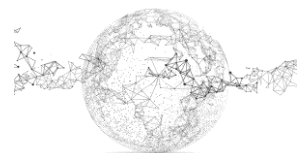


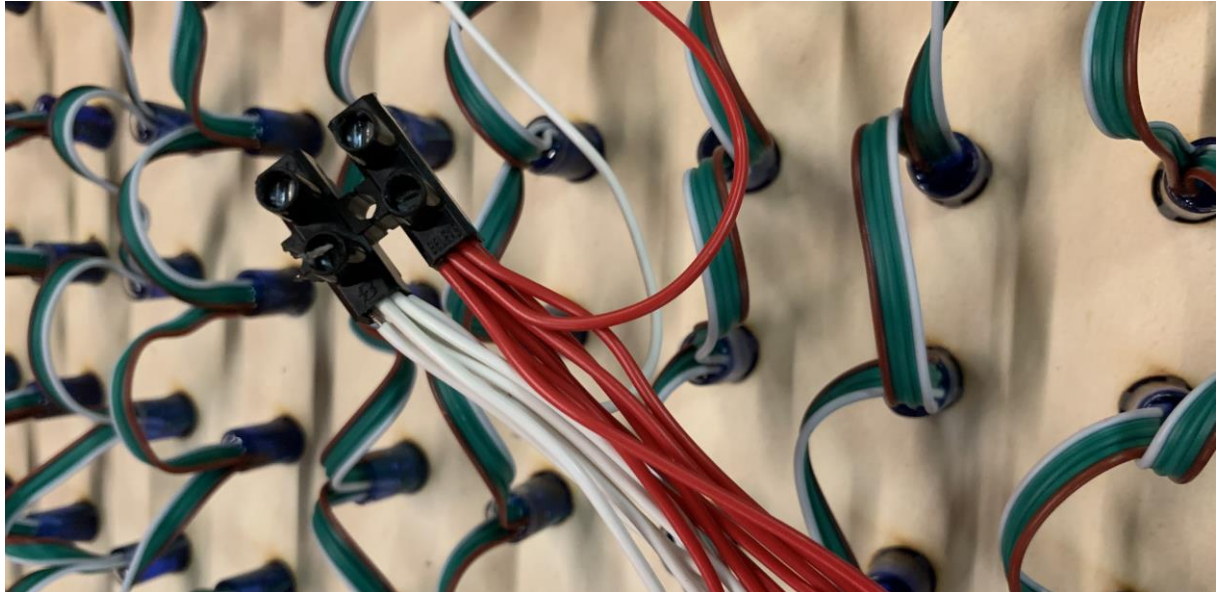
Mit Flüssigleim die LEDs von oben verleimen.

Einen Testlauf machen, ob alle LEDs brennen, lohnt sich vor dem Weiterbau!

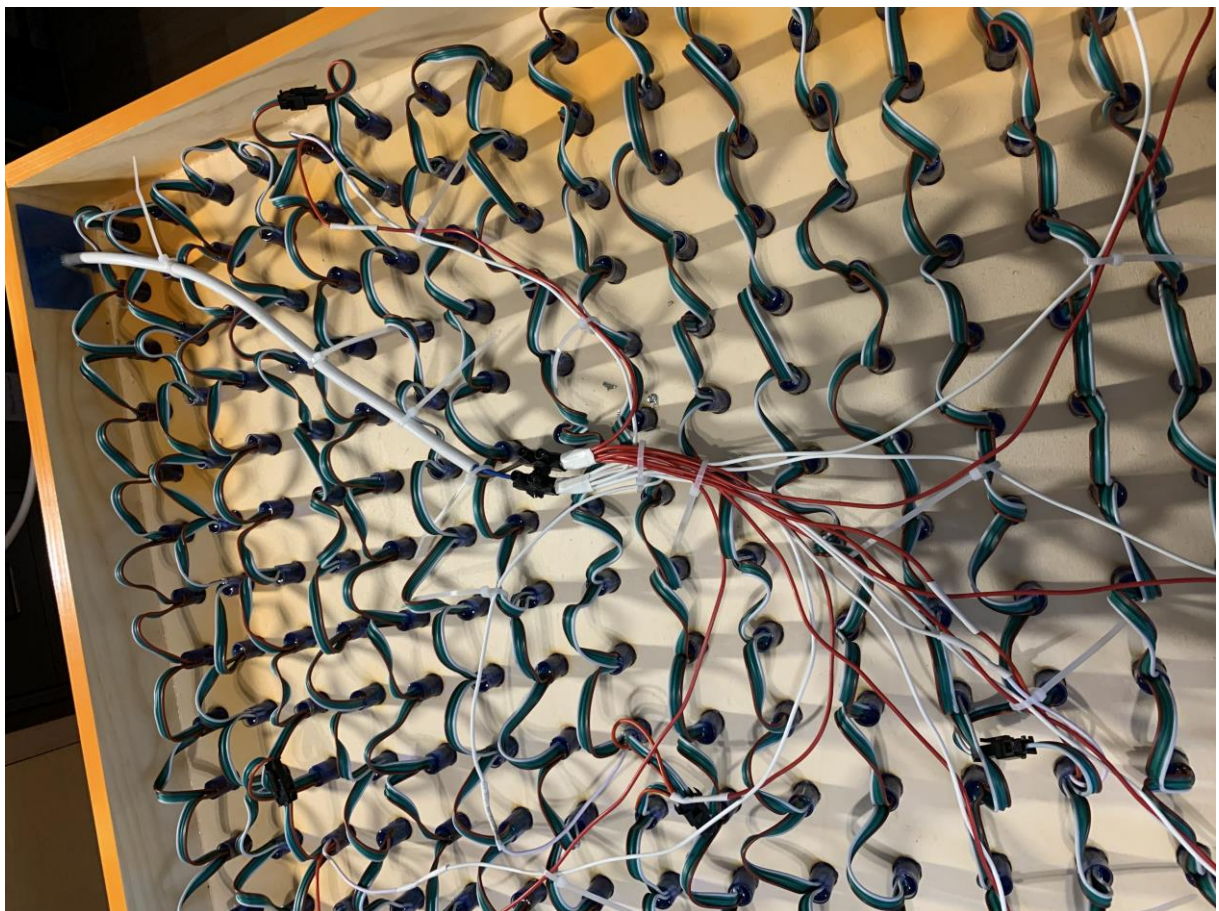
Für eine bessere Stromversorgung alle Plus- und Minus-Drähte der LEDs verlängern (nicht zu dünnen Draht wählen) und zusammennehmen. Plus: rot, Minus: weiss (grün Daten-Kabel).

Klemme verwenden.

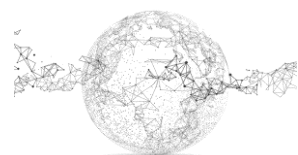




Kabel anschliessen für Strom, der aus einem Transformator kommt (30 A, 5V). Die Erdung kann man abschneiden. braun: Plus, blau: Minus

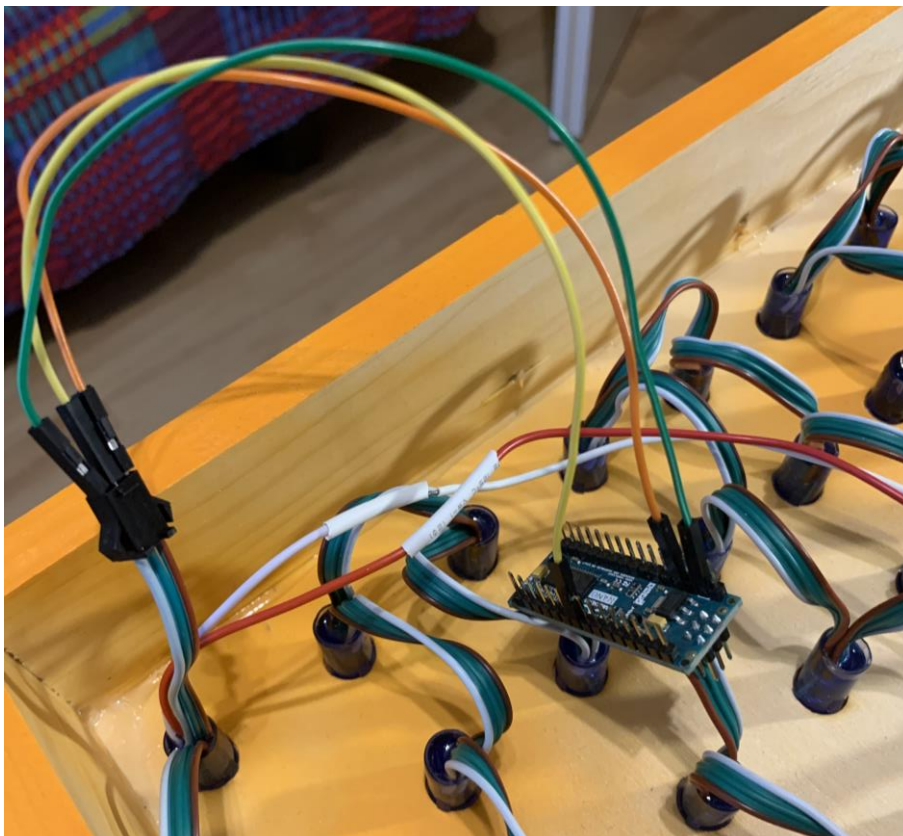


Kabel mit Kabelbinder befestigen.

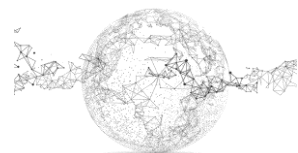




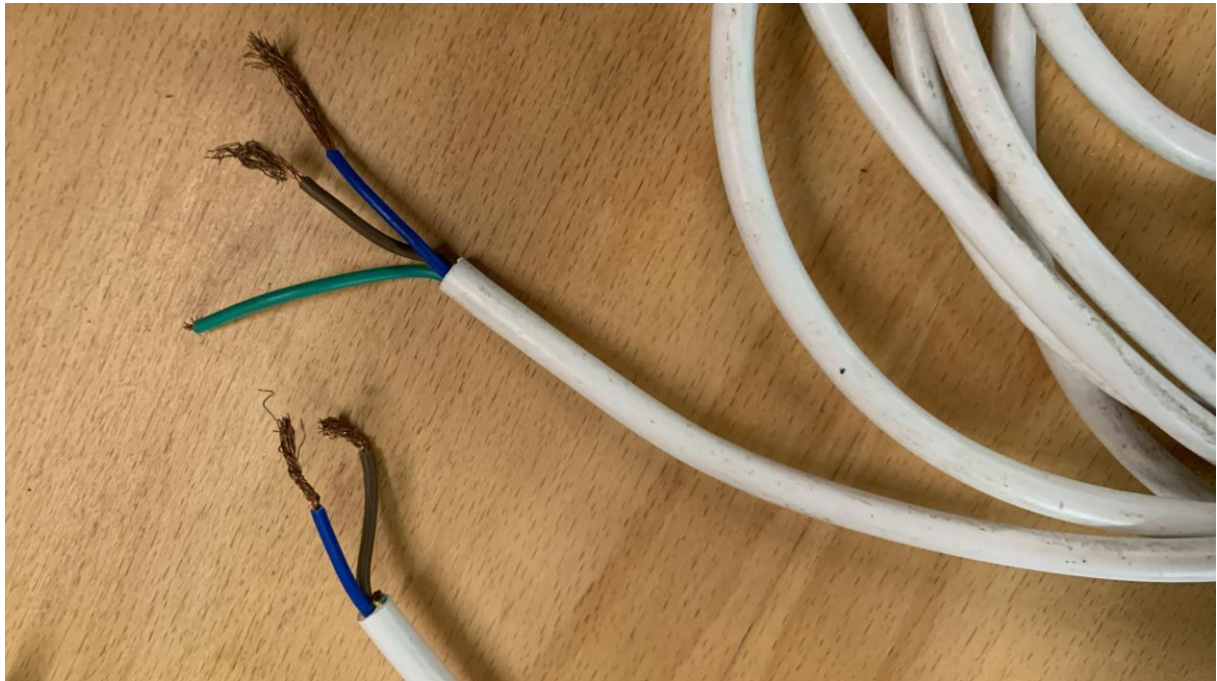
Die Anschlüsse am Arduino Nano: PIN 6 Daten (von grün auf gelb), Plus (5V) (von rot auf orange), GND (von weiss auf grün)



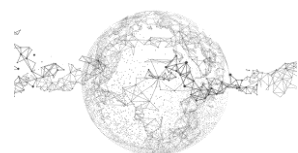
Dies ist dann die erste LED (von vorne gesehen oben links). Der Arduino Nano braucht keine weitere Stromversorgung mehr. Um die Daten zu laden, das USB Kabel am Nano mit dem Computer verbinden. Ich habe ein sehr langes USB Kabel genommen (10m-Verlängerung), damit ich später das Bild nicht mehr abhängen muss, um neue Programme zu laden.



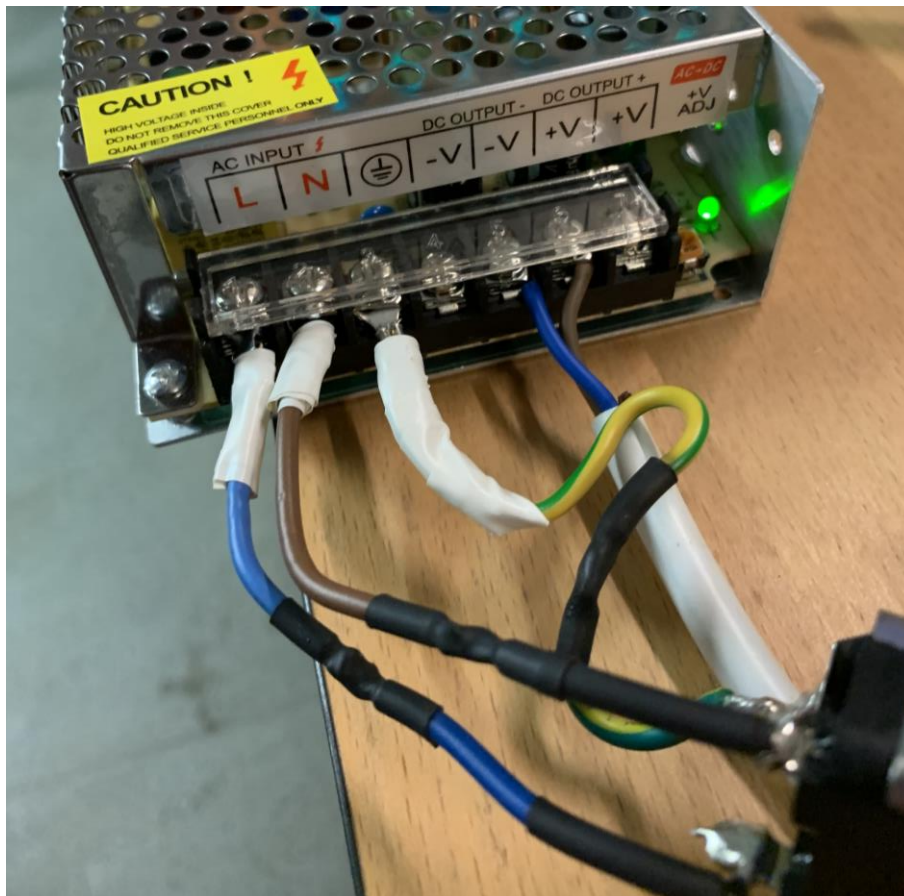
Ein altes Stromkabel dient für die Stromversorgung zwischen LEDs und Transformer:



Für die Stromversorgung zwischen Steckdose und Transformer einen Gerätestecker verwenden, wo später ein normales PC-Powerkabel eingesteckt wird.

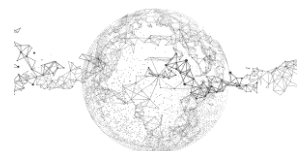


Das sieht schliesslich am Transformator so aus:



Für mehr Sicherheit: Ein altes Transformergehäuse ausschlachten und den Schalter nutzen. Falls man kein solches zur Hand hat, selber ein Gehäuse bauen. Am besten auch mit Kippschalter.

Vorne:



Hinten:

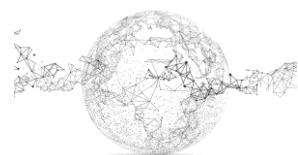
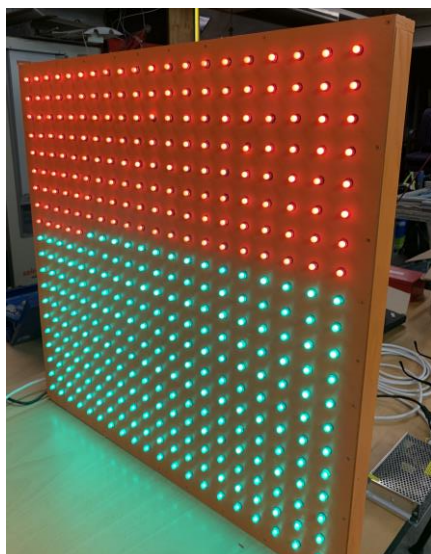


Zur Funktionskontrolle am besten den Strandtest von Adafruit laufen lassen:

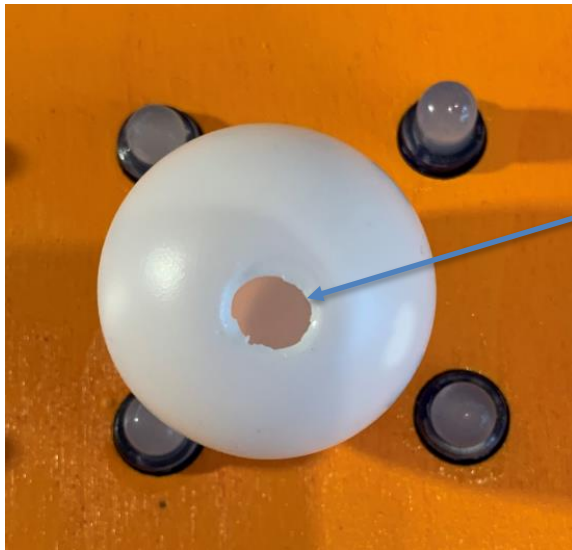
[https://github.com/adafruit/Adafruit\\_NeoPixel/blob/master/examples/strandtest/strandtest.ino](https://github.com/adafruit/Adafruit_NeoPixel/blob/master/examples/strandtest/strandtest.ino)

```
// Which pin on the Arduino is connected to the NeoPixels?  
// On a Trinket or Gemma we suggest changing this to 1:  
#define LED_PIN 6  
  
// How many NeoPixels are attached to the Arduino?  
#define LED_COUNT 60
```

auf 400 ändern!



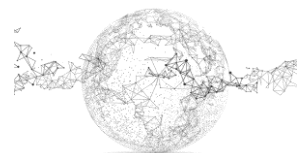
Dann die 400 Ping Pong Bälle mit einer Heissleim-Pistole lochen und mit etwas Leim aufkleben.



Leim entlang  
des Randes  
auftragen.



Dies ist eine ziemlich Geduldsarbeit, bis alle 400 Bälle gelocht und geklebt sind ... 😊





## Einkaufsliste:

Arduino NANO (*alternativ ginge auch chinesischer Clone → günstiger*)

[https://www.reichelt.com/ch/de/arduino-nano-v3-atmega-328-mini-usb-arduino-nano-p142943.html?CTYPE=0&MWSTFREE=0&PROVID=2788&wt\\_gacha=24387224682\\_413365094289&PROVID=2788&qclid=EALaIqobChMlieWj6pj86glVBSF7Ch2Q2QFoEAAYAiAAEgKQ-vD\\_BwE&r=1](https://www.reichelt.com/ch/de/arduino-nano-v3-atmega-328-mini-usb-arduino-nano-p142943.html?CTYPE=0&MWSTFREE=0&PROVID=2788&wt_gacha=24387224682_413365094289&PROVID=2788&qclid=EALaIqobChMlieWj6pj86glVBSF7Ch2Q2QFoEAAYAiAAEgKQ-vD_BwE&r=1)

LED-Reihen

<https://de.aliexpress.com/item/32625011824.html?spm=a2g0s.9042311.0.0.bae44c4dtr0Vxj>

Transformer

<https://de.aliexpress.com/item/32983648084.html?spm=a2g0s.9042311.0.0.bae44c4dtr0Vxj>

Ping Pong Bälle

<https://de.aliexpress.com/item/32796409392.html?spm=a2g0s.9042311.0.0.bae44c4dtr0Vxj>

Weiteres:

- MDF 4mm (80x80 cm)
- Fichtenlatten für Rahmen
- Jumperwire und Draht
- Draht, Lötkolben, Lötzinn, nötiges Kleinwerkzeug
- Leim (sehr flüssig für LEDs, zähflüssig für Bälle)
- Laser Cutter
- Stromkabel und Stecker

Eine günstigere Arduino-Alternative gibt es bei den Chinesen bei *AliExpress*. Der braucht dann aber einen anderen Treiber.

(siehe: <https://www.schularena.com/ict/informatik/robotik/arduino> → Downloads)

## Installation der Libraries:

Zuerst die Software für die Programmierung des Arduino installieren. Arduino IDE:

<https://www.arduino.cc/en/main/software>

Dann die entsprechende *Library* installieren. Dazu wie folgt vorgehen:

In der IDE auf Menü *Sketch* → *Include Library* → *Manage Libraries* → „*NeoPixel*“ bei Suche eingeben → *Name der Library* wählen

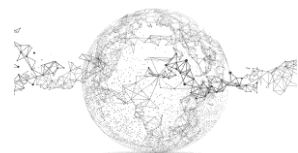
Je nach Code gibt es da verschiedene Libraries, die geladen werden müssen.

Beispiel:



```
pixel_bild | Arduino 1.8.7
File Edit Sketch Tools Help
pixel_bild
// Code by Marcel Isler
// You can find the project on www.
#include <Adafruit_NeoPixel.h>
```

Wichtig: Arduino kann nur immer ein Programm ausführen!



Code-Beispiele, die mit diesem Projekt funktionieren (selber ausprobieren, welche funktionieren und Code anpassen: PIN und Anzahl LEDs):

<https://platformio.org/lib/show/126/FastLED/examples?file=XYMatrix.ino>

Im Downloadbereich unserer Webseite gibt es als Zip-File einige Codes, die funktionieren. Man kann immer nur ein Programm laden und ausführen.

## Die Pixelbilder:

Analog zum Projekt *LED-Pixel Bild* kann man auch selber Bilder aufzeichnen und LEDs entsprechend programmieren. In unserem Projekt wäre jedoch die 0 oben links und ein Raster mit 400 Feldern wäre nötig. Vergleiche dazu:

<https://www.schularena.com/ict/informatik/make-it/led-pixel-bild>

72	73	74	75	76	77	78	79	80
71	70	69	68	67	66	65	64	63
54	55	56	57	58	59	60	61	62
53	52	51	50	49	48	47	46	45
36	37	38	39	40	41	42	43	44
35	34	33	32	31	30	29	28	27
18	19	20	21	22	23	24	25	26
17	16	15	14	13	12	11	10	9
0	1	2	3	4	5	6	7	8

Man kann auch Bilder in Array-Werte konvertieren. Dazu mal folgendes probieren:

<https://www.brainy-bits.com/create-arduino-array-from-pictures/>

Viel Spass!

