

FINGERZINKENFRÄSE mit SCHRITTMOTOR

RALF HILDEBRANDT



Lt. [Building The Ultimate Box Joint Jig - IBUILDIT.CA](https://www.ibuildit.ca) habe ich die Fingerzinkenfräse nachgebaut. Leider nicht mit dem gewünschten Erfolg, da das Versetzen an der Gewindestange bei mir nicht immer exakt verlief. Trotz etlicher Umstellungen war ich mit meiner Ausführung nicht zufrieden.



Ich habe daraufhin die Grundvorrichtung übernommen und mit einem Schrittmotorantrieb gekoppelt.

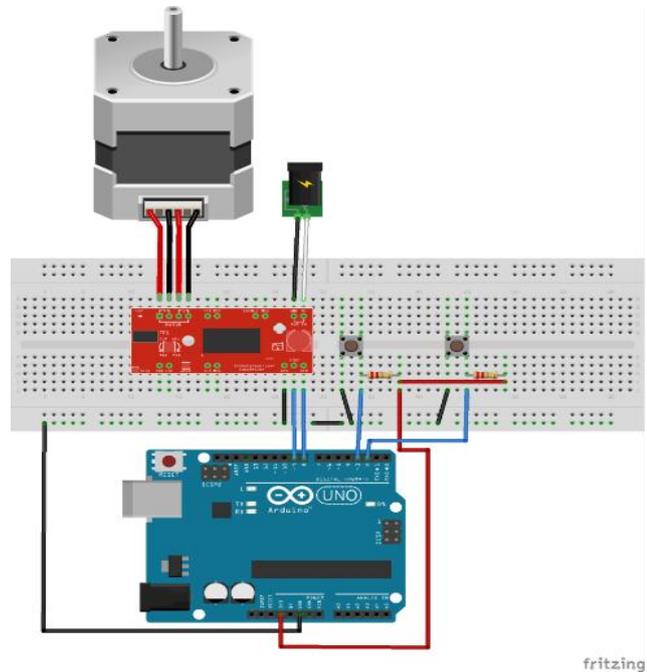
Hierbei lässt sich durch die exakte Steuerung mittels eines Microprozessors die Schrittlänge sehr genau einstellen.

Gewählt habe ich hierzu einen Seeeduno XIAO - kaum größer als ein Daumennagel - und nachgeschalteten Motortreiber (easy driver) zur Schrittmotorsteuerung.

Die Programmierung habe ich von [Easy Driver Examples \(schmalzhaus.com\)](http://schmalzhaus.com) übernommen.

Zur Ausführung kommt die Variante Beispiel 1.6 :

Verwenden von zwei Tasten, um vorwärts und rückwärts zu bewegen



Sketch von Schmalzhaus:

```
#define DISTANCE 3200

int StepCounter = 0;
int Stepping = false;

void setup() {
  pinMode(8, OUTPUT);
  pinMode(9, OUTPUT);
  digitalWrite(8, LOW);
  digitalWrite(9, LOW);

  pinMode(2, INPUT);
  pinMode(3, INPUT);
}

void loop() {
  if (digitalRead(3) == LOW && Stepping == false)
  {
    digitalWrite(8, LOW);
    Stepping = true;
  }
  if (digitalRead(2) == LOW && Stepping == false)
  {
    digitalWrite(8, HIGH);
    Stepping = true;
  }
}
```

```

}

if (Stepping == true)
{
    digitalWrite(9, HIGH);
    delay(1);
    digitalWrite(9, LOW);
    delay(1);

    StepCounter = StepCounter + 1;

    if (StepCounter == DISTANCE)
    {
        StepCounter = 0;
        Stepping = false;
    }
}
}

```

GESCHWINDIGKEIT

Hierbei ist lediglich eine Variable anzupassen:

es ist in der Klammer hinter delay, die Zahl zu variieren.

Will man es erheblich schneller, ist Microseconds einzufügen

z. B. delayMicroseconds(10)

Wird man zu schnell, kann darunter die Genauigkeit leiden.

Da ist eine kleine Testphase zum Optimieren erforderlich.

VORSCHUB FÜR BELIEBIGE FINGERZINKENBREITE

die Zweite und weitaus wichtigere Variable ist die Schrittlänge

diese wird in der 1. Zeile eingestellt `define DISTANCE 3200`

Will man bei den Fingerzinken alle möglichen Fingerbreiten ermöglichen, ist der Vorschub so einzustellen,

dass die Schrittlänge genau der Breite des Sägeblattes entspricht.

Das ist bei mir eine Schrittlänge von 460 (abhängig von der Steigung der Spindelachse und Wahl des Schrittmotors).

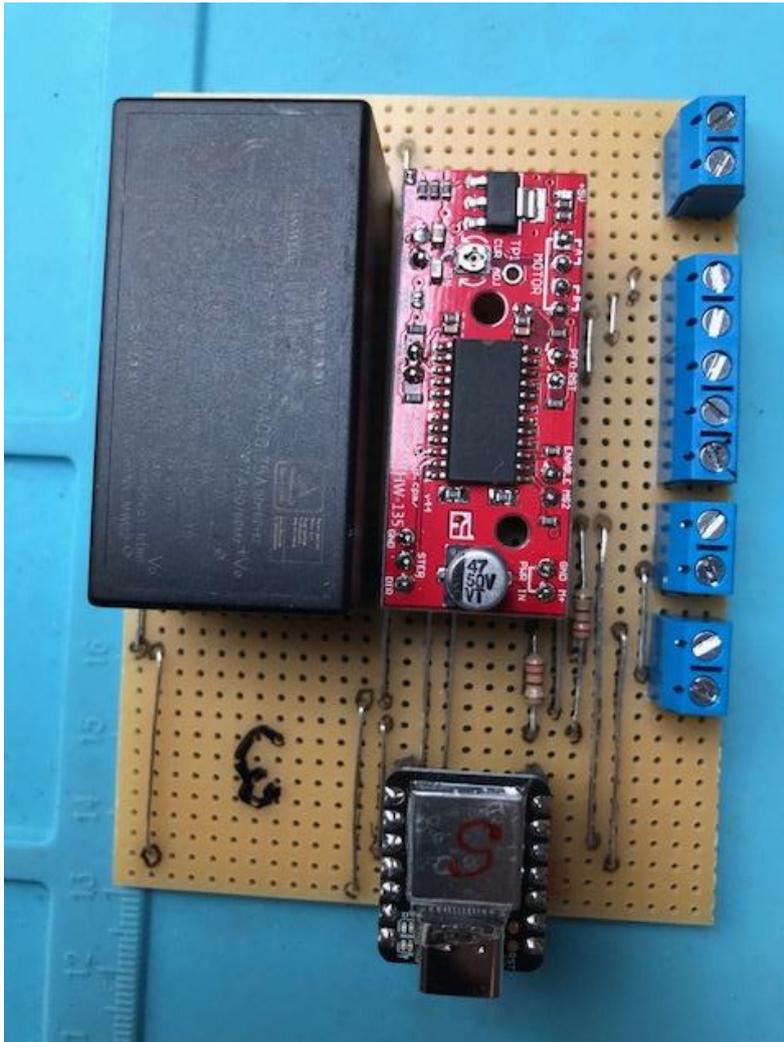
Nun muss man beim Vorschub immer sooft die Vorwärts-Taste drücken bis man die gewünschte

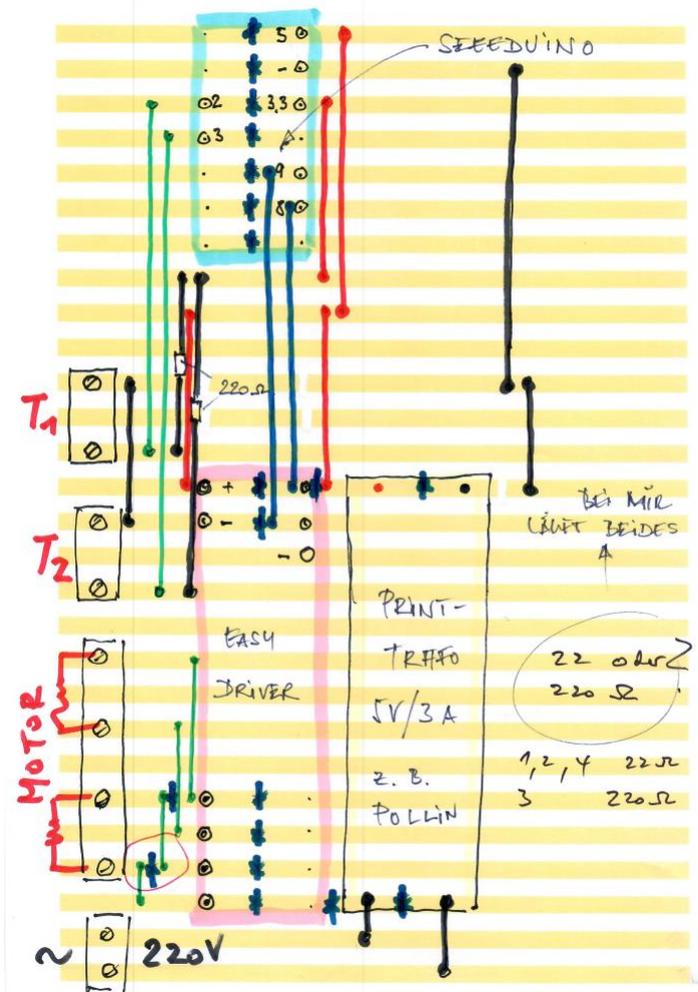
Fingerzinkenstärke erreicht hat. Zu beachten ist, dass man beim Vorschub ohne Sägen ein Tastendruck mehr als für die Sägeschnitte braucht.

Die Rückwärtstaste dient dazu, die Vorschubeinheit wieder auf die Ursprungsposition zu fahren.

Hierzu kann man die Taste einfach gedrückt halten, bis man ungefähr wieder am Anfang ankommt.

Aufgebaut habe ich die ganze Schaltung auf einer Streifenrasterplatine:





Die kurzen blauen „ge-ixten“ Striche sind Trennungen der Rasterstreifenplatine.

Etwas Mühe kostet noch die Installation des Seeeduino, um ihn über die arduino software laufen zu lassen.

Hier muss man den Microprozessor in der Arduino ID unter Voreinstellungen anmelden und unter Werkzeuge in die Bibliothek einfügen.

Unter Werkzeuge – Boards - Boardverwalter den Seeeduino XIAO anmelden und unter Datei - Voreinstellungen unter URLs https://files.seeedstudio.com/arduino/package_seeeduino_boards_index.json eingeben.

Schwer erkennbar: zwischen Package,Seeeduino,boards und index ist jeweils ein Unterstrich.

Unter [Seeeduino XIAO - Seeed Wiki \(seeedstudio.com\)](https://wiki.seeedstudio.com/Seeeduino-XIAO/) findet man eine ausführliche Anleitung.

Man kann natürlich auch arduino Boards nehmen.

Zum Schluss noch der notwendige Hinweis:

Anschluss an den Netzstrom nur durch eine zugelassene Fachkraft !



Taste grün: vorwärts / Taste rot: zurück / Schalter recht: An, Aus

