

## Lokschuppentore mit Stepper betätigen

Die Torbetätigung im ersten Lokschuppen habe ich mithilfe des neuen 3D Druckers realisieren können.

Die Elektronik basiert auf dem Arduino. Der Antrieb erfolgte über ein Servo.

Bei einem weiteren Lokschuppen musste ich einen anderen Weg gehen. Unter dem Tor wurde die Platte auf 10 mal 10 cm geöffnet. Auf das dort passende Brett kommt ein Türrahmen, den ich aus einem 8 mm Sperrholz aussägte. Er passt genau in den Lokschuppeneingang und ist von vorn nicht zu sehen.

Die beiden Tore werden nun nicht mehr direkt wie in den Lokschuppen von Faller drehbar eingesetzt, sondern in diesen Holzrahmen.

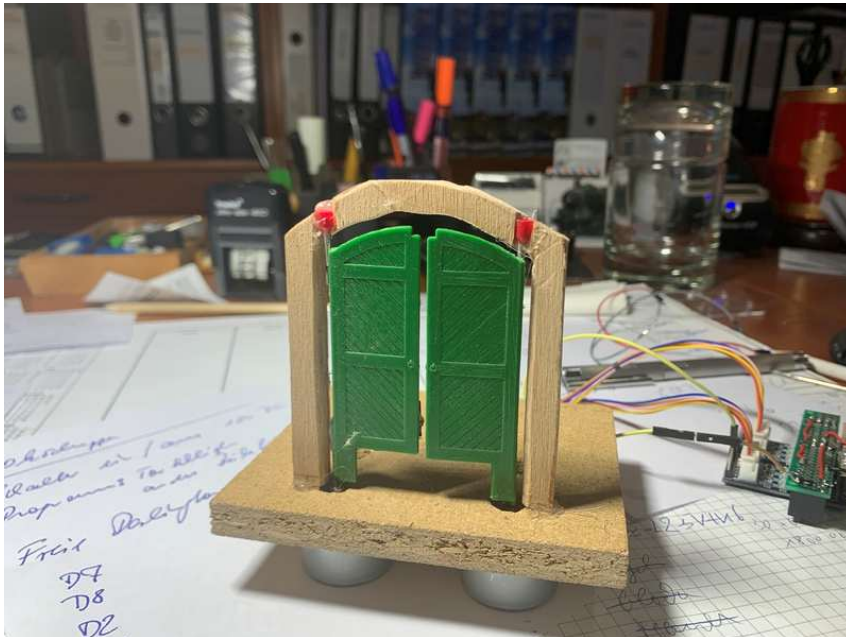


Bild 1

Grundplatte mit  
Torrahmen

Aus Fahrradspeichen wurden Achsen geschnitten, die an die Tore geklebt wurden. Oben werden die Achsen in eingeklebten Aderendhülsen gelagert. Unten ragen die Achsen durch Löcher bis unter die Einsatzplatte. Dort werden zwei Steppermotoren angeschraubt. In deren Achse habe ich Löcher gebohrt, in die die Achsen press passen.

Darunter kommt auch die Elektronikplatte. Sie besteht aus einem Arduino UNO mit einem selbst angefertigten Shield.

Der Arduino wird mit 12 V DC versorgt. Die Steppermotoren vertragen nur 5 V DC, also sitzt auf dem Shield ein Spannungswandler 7805. Bei Versuchen habe ich festgestellt, dass der 7805 sehr heiß wird, wenn die Stepper permanent angeschlossen bleiben. Dem bin ich mit zwei Maßnahmen begegnet.

- Der 7805 sitzt auf einem Blech, welches als Kühlkörper dient
- Die 5 V für die Stepper werden nur eingeschaltet, wenn die Stepper laufen sollen.

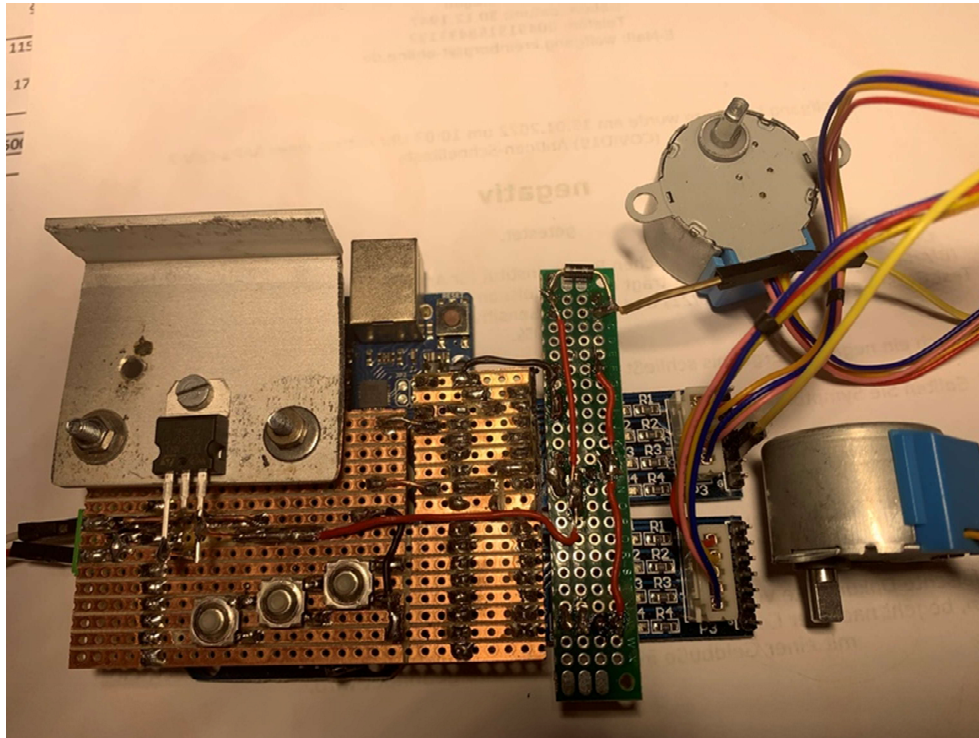


Bild 2  
Arduino mit  
Shield

Auf dem Shield sitzen drei Taster. Je einer öffnet bzw schließt die Tore, ein Taster simuliert das automatische Öffnen / Schließen. Dazu bewegt der eine Stepper das linke Tor gefolgt vom zweiten Stepper mit dem echten Tor. Die Drehgeschwindigkeit ist auf langsam eingestellt. Der Öffnungswinkel beträgt 90 Grad.

Vor dem Shield ist eine grüne schmale Leiterplatte. Sie steckt in den jeweils 4 Kontakten des Steppertreibers und stellt Masse (GND), 5 V DC und die beiden Jumperbuchsen zum Trennen der Motorversorgung bereit.

Unter diesem Streifen sitzt auch ein 5 V DC Relais, welches die 5 V DC auf die Motoren schaltet. Das Relais kann nicht direkt von dem Arduino angesteuert werden. Die Treiberplatine des Steppers hat jedoch ein Darlington Array mit 7 Kanälen, von denen nur vier für den Steppermotor dienen.

Also habe ich einen Kanal für das Relais genommen. Der Darlington schaltet dann gegen Masse. Über die Relaisspule ist eine Freilaufdiode gelötet.

Damit sind noch 2 bzw 3 Darlington Transistoren frei verfügbar.

Die Türöffnung soll

- einerseits zufallsgesteuert alle 20 bis 30 Minuten erfolgen.
- Über einen Taster auszulösen sein
- vom Steuerstand (CS2) geöffnet werden können.

Auf dem Schreibtisch (noch mit Drahtverhau) läuft der Sketch zufriedenstellend. Ich habe den Anschluss an A0 rausgeführt und provisorisch einen Taster angeschlossen. Da der Arduino weiß, ob das Tor offen oder geschlossen ist, reicht der eine Taster aus. Die Betätigung des Tasters macht je nach vorgefundenem Zustand die Tore auf oder zu.

Dieser Taster erhält später ein parallel geschaltetes Relais. Damit ist das Tor von der CS2 aus zu öffnen bzw. zu schließen.

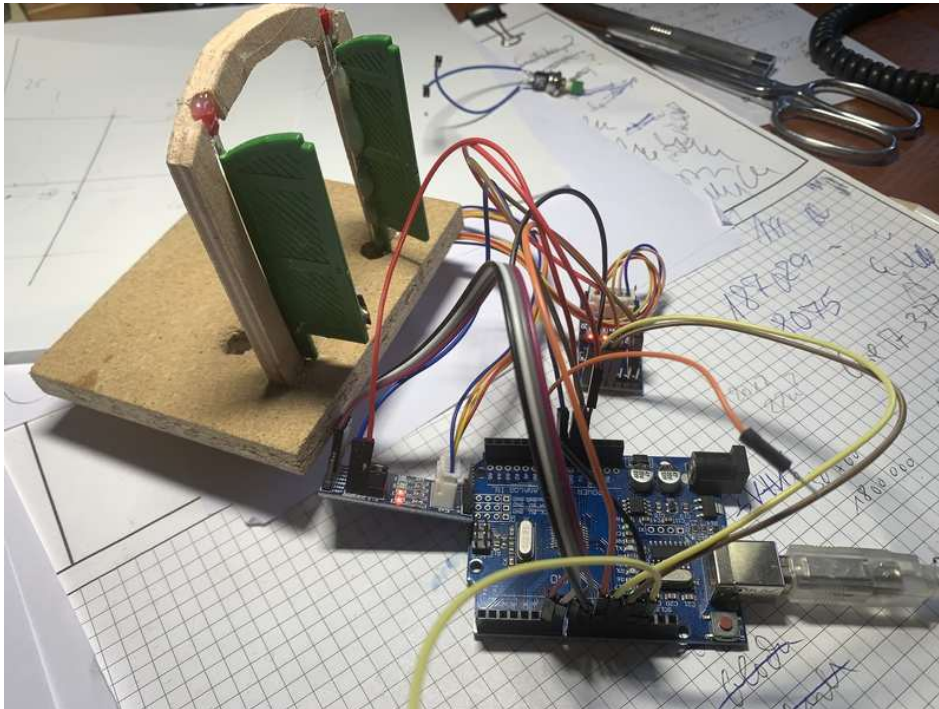
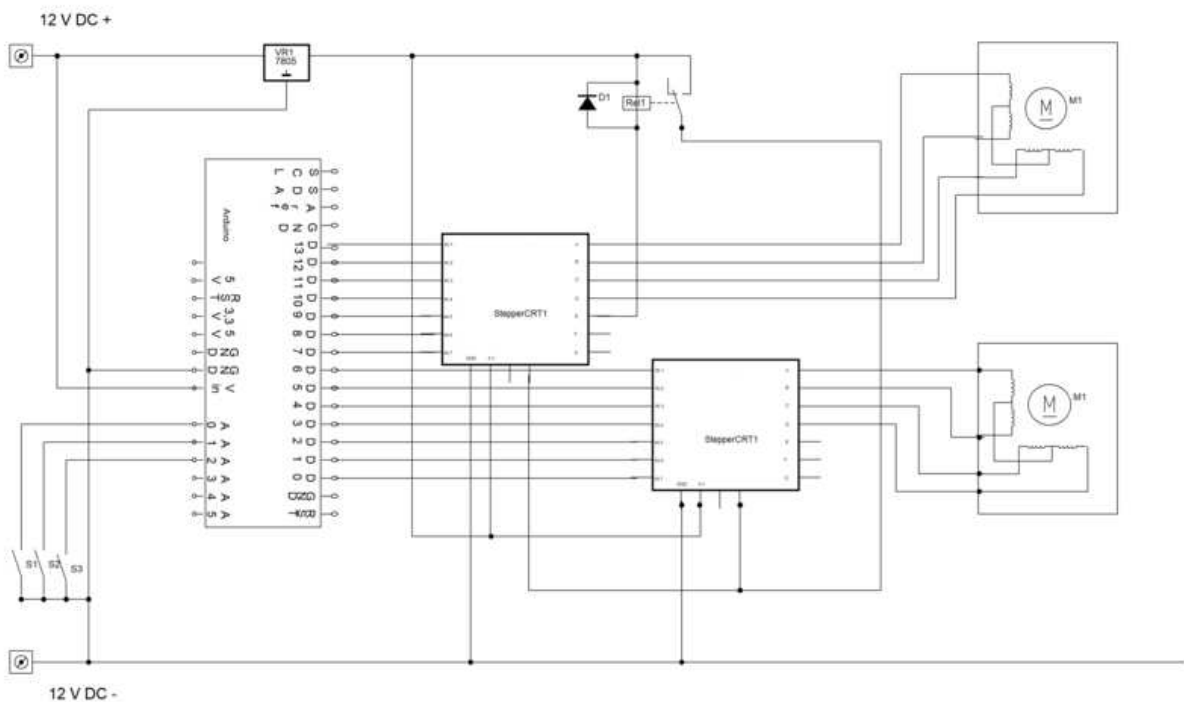


Bild 3  
Testaufbau

Bild 4  
Schaltplan



lokschuppen\_links.spl

17.01.2022



## Lokschuppen-mit-Stepper.docx

Nun wurde der Rahmen in der Baugrube eingepasst und der Arduino so eingebaut, dass er weiterhin programmiert werden kann, ohne ihn auszubauen.

Jetzt muss das Tor noch farblich behandelt werden, ebenso der Untergrund mit der Einsatzplatte.

Im Lokschuppen selbst steckt eine LED Lichtleiste in der Decke sowie ein LED in dem Anbau. Beide werden von einem anderen Arduino angesteuert und leuchten nicht permanent. Im nächsten Schritt soll die Lichtleiste an den Öffnungs-/Schließvorgang zeitlich angepasst werden.



Bild 5  
Einbau



Bild 6  
Einbau

Bild 7  
Der Rahmen passt, der Arduino hängt darunter.

Einige Pflastersteine liegen schon da.

