

Bauanleitung



Benötigtes Material

- Bauteile gemäss Einkaufsliste (Bauteile_und_Verbindungen.xlsx)
- ca. 1 kg Filament PLA
- Ping Pong Bälle
- 6 AA-Batterien (wiederaufladbar)
- Klebstoff für Kunststoff (z.B. UHU hart)
- Schrumpfschlauch / Isolierband
- Software/Arduino Sketch zum Projekt

Benötigtes Werkzeug

- 3D Drucker mit Druckraum 200mm in allen Richtungen (z.B. Original Prusa i3 MK3)
- Multimeter / Spannungsmessgerät
- USB-Kabel, passend zum Anschluss des Arduino NANO
- Lötkolben, Lötzinn
- Schraubenzieher / Schraubenschlüssel / Flachzange / Klebeband

Vorausgesetzte Vorkenntnisse

- 3D Druck von einfachen Bauteilen mit PLA
- Löten: Verlängern von Kabeln; Litzen an elektronische Bauteile anlöten; Isolieren mit Schrumpfschlauch/Isolierband
- Elektronik/Microcontroller: Herunterladen und installieren von Libraries; Upload von Programmcode auf Arduino NANO; Spannungen messen mit einem Multimeter

1. Druck der Bauteile

Drucke alle Bauteile gemäss den Anweisungen der Tabelle «Druckbare Bauteile» (Bauteile_und_Verbindungen.xlsx). Im Anhang 1 am Ende dieser Bauanleitung findest du eine Abbildung jedes 3D gedruckten Bauteils. Du kannst in den Abbildungen gut erkennen, wie diese Bauteile für optimale Druckergebnisse auf dem Drucktisch ausgerichtet werden sollten.

Wer den PrusaSlicer für das Slicing verwendet, kann direkt die 3MF-Dateien öffnen und von voreingestellten Parametern und Druckausrichtungen profitieren.

Da der Druck aller Bauteile relativ lange dauert, empfiehlt sich, mit dem Druck gleich zu Beginn des Projekts zu starten und sich während des Druckprozesses den nächsten Schritten zu widmen.

Tipp: Basis, Schriftzüge, Ecke und Deko zuerst drucken und anschliessend gleich miteinander verkleben gemäss Kapitel «Zusammenbau». Währenddem der Kleber trocknet, können die restlichen Bauteile gedruckt werden.

2. Lötarbeiten

In der Tabelle «Verbindungsplan» (Bauteile_und_Verbindungen.xlsx) ist für jede Verbindung beschrieben, welche Anschlüsse von welchem Bauteil auf welche Weise miteinander verbunden werden müssen (löten, schrauben, stecken).

Für die Lötarbeiten filterst du am besten die Tabelle nach den «X» in den Spalten «Lötv. A» respektive «Lötv. B», um nur jene Verbindungen anzuzeigen, welche Lötarbeiten erforderlich machen. Arbeitet dich anschliessend zeilenweise durch die Tabelle, bis alle Lötverbindungen erledigt sind.

Im Anhang 2 am Ende dieser Bauanleitung findest du einige Bilder der fertigen Lötverbindungen.

Achtung: Diverse Bauteile kommen mehrfach vor, wie zum Beispiel die Mini Kippschalter. In diesem Fall ist auch die in der Klammer stehende Funktion des Bauteils zu beachten, um die Bauteile voneinander unterscheiden zu können.

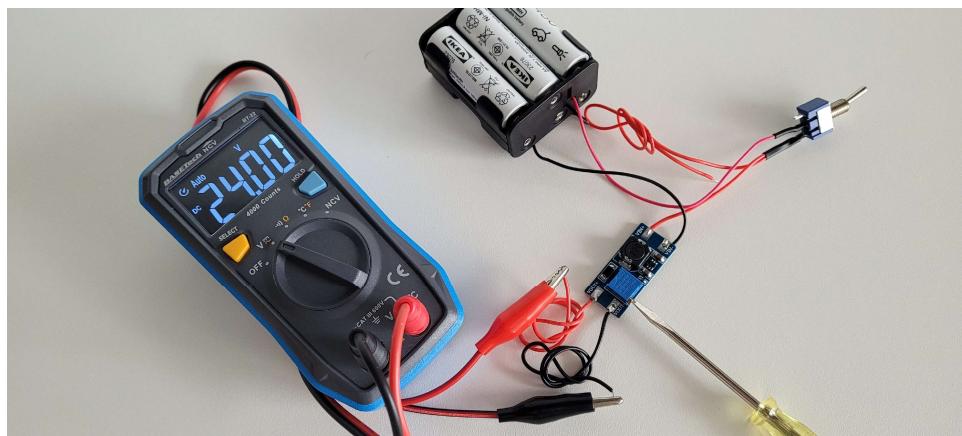
Beispiel: Verbindung Nr. 10

Nr.	Bauteil A	Anschluss Bauteil A	Lötv. A	Verbindung	Lötv. B	Anschluss Bauteil B	Bauteil B
10	Mini Kippschalter MTS-102 SPDT (Schussrichtung rechts)	Mittlerer Kontakt	X	10cm Kabel (schwarz), DuPont-Buchse einseitig: An Schalter anlöten und am Pin des Boards einstecken.		GND	NANO I/O Expansion Board

Nimm ein schwarzes Kabel mit DuPont-Buchsen (female). Dieses trennst du so in zwei Teile, dass du ein Kabel mit 10 cm Kabellänge erhältst, welches einseitig eine DuPont-Buchse aufweist. Das freie Ende dieses Kabels lötest du nun an den mittleren Kontakt des Mini Kippschalters, welcher für die «Schussrichtung rechts» vorgesehen ist. (Das andere Ende mit der DuPont-Buchse wird zu einem späteren Zeitpunkt im Schritt «Verkabelung» in das Arduino NANO Expansion Board an den GND-Pin gesteckt.)

3. StepUp-Converter einstellen

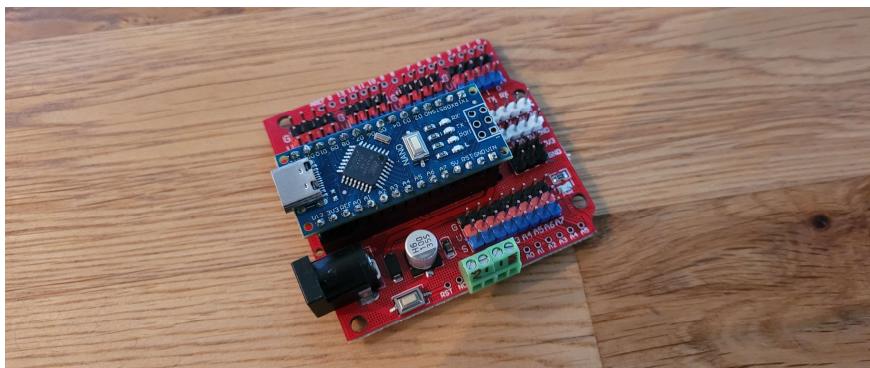
Da für den Betrieb der Motoren eine höhere Spannung als für den Arduino Microcontroller benötigt wird, muss die von der Stromversorgung gelieferte Spannung mit Hilfe eines StepUp-Converters erhöht werden. Der StepUp-Converter soll für unsere Zwecke auf eine Ausgangsspannung von 24V eingestellt werden. Gehe hierzu wie folgt vor:



1. Stelle dein Multimeter auf Spannungsmessung ein und lege das Messgerät an die Anschlüsse VOUT- und VOUT+ des StepUp-Converters an.
2. Stelle sicher, dass keine Kurzschlussgefahr besteht. **Es dürfen sich keine freiliegenden Kabelenden berühren!**
3. Lege sechs AA-Batterien in das Batteriefach.
4. Lies die Spannung am Multimeter ab. Falls das Messgerät keine Spannung anzeigt, dann stelle sicher, dass der Mini Kippschalter, der Ein-/Ausschalter des PING PONG TRAINERS, eingeschaltet ist.
5. Drehe mit einem Schraubenzieher an der kleinen Stellschraube des StepUp-Converters so lange, bis eine Ausgangsspannung von 24V im Multimeter angezeigt wird.
6. Unterbrich die Stromversorgung mit dem Mini Kippschalter. Die angezeigte Spannung sollte nun auf 0V zurückfallen. Danach kannst du die Batterien wieder entfernen.

4. Software

Die Software für den PING PONG TRAINER steht dir vollumfänglich zu Verfügung. Sie ist mit diversen Kommentaren versehen, so dass du dich bald darin zurechtfinden wirst. Im Anhang 3 wird auf ein paar wichtige Parameter und Codezeilen hingewiesen, die für das Feintuning für dich von Interesse sein könnten. Um die Software auf dem Arduino NANO ausführen zu können, werden die Libraries «Toggle» und «Servo» vorausgesetzt.



1. Stelle sicher, dass du die beiden Libraries «Servo» und «Toggle» heruntergeladen und in deiner Arduino Entwicklungsumgebung («Arduino IDE») installiert hast. Du findest die beiden Libraries im Bibliotheksmanager der Arduino IDE:
Servo: <https://www.arduino.cc/reference/en/libraries/servo>
Toggle: <https://www.ardu-badge.com/Toggle>
2. Stecke den Arduino NANO auf das Expansion Schild. Beachte dabei unbedingt die korrekte Ausrichtung respektive die korrekte Polung (siehe Abbildung).
3. Verbinde den Arduino Nano des PING PONG TRAINERS per USB-Kabel mit deinem Computer und lade die Software auf den Microcontroller. Sofort nach dem Upload, und auch bei jedem Programmstart, blinkt die interne LED des Microcontrollers dreimal. Damit weisst du, dass die Software für den PING PONG TRAINER korrekt auf den Microcontroller geladen wurde.

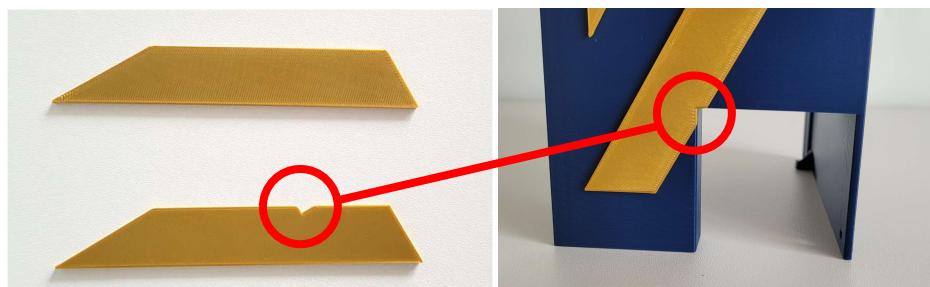
5. Zusammenbau

Als nächstes können alle Bauteile zusammengebaut werden. In den folgenden Schritten sind, wenn nicht anders angegeben, jeweils M3 Schrauben und Vierkantmuttern für die Schraubverbindungen zu verwenden. In den meisten Fällen sind Schlitze oder Vertiefungen zum Halten der Vierkantmuttern vorhanden.

1. Dekoration aufkleben



- Klebe mit Klebstoff die Schriftzüge auf die Basis, sowie die Dekorationen auf die Ecke. Die Prägungen in den Bauteilen helfen bei der Platzierung der Elemente. Dieser Schritt wird am besten gleich nach dem 3D Druck gemacht, damit der Klebstoff Zeit zum Aushärten hat.



Achtung: Zwei Deko-Bauteile sind beinahe identisch. Sie unterscheiden sich nur durch eine kleine Kerbe. Jenes Deko-Bauteil mit der Kerbe muss auf der Vorderseite, beim Ausgang der Schussvorrichtung, angebracht werden!

2. Micro LimitSwitch in Adapter schrauben und in Basis einsetzen



- Lege den Micro LimitSwitch in den Adapter. Beachte die korrekte Ausrichtung des Schalters und die Kabelrichtung: Der Hebel des Schalters muss in Richtung des in den Adapter geprägten Pfeils zeigen und die Kabel sind nach hinten hinauszuführen (siehe Abbildungen).
- Befestige den Schalter mit zwei M2.5 Schrauben (idealerweise 12mm Länge) am Adapter. Achte darauf, dass du den Schalter beim Eindrehen der Schrauben nicht einklemmst. Es sind keine Muttern als Gegenstück notwendig.
- Lege den Micro LimitSwitch in die Basis ein und beachte dabei erneut die korrekte Ausrichtung: Der Pfeil auf dem Adapter sollte in Schussrichtung zeigen (siehe Abbildungen).

3. Motor für Ballvorschub an Basis schrauben



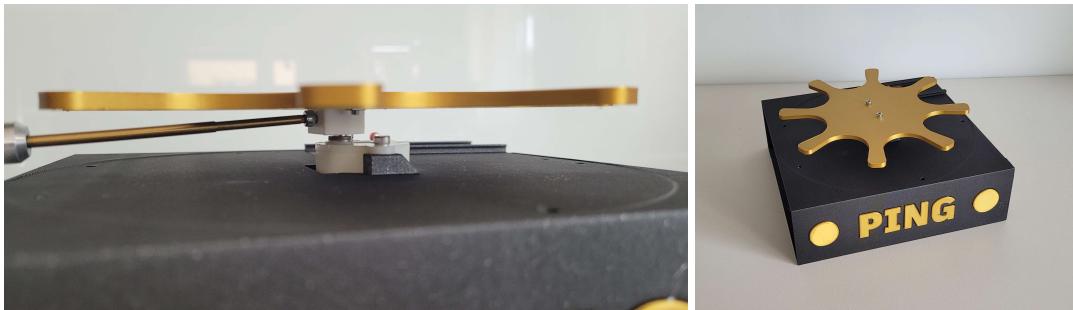
- Führe zwei Vierkantmuttern in die innenliegenden Schlitte der Aussparung für den Motor ein.
- Setze den Motor in die Basis ein und beachte dabei die korrekte Ausrichtung der Motorwelle (siehe Abbildung).
- Befestige den Motor mit zwei Zylinderkopf-Schrauben.

4. Adapter mit Scheibe verbinden



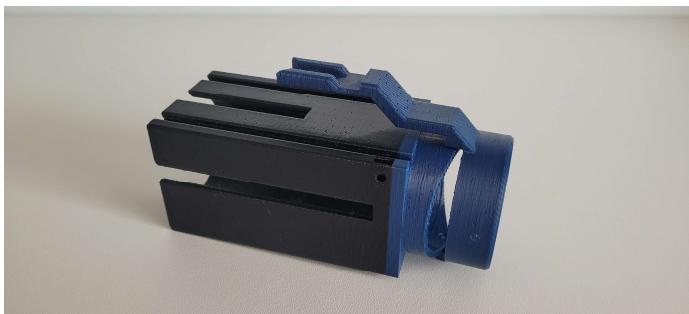
- Setze eine Vierkantmutter in den vorgesehenen Schlitz des Adapters.
- Drehe eine Zylinderkopf-Schraube soweit seitwärts in den Adapter, bis sie von der Vierkantmutter gefasst wird.
- Setze den Adapter in die Aussparung der Scheibe und verbinde diesen mit zwei Zylinderkopf-Schrauben und Muttern.

5. Scheibe mit Basis verbinden



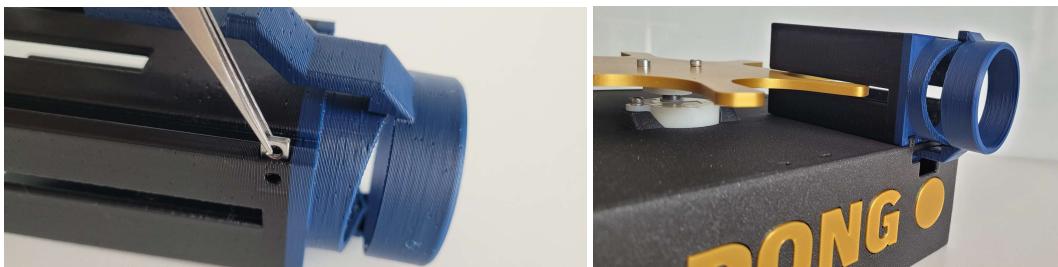
- Setze die Scheibe mit dem Adapter nach unten auf die Welle des Motors und arretiere diese mit der Schraube.

6. Schwenkarm auf Zuführung setzen



- Setze den Schwenkarm auf die Zuführung und beachte dabei die korrekte Ausrichtung des Schwenkarms (siehe Abbildung). Die beiden Noppen des Schwenkarms sollten so in der Zuführung einrasten, dass sich der Schwenkarm ganz einfach hin- und herbewegen lässt.

7. Zuführung mit Basis verbinden



- Setze eine Vierkantmutter in den Schlitz unterhalb der Zuführung.
- Schiebe nun die Zuführung von vorne auf die Basis. Damit wird gleichzeitig auch der Schwenkarm in die vordere Aussparung der Basis geführt. Achte darauf, dass die vorherig eingesetzte Vierkantmutter nicht herausfällt.

8. Ecke festschrauben



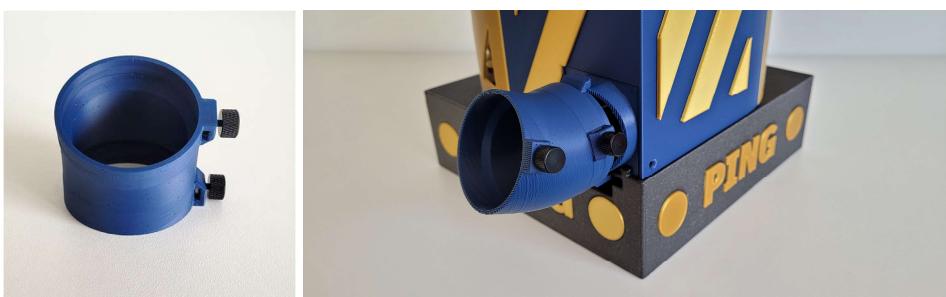
- Setze die Ecke auf die Basis und schraube diese mit einer Senkkopf-Schraube auf die Zuführung. Evtl. musst du dafür die in der Zuführung liegende Mutter etwas ausrichten.
- Schraube anschliessend die Ecke mit zwei Senkkopf-Schrauben und Muttern an die Basis.

9. Korb festschrauben



- Fahre den Korb in die Nut der Ecke ein und setze ihn auf die Basis.
- Schraube den Korb mit fünf Senkkopf-Schrauben und Muttern an die Basis.

10. Biegung aufsetzen



- Setze zwei Muttern in die Biegung und drehe danach zwei Rändelschrauben in die Bohrlöcher.
- Stülpe die Biegung über den Schwenkarm bis zum Anschlag und arretiere sie mit der Rändelschraube.

11. Motorbefestigung auf Lauf schrauben



- Setze zwei Muttern in die Schlitze des Laufs und befestige die untere Motorschale mit zwei Senkkopf-Schrauben. Beachte dabei die korrekte Ausrichtung der Motorschale (siehe Abbildung).

12. Motor mit Gummi umwickeln



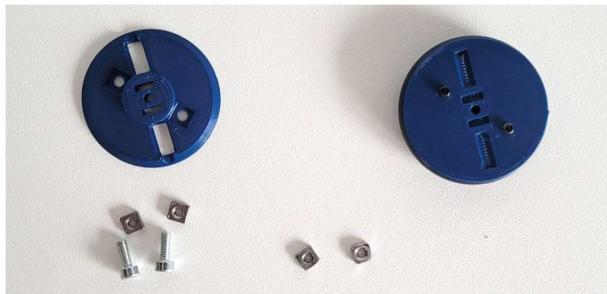
- Umwickle einen Motor mit einem Stück Gummi aus einem alten Fahrradschlauch oder einem ähnlichen, weichen, rutschfesten Material. Befestige die Gummiummantelung etwas mit Klebeband. **Die Gummischicht sollte ca. 2 mm dick sein.**

Hinweis: Die Gummiummantelung dient einerseits dazu, dass der Motor in der Motorschale weniger herumrutscht. Anderseits lässt sich mit der Dicke der Ummantelung auch der Abstand des Rades zum Ball einstellen. Du kannst die Gummiummantelung also später etwas dicker oder dünner wickeln, falls das Rad den Ping Pong-Ball zu wenig oder gar nicht herausschleudert oder der Ball im Abschlussrohr vom Rad blockiert wird.

13. Rad und O-Ring zusammenschrauben



- Befreie in Rad1 die beiden Bohrlöcher von den Stützen, die vom 3D Drucker gedruckt wurden. Am besten erledigst du dies, indem du mit einem spitzen Gegenstand die Stützen von der anderen Seite des Lochs herausdrückst. Vorsicht: Verletzungsgefahr!



- Stecke zwei Zylinderkopf-Schrauben in die beiden Bohrlöcher von Rad1
- Lege den O-Ring auf das Rad1 und anschliessend Rad2 darauf. Beachte dabei die korrekte Ausrichtung: Der O-Ring muss von beiden Teilen gut umfasst werden.



- Lege Rad3 auf die anderen Teile und platziere zwei Vierkantmuttern in die vorgesehenen Aussparungen.
- Schraube nun die drei Radteile fest zusammen, so dass der O-Ring kräftig eingequetscht wird.

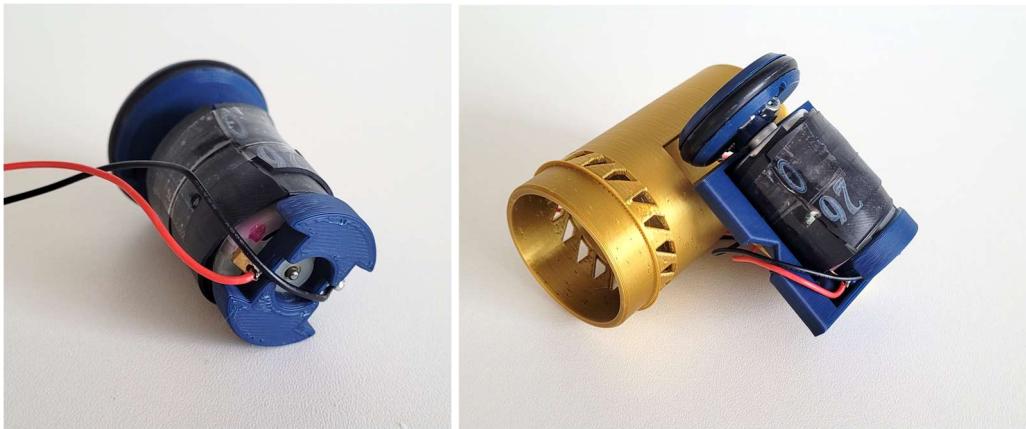


- Jetzt legst du zwei Muttern in die beiden verbleibenden Aussparungen und schraubst zwei Zylinderkopf-Schrauben lose in die quer zur Achse liegenden Bohrlöcher, gerade genug, damit die Schrauben die Muttern fassen.



- Setze das Rad auf die Welle des Motors und befestige es, indem du die beiden Arretierungsschrauben gleichmässig anziehst.

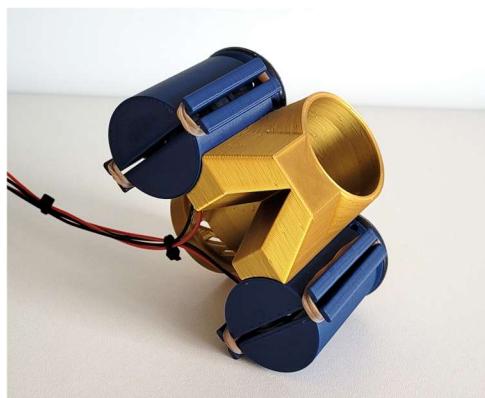
14. Motor einsetzen



- Setze den Motoreinsatz hinten auf den Motor, dort wo du die Kabel angelötet hast.
- Lege den Motor in die angeschraubte Motorschale.
- Führe die beiden Kabel des Motors durch die Kerbe in der Schale.



- Lege den Deckel der Motorschale auf den Motor und befestige die beiden Motorschalen mit je einem Gummiband auf jeder Seite. Stelle sicher, dass der Motor gut in der Motorschale liegt und die Kabel nicht gequetscht werden.
- Drehe das Rad am Motor ein paar Mal von Hand. Stelle damit sicher, dass sich das Rad frei drehen kann und nirgends mit anderen Bauteilen kollidiert.



Wiederhole die Schritte 11-14 mit dem zweiten Motor, so dass der Lauf schliesslich mit zwei Motoren bestückt ist.

15. Lauf einsetzen



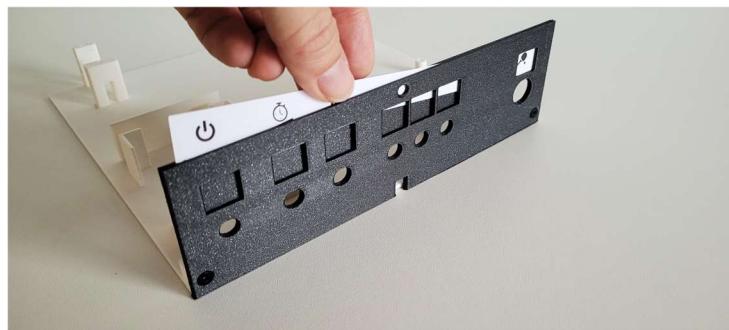
- Setze den Lauf bis zum Anschlag in die Biegung ein und arretiere ihn mit der Rändelschraube.
- Führe die Kabel der Motoren durch die Öffnung unterhalb der Zufuhr in die Basis hinein.

16. Labels ausdrucken



- Drucke die Datei «Labels.pdf» auf ein A4 Papier aus. Achte bei den Druckeinstellungen darauf, dass das Label beim Drucken nicht verkleinert oder vergrössert wird.
- Schneide das Label dem Rahmen entlang aus.

17. Schubladenblende mit Schubladenboden verbinden



- Verbinde die Schubladenblende mit dem Schubladenboden mit Hilfe von zwei Senkkopf-Schrauben und Muttern.
- Legt die ausgedruckten Labels in den Schlitz oberhalb der Schubladenblende.

18. Schalter und Regler einbauen



- Setze alle Schalter und Regler in die Blende ein und schraube sie fest. Beachte dabei, dass insbesondere der Kippschalter zum Ein-/Ausschalten an der richtigen Position gemäss dem Label festgemacht ist. Du solltest alle Kippschalter so ausrichten, dass wenn der Schalter nach oben zum Label zeigt, der Schalter geschlossen ist. Nötigenfalls kannst du das auch später noch korrigieren.
- Stecke die Drehknöpfe auf die Potentiometer.

19. Komponenten anschrauben

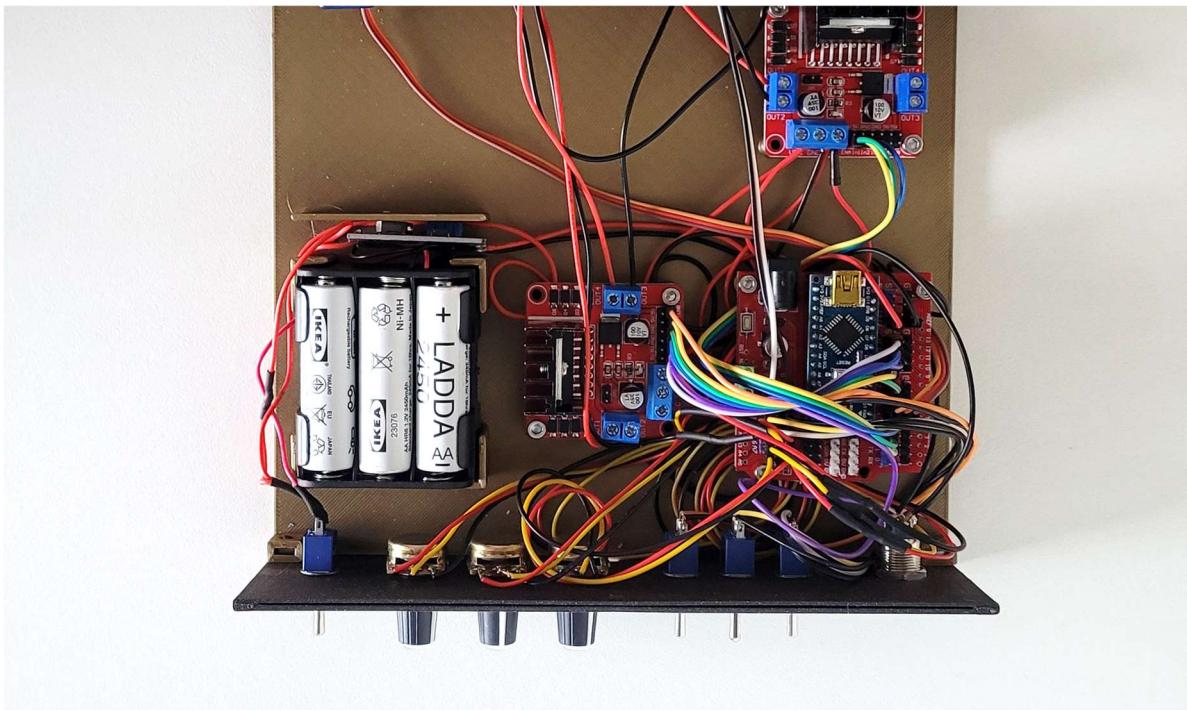


- Befestige den Servomotor mit zwei **M2 Schrauben** auf dem Schubladenboden. Achte dabei auf die korrekte Ausrichtung des Servomotors (siehe Abbildung). Führe das Kabel mit dem Stecker durch die Lücke der Befestigungsvorrichtung.
- Schraube das NANO Expansion Schild mit Hilfe von vier Zylinderkopf-Schrauben in die Schublade.
- **Entferne alle Jumper auf den beiden Motortreibermodulen L298N (rote Version).** Schraube anschliessend die Module mit Hilfe von jeweils vier Zylinderkopf-Schrauben in die Schublade.

Ein Bauteil bleibt nun noch übrig: Der Servoarm. Wir warten aber noch mit der Montage dieses Bauteils, denn vorher müssen alle Verkabelung gemacht werden.

6. Verkabelung

Nun werden alle elektronischen Komponenten miteinander verbunden. Hierzu wird nochmals anhand der Tabelle «Verbindungsplan» (Bauteile_und_Verbindungen.xlsx) vorgegangen. Wenn der Filter auf die Lötverbindungen aus dem Kapitel «Lötarbeiten» noch gesetzt ist, so solltest du diesen nun entfernen, damit alle Einträge in der Tabelle sichtbar werden.

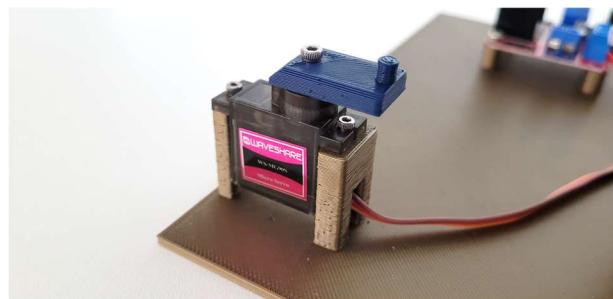


Arbeite dich nun erneut Zeile für Zeile durch die Tabelle und verbinde den angegebenen Anschluss des Bauteils A mit dem Anschluss des Bauteils B. Der Fokus liegt nun auf den Steck- und Schraubverbindungen, da ja alle Lötarbeiten bereits erledigt sind.

Wenn du möchtest, kannst du die Kabel unter den Boards durchführen, um den Blick auf die Boards und deren Steckplätze freizugeben. Achte aber darauf, dass keine Kabel eingeklemmt werden.

7. Servoarm anschrauben

Bitte überprüfe nun alle elektronischen Verbindungen nochmals. Bist du sicher, dass du alles richtig zusammengesetzt und die Verkabelung korrekt gemacht hast? Sind die Kabel in den Schraubterminals gut festgemacht? Wenn ja, dann kannst du nun den Zusammenbau wie folgt abschliessen:



- Lege sechs AA-Batterien in das Batteriefach.

- Schalte den PING PONG TRAINER ein, indem du den Ein-/Ausschalter betätigst. Da du die Software ja bereits auf den Microcontroller geladen hast, führt dieser beim Einschalten die Setup-Routine durch. In dieser Routine wird der Servomotor in die Mittelposition gefahren. Du solltest deshalb ein kurzes Surren des Servomotors wahrnehmen. Anschliessend blinkt die interne LED des Microcontrollers dreimal kurz auf.
- Nun, da der Servomotor in der Mittelposition ist, kannst du den PING PONG TRAINER am Ein-/Ausschalter wieder ausschalten.
- Stecke jetzt den Servoarm auf die Welle des Servomotors und befestige ihn mit einer passenden Schraube, in der Regel eine M2.5 Schraube. Beachte dabei unbedingt die korrekte Ausrichtung des Noppen auf dem Bauteil (siehe Abbildung). Achte auch darauf, dass sich die Welle des Servomotors während der Montage nicht verstellt.

8. Schublade schliessen



- Setze eine Vierkantmutter in den Schlitz, dort wo du danach die Schublade festschrauben wirst. **Achtung:** Die Schraube könnte beim Öffnen der Schublade wieder herausfallen und einen Kurzschluss verursachen! Deshalb kannst du diesen Schritt entweder weglassen (die Schraube sollte auch ohne Mutter genügend halten, um die Schublade geschlossen zu halten) oder du kannst die Vierkantmutter mit etwas Leim vor dem Herausfallen sichern.
- **Halte das Abschussrohr mit einer Hand geradeaus** und schiebe dann vorsichtig die Schublade in die Basis. Der Noppen auf dem Servoarm muss dabei in die Nut des in der Basis liegenden Schwenkarms des Abschussrohrs eingeführt werden! Wenn sich das Abschussrohr bei geschlossener Schublade nicht mehr von Hand schwenken lässt, dann kannst du davon ausgehen, dass das gelungen ist. Achte beim Schliessen der Schublade auch darauf, dass du keine Kabel einklemmst.
- Sichere die Schublade, indem du sie mit einer Senkkopf-Schraube mit der Basis verschraubst.

Fertig! Nun kannst du mit dem Training loslegen. Schau nun in die Bedienungsanleitung, um alle Funktionen des PING PONG TRAINERS kennen zu lernen.

Funktioniert etwas nicht wie gewünscht? Dann konsultiere das nächste Kapitel «Fehlerbehebung».

9. Fehlerbehebung

Die folgenden Tipps können dir bei der Fehlersuche und deren Behebung behilflich sein.

Ein Motor dreht in die falsche Richtung

Wenn einer der Motoren in die falsche Richtung dreht, dann hast zu wahrscheinlich die Kabel des Motors verkehrtherum angeschlossen.

Lösung: Tausche die Anschlüsse der Motorenkabel oder passe die Software an (siehe Anhang 3: «Drehrichtung der Motoren ändern»).

Es fehlt an Schusskraft. Die Bälle werden zu wenig weit geschossen.

Versuche das Problem wie folgt zu lösen:

1. Stelle sicher, dass du den Regler für die Schussweite auf voller Kraft hast, der Motor also mit voller Geschwindigkeit dreht. (Wenn du den von mir in der Bauanleitung empfohlenen Motor benutzt, so sollte die Schussstärke bereits bei halb offenem Drehregler genügend sein.)
2. Stelle sicher, dass du vollgeladene Batterien verwendest.
3. Wenn die Punkte 1 und 2 erfüllt sind und die Schusskraft immer noch nicht ausreicht, so liegt das wahrscheinlich daran, dass das Rad zu wenig tief in den Lauf ragt, um die Bälle zu erfassen und zu beschleunigen. In diesem Fall solltest du die Gummiummantelung etwas verringern, so dass die Motoren etwas tiefer in der Motorschale liegen. Siehe hierzu Kapitel «Zusammenbau» Schritt 12.

Der Servomotor dreht zwar, aber das Abschussrohr bewegt sich nicht.

Bitte stelle sicher, dass beim Einfahren der Schublade der Servoarm in den Schwenkarm geführt wird. Du erreichst das, indem du das Abschussrohr während des Einschiebens der Schublade gerade hältst. Siehe hierzu Kapitel «Schublade schliessen».

Kontakt

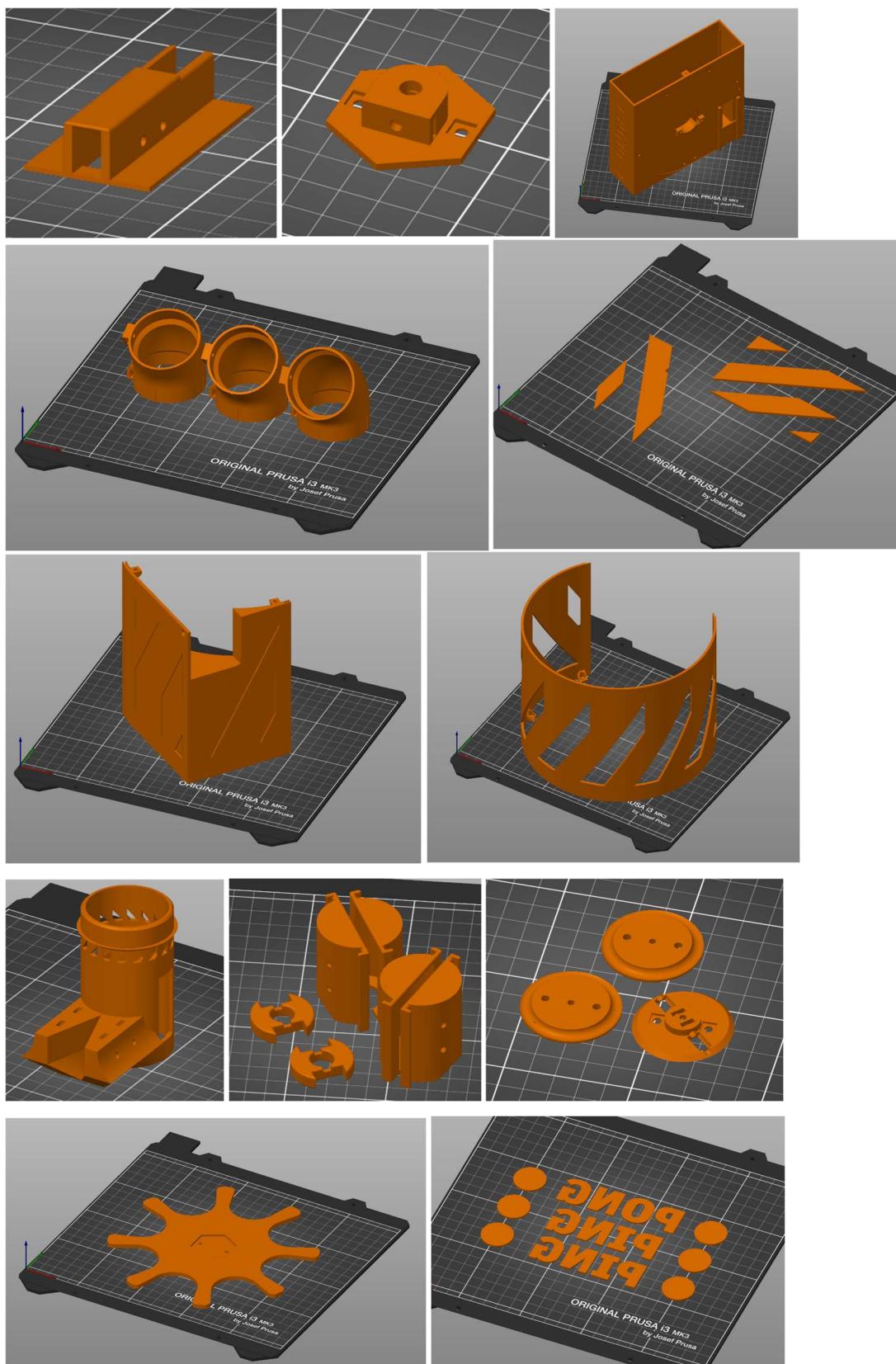


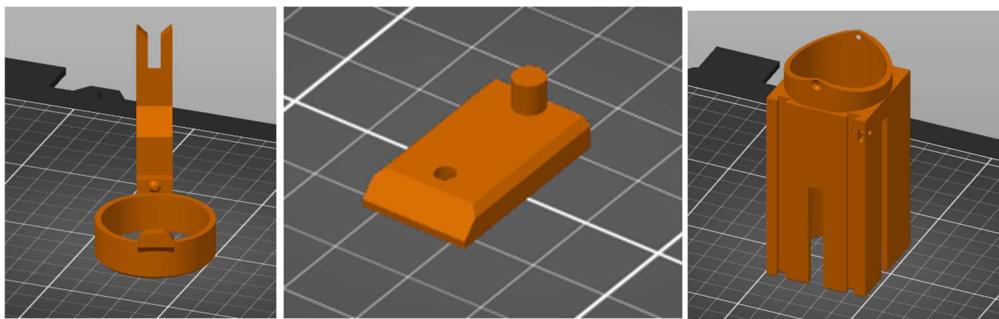
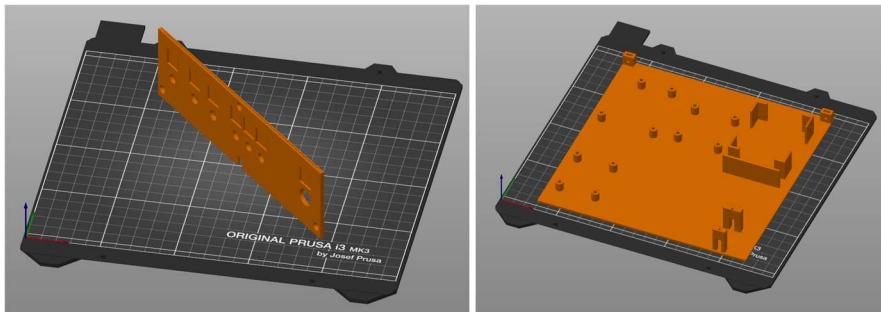
Benno Lottenbach, CH-6055 Alpnach Dorf

bingobricks@bluewin.ch

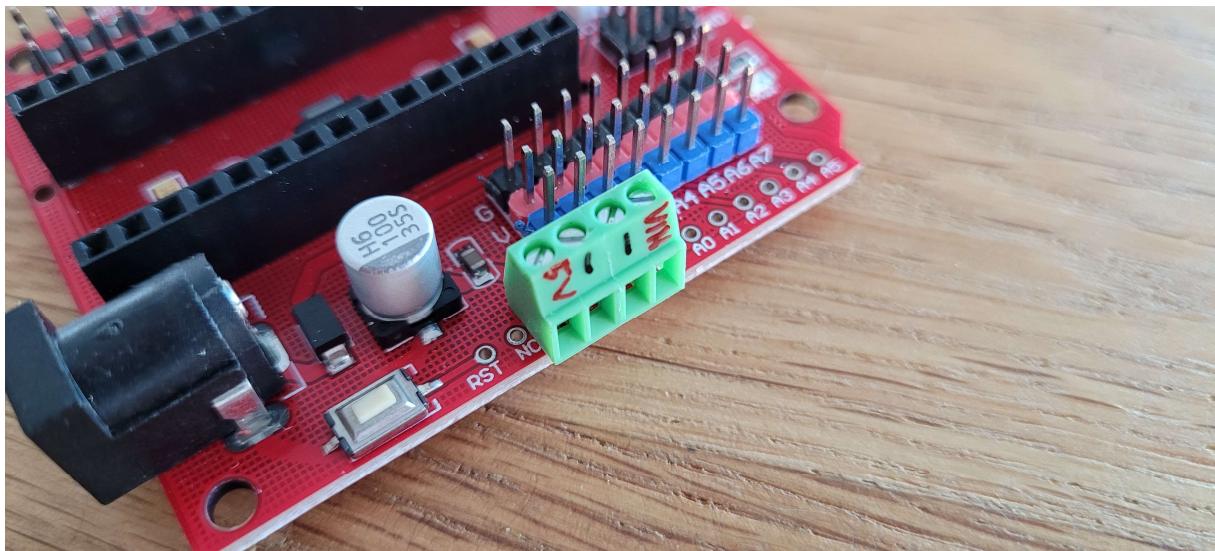
<https://youtube.com/@BINGOBRICKS>

Anhang 1: Ausrichtung der Druckobjekte

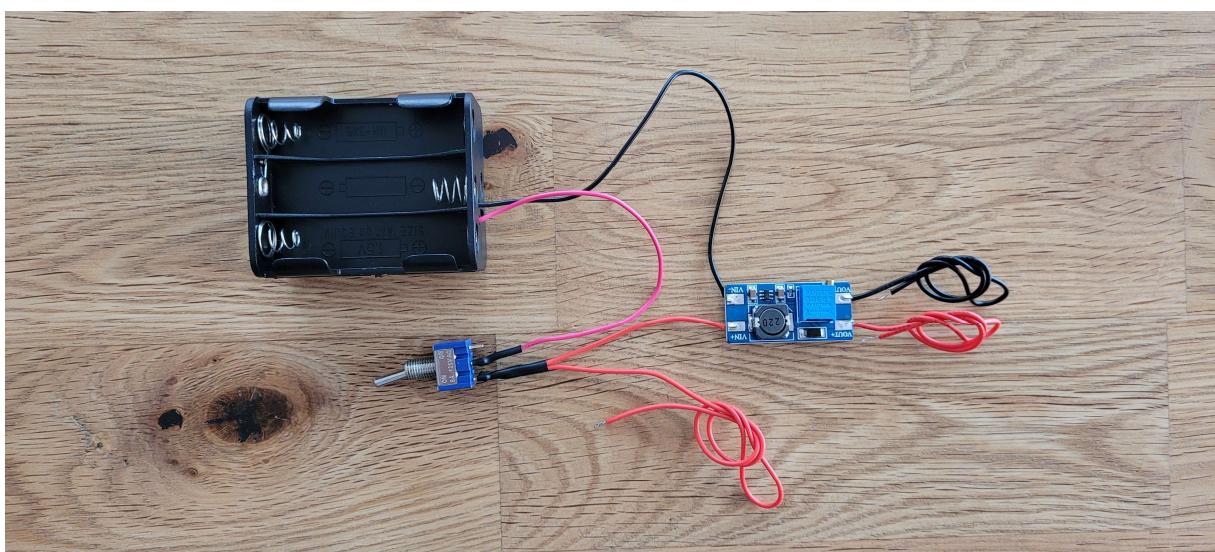




Anhang 2: Lötverbindungen



Terminal Block auf NANO I/O Expansion Board gelötet und beschriftet (VerbindungNr. 0)



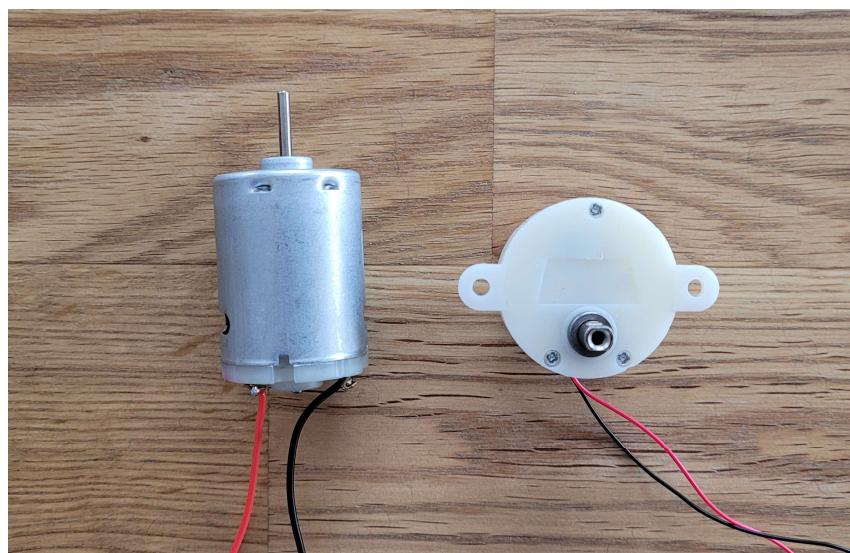
Die zwei Kabel des Batteriefachs sind verlötet mit Kippschalter und StepUp-Converter (VerbindungNr. 1 und 2).

Am zweiten Anschluss des Kippschalters ist ein zweites und drittes Kabel angelötet (VerbindungNr. 3 und 4).

Am Converter-Ausgang sind zwei Kabel angelötet (VerbindungNr. 20 und 21).



Kippschalter, Micro LimitSwitch und Momentary Switch mit LED (VerbindungNr. 5-16); Potentiometer (VerbindungNr. 34-39)



DC Motor 380 und DC Motor S30K (VerbindungNr. 24-27, die Kabelverlängerungen sind im Bild nicht sichtbar).

Anhang 3: Software

Die Software (Arduino Sketch) teilt sich in zwei Dateien plus Libraries auf:

PingPongTrainer.ino: Enthält die Deklaration aller Konstanten und Variablen, die Setup-Routine und die Loop-Routine. Alle nachfolgend dokumentierten Parameter sind in dieser Datei zu finden.

Functions.ino: Enthält alle Funktionen, die in der Loop-Routine aufgerufen werden.

Library «Servo»: Enthält Funktionen für die Steuerung für den Servomotor.
<https://www.arduino.cc/reference/en/libraries/servo>

Library «Toggle»: Enthält Funktionen für das Auslesen von Tastern und Schaltern.
<https://www.ardu-badge.com/Toggle>

Im Folgenden wird auf die relevantesten Codezeilen hingewiesen, um die wichtigsten Parameter der Software bei Bedarf anpassen zu können.

Servopositionen / Schussrichtungen verändern

```
44 //Servo deklarieren
45 #define cServoPin 5
46 #define cServoPosLeft 160           //Servoposition für Schussrichtung links
47 #define cServoPosCenter 90          //Servoposition für Schussrichtung mitte
48 #define cServoPosRight 20          //Servoposition für Schussrichtung rechts
```

PingPongTrainer.ino

Die Konstanten `cServoPosLeft` und `cServoPosRight` in den Zeilen 46 und 48 geben die Servoposition für die Schussrichtungen nach links und rechts an. Du kannst den Winkel zwischen links und rechts vergrößern respektive verkleinern, indem du die Werte entsprechend anpasst.

Hinweis: Falls es dir Spass macht, kannst du den PING PONG TRAINER statt an der Kantenmitte an der rechten Ecke des Tisches positionieren und die Werte der Servopositionen stattdessen auf «geradeaus», «leicht links zur Mitte» und «stark links zur gegenüberliegenden Ecke» ändern.

Maximale Drehzahlen der Motoren für die Schussweite/Spin verändern Drehzahlen der Motoren für die Schussweite/Spin synchronisieren

```
53 // Motor B1 (Backspin) deklarieren
54 #define cMB1Pin 3
55 #define cMB1In1Pin 2
56 #define cMB1In2Pin 4
57 #define cMB1Throttle 0.50
58 bool vMB1Invert = false;
59 byte vMB1State = LOW;
60
61 // Motor A1 (Topspin) deklarieren
62 #define cMA1Pin 11
63 #define cMA1In1Pin 10
64 #define cMA1In2Pin 9
65 #define cMA1Throttle 0.50
66 bool vMA1Invert = false;
67 byte vMA1State = LOW;
```

PingPongTrainer.ino

Die Konstanten `cMB1Throttle` respektive `cMA1Throttle` auf den Zeilen 57/65 senken die Spannungen, die an den Motoren für die Schussweite/Spin anliegen.

Gemäss Materialliste setzen wir einen Motor mit einer Drehzahl von 16'000 RPM bei 24V ein. Meine Versuche haben aber ergeben, dass 8'000 Umdrehungen/Minute bereits mehr als genug sind. Der obig eingestellte Wert von 0.5 bewirkt also, dass bei voll geöffnetem Drehregler nicht 24V am Motor anliegen, sondern nur 12V, was die maximale Drehzahl somit ungefähr halbiert ($24V * 0.5 = 12V$).

Umgekehrt kann natürlich der Wert dieser Konstante erhöht werden, wenn ein Motor mit geringerer Umdrehungszahl eingesetzt wird und die Kraft nicht ausreicht, um den Ball ans andere Tischende zu befördern.

Falls du für Topspin und Backspin zwei unterschiedliche Motoren mit unterschiedlichen Drehzahlen verwendest, so kannst du die beiden Konstanten ebenfalls mit unterschiedlichen Werten versehen, damit die Motordrehzahlen aneinander angeglichen werden.

Betriebsspannung des Motors für den Ballvorschub verringern

```
69 // Motor A2 (Ballzuführung) deklarieren
70 #define cMA2Pin 6
71 #define cMA2In3Pin 7
72 #define cMA2In4Pin 8
73 #define cMA2Throttle 0.50
74 bool vMA2Invert = false;
75 byte vMA2State = LOW;
76 unsigned long vLastMotorstart = 0;
```

PingPongTrainer.ino

Die Konstante `cMA2Throttle` auf Zeile 73 senkt die Spannung, die am Motor für den Ballvorschub anlegt.

Gemäss Materialiste setzen wir einen Motor ein, der für 12V vorgesehen ist. Die am Motortreibermodul anliegende Spannung ist aber 24V. Würden die vollen 24V an diesen Motor weitergegeben, so würde der Motor zu schnell drehen.

Der in dieser Konstante eingestellte Wert von 0.5 bewirkt daher, dass von den am Motortreibermodul anliegenden 24V nur die Hälfte davon an den Motor weitergegeben wird ($24V * 0.5 = 12V$).

Wartezeit bis zur ersten Schussabgabe verändern

Wartezeit bis zum Abschalten der Motoren verändern

```
104 //Weitere Variablen und Konstanten
105 #define cStartDelay 5000
106 #define cStopDelay 6000
107 unsigned long vTimeStamp = 0;
108 byte vState = 0;
```

PingPongTrainer.ino

Die Konstante `cStartDelay` auf Zeile 105 beinhaltet die Wartezeit in Millisekunden, die zwischen dem Drücken des Startknopfes und der ersten Schussabgabe verstreichen soll. In dieser Zeitspanne hat der Benutzer die Gelegenheit, sich vom PING PONG TRAINER an die gegenüberliegende Tischseite in Spielposition zu begeben. Du kannst diese Wartezeit gerne verlängern oder verringern.

Die Konstante `cStopDelay` auf Zeile 106 beinhaltet die Zeit in Millisekunden, die der Motor für den Ballvorschub drehen darf, bevor sich alle Motoren des PING PONG TRAINERS abschalten. Wird innerhalb dieses Zeitraums kein Ball in die Ballzufuhr mehr geschoben, so geht PING PONG TRAINER davon aus, dass der Korb leer ist.

Drehrichtung der Motoren ändern

```
53 // Motor B1 (Backspin) deklarieren
54 #define cMB1Pin 3
55 #define cMB1In1Pin 2
56 #define cMB1In2Pin 4
57 #define cMB1Throttle 0.43
58 bool vMB1Invert = false;
59 byte vMB1State = LOW;
60
61 // Motor A1 (Topspin) deklarieren
62 #define cMA1Pin 11
63 #define cMA1In1Pin 10
64 #define cMA1In2Pin 9
65 #define cMA1Throttle 0.50
66 bool vMA1Invert = false;
67 byte vMA1State = LOW;
68
69 // Motor A2 (Ballzuführung) deklarieren
70 #define cMA2Pin 6
71 #define cMA2In3Pin 7
72 #define cMA2In4Pin 8
73 #define cMA2Throttle 0.50
74 bool vMA2Invert = false;
75 byte vMA2State = LOW;
76 unsigned long vLastMotorstart = 0;
```

PingPongTrainer.ino

Solltest du die Motoren verkehrtherum angeschlossen haben, so drehen sich diese in die falsche Richtung. In den Codezeilen 58, 66 und 74 kannst du die Drehrichtung softwareseitig umkehren. Setze hierzu die entsprechenden Variablen `vMB1Invert`, `vMA1Invert` oder `vMA2Invert` auf `true`.

Tipp: USB Kabel hinausführen



Um die Software auch bei geschlossener Schublade auf den Microcontroller zu laden, hat die Schubladenblende eine kleine Aussparung für das USB Kabel.