

Beckenbauer, Daniel

ArduWood-IllumiSphere. Eine mikrocontrollergestützte Weiterentwicklung eines Seifenblasenautomaten

technik-education (tedu). Fachzeitschrift für Unterrichtspraxis und Unterrichtsforschung im allgemeinbildenden Technikunterricht 4 (2024) 1, S. 45-72



Quellenangabe/ Reference:

Beckenbauer, Daniel: ArduWood-IllumiSphere. Eine mikrocontrollergestützte Weiterentwicklung eines Seifenblasenautomaten - In: *technik-education (tedu). Fachzeitschrift für Unterrichtspraxis und Unterrichtsforschung im allgemeinbildenden Technikunterricht 4 (2024) 1, S. 45-72* - URN: urn:nbn:de:0111-pedocs-300702 - DOI: 10.25656/01:30070

<https://nbn-resolving.org/urn:nbn:de:0111-pedocs-300702>

<https://doi.org/10.25656/01:30070>

in Kooperation mit / in cooperation with:



<https://tec-edu.net/tedu>

Nutzungsbedingungen

Dieses Dokument steht unter folgender Creative Commons-Lizenz: <http://creativecommons.org/licenses/by/4.0/deed.de> - Sie dürfen das Werk bzw. den Inhalt vervielfältigen, verbreiten und öffentlich zugänglich machen sowie Abwandlungen und Bearbeitungen des Werkes bzw. Inhaltes anfertigen, solange Sie den Namen des Autors/Rechteinhabers in der von ihm festgelegten Weise nennen.

Mit der Verwendung dieses Dokuments erkennen Sie die Nutzungsbedingungen an.

Terms of use

This document is published under following Creative Commons-License: <http://creativecommons.org/licenses/by/4.0/deed.en> - You may copy, distribute and render this document accessible, make adaptations of this work or its contents accessible to the public as long as you attribute the work in the manner specified by the author or licensor.

By using this particular document, you accept the above-stated conditions of use.



Kontakt / Contact:

peDOCS
DIPF | Leibniz-Institut für Bildungsforschung und Bildungsinformation
Informationszentrum (IZ) Bildung
E-Mail: pedocs@dipf.de
Internet: www.pedocs.de

Mitglied der


Leibniz-Gemeinschaft

technik – education

4. Jahrgang

Fachzeitschrift für Unterrichtspraxis und Unterrichtsforschung
im allgemeinbildenden Technikunterricht

1 | 2024

|| 1 a n e o n Arial

|| 1 n n n e n Futura PT

|| 1 a n e o Source Sans Pro

www.tec-edu.net

tedu

Fachzeitschrift für Unterrichtspraxis und Unterrichtsforschung im allgemeinbildenden Technikunterricht

[HTTPS://TEC-EDU.NET/TEDU](https://tec-edu.net/tedu)

HERAUSGEBER

Prof. Dr. Hannes Helmut Nepper
Dr. Armin Ruch, OStR
Dr. Dr. Dierk Suhr

Mail

herausgeber@tec-edu.net

Anschrift

Pädagogische Hochschule Schw. Gmünd
Institut für Bildung, Beruf und Technik
Abteilung Technik
Oberbettringer Straße 200
73525 Schwäbisch Gmünd
www.tec-edu.net

AUTOR*INNEN IN DIESEM HEFT

Simon Baier
Lucas Bareis
Daniel Beckenbauer
Fabian Csoch
Maximilian Fuchs
Markus Hummel
Lara Mayer
Jochen Pfeifer
Dierk Suhr
Katrin Wohlfromm

Inhalt

GRUSSWORT DER HERAUSGEBER.....2

UNTERRICHTSFORSCHUNG

D. SUHR

KONZEPTE EINER MINT-DIDAKTIK.....3

UNTERRICHTSFORSCHUNG

F. CSOSCH

SOZIALE ROBOTER IM TECHNIKUNTERRICHT16

UNTERRICHTSPRAXIS

K. WOHLFROM

KI-GENERIERTE PÄDAGOGISCHE AGENTEN30

UNTERRICHTSPRAXIS

M. FUCHS

EIN UNTERRICHTSTAUGLICHES 3D-DRUCKERGEHÄUSE37

UNTERRICHTSPRAXIS

D. BECKENBAUER

ARDUWOOD-ILLUMISPHERE45

UNTERRICHTSPRAXIS

M. HUMMEL

SMARTER BLUMENTOPF73

UNTERRICHTSPRAXIS

S. BAIER, L. BAREIS, LARA MAYER & J. PFEIFER

FERTIGUNG EINES NISTKASTENS IN DER GRUNDSCHULE..86

Namentlich gekennzeichnete Beiträge geben nicht unbedingt die Meinung der Herausgeber wieder.

Insbesondere bei unterrichtspraktischen Artikeln wird darauf hingewiesen, dass es unterschiedliche Sicherheitsbestimmungen gibt und jede Lehrkraft bei der Umsetzung selbst dafür verantwortlich ist, die Gefährdung zu beurteilen und die Vorschläge für die eigene Praxis entsprechend der jeweilige Vorschriftenlage anzupassen.

Titelbild: Armin Ruch

ISSN: 2748-2022

ArduWood-IllumiSphere

Eine mikrocontrollergestützte Weiterentwicklung eines Seifenblasenautomaten

Daniel Beckenbauer

SCHLAGWORTE

Seifenblasenautomat
Elektrotechnik
Mikrocontroller
Fertigung

ABSTRACT

In diesem Artikel wird die Weiterentwicklung einer Seifenblasenautomatenvariante vorgestellt, die auf der in tedu Ausgabe 5, 1|2023 (S. 24-37) veröffentlichten Variante aufbaut und durch die Integration eines Mikrocontrollers weitere didaktische Zugänge ermöglichen soll. Dabei wird zunächst exemplarisch aufgezeigt, welche differenzierenden Möglichkeiten und Zugänge bestehen, diese Variante im Unterricht umzusetzen. Im Anschluss daran wird mithilfe einer umfassenden Schritt-für-Schritt-Anleitung die Fertigung des mikrocontrollergestützten Seifenblasenautomaten aufgezeigt, bei deren Umsetzung neben verschiedenen Fertigungsverfahren und Planungsmitteln weitere Inhalte der Elektro- und Informationstechnik einbezogen werden.

EINLEITUNG

Seifenblasen, ein Naturschauspiel, das die Menschen schon seit Jahrtausenden fasziniert, haben ihren festen Platz in der menschlichen Kultur gefunden und sind seit jeher ein Bestandteil davon. Die Herstellung von Seife durch die Sumerer führte um 3000 v. Chr. zur ersten Erzeugung von Seifenblasen als Nebenprodukt, die besonders bei Kindern aufgrund ihrer Langlebigkeit und ihres Unterhaltungswertes beliebt waren (Pustefix, o.J.). Mit der Zeit symbolisierten Seifenblasen in diversen Kulturen unter anderem sowohl Vergänglichkeit als auch Unbeschwertheit und Freude (ebd.). Sie nahmen ihren Weg in die Welt der Kunst, Literatur und Wissenschaft, wobei bedeutende Gelehrte wie Leonardo da Vinci und Isaac Newton ihre physikalischen Merkmale studierten (ebd.). In der heutigen Zeit erfreuen sich Seifenblasen weiterhin großer Beliebtheit als Spielzeug für Jung und Alt und inspirie-

ren nach wie vor die wissenschaftliche Forschung sowie die Architektur, wo sie beispielsweise als Inspiration für Dachkonstruktionen dienen (ebd.).

Die Faszination, die Seifenblasen auch heute noch hervorrufen, spiegelt sich in der Entwicklung des hier vorgestellten Seifenblasenautomaten wider. Von der ersten Upcycling-Variante (kurze Beschreibung in der tedu Ausgabe 5, 1|2023) bis hin zu der hier vorgestellten Mikrocontroller-Variante wurde deutlich, dass dieses einfache Konstrukt „Seifenblasenautomat“ technisch gesehen zahlreiche Möglichkeiten für stetige Weiterentwicklungen bietet. Er erlaubt es, vielfältige Inhalte im Technikunterricht aufzugreifen und zu integrieren, was ihn zu einem facettenreichen Zugangs-Werkstück bzw. Lehrmittel macht.

Die im Folgenden beschriebene Variante des mikrocontrollergestützten Seifenblasenautomaten (Abb. 1) wurde

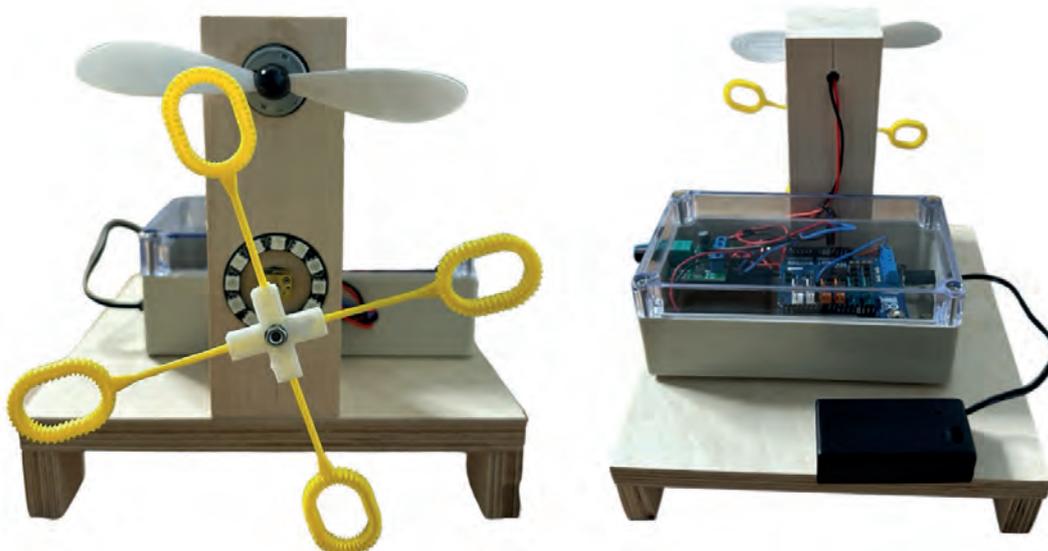


Abbildung 1: Seifenblasenautomat mit Arduino-Steuerung

dabei als Fertigungsaufgabe für die Klassenstufen 8/9/10 der Sekundarstufe I konzipiert. Dabei bietet das Werkstück zum einen die Möglichkeit zur Wiederholung sämtlicher Kompetenzen wie beispielsweise dem sicheren Umgang mit Werkzeugen und Maschinen, der Nutzung technischer Planungsmittel und der Umsetzung einfacher elektronischer Schaltungen (Bildungsplan Sekundarstufe I Baden-Württemberg Technik, 2016, S. 14 ff.). Zum anderen kann das Werkstück zur Anbahnung weiterer Kompetenzen wie beispielsweise der Fügetechnik Löten dienen (ebd., S. 16). Weiterhin ist es auch möglich, speziell durch die Integration eines Mikrocontrollers als Zugangs-Werkstück eine Brücke zwischen der Elektro- und der Informationstechnik zu schlagen. Ebenso können durch den bereitgestellten Programmcode einfache Steuerungen und Regelungen mithilfe von Blockdiagrammen thematisiert (ebd., S. 22) sowie die Realisierung von additiv gefertigten Elementen (3D-Druck) für den Seifenblasenautomaten computergestützt (CAD, CAM) (ebd., S. 17) geplant und erstellt werden.

FUNKTION, AUFBAU UND ANFORDERUNGEN

Der Seifenblasenautomat ist so gestaltet, dass er mit minimalem Materialaufwand maximale Funktionalität bieten soll. Zusätzlich ist er für individuelle Anpassungen oder Erweiterungen wie beispielsweise die Integration von LED/OLED-Displays zur Anzeige bestimmter Betriebsparameter offen konzipiert.

Die Funktionsweise des Seifenblasenautomaten ist denkbar einfach. Ein motorbetriebenes Blasringrad taucht während seiner Drehung kontinuierlich in ein Behältnis ein und nimmt Seifenlauge auf. In der oberen Hälfte des Blasringrades erzeugt ein von einem weiteren Motor angetriebenes Rotorblatt einen gleichmäßigen Luftstrom, der in Abhängigkeit von der Rotationsgeschwindigkeit des Rades Seifenblasen in hoher Frequenz und ohne manuelles Zutun produziert. Ein um das Rad angebrachter LED-Ring illuminiert dabei die Seifenblasen mit verschiedenen Lichtanimationen. Diese Beleuchtung verstärkt die Lichtbrechung der Seifenblasen, wodurch sie schimmern und besonders bei Dunkelheit einen schönen Effekt erzeugen.

Der Grundaufbau über die Bodenplatte mit Standfüßen soll eine ausreichende Stabilität für vielfältige Betriebsbedingungen schaffen. So können über die Ausrichtung der Standfüße unterschiedliche Erhöhungen der Grundplatte realisiert werden, die an unterschiedliche Durchmesser des später montierten Seifenblasenringrades anpassbar sind, sodass das Rad später sauber und störungsfrei im Seifenlaugebehälter drehen kann.

Auf der Grundplatte sind ein Motorengehäuse und eine Verteilerdose montiert, um die elektronischen Komponenten vor Feuchtigkeit zu schützen. Im Motorengehäuse ist ein $\varnothing 24$ mm (3-6 V) DC-Motor integriert, der zusammen mit einem Rotorblatt einen Luftstrom erzeugt. Hinzu kommt ein DC-Getriebemotor ($\varnothing 10$ mm, 30-60 U/min bei 3-6 V) mit einer fest montierten M3-Gewindewelle. Um diese Welle ist ein 12-Bit-LED-Ring in das Gehäuse eingelassen. Auf der Welle

des Getriebemotors ist ein Blasringrad formschlüssig befestigt.

Für die Herstellung dieses Rades bestehen mindestens drei Optionen: (1) die Nutzung eines fertigen Blasringrades, (2) die Eigenfertigung aus wenigen Komponenten – speziell einem Fugenkreuz und (3) vier einzelnen Blasringen oder die Anfertigung mittels 3D-Druck.

Der Antrieb des Getriebemotors und die Animation des LED-Rings werden über ein Arduino Uno R3 Board gesteuert. Der Mikrocontroller wird dabei mit einem Motortreiber-Shield ergänzt, um eine konstante Steuerung des Motors zu gewährleisten, da dieser im Betrieb auch einen Richtungswechsel vollziehen soll. Das Board wird während des Betriebs durch ein 9V-Blockbatteriefach mit Klinkenstecker und integriertem An-/Aus-Schalter mit der notwendigen Spannung versorgt. Zudem können sowohl die Animation des LED-Rings als auch Geschwindigkeits- und Richtungs-werte des Getriebemotors über den bereitgestellten Arduino-Code verändert werden.

Bei der eigenen Erprobung der dualen Motorsteuerung über das Arduino-Board bzw. des Motortreiber-Shields konnte keine ausreichende Spannung über den Arduino bzw. das Motortreiber-Shield mit externer Spannungsversorgung erzeugt werden, sodass der $\varnothing 24$ mm DC-Motor eine konstant ausreichende Drehzahl für eine angemessene Windstärke generieren konnte. Deshalb wurde konstruktiv ein DC-Drehzahlregler mit eigener Spannungsversorgung realisiert, der in der Verteilerdose an- bzw. untergebracht ist. Dadurch kann die Windstärke in sehr feinen Schritten so reguliert werden, dass ein ausreichendes Verhältnis der Windstärke bei unterschiedlichen Zusammensetzungen und weiteren Einflussfaktoren der verwendeten Seifenlauge eingestellt werden kann, um eine konstante Funktionalität unter allen Bedingungen zu gewährleisten.

Die Verwendung einer flüssigen Seifenlauge erfordert eine sorgfältige Auswahl der Materialien und Hilfsstoffe sowie Schutzmaßnahmen für die elektronischen Komponenten, um Kontakt mit der Lauge zu vermeiden oder das Risiko dafür zu minimieren. Sowohl der Mikrocontroller als auch der Drehzahlregler und das Batteriefach des Reglers wurden zum besonderen Schutz und zur Erleichterung der Installation in eine Verteilerdose eingebaut, die gemäß der Schutzklasse IP66 vor starkem Spritzwasser schützen soll. Das Gehäuse bzw. die Verteilerdose dient neben dem Schutz sämtlicher elektronischer Komponenten auch dazu, diese Komponenten möglichst sicher und kompakt auf der Grundplatte zu fixieren.

Durch die unumgängliche Notwendigkeit der Verwendung einer flüssigen Seifenlauge ist die Wahl der verwendeten Materialien und Hilfsstoffe ebenso von Bedeutung wie der Einfluss der Flüssigkeit auf die Funktion der elektronischen Komponenten, die nicht in der Verteilerdose, sondern im Motorengehäuse untergebracht werden mussten. Deshalb wurden diese Komponenten so angebracht, dass die elektronischen Bauteile möglichst nicht mit der flüssigen Seifenlauge in Kontakt treten.

DIFFERENZIERUNGSMÖGLICHKEITEN

Das Fertigen des mikrocontrollergestützten Seifenblasenautomaten bietet, ähnlich wie seine Vorgängervarianten, vielfältige Anpassungsmöglichkeiten für die Umsetzung im Technikunterricht. Durch eine gezielte Aufbereitung der Materialien können Schüler*innen ausgehend von ihren individuellen Fähigkeiten und Möglichkeiten in verschiedene Phasen der Fertigung einsteigen. Zum Beispiel können komplexere Bauteile und Fertigungsschritte wie das Fertigen des Motorengehäuses vorab von der Lehrkraft vereinfacht oder durch die Bereitstellung von Musterteilen erleichtert werden. Der Einsatz von Erklärvideos oder angepassten Materialien wie Checklisten kann ebenso einen weiteren Differenzierungsansatz ermöglichen.

Der Seifenblasenautomat kann sowohl als ganzheitliche Fertigungsaufgabe mit eigenständiger Umsetzung durch die Schüler*innen realisiert werden, als auch als Bausatz, der ausreichend Zeit ermöglicht, besonders die elektro- und informationstechnischen Inhalte sowie die Thematisierung des Mikrocontrollers (z.B. Blockdiagramme zur Programmierung etc.) vertiefend zu behandeln. Eine weitere Schwerpunktsetzung wäre, auch die additive Fertigungsmethode des 3D-Drucks zusätzlich einzuführen. So kann beispielsweise der Standardbehälter (Schüssel) für die Seifenlauge durch ein speziell für diese Variante entworfenes Muster eines Seifenlaugenbehälters, der fest mit der Grundplatte montiert wird, im 3D-Druck hergestellt und ersetzt werden. Darüber hinaus ist es auch denkbar, dass die Schüler*innen eigene Designs des Laugenbehälters mittels CAD-Software erstellen und personalisieren können, was auch die Fertigung eines angepassten Seifenblasenringrads und/oder eines Gehäuses für die elektronischen Komponenten mittels 3D-Druck umfassen kann.

KLEINER EXKURS: 3D-DRUCK

Die im Folgenden vorgestellte Schritt-für-Schritt-Anleitung gibt nur einen Einblick in das Themenfeld des 3D-Drucks. An dieser Stelle soll kurz vorgestellt werden, wie der 3D-Druck speziell für die Fertigung des Seifenlaugenbehälters (Abb. 2) aussehen kann und wie dieser exemplarisch umgesetzt wurde.

Ein Seifenlaugenbehälter lässt sich mithilfe des 3D-Drucks optimal an die Ausführung des Seifenblasenautomaten anpassen. So kann der Behälter an unterschiedliche Durchmesser des verwendeten Seifenblasenringrades individualisiert und additiv gefertigt werden. Ein Muster eines Laugenbehälters für ein gebrauchsfertiges Seifenblasenringrad mit einem Durchmesser von 200 mm wurde mithilfe des Programms SHAPR3D modelliert, wobei grundsätzlich sämtliche CAD-Programme aufgrund der vergleichsweise geringen Komplexität des Modells geeignet sein sollten. Das Programm SHAPR3D zeichnet sich erfahrungsgemäß durch seine intuitive Bedienung aus, insbesondere in Kombination mit der App auf einem Tablet und der Nutzung eines Tablet-Stifts.

In der exemplarischen Konstruktion wurden Befestigungen des Laugenbehälters an der Grundplatte berücksichtigt

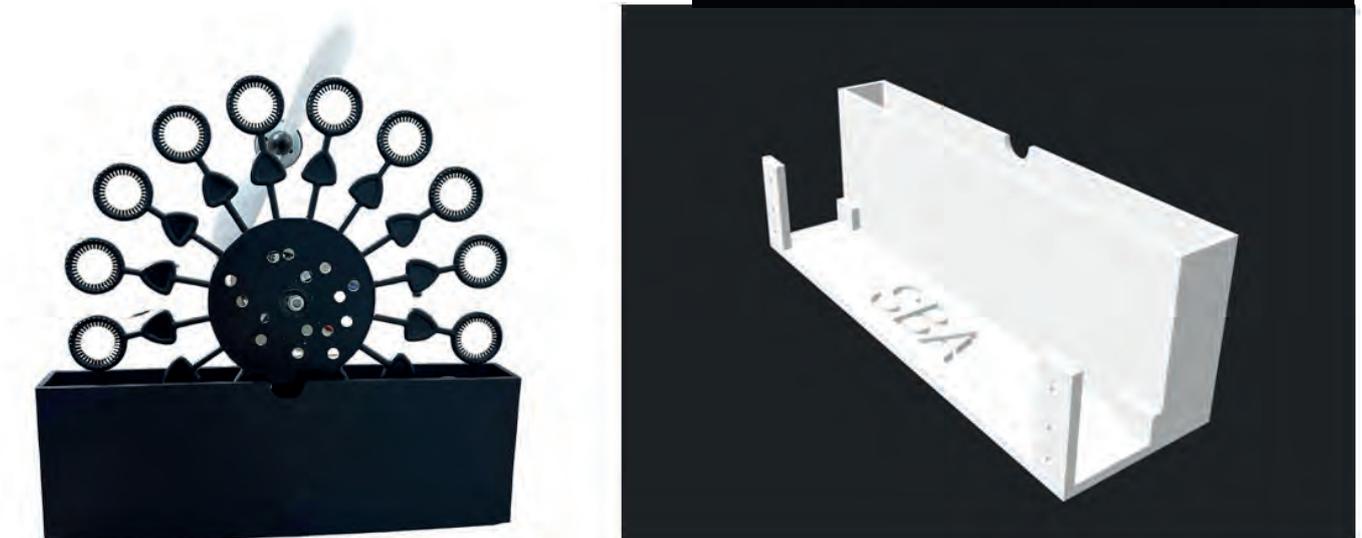


Abbildung 2: Seifenlaugenbehälter und dazugehörige STL-Datei (unten), sowie STL-Datei: Ergänzungsbogen (oben)



Abbildung 3: STL-Datei: Ergänzung Bogen

(Abb. 3). Eine weitere Konstruktionsmöglichkeit ist ein zusätzlich über das Blasringrad verlaufender Bogen, der das Abfließen überschüssiger Seifenlauge zurück in den Behälter ermöglicht (Abb. 2b). Hierbei ist allerdings zu beachten, dass dies aufgrund des benötigten Radius nur in 3D-Druckern gedruckt werden kann, die eine ausreichend große Druckbettfläche aufweisen. Darüber hinaus sollten sämtliche, vom verwendeten 3D-Drucker abhängige, einstellbare Parameter, wie z.B. das verwendete Druckmaterial und die Druckgeschwindigkeit, berücksichtigt werden, um sicherzustellen, dass der Seifenlaugebehälter wasserdicht und funktional ist. Es wäre außerdem denkbar, auch das Seifenblasenringrad selbst zu konstruieren, zeichnen und drucken oder auch weitere Gehäuseteile und Komponenten.

ARBEITSSICHERHEIT UND HILFSMITTEL BEI DER FERTIGUNG

Die Lehrkraft entscheidet, welche Werkzeuge und Maschinen in Abhängigkeit von der Klasse und der Schwerpunktsetzung verwendet werden. Für eine entsprechende Auswahl sollten unbedingt die in Tabelle 3 im Anhang aufgelisteten

Gefährdungen und Maßnahmen berücksichtigt werden. Insbesondere erfordert die Fertigung des Werkstücks Bohrungen, welche mit einer möglichst hohen Maßgenauigkeit mit der Ständer-/Tischbohrmaschine erfolgen sollten. Die Bohrungen sollten dabei mithilfe des Tiefenanschlags unter Berücksichtigung der für die jeweiligen Bohrergrößen vorgesehenen Drehzahlen durchgeführt werden. Dies gewährleistet den Schutz der Werkzeuge, die Qualität und die Maßgenauigkeit des Werkstücks. Jedoch werden auch bei der Vorbereitung der Verteilerdose und bei optionaler Fertigung eines Seifenblasenringrades Bohrungen mithilfe eines Akkuschraubers notwendig. Hier ist es wichtig, eine sichere Fixierung der jeweils zu bohrenden Bauteile, z.B. mit einem Schraubstock oder Schraubzwingen, vorzunehmen und ebenso bei Bohrungen auf glatten Kunststoffoberflächen die Bohrpunkte vorab mithilfe eines Körners vorsichtig zu markieren, um das Risiko des Abrutschens zu minimieren. Durch die beim Bohren und Schleifen anfallende Holzspäne herrscht Rutsch- und Gesundheitsgefahr. Schleifarbeiten sollten daher möglichst auf ein Minimum beschränkt werden. Der bei der Fertigung entstehende Holzstaub sollte beim Schleifen bereits während des Arbeitsvorgangs und beim Bohren nach dem Arbeitsvorgang mit einer Absaugmaschine der Holzstaubklasse M entfernt werden (Deutsche Gesetzliche Unfallversicherung e.V. [DGUV], 2006, S. 11). Ebenso werden Lötarbeiten notwendig. Dabei muss darauf geachtet werden, dass die Schüler*innen sicher im Umgang mit allen zum Löten erforderlichen Regeln und Schutzausrüstungen vertraut sind. Neben der Verwendung einer feuerfesten Unterlage sind dabei Regeln zur Einrichtung des Arbeitsplatzes beim Löten zu beachten, wie z.B. das Anbringen der Kabel der Lötstation. Weiterhin ist beim Löten zu beachten, dass die Konzentration an gesundheitsgefährdenden Stoffen in der Atemluft minimiert werden muss (Deutsche Gesetzliche Unfallversicherung e.V. [DGUV], 2010, S. 28), indem für eine ausreichende Raumbelüftung und/oder die Verwendung einer Absaugung gesorgt wird. Bleihaltiges Lot darf dabei nicht im Unterricht verwendet werden (ebd.). Darüber hinaus sollten Kenntnisse für das Praktizieren des Lötens bekannt sein, um elektronische Bauteile nicht zu beschädigen. Sollten bisher keine Lötarbeiten von den Schüler*innen durchgeführt worden sein, bietet es sich an, vorab Übungen durchzuführen. Ebenso muss beachtet werden, dass die Platine des Arduino-Boards nicht isoliert ist und deshalb nicht mit metallischen oder anderen leitenden Oberflächen in Kontakt kommen darf, da dies sonst Kurzschlüsse auslösen und den Mikrocontroller beschädigen kann. Ebenso muss sichergestellt werden, dass alle potenziellen Kurzschlussstellen in der gesamten Elektronik vor Inbetriebnahme überprüft und ggf. ausreichend nachisoliert werden.

Für den späteren Betrieb gilt es darüber hinaus zu beachten, dass das Werkstück nur im Freien in Betrieb genommen werden sollte, da durch die erzeugten Seifenblasen Rutschgefahr bestehen kann.

BENÖTIGTE MATERIALIEN, WERKZEUGE UND MASCHINEN

Die für die Fertigung benötigten Materialien werden in Tabelle 1, die benötigten Werkzeuge und Maschinen werden in Tabelle 2 aufgeführt.

Menge	Material	Maße/Durchmesser/ Länge/Art
1	Alleskleber (z.B. von UHU)	
1	Alleskleber wasserdicht (z.B. von Pattex)	
1	Arduino-Mikrocontroller Uno R3	
1	Arduino Motortreiber-Shield Rev3	
1	Batteriefach	für 4x AAA Batterien
1	Batteriehalter	9V Blockbatterie - Hohlstecker
4	Blasring z.B. von Pustefix	
1	DC-Getriebemotor	ø 10 mm: 30-60 U/ min (6V), Welle mit M3 Gewinde, Länge Gewinde 55 mm
1	DC-Motor	ø 24 mm (3-6 V)
2	Flachkopfschraube	2,2x8 mm
2	Flachkopfschraube	4x12 mm
2	Flachkopfschraube	Optional zur Fixie- rung des LED Rings
1	Fugenkreuz	30x30x5 mm
1	Holzleim	
1	Holzplatte	200x200x15 mm
4	Holzquadrate	30x30x15 mm
3	Jumper-Kabel (rot/schwarz/blau)	
4	Kabelbinder	
1	Kantholz	150x50x50 mm
2	M3-Mutter	
1	M3-Unterlagscheibe	
1	PWM-Motor-Drehzahlsteller	
1	RGB LED Ring 12-bit	5V, außen ø 38 mm, innen ø 27 mm
1	Rotorblatt	ø 120 mm
3	Schaltdraht, rot	400 mm
3	Schaltdraht, schwarz	400 mm
1	Schaltdraht, blau	400 mm
4	Schrumpfschlauch	2,5x20 mm
1	Verteilerdose/Gehäuse	158x90x60 mm

Tabelle 1: Benötigtes Material

Benötigte Werkzeuge und Maschinen	Größe/Durchmesser/Körnung
Abisolierzange	
Akkuschrauber	
Bleistift	
Dritte Hand (Löthilfe)	
Fön/Heißluftfön	
Forstnerbohrer	ø 40 mm und 25 mm
Hitzebeständige Handschuhe	
Holzbohrer	ø 10 mm und 7 mm
HSS-Bohrer	ø 10 mm, 7 mm und 3 mm
Leim- oder Schraubzwinge(n)	
LötKolben/Lötstation	
Feuerfeste Lötunterlage	
Lötzinn	Bleifrei
Maschinenschraubstock	
Rundfeile klein (ggf.)	
Senker	Handsenker
Schleifpapier	120er
Schlosser- oder Schreinerwinkel	
Schraubendreher	
Seitenschneider	
Ständer-/Tischbohrmaschine	
Stahlmaßstab/Lineal/Streich- maß	
Vorstecher	

Tabelle 2: Benötigte Werkzeuge und Maschinen

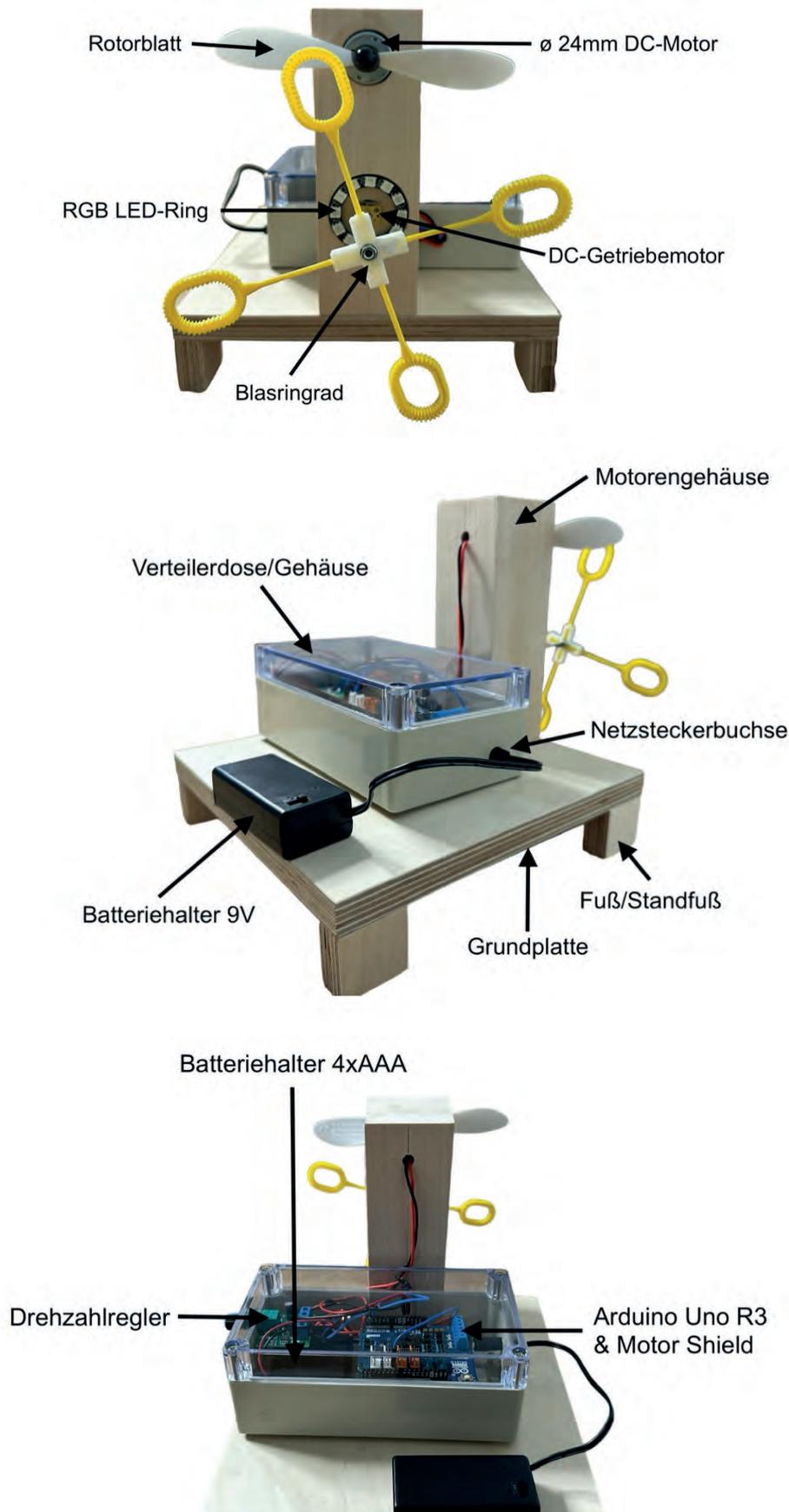


Abbildung 4: Bauteile und Aufbau (3 Ansichten)

SCHRITT-FÜR-SCHRITT-ANLEITUNG

Im Folgenden werden alle Arbeitsschritte zur Fertigung des mikrocontrollergestützten Seifenblasenautomaten aufgezeigt und zur besseren Verständlichkeit mit Fotos visualisiert. Vorab soll jedoch die in Abbildung 4 dargestellte Übersicht den Aufbau und die einzelnen Bauteile des Seifenblasenautomaten beschreiben, sodass die Bezeichnungen der einzelnen Arbeitsschritte besser nachvollziehbar sind.

Der im Weiteren beschriebene Arbeitsschritt zur Fertigung des Blasringrads ist dabei ein optionaler Schritt: Er ist nur notwendig, wenn das Seifenblasenringrad selbst gefertigt wird. Alternativ kann auch ein fertiges Seifenblasenringrad, z.B. von EUROLITE, verwendet oder ein 3D-Druck realisiert werden.

Motorengehäuse: Bohrpunkte Anzeichnen Vorder- und Rückseite

Beginnend mit dem ersten Arbeitsschritt wird für die Fertigung des Motorengehäuses das 50 x 50 x 150 mm große Kantholz benötigt. Die Maße aus der Technischen Zeichnung* (TZ) „Motorgehäuse“ sind dabei genau auf das Kantholz zu übertragen. Sobald die Maße präzise übernommen wurden (Abb. 5), können die Bohrungen erfolgen. Die genaue Übertragung ist essenziell, um eine Kollision des Rotorblattes mit der Getriebewelle des Getriebemotors während des späteren Betriebs zu vermeiden. An der Rückseite des Motorengehäuses sind drei weitere Bohrungen vorzunehmen. Auch hier ist es wichtig, dass die Maße aus der TZ „Motorgehäuse“ präzise übertragen werden (Abb. 6), um die Zentrierung mit den vorderen Bohrlöchern zu gewährleisten, die später als Kabelschächte für die Motoren und den LED-Ring dienen.



Abbildung 5: Vorderseite Motorengehäuse anzeichnen



Abbildung 6: Rückseite Motorengehäuse anzeichnen

Motorengehäuse: Bohren

Bei den folgenden Bohrarbeiten sind die Sicherheitsvorschriften der Ständerbohrmaschine zu beachten. Vor dem Bohren kann zusätzlich zur Zentrierspitze des Holzbohrers ein Vorstecher verwendet werden (Abb. 7), um eine höhere Genauigkeit zu erreichen. Das Werkstück ist fest in einen Maschinenschraubstock einzuspannen. Für die verschiedenen Bohrergrößen sind die passenden Drehzahlen an der Tisch- oder Ständerbohrmaschine einzustellen. Das Werkstück darf erst nach Stillstand der Maschine ausgespannt werden. Ebenso ist beim Ausspannen zu beachten, dass eine Holzunterlage verwendet wird, um Beschädigungen am Bohrer zu vermeiden, falls dieser aus der Hand entgleiten sollte¹. Beim Bohren ist zudem zu beachten, dass die Tiefenanschläge für die jeweiligen Bohrpunkte der Vorderseite unterschiedlich angepasst werden müssen:

- Motor (\varnothing 24 mm): Bohrung \varnothing 25 mm – Tiefenanschlag: 45 mm (Forstnerbohrer)²
- Getriebemotor: Bohrung \varnothing 10 mm – Tiefenanschlag: 30 mm
- LED-Ring: Bohrung \varnothing 40 mm – Tiefenanschlag: 5 mm (Forstnerbohrer)

Alle drei Bohrungen an der Rückseite des Motorengehäuses erfolgen mit einem \varnothing 7 mm Holz- oder HSS-Bohrer. Der Tiefenanschlag ist auf 45 mm einzustellen, um eine ausreichende Bohrtiefe zu erreichen und gleichzeitig Beschä-

- 1 Diese Hinweise gelten für alle weiteren Bohrarbeiten in dieser Schritt-für-Schritt Anleitung.
- 2 Abhängig von der Länge der verwendeten Motoren bzw. dem Durchmesser des LED-Rings. Abweichungen von der Materialliste sind möglich.

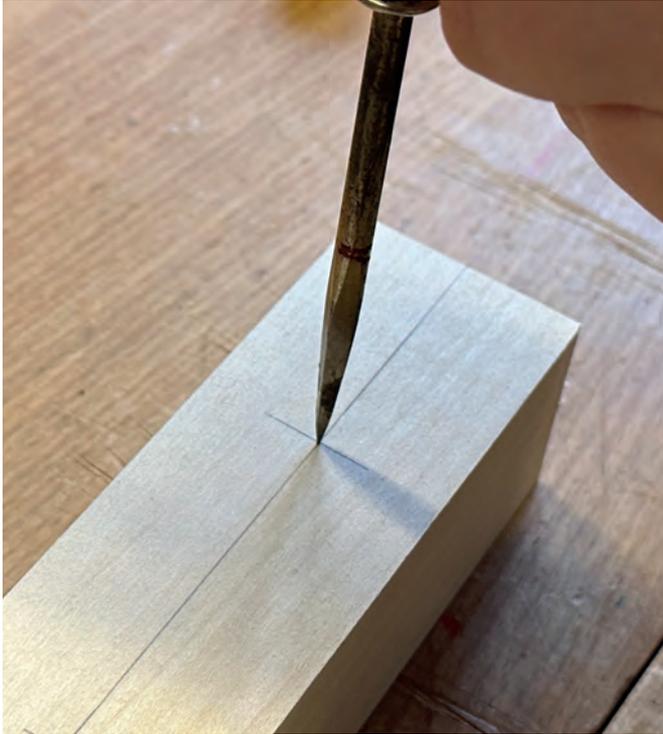


Abbildung 7: Bohrpunkte vorstechen

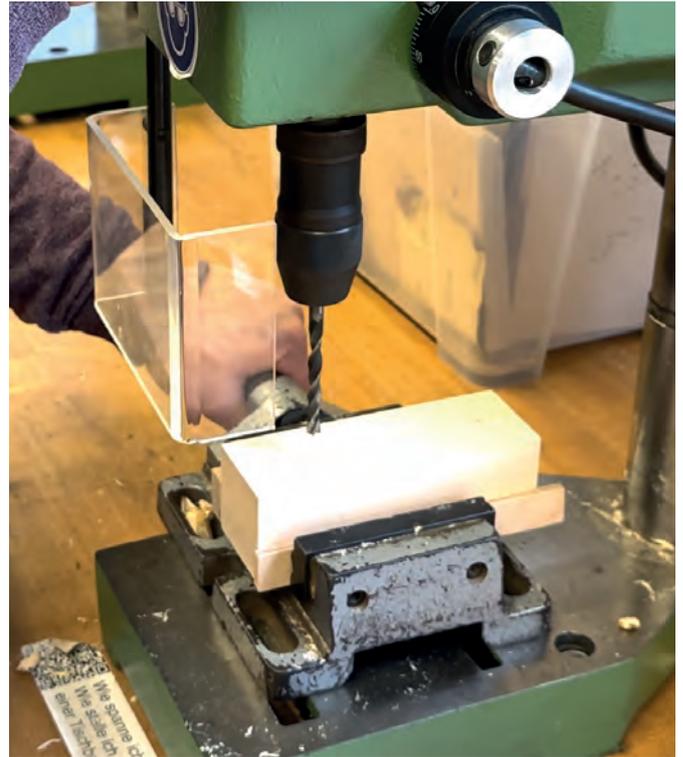


Abbildung 10: Bohren der Rückseite



Abbildung 8: Bohren der Vorderseite



Abbildung 9: Fertigstellen der Vorderseite



Abbildung 11: Fertigstellen der Rückseite

digungen des Maschinenschraubstocks oder anderer Bauteile zu vermeiden. Das Werkstück ist vor dem Bohren fest einzuspannen und erst nach Stillstand der Maschine wieder auszuspannen. Die Drehzahl der Ständer- oder Tischbohrmaschine muss an den $\varnothing 7$ mm Bohrer angepasst werden.

Nachdem alle Bohrungen durchgeführt wurden, können die Bohrlöcher, wenn nötig, mit einem Schleifpapier oder einer Rundfeile entgratet werden (Vorder- und Rückseite des Motorengehäuses).

Motorengehäuse: Anzeichnen der Position auf der Grundplatte

In diesem Arbeitsschritt wird zunächst die Position des Motorengehäuses auf der Grundplatte markiert. Die entsprechenden Maße sind der TZ „Grundplatte“ zu entnehmen. Eine präzise Maßübertragung ist essentiell, um sicherzustellen, dass das Motorengehäuse später korrekt zentriert und kantenbündig auf der Grundplatte sitzt. Dies ist für die Funktionsfähigkeit der späteren Konstruktion ausschlaggebend.

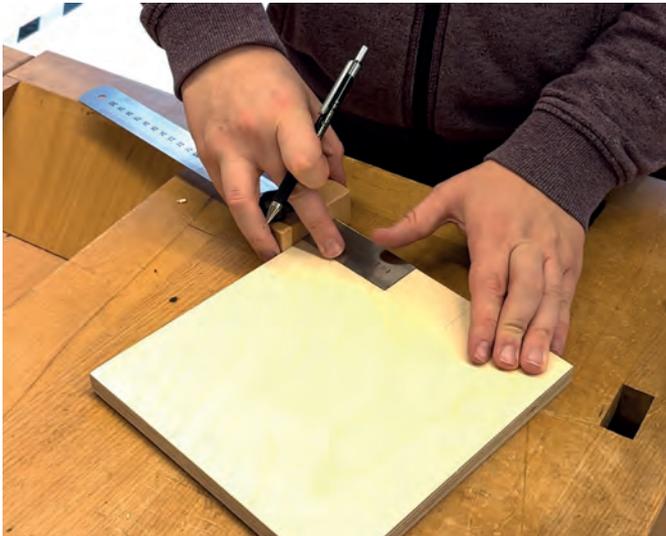


Abbildung 12: Grundplatte anzeichnen

Grundplatte: Ausrichten und Verleimen der Standfüße

Die Ausrichtung der Standfüße hängt vom Durchmesser des gewählten Seifenblasenringrades ab:

Die Standfüße können so angepasst werden, sodass eine Erhöhung der Grundplatte um 15 mm (Abb. 13) oder um 30 mm (Abb. 14) möglich ist. Bei einem Seifenblasenringrad mit einem Durchmesser von 160 mm reicht es aus, die Füße in horizontaler Ausrichtung zu verleimen, um eine Erhöhung von 15 mm zu erreichen. Bei einem Durchmesser von 200 mm des Seifenblasenringrads ist hingegen eine Erhöhung von 30 mm erforderlich. In diesem Fall müssen die Standfüße vertikal mit der Grundplatte verleimt werden. In beiden Fällen wird so sichergestellt, dass das Seifenblasenringrad während des Betriebs im Seifenlaugenbehälter frei drehen kann.

Wenn die korrekte bzw. erforderliche Höhe der Standfüße feststeht, werden die Standfüße in der entsprechenden Position an die Grundplatte angebracht. Hierfür werden die vier Holzquadrate an den äußeren Kanten der Unterseite mit

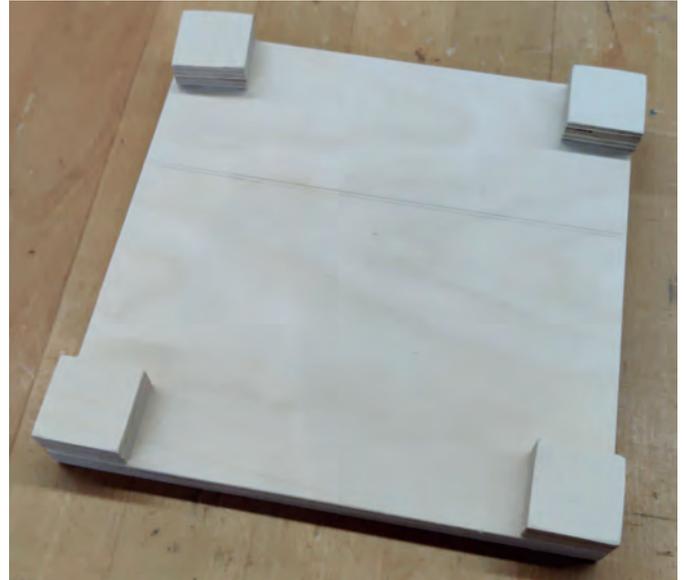


Abbildung 13: Horizontale Ausrichtung

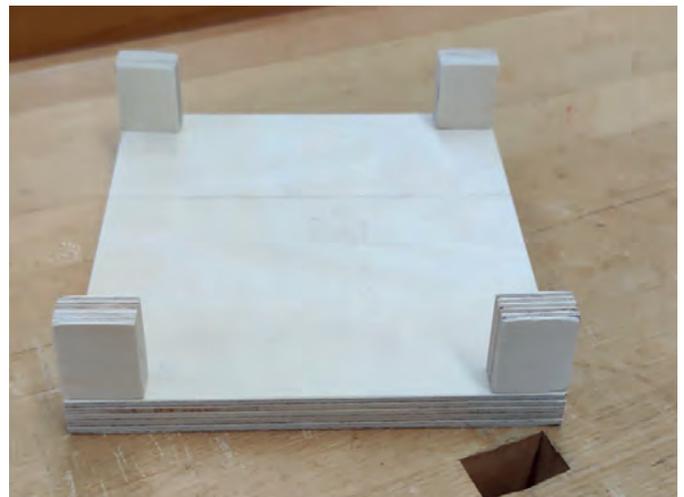


Abbildung 14: Vertikale Ausrichtung



Abbildung 15: Verleimen der Standfüße

der Grundplatte verleimt und mit Leim- oder Schraubzwingen (Abb. 15) für mindestens 20 Minuten fixiert³.

Grundplatte: Ausrichten und Verleimen des Motorengehäuses

Nun kann das Motorengehäuse auf der in Schritt „Motorengehäuse: Anzeichnen der Position auf der Grundplatte“ (S. 16) markierten Position mit der Grundplatte verleimt und mit einer Schraubzwinde fixiert werden (Abb. 16). Der Holzleim sollte für mindestens 20 Minuten⁴ trocknen, bevor der nächste Arbeitsschritt erfolgt. Bei der Ausrichtung ist zu beachten, dass die Öffnungen der Motorschächte korrekt in Richtung der Kante zeigen. Eine spätere Korrektur ist nach dem Austrocknen des Holzleims nicht mehr möglich.

Motoren: Verkabelung

In diesem Schritt erfolgt die Verkabelung beider Motoren. Zu-

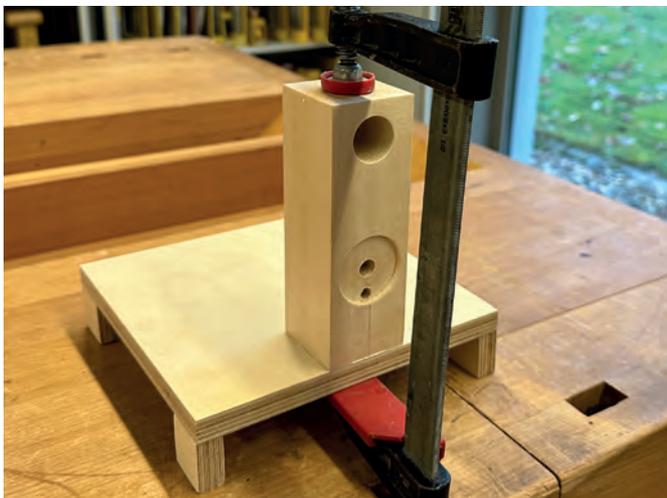


Abbildung 16: Verleimen des Motorengehäuses

erst sind je zwei schwarze und rote Schaltdrähte (SD) an jeweils einem Ende abzuisolieren. Danach werden Schrumpfschläuche über die SD geschoben. Anschließend sind die SD in die entsprechenden Ösen (roter SD: +Pol und schwarzer SD: -Pol) an den Motoranschlüssen einzuführen und diese leicht zu verdrehen. Zum Abschluss werden die Schrumpfschläuche über die offenen Kontaktstellen der SD und Ösen geschoben, um diese zu isolieren. Sie werden dann kurzzeitig mit einem Heißluftföhn oder einem haushaltsüblichen Föhn erhitzt, bis sie fest sitzen und somit die Anschlüsse korrekt isoliert sind. Alternativ können die SD ebenso fest mit den Kontaktstellen an den Motoren verlötet werden.

Aufgrund der starken Hitzeentwicklung im Luftstrom des Föns ist es ratsam, hitzebeständige Handschuhe oder eine Zange zu verwenden, um Verbrennungen zu verhindern.

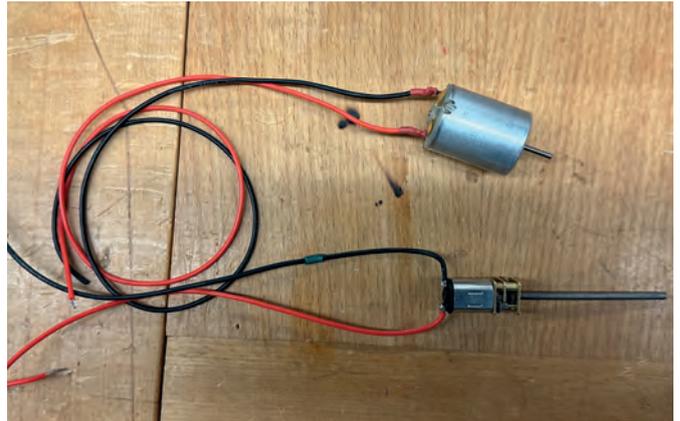


Abbildung 17: Verkabelung der Motoren

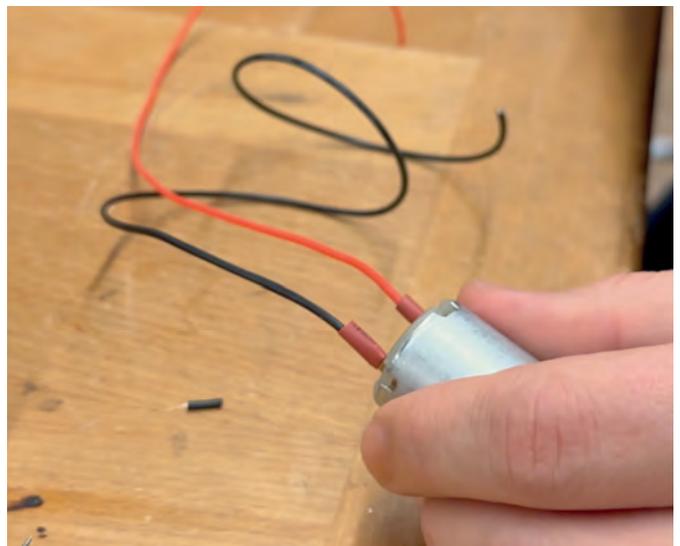


Abbildung 18: Verkabelung des einen Motors

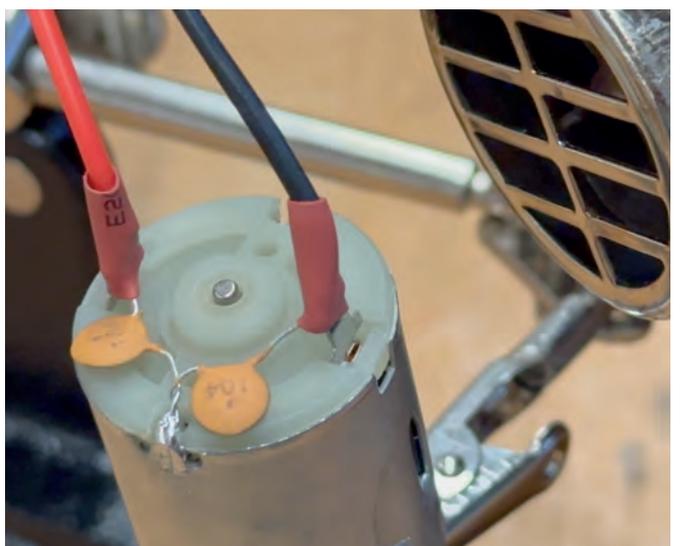


Abbildung 19: Isolierung des einen Motors

³ Je nach verwendetem Holzleim kann die Trockenzeit variieren.

⁴ Je nach verwendetem Holzleim kann die Trockenzeit variieren.

Motoren: Installation

Nun werden die SD der Motoren durch die jeweiligen Kabelschächte zur Rückseite des Motorengehäuses geführt. Wenn nun alle Schaltdrähte verlegt sind, können die beiden Motoren im Motorengehäuse fixiert werden. Dazu wird eine geringe Menge Alleskleber auf die Außenseite der Motoren aufgetragen und diese vorsichtig in den jeweils vorgesehenen Motorschacht eingesetzt (Abb. 20). Es ist wichtig, darauf zu achten, dass der Kleber vollständig aushärtet, bevor der Seifenblasenautomat in Betrieb genommen wird. Dies gewährleistet, dass die Motoren fest und sicher im Gehäuse sitzen. Die Trocknungszeit des Klebers hängt von dessen Zusammensetzung ab und sollte gemäß den Herstellerangaben eingehalten werden.

Zudem gilt es während des Verklebens und der Trocknungsphase zu beachten, dass keine Klebstoffreste in die Motoren (z.B. Lüftungsschlitze) oder auf die Schaltdrähte gelangen, da dies deren Funktion beeinträchtigen könnte. Zudem sollte das Holzgehäuse vor dem Auftragen des Klebers sauber und trocken sein, um eine optimale Haftung zu gewährleisten.



Abbildung 20: Einführen des einen Motors in das Gehäuse

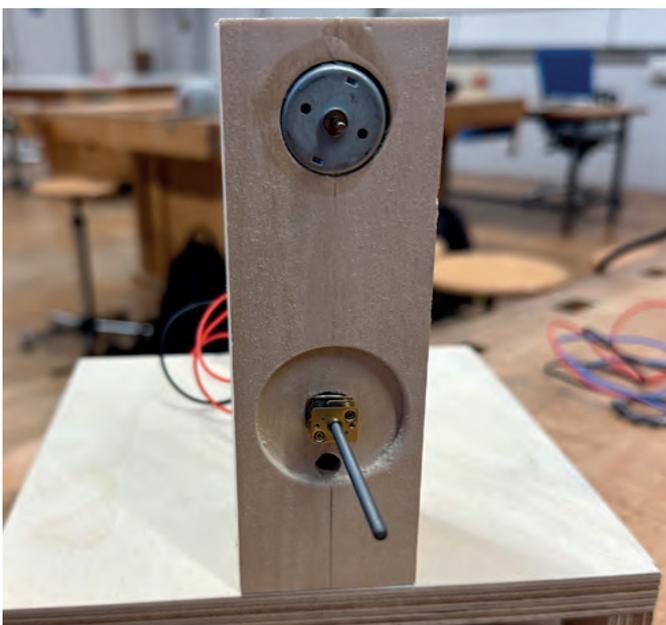


Abbildung 21: Installierte Motoren im Gehäuse

LED-Ring: Verlöten

In diesem Schritt müssen drei SD (schwarz für GND, rot für VCC und blau für IN⁵) an den LED-Ring angelötet werden.

Dafür ist es zunächst notwendig, die Enden der Drähte abzuisolieren (ca. 5 mm Isolierung entfernen, sodass der blanke Draht freiliegt). Im Anschluss daran, werden die Drahtenden leicht verdrillt, sodass keine Drahtadern abstehen.

Vor dem Lötten empfiehlt es sich, den LED-Ring vorsichtig mit einer Krokodilklemme in einer Löthilfe (z.B. „Dritte Hand“) zu fixieren. Dies verhindert Beschädigungen und erleichtert den Lötvorgang.

Die ideale Temperatur des LötKolbens liegt zwischen 300 und 350°C⁶. Ohne Lötterfahrung sollten vorab Übungen durchgeführt werden. Des Weiteren ist es wichtig zu beachten, sowohl bleifreies Lötzinn und eine feuerfeste Lötunterlage zu verwenden als auch für eine ausreichende Frischluftzufuhr zu sorgen.

Beim Lötten der Kabel sollten die abisolierten Drahtenden sauber verzinkt sein. Dazu berührt man mit der Spitze des LötKolbens das Ende eines Drahtes und hält das Lötzinn an den Draht, sodass es schmilzt und den Draht gleichmäßig bedeckt (Abb. 22). Dies wird für alle drei Drahtenden wiederholt.



Abbildung 22: Verzinnen der SD

Die drei Kontakte am LED-Ring müssen ebenfalls verzinkt werden (Abb. 23). Es ist wichtig, die Lötzeit hierbei kurz zu halten, um Schäden am Bauteil zu vermeiden. Beim Anlöten der SD müssen sowohl jeweils der SD als auch die jeweilige Anschlusskontaktfläche am LED-Ring gleichzeitig erhitzt werden (Abb. 24).

Die Drähte sollten mit einer Lötpinzette gehalten werden, da sie sich schnell erhitzen. Darüber hinaus kann man die Drähte mithilfe der Lötpinzette genauer fixieren.

Für das Anlöten der SD ist es ebenso zu beachten, dass die SD nach innen, d.h. in Richtung des Innendurchmessers (Abb. 25) gelegt bzw. angelötet werden müssen. Dies ist deshalb wichtig, sodass im weiteren Verlauf die SD sauber in die

5 GND = Ground/Masse, VCC = Voltage at the Common Collector/ positive Versorgungsspannung IN = Input/Eingang.

6 Die optimale Temperatur hängt von der Zusammensetzung des Lötzinns ab. Bleifreies Lötzinn erfordert oft eine höhere Temperatur. Im Zweifel immer Herstellerangaben beachten.

Kabelschächte des Motorengehäuses gelegt werden können und diese nicht das Einlassen des LED-Rings in die vorgesehene Öffnung (\varnothing 40 mm) behindern.

Nach dem Anlöten ist es ratsam, die Verbindungen auf potenzielle Kurzschlussstellen oder kalte Lötstellen zu überprüfen und bei Bedarf nachzubessern.

Um sicherzustellen, dass die jeweiligen Verbindungen gut gelungen sind, wird überprüft, ob das Lötzinn gleichmäßig um die jeweilige Verbindungsstelle geflossen ist und eine glänzende Oberfläche aufweist. Eine gute Lötstelle sollte keine Klumpen oder kalten Lötstellen aufweisen.

Isolation der Verbindungsstelle: Sofern die offenen Lötstellen am LED-Ring zu nah aneinander liegen und somit Kurzschlussgefahr besteht, können diese auch mit etwas Isolierband isoliert werden.



Abbildung 23: Verzinne der Kontakte

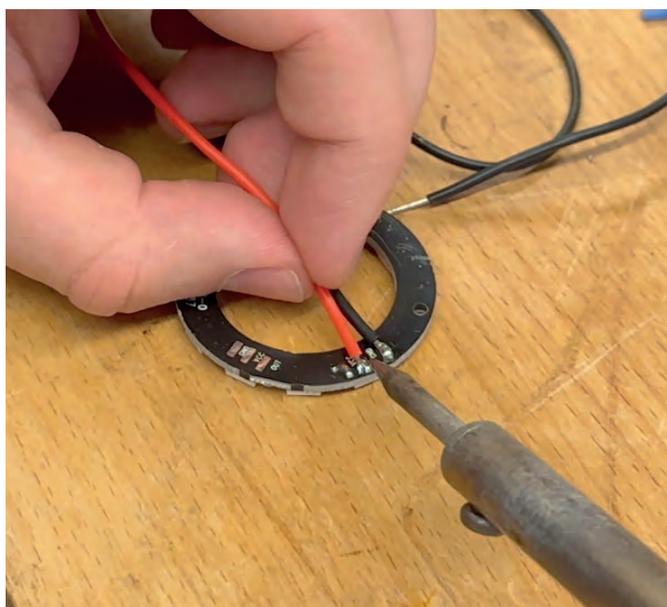


Abbildung 24: Verbinden der SD mit den Kontakten

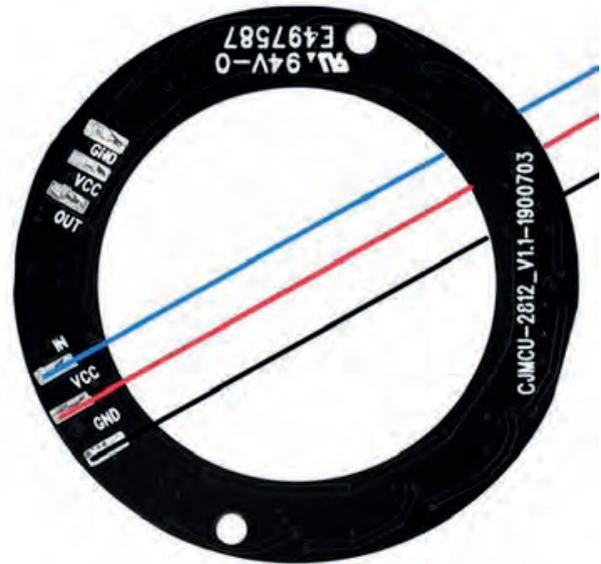


Abbildung 25: Anlötrichtung der SD an den LED-Ring

LED-Ring: Installation

In diesem Schritt wird der LED-Ring vorsichtig in die vorgesehene Öffnung des Motorengehäuses (\varnothing 40 mm) eingelassen, indem die drei Kabel zunächst durch den Kabelschacht nach hinten eingeführt werden, bis der LED-Ring fest in der Öffnung sitzt. Abschließend kann dieser, wenn nötig, mit einer Flachkopfschraube in der dafür vorgesehenen Öffnung am Ring mit dem Motorengehäuse fest fixiert oder alternativ vorsichtig eingeklebt werden.

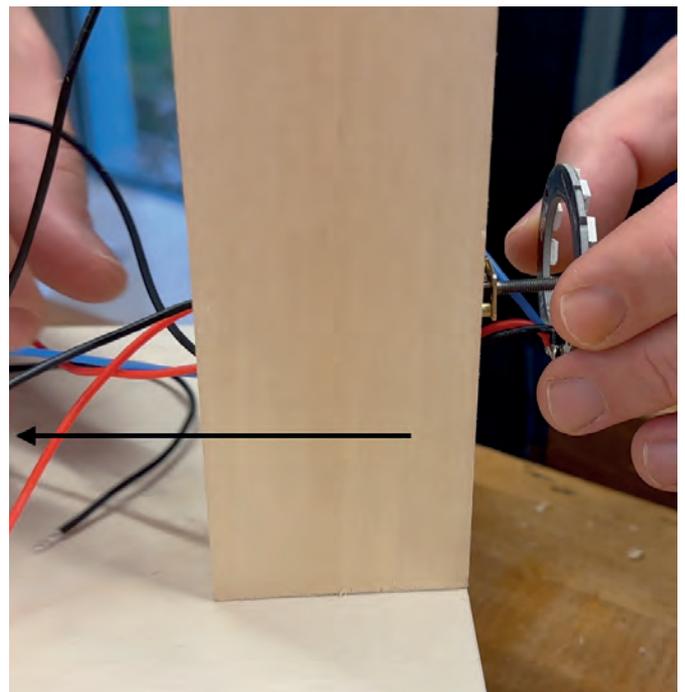


Abbildung 26: Einlassen des LED-Rings in das Gehäuse



Abbildung 27: Gehäuse mit Motoren und LED-Ring

LED-Ring und Jumperkabel: Verlöten

In diesem Schritt werden die Jumper-Kabel (rot, schwarz und blau; Abb. 28) an die gleichfarbigen SD, die zuvor an den LED-Ring angelötet wurden, verlötet. Dabei sind alle Hinweise zum Lötten aus dem vorausgehenden Arbeitsschritt zum Lötten zu beachten.

Dazu werden zuerst die Schaltdrähte des LED-Rings gegebenenfalls gekürzt und die Enden abisoliert. Dann werden die Drahtenden leicht verdrillt, um sie zu stabilisieren.

Löten der Kabel: Die abisolierten Drahtenden werden sauber verzinnt (sowohl SD als auch Jumper-Kabel-Enden). Dazu berührt man mit der Spitze des LötKolbens das Ende eines Drahtes und hält das Lötzinn an den Draht, sodass er schmilzt und den Draht gleichmäßig bedeckt. Dies wird für alle sechs Drahtenden wiederholt.

Verbinden der Drähte: Die verzinnten Enden der Drähte werden so zusammengehalten, dass sie sich überlappen. Eine Löthilfe oder Lötpinzette können hier nützlich sein. Dann erhitzt man beide verzinnten Enden gleichzeitig mit der Spitze des LötKolbens und trägt zusätzliches Lötzinn an die Verbindungsstellen auf (Abb. 29).

Um sicherzustellen, dass die jeweiligen Verbindungen gut gelungen sind, wird überprüft, ob das Lötzinn gleichmäßig um die jeweilige Verbindungsstelle geflossen ist und eine glänzende Oberfläche aufweist. Eine gute Lötstelle sollte keine Klumpen oder kalten Lötstellen aufweisen.

Isolation der Verbindungsstelle: Nach dem Abkühlen der Lötstellen werden diese mit einem Schrumpfschlauch (vgl. Arbeitsschritt: Motoren: Verkabelung) oder Isolierband isoliert (Abb. 30), um Kurzschlüsse zu verhindern.

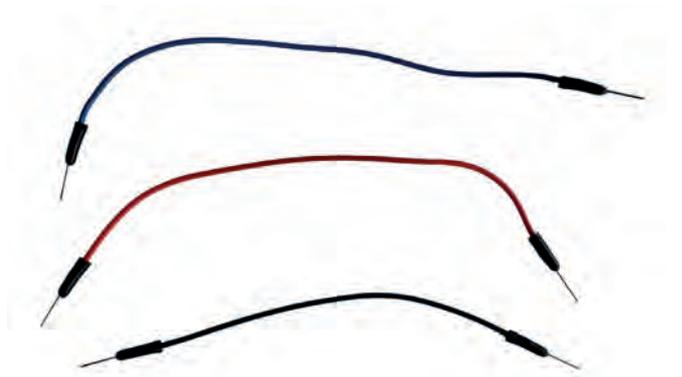


Abbildung 28: Jumper-Kabel



Abbildung 29: Verlöten der Jumper-Kabel mit den SD

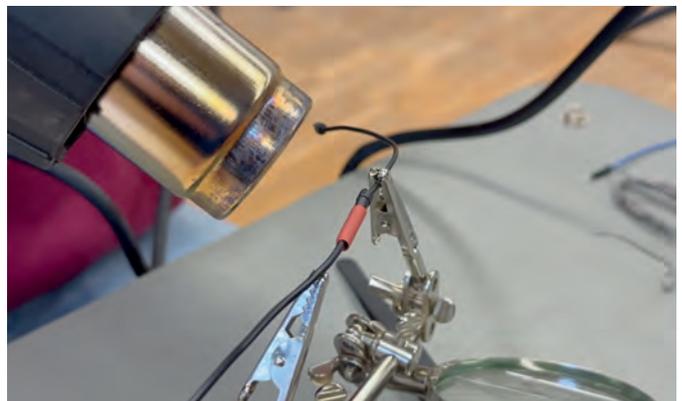


Abbildung 30: Isolation der Verbindungsstelle

Verteilerdose/Gehäuse: Anzeichnen und Bohren

In diesem Arbeitsschritt wird die Verteilerdose vorbereitet. Zuerst wählt man dazu eine geeignete Stelle an einer der Außenseiten aus, um den Drehzahlregler des Motors zu befestigen. Im vorliegenden Beispiel wurde der Regler an einer Breitseite des Gehäuses angebracht. Zuerst ist der Bohrpunkt mit einem Lineal und Permanentmarker zu markieren. Anschließend wird mit einem $\varnothing 7$ mm HSS-Bohrer das Loch für den Drehzahlregler gebohrt (Abb. 31), in das er eingeführt und mit einer Mutter abschließend fixiert wird. Hierbei sollte, wenn möglich, das Batteriefach später unterhalb des Drehzahlreglers Platz finden, wodurch die Höhe des Drehzahlreglers mitbedacht werden muss. Weiterhin ist eine Öffnung für die Netzsteckerbuchse des Arduinos

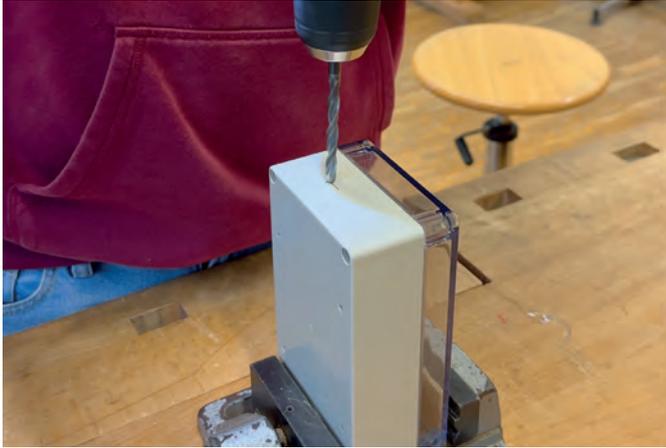


Abbildung 31: Bohren der Verteilerdose

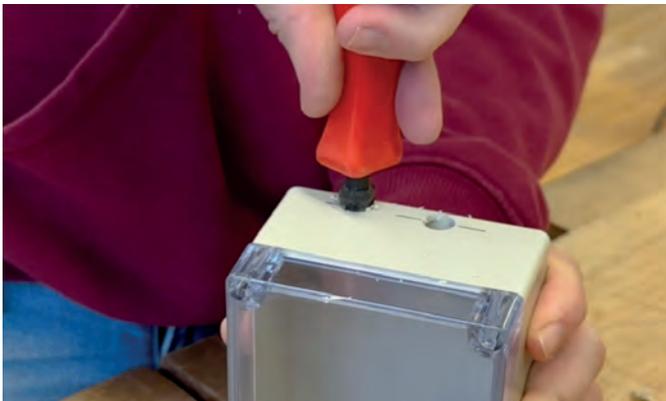


Abbildung 32: Entgraten der Bohrlöcher

notwendig. Hierfür wird das Arduino-Board auf der gegenüberliegenden Seite des Drehzahlreglers installiert, wozu eine $\varnothing 10$ mm Bohrung notwendig ist. Die Positionen der Bohrpunkte können aus der TZ „Verteilerdose/Verteilerdose Unterteil“ entnommen werden, sofern eine identische Verteilerdose verwendet wird. Abschließend wird noch ein Loch für die Kabel der elektronischen Komponenten im Motorengehäuse (Motor ($\varnothing 24$ mm), Getriebemotor und LED-Ring) mit einem $\varnothing 10$ mm HSS-Bohrer gebohrt. Beim Bohren ist darauf zu achten, dass die Verteilerdose bei allen Bohrungen fest in einem Maschinenschraubstock eingespannt wird und die Bohrlöcher zuvor mit einem Körner markiert werden, da die HSS-Bohrer auf der glatten Kunststoffoberfläche leicht verrutschen können. Dies verhindert neben möglichen Verletzungen durch Abrutschen auch ungenaue Bohrungen. Nach dem Bohren können die Bohrlöcher bei Bedarf ideal mit einem Handsenker entgratet werden (Abb. 32).

Verteilerdose/Gehäuse: Installation DC-Drehzahlregler

Nachdem die Bohrungen erfolgt sind, kann der DC-Drehzahlregler an der vorgesehenen Stelle montiert werden. Zuvor müssen die Mutter und die Drehkappe entfernt werden. Zuerst muss eine Unterlagscheibe auf den Regler aufgeschoben werden, dann wird dieser von innen nach außen (Gehäuse) durch die Bohrung geschoben (Abb. 33) und von außen mit der dazugehörigen Mutter fixiert, sodass der Drehzahlregler



Abbildung 33: Einsetzen des Drehzahlreglers

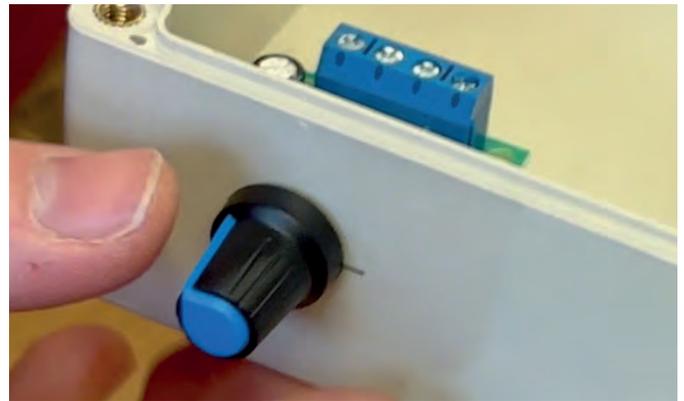


Abbildung 34: Fixieren des Drehzahlreglers

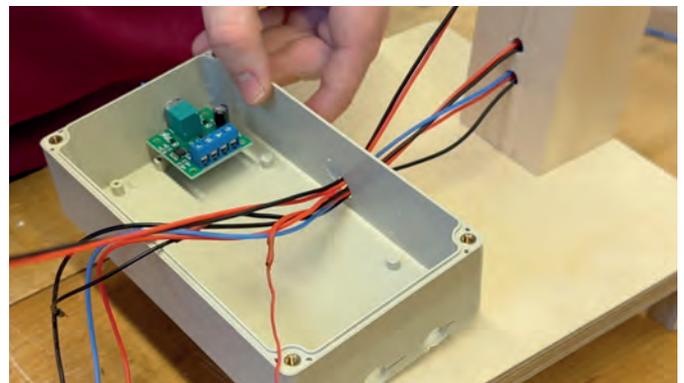


Abbildung 35: Einführen der SD

fest mit dem Gehäuse verbunden ist. Abschließend wird die Drehkappe wieder auf den Drehzahlregler gesteckt (Abb. 34).

Verteilerdose/Gehäuse: Fixieren auf der Grundplatte

Nun wird die Verteilerdose auf der Grundplatte fixiert. Zuerst müssen dafür die aus dem Motorengehäuse kommenden Schaltdrähte durch die Öffnung geführt werden (Abb. 35). Anschließend wird die Verteilerdose mit zwei Flachkopfschrauben und einem Akkuschauber oder Schraubendreher, etwas in Richtung des Drehzahlreglers versetzt, auf der Grundplatte festgeschraubt (Abb. 36).

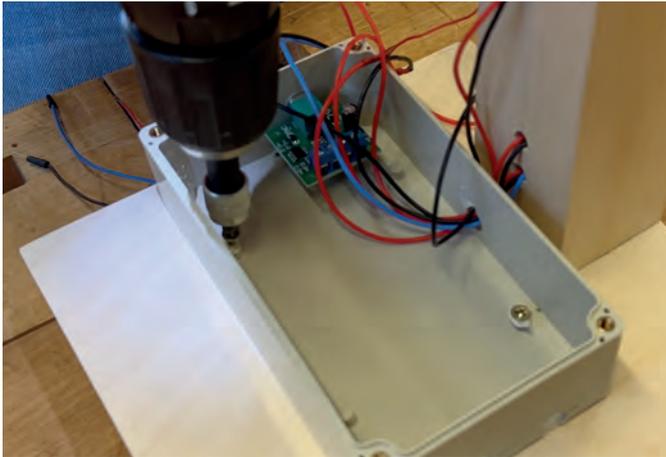


Abbildung 36: Fixieren der Verteilerdose

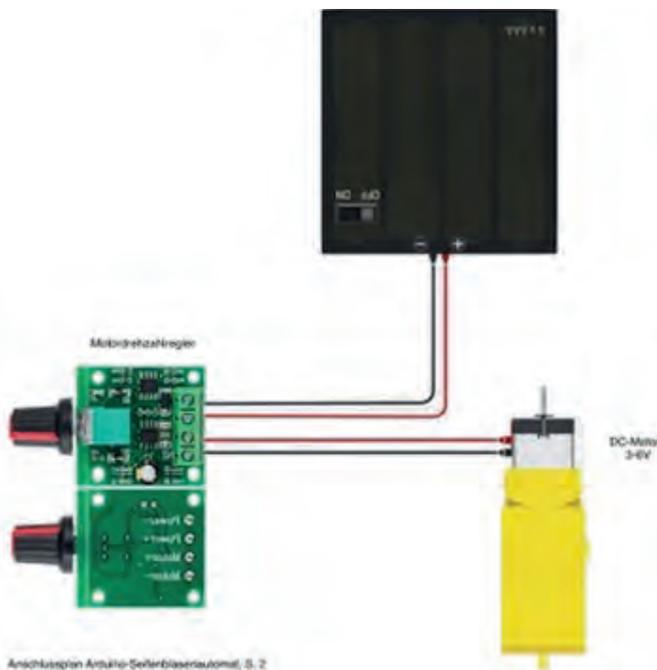


Abbildung 37: Anschlussplan Drehzahlregler

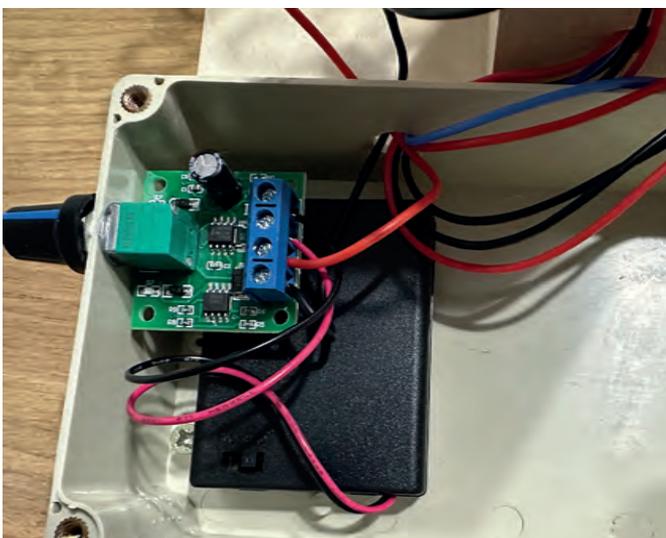


Abbildung 38: Installierter Drehzahlregler

Verteilerdose/Gehäuse: Anschließen des DC-Drehzahlreglers

In diesem Arbeitsschritt wird der DC-Drehzahlregler mit der Spannungsversorgung und dem $\varnothing 24$ mm DC-Motor, entsprechend des Anschlussplans aus Abb. 37, angeschlossen.

Zunächst verbindet man dazu die Schaltdrähte (rot und schwarz) des $\varnothing 24$ mm Motors mit dem Drehzahlregler. Falls notwendig, werden die Schaltdrähte gekürzt, um unnötigen Platzverbrauch in der Verteilerdose zu vermeiden. Dann werden die Enden der Schaltdrähte abisoliert und an die Anschlüsse des Drehzahlreglers „M +“ (rot) und „M -“ (schwarz) angeschlossen. Die Schaltdrähte werden so eingeführt, dass keine Kurzschlussstellen entstehen können, und zuletzt die Schrauben festgezogen. Durch leichtes Ziehen kann überprüft werden, ob die Kabel fest sitzen.

Nun verbindet man die SD des Batteriepacks (4xAAA) mit den beiden anderen Anschlüssen des Motor-Drehzahlreglers. Die SD werden entsprechend an die Anschlüsse „P +“ (rot) und „P -“ (schwarz) angeschlossen und verschraubt. Nach der Überprüfung der Kabel, wie in den vorherigen Schritten beschrieben, wird das Batteriepack unter den Drehzahlregler seitlich in die Verteilerdose gelegt (Abb. 38).

Blasringrad (optional): Anzeichnen und Bohren

Falls kein fertiges Seifenblasenringrad verwendet wird, kann dieses einfach selbst hergestellt werden. Benötigt werden ein Fugenkreuz, vier Blasringe, eine Schere/ein Seitenschneider, eine Unterlage, wasserfester Alleskleber, ein Maschinenschraubstock, Schraubzwingen, ein Vorstecher und ein Akkuschauber mit einem 3 mm HSS-Bohrer.

Zunächst werden dazu die Bohrpunkte am Fugenkreuz markiert (Abb. 39). Insgesamt sind fünf Bohrungen nötig: Neben einer Bohrung im Zentrum⁷, durch die später das Blasringrad auf die Getriebewelle aufgeschoben wird, sind vier Bohrungen an den Armen/Balken des Kreuzes an den Außenseiten notwendig, in die später die Blasringe eingeführt werden sollen. Entsprechend sind diese Bohrpunkte anzuzeichnen und mit einem Vorstecher vorzustechen (Abb. 40). Dann werden die Öffnungen für die Blasringe gebohrt. Hierfür wird das Fugenkreuz horizontal im Maschinenschraubstock eingespannt und vorsichtig gebohrt (Abb. 41). Die zentrierte Bohrung kann so vorgenommen werden, dass das Kreuz zwischen zwei Schraubzwingen auf einer Unterlage eingespannt wird oder das Kreuz beispielsweise mit einer Schraubzwinde an einem der Arme/Balken leicht fixiert wird (Abb. 42).

Blasringrad (optional): Verkleben der Blasringe mit dem Fugenkreuz

Zunächst wird eine Unterlage benötigt, auf die das Fugenkreuz gelegt wird. Dann müssen die vier Blasringe mit einem Seitenschneider gleichmäßig abgelängt werden (Abb. 43). Ist dies erfolgt, werden die vier Blasringe gleichmäßig durch

⁷ Die mittige Bohrung kann auch erst nach dem Verkleben der Blasringe (folgender Schritt) erfolgen, um zu vermeiden, dass der flüssige Kleber die Öffnung füllt und das Fugenkreuz an der Unterlage festklebt.



Abbildung 39: Bohrpunkte anzeichnen



Abbildung 41: Fugenkreuz bohren

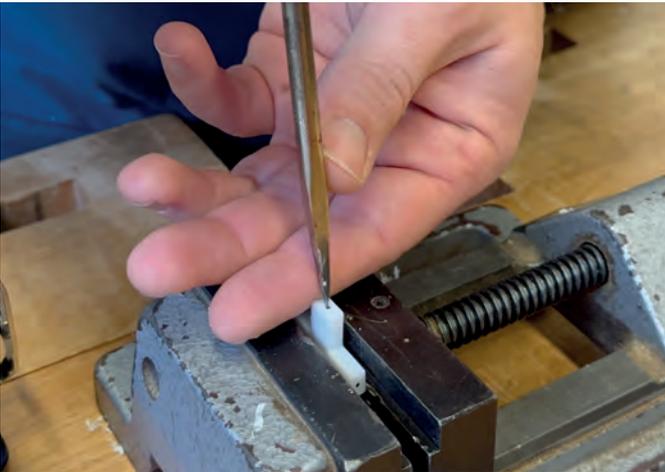


Abbildung 40: Bohrpunkte vorstechen



Abbildung 42: Fugenkreuz für mittige Bohrung einspannen



Abbildung 43: Ablängen der Blasinge

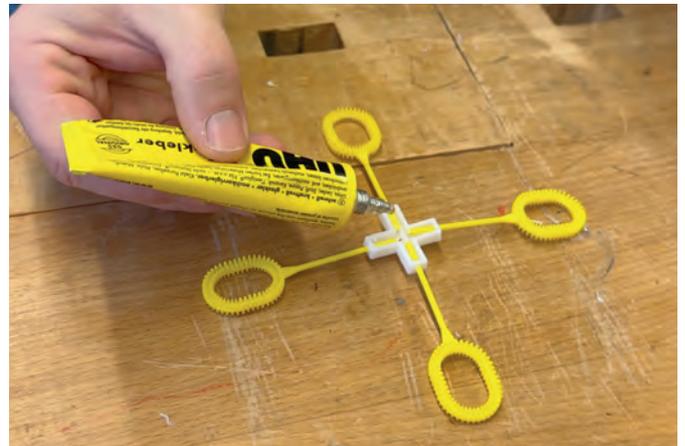


Abbildung 45: Fugenkreuz bohren

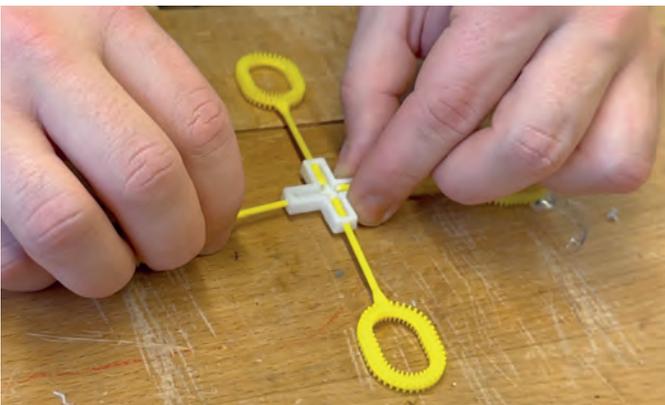


Abbildung 44: Bohrpunkte vorstechen

die Bohrungen an den Balken des Kreuzes eingeführt, sodass sie gleich lang sind und die Mitte des Kreuzes freibleibt (Öffnung für die Getriebewelle) (Abb. 44).

Anschließend wird etwas wasserfester Alleskleber vorsichtig auf die Stäbe der Blasinge aufgetragen, damit diese nach dem Trocknen fest mit dem Fugenkreuz verbunden sind (Abb. 45). Während des Trocknens sollte das Fugenkreuz nicht bewegt werden.

Arduino-Board: Verbinden des Uno R3 Boards mit dem Motor Shield

In den folgenden Schritten werden die Installation und der Anschluss der Komponenten am Arduino-Board vorgenommen.

Im ersten Schritt wird das Arduino Uno Board mit dem Motor-Shield verbunden, indem dies auf den Arduino Uno aufgesteckt wird (Abb. 46), wobei die Stiftleisten des Shields in die entsprechenden Buchsen auf dem Arduino passen. Hier muss beachtet werden, dass die Stiftleisten leicht biegsam sind, weshalb das Aufstecken vorsichtig erfolgen muss. Die Beschreibungen der einzelnen Buchsen stimmen auf beiden Platinen überein.

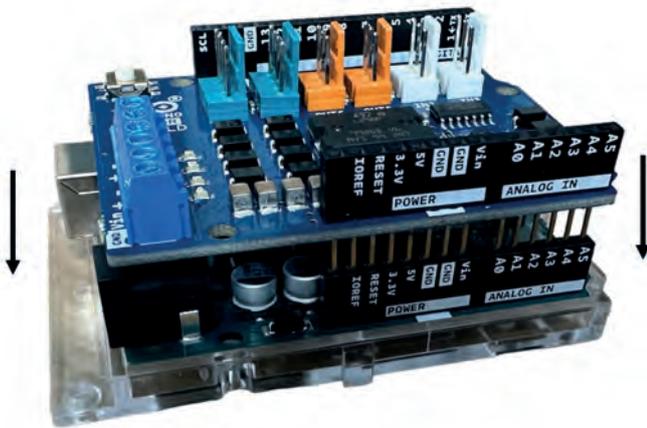


Abbildung 46: Arduino und Motor-Shield verbinden

Arduino-Board: Anschließen des Getriebemotors

Nun wird der Getriebemotor an das Motor Shield des Arduinos angeschlossen, indem die beiden Schaltdrähte des Getriebemotors mit den entsprechenden Buchsen verbunden werden (Abb. 47). Es ist darauf zu achten, dass der Steckplatz „A“ auf dem Motor Shield verwendet wird, da dieser im Code weiter konfiguriert ist. Die passenden Bezeichnungen der Pole („+“ und „-“) des jeweiligen Steckplatzes können auf dem Motor Shield eingesehen werden. Weiterhin ist sicherzustellen, dass die Drahtenden korrekt verdrillt und sauber in die Steckplätze eingeführt sowie fest verschraubt werden (Abb. 48). Abschließend ist zu überprüfen, dass keine einzelnen Drahtadern abstehen und die Isolierung der Schaltdrähte bündig mit den Buchsen abschließt, um spätere Kurzschlüsse zu vermeiden. Durch leichtes Ziehen an den Kabeln lässt sich zudem feststellen, ob sie fest sitzen.

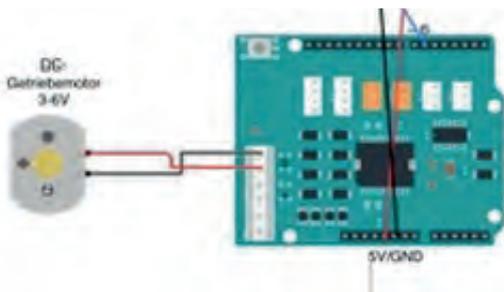


Abbildung 47: Anschlussplan des Getriebemotors

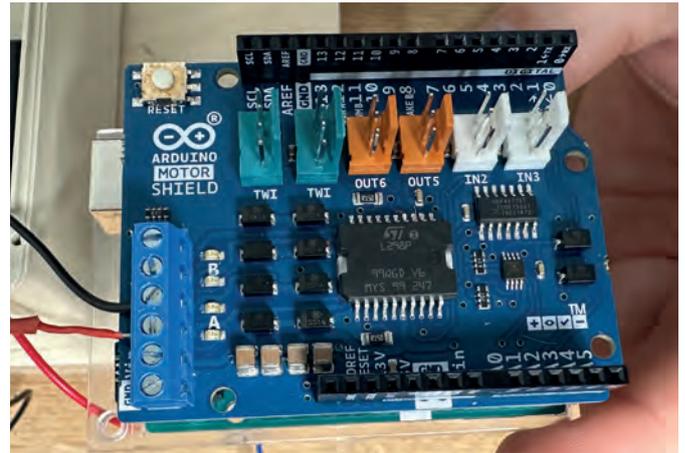


Abbildung 48: Anschluss des Motors an das Motor-Shield

Arduino-Board: Anschließen des LED-Rings

Der RGB LED-Ring wird über die PIN-Steckplätze des Motor-Shields verbunden, indem die drei angelöteten Jumper-Kabel in folgende Steckplätze auf dem Motortreiber-Shield eingesteckt werden (Abb. 49 & 50):

- Der GND (Ground) Draht des LED-Rings wird mit einem GND-Pin auf dem Motor-Shield verbunden.
- Der VCC (Spannungsversorgung) Draht des LED-Rings wird an den 5V-Pin auf dem Motor Shield angeschlossen.
- Der IN (Signal-Eingang) Draht des LED-Rings wird an den digitalen Signal-Pin 6 auf dem Motor Shield angeschlossen, der die Signale vom Arduino Uno zum LED-Ring überträgt. Pin 6 deshalb, da dieser im folgenden Code für den LED-Ring konfiguriert ist.

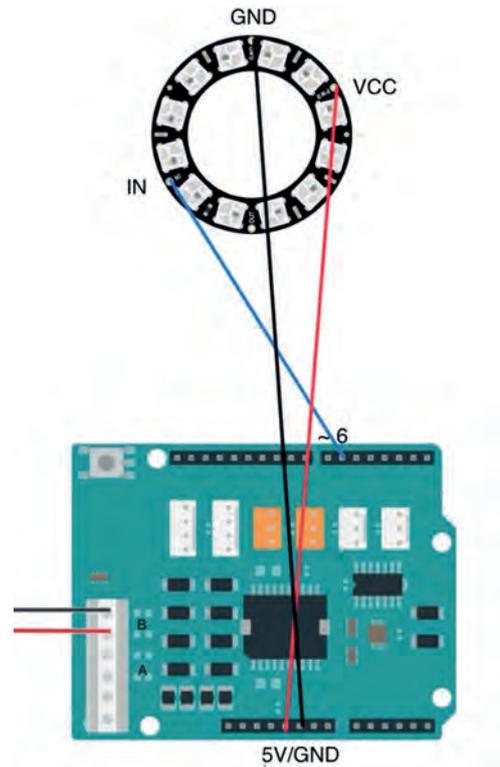


Abbildung 49: Anschlussplan des LED-Rings

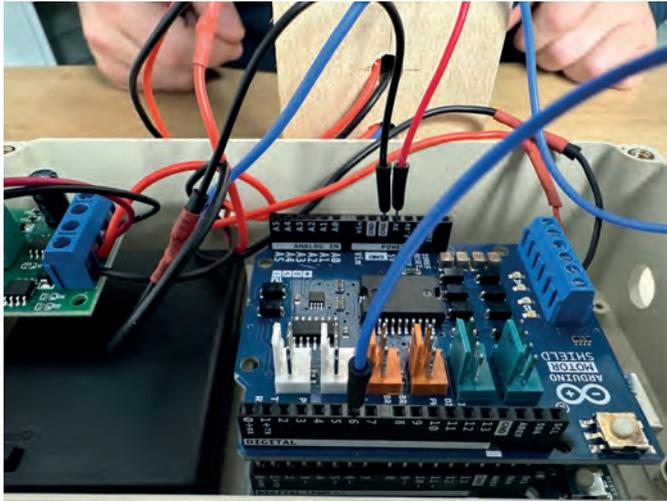


Abbildung 50: Anschluss des Motors an das Motor-Shield

Arduino-Board: Abschließende Installation

Abschließend können die SD in der Verteilerdose separiert werden, indem sie mit Kabelbindern sauber zusammengebunden werden. So liegen sie ordentlich in der Verteilerdose, ohne weitere Störungen oder potenzielle Kurzschlussstellen zu verursachen. Zuletzt ist es wichtig, optisch zu prüfen, dass alle Verbindungen sauber absolviert sind und keine offenen Schaltdrahtenden oder andere metallische Oberflächen in Kontakt mit offenen Stellen der Arduino-Platine kommen können. Dies könnte zu Kurzschlüssen oder Beschädigungen am Arduino-Board oder am Motortreiber-Shield führen. Ebenso können in diesem Schritt bereits die Batterien (4xAAA) in den entsprechenden Batteriehalter eingelegt werden.

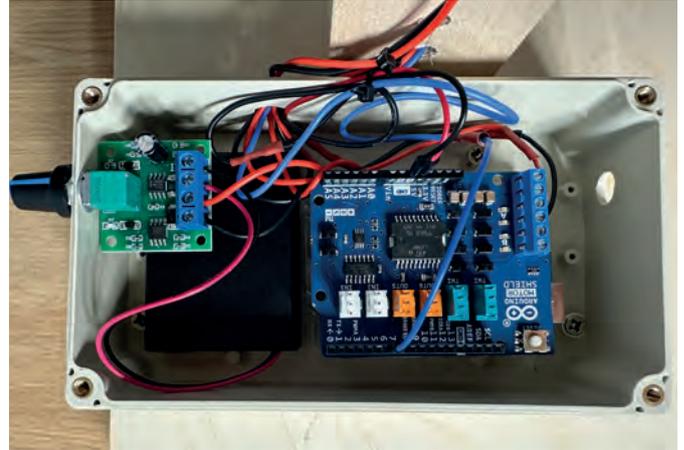


Abbildung 51: Finale Installation der Verteilerdose

Programmierung: Installation „Arduino IDE“

Nun werden im Weiteren die nötige Software installiert und der Code auf den Arduino hochgeladen.

Die Arduino IDE ist die Schlüsselsoftware für das Programmieren des Arduino-Boards. Sie ist auf der offiziellen Website von Arduino verfügbar (<https://www.arduino.cc>) und muss für das entsprechende Betriebssystem heruntergeladen und installiert werden (Abb. 52). Die Installation hängt vom Betriebssystem ab und wird deshalb nicht im Weiteren beschrieben. Infos und Schritt-für-Schritt-Anleitungen dazu werden ebenfalls auf der Website von Arduino zur Verfügung gestellt.

Es gibt darüber hinaus weitere Programme zur Programmierung des Arduino-Mikrocontrollers. Ein Beispiel hierfür ist ArduBlock, welches eine vereinfachte Darstellung und Programmierung ermöglicht.

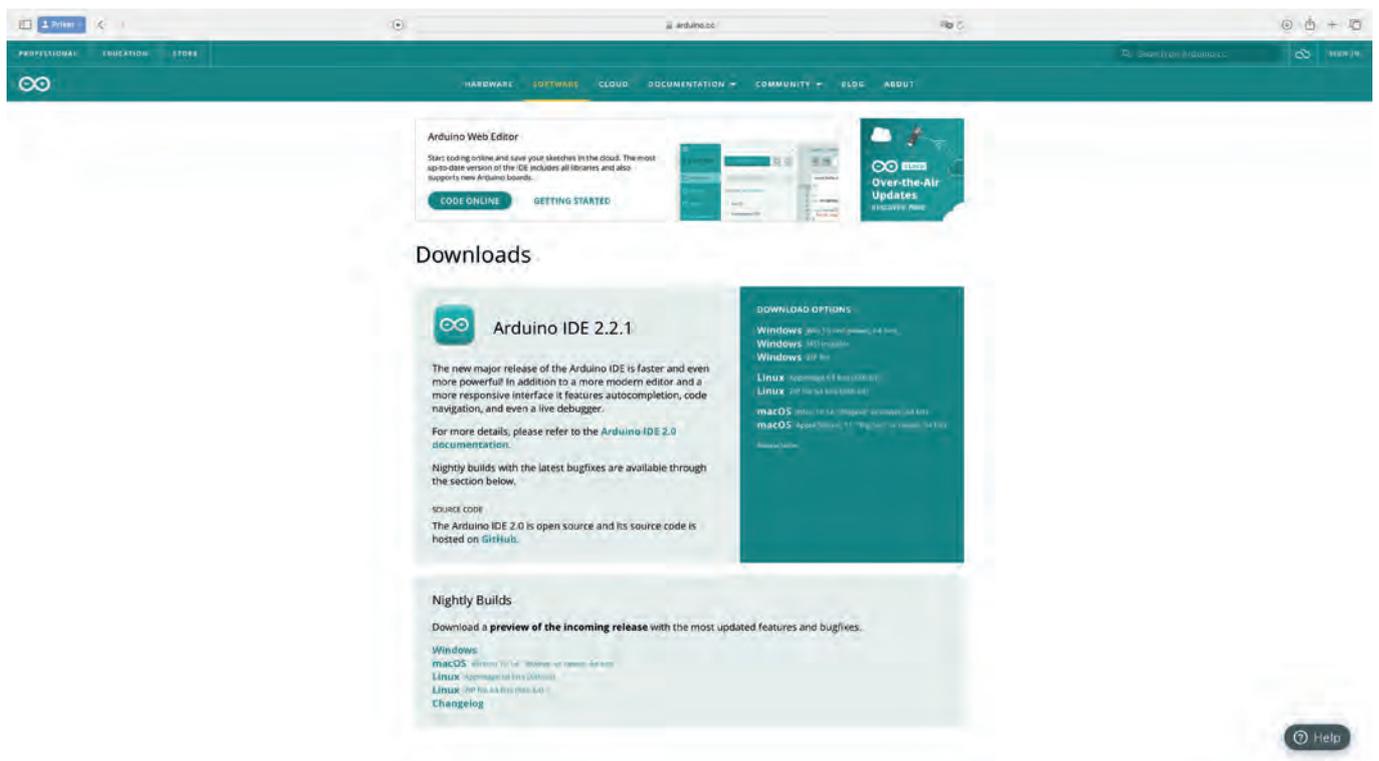


Abbildung 52: Installation der Arduino IDE

Programmierung: Installieren benötigter Bibliotheken

Nach der Installation der Software öffnet sich die Benutzeroberfläche der Arduino IDE, die neben einem Texteditor für das Schreiben von Codes, ein Menü für verschiedene Funktionen und eine Symbolleiste mit wichtigen Werkzeugen enthält (Abb. 53).

Für den bereitgestellten Code sind spezielle Bibliotheken erforderlich, wie `Servo.h` und `Adafruit_NeoPixel.h`. Diese lassen sich über den Bibliotheksmanager in der Arduino IDE installieren, indem man nach dem Bibliotheksnamen sucht und die entsprechende Bibliothek über die entsprechenden Schaltflächen installiert (Abb. 54).

Die Bibliothek „Servo.h“ muss ggf. nicht installiert werden, da diese zu den Standard-Bibliotheken zur Motorsteuerung zählt. Jedoch sollte der Installationsstatus dennoch überprüft werden.

Programmierung: Verbinden des Arduino-Boards mit einem PC

Bevor der Code kopiert wird, muss das Arduino-Board mit einem PC verbunden werden (Abb. 55). Zur Verbindung des Arduino-Boards mit der IDE-Software verbindet man das Board per USB mit dem Computer, wählt den entsprechenden Port (z.B. „COM3“ oder „...usbmodem“) und das Board-Modell (hier „Arduino Uno“) unter dem Menüpunkt „Tools“ aus (Abb. 56).

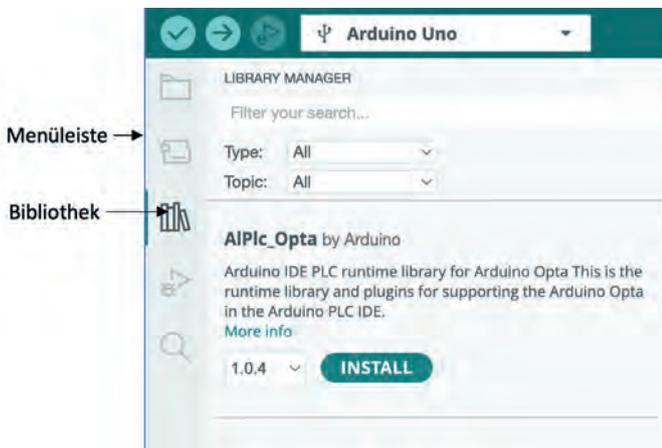


Abbildung 53: Benutzeroberfläche Arduino IDE

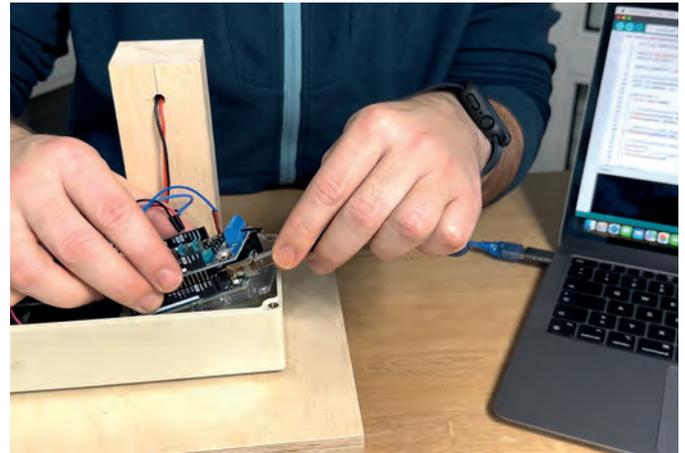


Abbildung 55: Verbinden des Arduino mit einem Computer

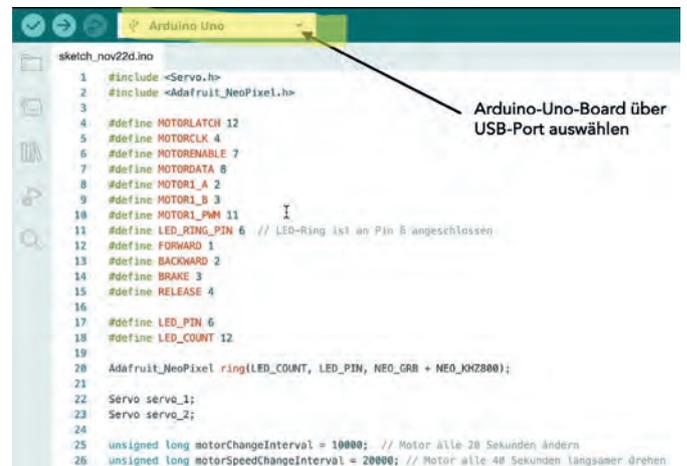


Abbildung 56: Auswahl des Ports

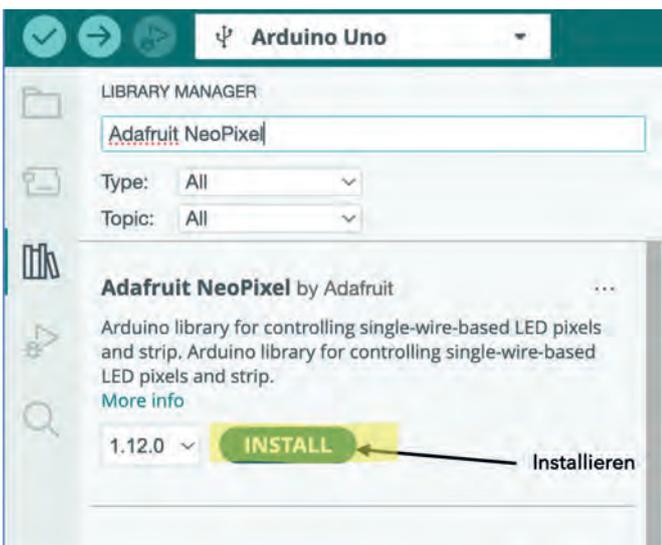


Abbildung 54: Bibliothek installieren

Programmierung: Einfügen und Upload des Programmcodes

Nun kann der bereitgestellte Code (Anhang) in den Editor der Arduino IDE kopiert werden (Abb. 57). Es ist sicherzustellen, dass das korrekte Board ausgewählt wurde (Abb. 58), entsprechend dem Port „Arduino Uno - usbmodem ...“. Unter Windows kann die Bezeichnung variieren, möglicherweise als „Port3“ oder „COM3“. Bei der Verwendung von Windows sollten auch weitere Einstellungen beachtet werden, die auf der Arduino-Website ausführlich beschrieben sind. Nach dem Einfügen des Codes und der erneuten Überprüfung des korrekten Ports kann der Upload mit der entsprechenden Schaltfläche auf das Arduino-Board gestartet werden (Abb.

59). Es ist darauf zu achten, dass beim Kopieren des Codes keine zusätzlichen Leerzeichen oder Ähnliches übertragen werden, da dies zu Fehlern beim Hochladen führen kann. Sobald der Code hochgeladen wurde, sollten alle mit dem Arduino-Board verbundenen Komponenten starten (LED-Ring und Getriebemotor).



Abbildung 57: Code einfügen

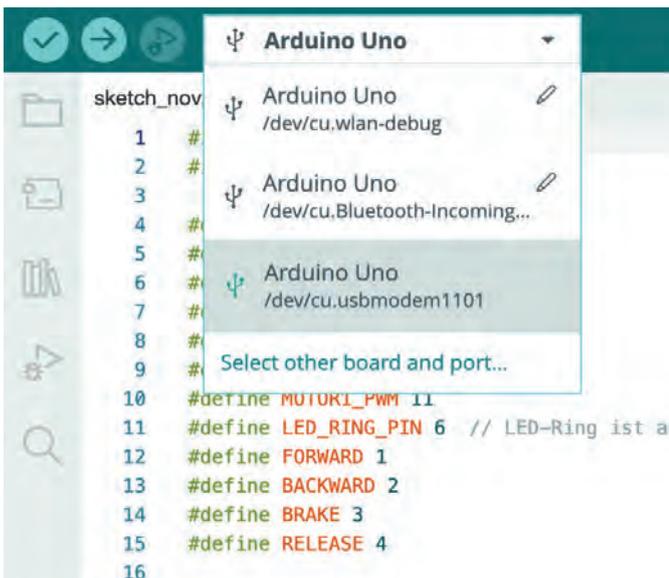


Abbildung 58: Board-Auswahl

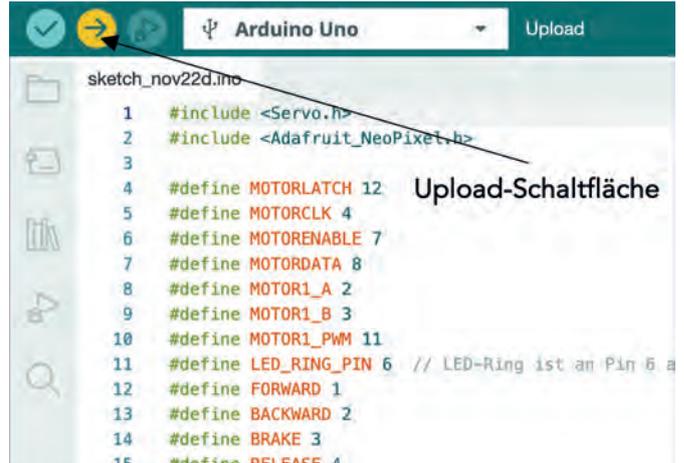


Abbildung 59: Code-Upload

Programmierung: Anpassung des Codes und Abschluss

Der Code steuert einen Getriebemotor und einen LED-Ring. Er umfasst das Einbinden von Bibliotheken, die Definition von Konstanten und Variablen, die setup()- und loop()-Funktionen sowie spezifische Funktionen für die Motor- und LED-Steuerung. Einige Werte können leicht verändert und somit an unterschiedliche Ansprüche angepasst werden.

Anpassung der Motorgeschwindigkeit: Die Motorgeschwindigkeit wird über die Variable motorSpeed gesteuert, die Werte von 0 (stillstehend) bis 255 (maximale Geschwindigkeit) annehmen kann. Durch Ändern des Wertes in der Variable motorSpeed lässt sich die Geschwindigkeit anpassen (Abb. 60). Hier wurde der Wert „75“ gewählt, um zu verhindern, dass die Seifenlauge im Behälter durch eine zu hohe Motordrehzahl zu stark aufschäumt und überläuft. Weiterhin bedarf es einiger Experimente, da das Verhalten auch von der Spannungsquelle abhängt; möglicherweise dreht der Motor bei diesem Wert nicht, wenn das Board per USB angeschlossen ist, wohl aber bei Anschluss einer 9-Volt-Blockbatterie.

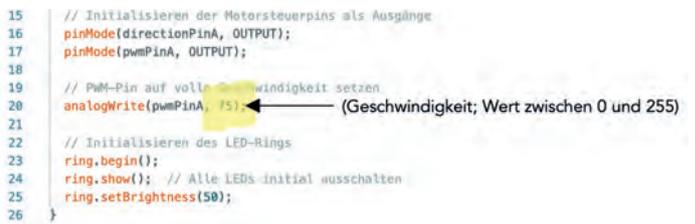


Abbildung 60: Board-Auswahl

Funktionen für Motor- und LED-Steuerung: Der Code enthält spezielle Funktionen, wie motor(), changeMotorDirection(), colorWipe() und theaterChase(). Die Anpassung der Parameter in diesen Funktionen ermöglicht die Kontrolle des Verhaltens von Motor und LEDs.

Zum Abschluss kann das USB-Kabel vom Arduino-Board getrennt werden. Dann wird der Netzstecker von außen in das Gehäuse eingeführt, mit dem Arduino verbunden und zum Schluss der Deckel mit der Verteilerdose fest verschraubt (Abb. 61).

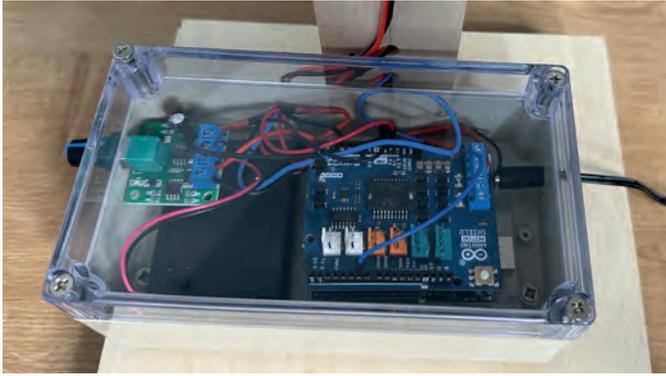


Abbildung 61: Installation Deckel

Anbringen des Blasringrades

In diesem Schritt wird das Blasringrad auf die Welle des Getriebemotors montiert. Die Art der Montage hängt von den Eigenschaften des jeweiligen Seifenblasenringrades ab. Bei einem selbst hergestellten Seifenblasenringrad gemäß Arbeitsschritt „Blasringrad (optional)“ erfolgt die Montage, wie folgt:

Zuerst wird eine M3-Gewindemutter auf die Gewindewelle aufgeschraubt. Anschließend wird das Seifenblasenringrad auf die Welle geschoben, sodass sich die Mutter in der hinteren Einkerbung des Kreuzes fixiert. Dann wird das Seifenblasenringrad mit einer Unterlagscheibe und einer weiteren Mutter in der gewünschten Position fixiert, indem die beiden Muttern leicht gegeneinander festgezogen werden (Abb. 62). Hierbei ist darauf zu achten, dass das Seifenblasenringrad möglichst gerade auf der Welle ausgerichtet ist, bevor es final fixiert wird.

Andere Seifenblasenringräder könnten gegebenenfalls andere Befestigungsarten oder -systeme erfordern.

Spannungsversorgung

Da der Arduino während der Verbindung mit dem PC einige Funktionen bereits ausgeführt hat, ist es wichtig, eine konstante und vom PC unabhängige Stromversorgung zu ge-

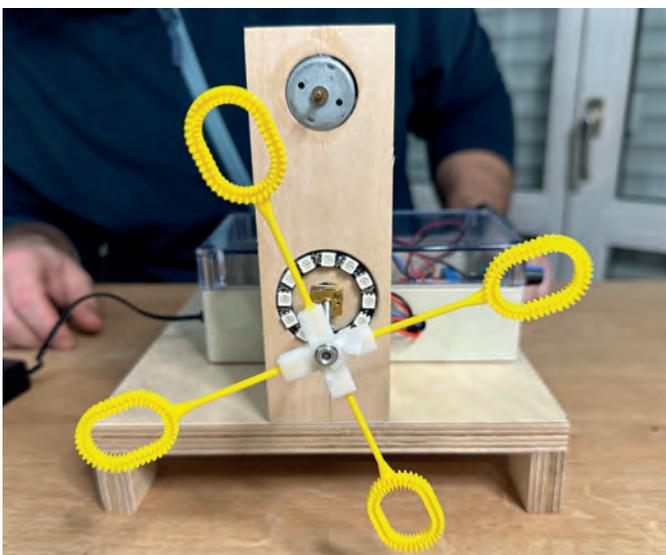


Abbildung 62: Installation des Seifenblasrings

währleisten. Hierfür wurde eine 9V-Blockbatteriebox über den 3,5 mm Hohlstecker an den Arduino angeschlossen. Das Ein- und Ausschalten des Arduino-Boards erfolgt, nach einlegen einer passenden Batterie, über den Schalter an der 9V-Blockbatteriebox. Die 9V-Blockbatteriebox kann, sofern gewünscht an einer Position nach Wahl bspw. auch mit einem wasserfesten Alleskleber an der Außenseite der Verteilerdose fixiert werden.

Der Motor, der das Rotorblatt und somit den Windstrom antreibt, lässt sich über den Motor-Drehzahlregler sowohl steuern als auch ein- und ausschalten. Beide Motoren sind mit unterschiedlichen Spannungsquellen versorgt. Diese Entscheidung basiert auf der Erkenntnis, dass eine Kombination beider Motoren bei der eigenen Erprobung nicht ausreichend Spannung für den \varnothing 24 mm Motor über das Arduino-Board bzw. das Motortreiber-Shield bereitstellen konnte. Zudem erwiesen sich die Geschwindigkeitseinstellungen über den Programmcode als zu ungenau. Ein externer Motor-Drehzahlregler bietet hier eine zuverlässigere und genauere Steuerung, was besonders wichtig ist, um auf verschiedene Seifenblasenlaugen und andere Einflussfaktoren reagieren zu können. So kann die Intensität der Windstärke genau eingestellt werden.

Darüber hinaus können die Geschwindigkeiten des Getriebemotors (vgl. Arbeitsschritt: „Programmierung: Anpassung des Codes und Abschluss“), neben der Problematik des Überschaümens, ebenfalls an die Beschaffenheit der Seifenblasenlauge angepasst werden. Bei geringerer Laugendichte ist eine niedrigere Geschwindigkeit notwendig, um eine ausreichende Oberflächenspannung in den Seifenblasenringen zu erzeugen.

Weitere Tipps:

Beim Betrieb des Werkstücks sollte besondere Aufmerksamkeit darauf gelegt werden, es ausschließlich im Freien zu verwenden. Der Grund dafür liegt in der Rutschgefahr, die von den produzierten Seifenblasen ausgehen kann. Zudem ist für die Herstellung der Seifenlauge ein Gefäß mit genügend Volumen erforderlich. Dieses soll nicht nur sicherstellen, dass die Blasringe vollständig in die Lauge eintauchen können, sondern auch, dass sie bei der Drehung die Seitenwände des Gefäßes nicht streifen. Ein Behältnis mit den Abmessungen 200x100x70 mm (Länge x Breite x Tiefe) hat sich während der eigenen Erprobung als optimal erwiesen, um allen Anforderungen gerecht zu werden. Rezepte für die Seifenlauge lassen sich leicht im Internet finden, und fertige Mischungen sind ebenfalls, beispielsweise im Spielwarenhandel, erhältlich.

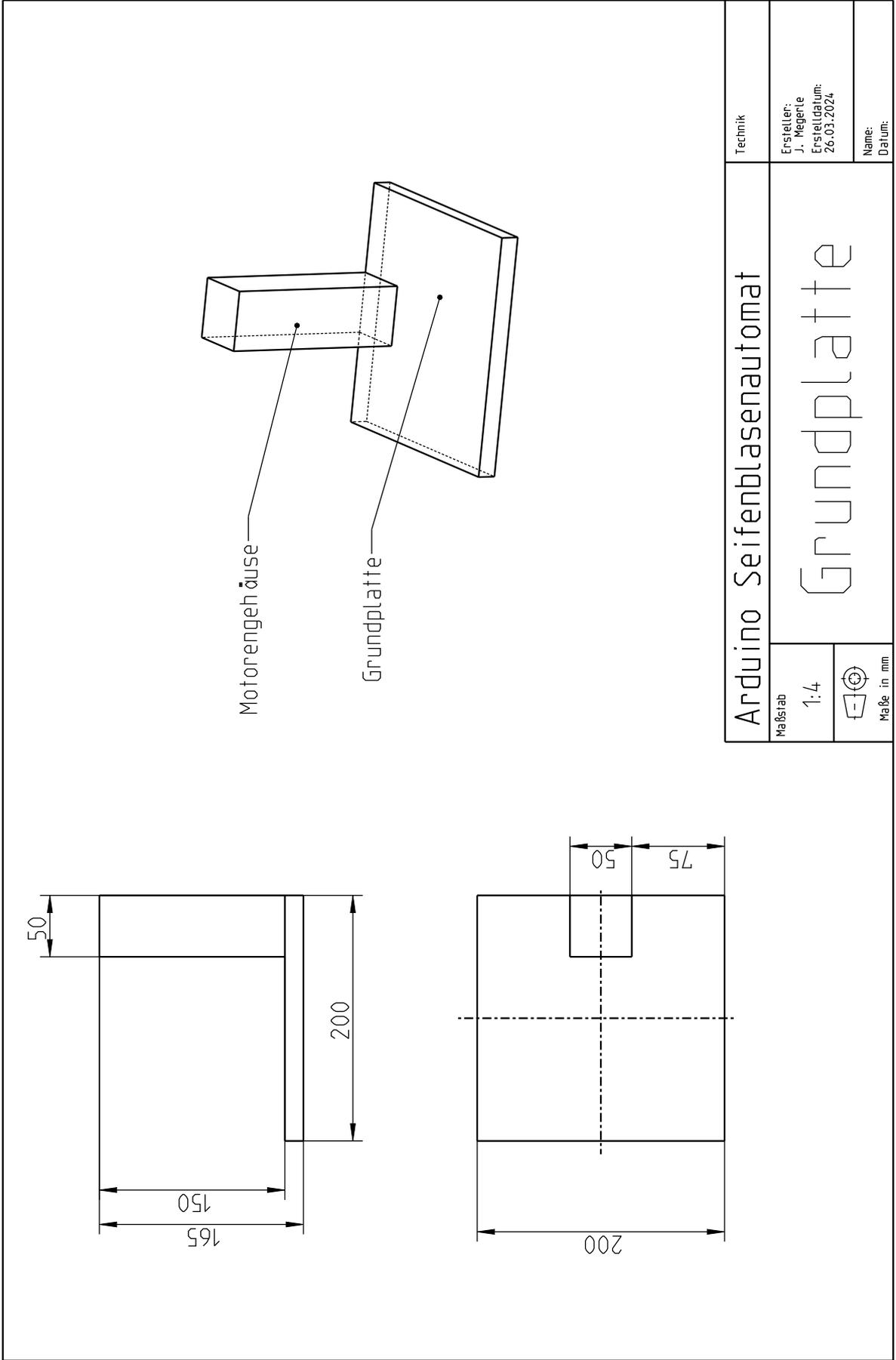
Hintergrund der Namensgebung: Der Name des Werkstücks „ArduWood-IllumiSphere - Seifenblasenautomat“ setzt sich zusammen aus „Arduino“ und „Wood“, was die Verwendung von Holz andeutet, „Illumi“ als Hinweis auf die Beleuchtungsfunktion und „Sphere“, was die runde Form der Seifenblasen widerspiegelt.

Abschließend möchte ich mich ganz herzlich bei meiner Kommilitonin Jessica Megerle für die Überarbeitung der technischen Zeichnungen bedanken.

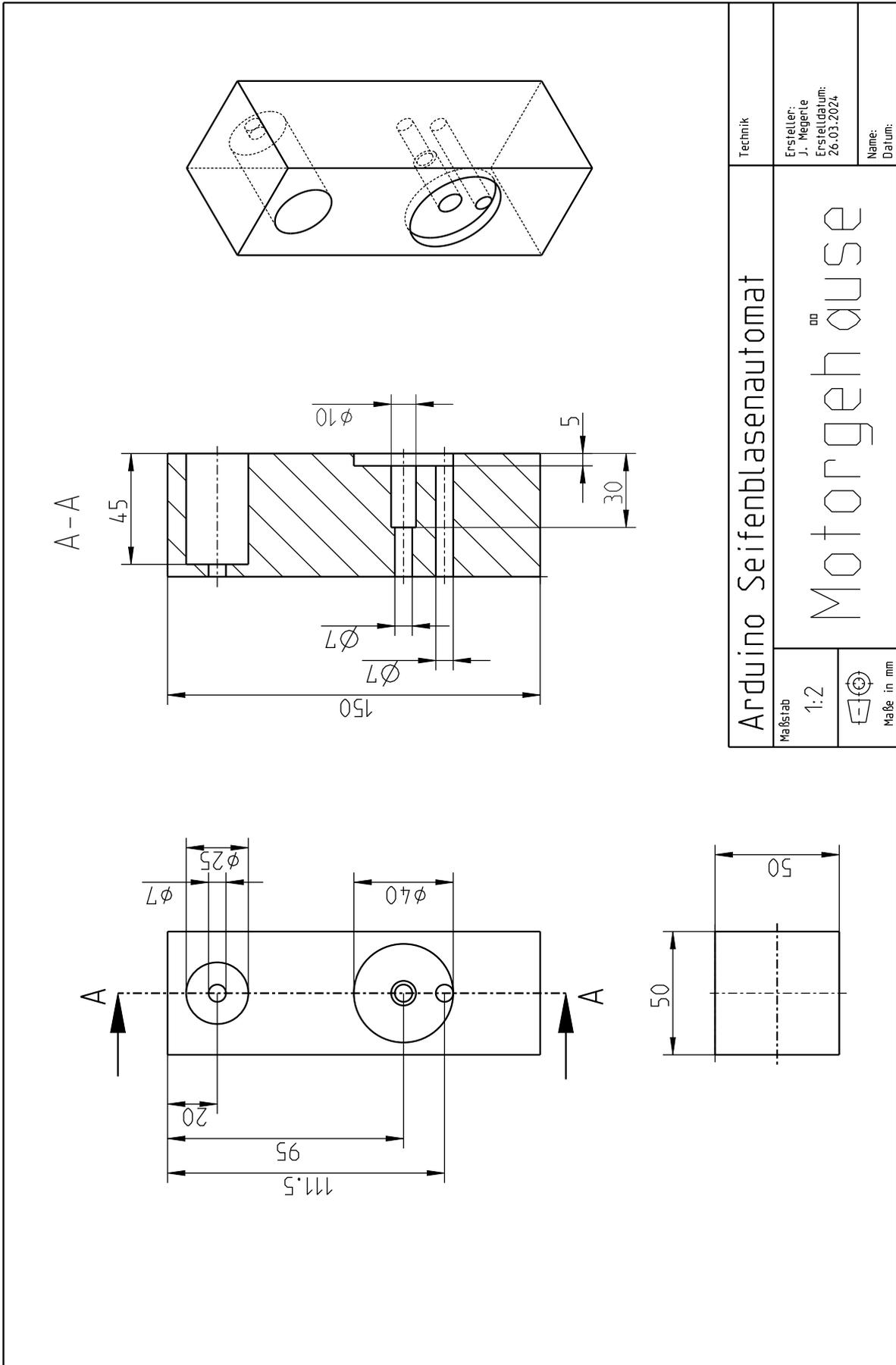
GEFÄHRDUNGEN UND MASSNAHMEN

Gefährdungsfaktor	Konkrete Gefährdung im Betrieb	Risiko	Maßnahme (technisch, organisatorisch, persönlich)
Mechanische Gefährdung	Ungeschützt bewegte Maschinenteile (Bohrmaschine)	gering	Lange Haare zum Zopf zusammenbinden, Schmuck und Uhren vor der Bearbeitung ablegen, enganliegende Kleidung tragen, ggf. Schutzbrille tragen
Spezielle physikalische Einwirkung	Lärm	gering	ggf. Gehörschutz tragen
Gefahrstoffe	Einatmen von Gefahrstoffen (Holzstaub)	gering	ggf. angefallenen Holzstaub mit Staubsauger (Staubschutzklasse M) absaugen, keinesfalls kehren
Gefahrstoffe	Hautkontakt mit Gefahrstoffen (Leim)	gering	Wasserlöslichen, ungiftigen Leim verwenden, nicht auf die Haut bekommen, direkt und gründlich Hände waschen
Thermische Gefährdungen	heiße Medien/Oberflächen (Föhn)	mittel	Zange oder hitzebeständige Handschuhe verwenden
Elektrische Gefährdungen	Elektrischer Schlag bei nicht gut isolierten, offenen Anschlussstellen	gering	Isolierung mit Schrumpfschlauch oder Isolierband ordentlich ausführen
Gefahrstoffe	Hautkontakt mit Gefahrstoffen (Klebstoff). Allergische Reaktionen/Haut- und Augenreizungen, Verschlucken von Gefahrstoffen (wasserf. Alleskleber)	gering	ungiftigen, lösungsmittelfreien Klebstoff verwenden, nicht auf die Haut und in die Augen bekommen, direkt und gründlich Hände waschen
Thermische Gefährdungen (Löten)	heiße Medien/Oberflächen (LötKolben)	mittel	den LötKolben nur am Griff anfassen, Zange oder hitzebeständige Handschuhe verwenden (Heißluftfön)
Gefahrstoffe	Einatmen von Gefahrstoffen (Lötdampf)	gering	Vermeidung des Einatmens von Lötdämpfen, Abstand halten und bei offenem Fenster bzw. guter Lüftung arbeiten, nur bleifreie Lote dürfen verwendet werden
Elektrische Gefährdungen	Elektrischer Schlag bei nicht gut isolierten, offenen Anschlussstellen	gering	auf die Lage des LötKabels achten, sauberes Löten, ggf. Isolierung an offenen Kontaktstellen vornehmen
Brände und Explosionen	Brandgefahr durch LötKolben	gering	LötKolben immer am Lötständer ablegen, Verwendung einer feuerfesten Lötunterlage
Mechanische Gefährdung	Bewegte Arbeitsmittel (Bohren)	mittel	Werkstück fest in einen Schraubstock einspannen und gegen Abrutschen des Bohrers Körner verwenden
Elektrische Gefährdung	Elektrischer Schlag bei nicht gut isolierten, offenen Anschlussstellen, Kurzschlussgefahr bei nicht korrekter Einhaltung der Anschlüsse	gering	Isolierung überprüfen und korrekte Anschluss-Reihenfolge einhalten
Mechanische Gefährdung	Ungeschützt bewegte Maschinenteile (Akkuschrauber)	gering	Lange Haare zum Zopf zusammenbinden, Schmuck und Uhren vor der Bearbeitung ablegen, enganliegende Kleidung tragen, ggf. Schutzbrille tragen, Werkstück gut fixieren, Holzunterlage verwenden, Bohrloch vorstechen

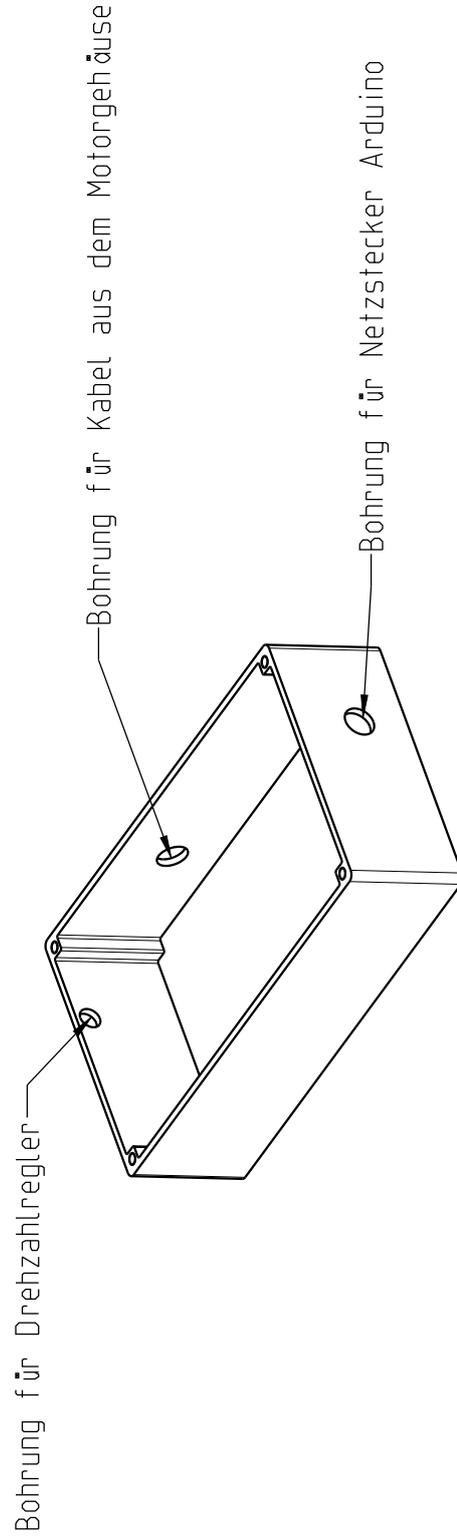
Tabelle 3: Gefährdungen und Maßnahmen



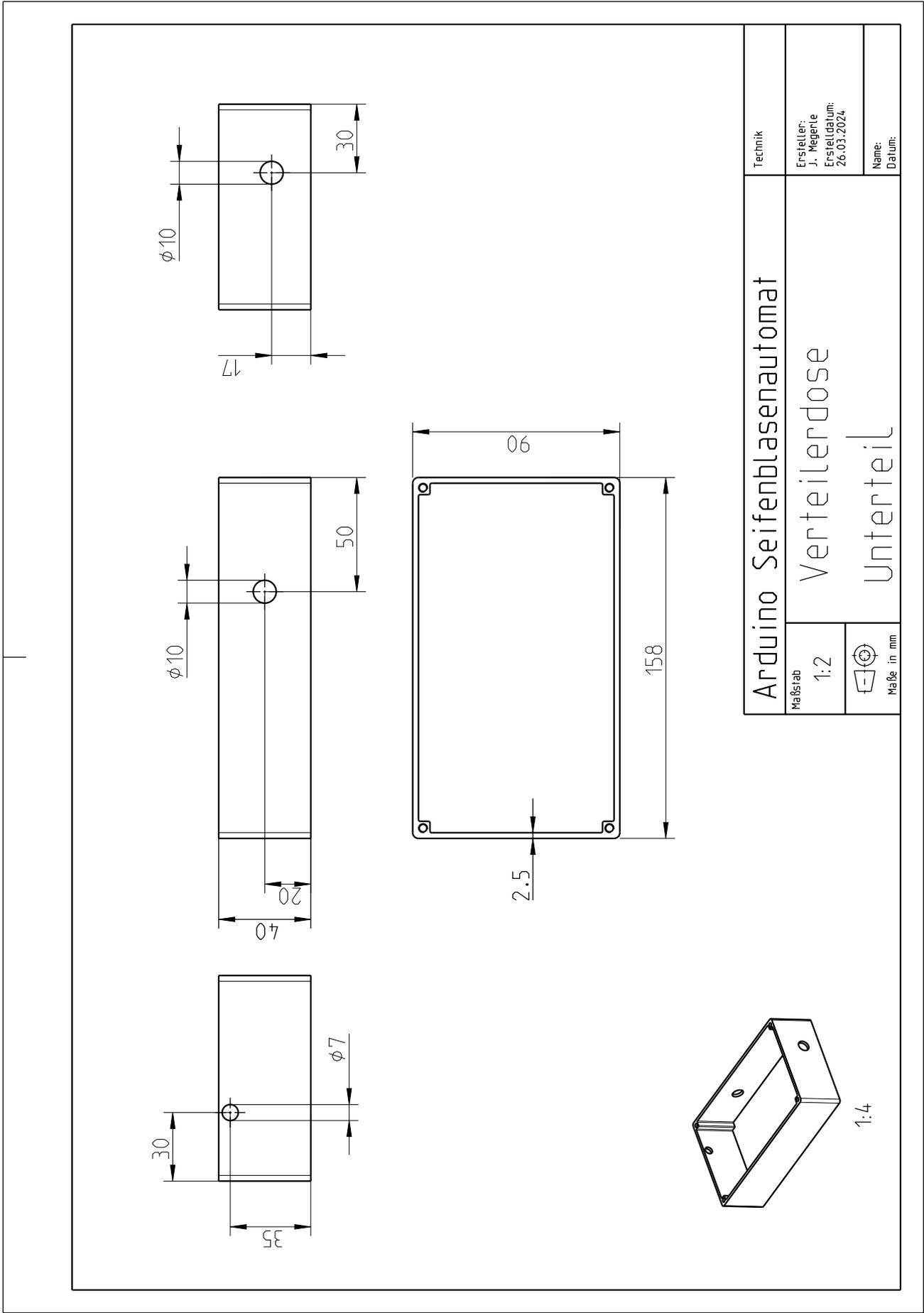
Technik	
Ersteller: J. Megerte	
Erstelldatum: 26.03.2024	
Name: Datum:	
Arduino Seifenblasenautomat	
Maßstab 1:4	Grundplatte
Maße in mm	



Arduino Seifenblasenautomat		Technik
Maßstab 1:2	Motorgehäuse	Ersteller: J. Wegente
Maße in mm		Erstelldatum: 26.03.2024
		Name: Datum:



<p>Arduino Seifenblasenautomat</p>		<p>Technik</p>
<p>Meßstab 1:2</p>	<p>Name: Datum:</p>	
<p>Maße in mm</p>	<p>Ersteller: J. Megertle Erstelldatum: 26.03.2024</p>	
<p>Verteilerdose</p>		



CODE ZUM KOPIEREN UND EINFÜGEN

Steuerungsmerkmale:

- Richtungswechsel des Seifenblasenringrades alle 20 Sekunden
- Verschiedene Farbanimationen des LED-Ringes

Der Code zum Kopieren beginnt mit „#include...“

```
#include <Adafruit_NeoPixel.h>
#define LED_COUNT 12
#define LED_PIN 6
Adafruit_NeoPixel ring(LED_COUNT, LED_PIN, NEO_GRB +
NEO_KHZ800);
// Definieren der Motorsteuerepins für Kanal A
const int directionPinA = 12; // Drehrichtung - DIR A
const int pwmPinA = 3; // Geschwindigkeit - PWM A
void setup() {
  Serial.begin(9600);
  // Initialisieren der Motorsteuerepins als Ausgänge
  pinMode(directionPinA, OUTPUT);
  pinMode(pwmPinA, OUTPUT);
  // PWM-Pin auf volle Geschwindigkeit setzen
  analogWrite(pwmPinA, 75);

  // Initialisieren des LED-Rings
  ring.begin();
  ring.show(); // Alle LEDs initial ausschalten
  ring.setBrightness(50);
}
void loop() {
  // Motorsteuerung
  static unsigned long lastMotorChange = 0;
  if (millis() - lastMotorChange >= 20000) { // Alle 20 Sekunden
    digitalWrite(directionPinA, !digitalRead(directionPinA)); //
Drehrichtung umkehren
    lastMotorChange = millis();
  }
  // LED-Ring-Steuerung
  colorWipe(ring.Color(255, 0, 0), 50); // Rot
  colorWipe(ring.Color(0, 255, 0), 50); // Grün
  colorWipe(ring.Color(0, 0, 255), 50); // Blau
  theaterChase(ring.Color(127, 127, 127), 50); // Weiß, halbe Hel-
ligkeit
  theaterChase(ring.Color(127, 0, 0), 50); // Rot, halbe Helligkeit
  theaterChase(ring.Color(0, 0, 127), 50); // Blau, halbe Hellig-
keit
  rainbow(10); // Fließender Regenbogenzyklus entlang des ge-
samten Rings
  theaterChaseRainbow(50); // Rainbow-verstärkte TheaterCha-
se-Variante
}
void colorWipe(uint32_t color, int wait) {
  for (int i = 0; i < ring.numPixels(); i++) {
    ring.setPixelColor(i, color);
    ring.show();
    delay(wait);
  }
}
void theaterChase(uint32_t color, int wait) {
```

```
for (int a = 0; a < 10; a++) {
  for (int b = 0; b < 3; b++) {
    ring.clear();
    for (int c = b; c < ring.numPixels(); c += 3) {
      ring.setPixelColor(c, color);
    }
    ring.show();
    delay(wait);
  }
}
void rainbow(int wait) {
  for (long firstPixelHue = 0; firstPixelHue < 5 * 65536; firstPixel-
Hue += 256) {
    for (int i = 0; i < ring.numPixels(); i++) {
      int pixelHue = firstPixelHue + (i * 65536 / ring.numPixels());
      ring.setPixelColor(i, ring.gamma32(ring.ColorHSV(pixel-
Hue)));
    }
    ring.show();
    delay(wait);
  }
}
void theaterChaseRainbow(int wait) {
  for (int j = 0; j < 256; j++) {
    for (int q = 0; q < 3; q++) {
      for (int i = 0; i < ring.numPixels(); i = i + 3) {
        int pos = i + q;
        if (pos < ring.numPixels()) {
          ring.setPixelColor(pos, Wheel((i + j) % 255));
        }
      }
    }
    ring.show();
    delay(wait);
    for (int i = 0; i < ring.numPixels(); i = i + 3) {
      int pos = i + q;
      if (pos < ring.numPixels()) {
        ring.setPixelColor(pos, 0);
      }
    }
  }
}
uint32_t Wheel(byte WheelPos) {
  if (WheelPos < 85) {
    return ring.Color(WheelPos * 3, 255 - WheelPos * 3, 0);
  } else if (WheelPos < 170) {
    WheelPos -= 85;
    return ring.Color(255 - WheelPos * 3, 0, WheelPos * 3);
  } else {
    WheelPos -= 170;
    return ring.Color(0, WheelPos * 3, 255 - WheelPos * 3);
  }
}
```

tedu

1|2024