



Ing. Manuel Natali, BSc

Entwicklung eines halbautomatischen Reinigungssystems für Optiken biomedizinischer Baugruppen

MASTERARBEIT

zur Erlangung des akademischen Grades

Diplom-Ingenieur

Masterstudium: Biomedical Engineering

eingereicht an der

Technischen Universität Graz

Betreuer

Ao.Univ.-Prof. Dipl.-Ing. Dr.techn. Hermann Scharfetter

Institute of Biomedical Imaging

Leitung: Univ.-Prof. Dipl.-Phys. Dr.rer.nat. Martin Uecker

Klagenfurt, Januar 2023

EIDESSTATTLICHE ERKLÄRUNG

Ich erkläre an Eides statt, dass ich die vorliegende Arbeit selbstständig verfasst, andere als die angegebenen Quellen/Hilfsmittel nicht benutzt und die den benutzten Quellen wörtlich und inhaltlich entnommenen Stellen als solche kenntlich gemacht habe. Das in TUGRAZonline hochgeladene Textdokument ist mit der vorliegenden Masterarbeit identisch.

Graz, am

.....

(Unterschrift)

Zusammenfassung

Optiken stellen in vielen Fällen einen zentralen Bestandteil biomedizinischer Baugruppen dar und haben auch nicht selten Einfluss auf Messergebnisse oder die Funktion des Endgerätes. In diesem Zusammenhang spielt auch die Sauberkeit dieser eine wesentliche Rolle. Ziel dieser Arbeit war es, ein Reinigungssystem für das Unternehmen *WILD GmbH* zu entwickeln und zu untersuchen, ob sich die bis dato manuell durchgeführte Reinigung runder Optiken damit teilautomatisieren lässt. Der entwickelte Roboter erlaubt es, unterschiedlich gekrümmte optische Oberflächen unter rotatorischer Bewegung gleichmäßig von innen nach außen zu reinigen. Ein integrierter Kraftsensor sorgt zudem dafür, dass der Druck des Reinigungsstäbchens (Polyester-Pad) auf die Oberfläche der Optik stets konstant gehalten wird. Die Ergebnisse der im Anschluss durchgeführten Versuchsreihen zeigen, dass der Prototyp aus funktionaler Sicht in der Lage ist, die manuelle Reinigung zu automatisieren und in Kombination mit geeigneten Lösungsmitteln Verschmutzungen wie Staub oder fettige Fingerabdrücke nahezu rückstandsfrei entfernt.

Schlüsselwörter: *Optik, Reinigung, Automatisierung, Python, CNC*

Abstract

In many cases, optics are a central component of biomedical assemblies and often have an influence on measurement results or the function of the end device. In this context, the cleanliness of these also plays an essential role. The aim of this work was to develop a cleaning system for the company *WILD GmbH* and to investigate whether the previously manual cleaning of round optics can be partially automated with it. The robot developed allows optical surfaces with different curvatures to be cleaned uniformly from the inside to the outside under rotational movement. An integrated force sensor also ensures that the pressure of the cleaning rod (polyester pad) on the surface of the optics is always kept constant. The results of the series of tests subsequently performed show that, from a functional point of view, the prototype is capable of automating manual cleaning and in combination with suitable solvents, removes soiling such as dust or greasy fingerprints almost without leaving any residue.

keywords: *optics, cleaning, automation, Python, CNC*

Inhaltsverzeichnis

1	Einleitung	1
1.1	Motivation	1
1.2	Der richtige Umgang mit optischen Komponenten	2
1.2.1	Arten von Verunreinigungen	2
1.2.2	Kontaminationsquellen	3
1.2.3	Die richtige Arbeitsumgebung	4
1.2.4	Persönliche Arbeitsausrüstung und Hilfsmittel	5
1.2.5	Lagerung und Transport	6
1.3	Manuelle Reinigungsmethoden	7
1.3.1	Reinigungsmittel	7
1.3.2	Druckluftreinigung	8
1.3.3	Mechanische Abschrubverfahren	8
1.3.4	Reinigung mit Flüssigpolymeren	10
1.4	Maschinelle Reinigungsmethoden	11
1.4.1	Ultraschallreinigung	11
1.4.2	Weitere maschinell unterstützte Reinigungsmethoden	12
1.5	Verifizierung der Sauberkeit	13
1.5.1	Quantitative Bestimmung von Verschmutzungen	13
1.5.2	Klassifizierung der Oberflächenreinheit	13
1.5.3	Visuelle Sichtprüfung vor dem Einbau	14
1.6	Einfluss von Verschmutzungen	17
1.6.1	Verdunkelung und Streuung	17
1.6.2	Absorptionsbanden	18
1.7	Ziel und Anforderungen	19
1.7.1	Anforderungen an die Mechanik	19
1.7.2	Funktionale Anforderungen	20
2	Methoden	21
2.1	Konzepterstellung	21
2.1.1	Herangehensweise	21
2.1.2	Mechanisches Basissystem	21
2.1.3	CAD Konstruktion	21
2.2	Werkstoffe und Fertigungstechniken	23
2.2.1	Mechanische Komponenten	23
2.2.2	Elektronische Komponenten	23
2.2.3	Normteile	24

2.3	Vorversuche	25
2.4	Systemüberblick	26
2.5	Hardware	29
2.5.1	Zentrale Steuerungseinheit	29
2.5.2	3-Achsen-Positionierungssystem	31
2.5.3	Drehtellereinheit	40
2.5.4	Reinigungskopf	44
2.5.5	Lösungsmittelfördereinheit	57
2.5.6	Bedien- und Anzeigeeinheit	62
2.5.7	Elektrische Versorgung und Leistungsverbrauch	63
2.6	Software	64
2.6.1	Vorbereitungen	64
2.6.2	Entwicklungswerkzeuge	66
2.6.3	Programmstruktur	67
2.6.4	Programmstart	71
2.6.5	Interrupts	71
2.6.6	Multithreading	71
2.6.7	Lock-Objekte	72
2.6.8	Signals	73
2.6.9	Übersicht der Programmfunktionen	73
2.6.10	Dokumentation	75
2.7	Versuchsreihen	76
2.7.1	Abtastsensoren	76
2.7.2	Wägezelle	77
2.7.3	Ermittlung der tatsächlichen Drehzahl	78
2.7.4	Testaufbau zur Evaluierung der Reinigung	80
2.7.5	Auswertung und Darstellung der Ergebnisse	82
3	Ergebnisse	83
3.1	Profilabtastung	83
3.1.1	Punktuelle Wiederholgenauigkeit	83
3.1.2	Wiederholgenauigkeit der Profilabtastung	85
3.2	Druckmessung	86
3.3	Drehteller	89
3.3.1	PWM-Signal	89
3.3.2	Linearität der Drehzahl	89
3.4	Reinigungsergebnisse	91
3.4.1	Partikuläre Kontaminationen	91

3.4.2	Fettige Schmutzfilme	93
3.4.3	Verunreinigung durch Schweißfinger	94
3.4.4	Speicheltröpfchen	95
3.4.5	Vergleich zwischen manueller und automatischer Reinigung . .	96
4	Diskussion	97
4.1	CNC-Positionierungssystem	97
4.2	Steuereinheit und Software	97
4.3	Lösungsmittelfördereinheit	98
4.4	Reinigungskopf	98
4.4.1	Messmethoden der Profilaufzeichnung	98
4.4.2	Präzision der Einzelpunktabtastung	99
4.4.3	Erfassung von Oberflächenprofilen	99
4.4.4	Präzision und Linearität der Druckmessung	99
4.4.5	Druckverläufe der Reinigungsmodi	100
4.5	Drehtellereinheit	101
4.5.1	Kraftübertragung und Platzbedarf	101
4.5.2	Drehzahl und Geräuschentwicklung	101
4.6	Reinigungserfolg	102
4.6.1	Partikuläre Verunreinigung	102
4.6.2	Schmutzfilme	102
4.7	Schlussfolgerung	103
5	Bedienungsanleitung	104
5.1	Grundlegendes	104
5.1.1	Aufbau der Benutzeroberfläche	104
5.1.2	In- und Außerbetriebnahme	105
5.2	Funktionen	106
5.2.1	Anlegen eines neuen Optikprofils	106
5.2.2	Erstellung eines Optikprofils	108
5.2.3	Durchführung einer Reinigung	109
5.2.4	Manuelle Kontrolle des Roboters	110
5.2.5	Anzeige & Änderung der Basiseinstellungen	112
5.3	Instandhaltung	113
5.3.1	Tausch des Reinigungsstäbchens	113
5.3.2	Befüllen der Lösungsmittelspritze	114
5.4	Fehlerbehebung	116
5.4.1	CNC-Controller reagiert nicht mehr	116

5.4.2	Tare-Wert außerhalb der Toleranz	116
5.4.3	Vorgehen bei Schrittverlusten	116
5.4.4	Alarm-Status beheben	117
A	Stücklisten	123
B	Verwendete Geräte und Software	126
C	Korrespondenzen	127

Symbole und Abkürzungen

PCR	Polymerase Chain Reaction
PTFE	Polytetrafluorethylen, gelegentlich auch Polytetrafluorethen
PET-G	Glycol modifiziertes Polyethylenterephthalat
PP	Polypropylen
HCl	Chlorwasserstoffsäure
HF	Fluorwasserstoffsäure
MEK	Methylethylketon
MEC	Methylethylcellulose
TCE	Trichlorethen
CaF₂	Calciumfluorid
MgF₂	Magnesiumfluorid
SF11	Schwerflintglas
SCP	Surface Cleanliness by Particle Concentration
NVR	Non Volatile Residue
CNC	Computerized Numerical Control
DIY	Do It Yourself
CAD	Computer Aided Design
STL	Standard Triangulation Language
FDM	Fused Deposition Modeling
PLA	Polylactid Acid
ABS	Acrylnitril-Butadien-Styrol-Copolymer
IC	Integrated Circuit
GPIO	General Purpose Input/Output

RPI4	Raspberry Pi der 4. Generation
UART	Universal Asynchronous Receiver/Transmitter
PWM	Pulsweitenmodulation
SPI	Serial Peripheral Interface
I²C	Inter-Integrated Circuit
GPU	Graphics Processing Unit
USB	Universal Serial Bus
HDMI	High Definition Multimedia Interface
TCP/IP	Transmission Control Protocol/Internet Protocol
LAN	Local Area Network
GTK+	GIMP Toolkit +
GUI	Graphical User Interface
NEMA	National Electrical Manufacturers Association
DC	Direct Current
CR	Carriage Return
LF	Line Feet
EEPROM	Electrically Erasable Programmable Read-Only Memory
POM	Polyoxymethylen
DMS	Dehnmessstreifen
SPS	Samples Per Second
V_{CC}	Voltage At The Common Collector
NO	Normally Open
DIP	Dual In-Line Package
PEP 8	Python Enhancement Proposal 8

IP	Internet Protocol
PIP	Packet Installer for Python
APT	Advanced Packaging Tool
PyPI	Python Package Index
IDE	Integrated Development Environment
SSH	Secure Socket Shell
XML	Extensible Markup Language
X11	X-Window System
DI-Wasser	Deionisiertes Wasser
HTML	Hypertext Markup Language
PDF	Portable Document Format

1 Einleitung

1.1 Motivation

Optische Komponenten sind mittlerweile zu unverzichtbaren Bauteilen vieler medizintechnischer Geräte und Instrumente geworden. Die daraus resultierenden optischen Systeme helfen bei der Prävention und Diagnose von Krankheiten, analysieren Proben auf molekularer Ebene und werden auch oft für therapeutische Zwecke eingesetzt. Die moderne Medizin wird somit in vielerlei Hinsicht unterstützt und könnte ohne Optiken nicht das leisten, wozu sie heute fähig ist. [1]

Als Auftragsfertiger medizinischer Geräte und Instrumente, verbaut das Unternehmen WILD (Völkermarkt, Österreich) eine Vielzahl von Optiken in unterschiedlichste medizinische Systeme. Eines seiner Spezialgebiete liegt in der *medizinischen Lasertechnik*. Die besonderen Eigenschaften eines Lasers machen die Lasertechnik zu einer unverzichtbaren Komponente in Fachbereichen wie der Dermatologie oder der Augenheilkunde [2]. Damit die Langzeitstabilität dieser Laserkomponenten gewährleistet werden kann, müssen sämtliche involvierte Optiken frei von Partikeln und filmischen Verschmutzungen sein [3].

Ein weiterer Fokus des Unternehmens liegt im Bereich der InVitro Diagnostik und der Analytik. WILD fertigt optomechatronische Baugruppen welche unter Anderem in Messkammern von PCR-Analysegeräten zum Einsatz kommen [4]. Auch in diesem Fall spielt die Sauberkeit eine wesentliche Rolle, damit sichergestellt ist, dass die Endgeräte mit wiederholter Genauigkeit und zuverlässig arbeiten.

Zu den typischsten Verschmutzungsarten zählen kleine Partikel, welche sich in der Umgebungsluft befinden und früher oder später unweigerlich den Weg auf die optische Oberfläche finden. Eine hohe Gefahr geht außerdem vom Menschen aus, welcher mit seinem Dasein und seinen Aktivitäten die größte Kontaminationsquelle darstellt [5]. Der falsche Umgang mit optischen Bauteilen führt nicht selten zu Fingerabdrücken, die unter Umständen wieder mit viel Aufwand entfernt werden müssen. Eine Kontamination kann aber während des gesamten Verarbeitungsprozesses einer optischen Komponente auftreten. Das beginnt bei einer falschen Lagerung und endet bei ungeeigneten Arbeitsumgebungen. Für die Herstellung eines optisch einwandfreien Gerätes sind viele Punkte zu beachten auf welche in Kapitel 1.2 näher eingegangen wird.

Im Unterschied zu herstellungsbedingten Fehlern, wie beispielsweise Lufteinschlüssen im Glaskörper oder mangelhafter Oberflächenbeschichtungen, lassen sich die Verunreinigungen in den meisten Fällen wieder vollständig entfernen. Dazu haben sich im Laufe der Zeit unterschiedliche manuelle Reinigungstechniken etabliert [6].

Die vorliegende Arbeit beschäftigt sich mit der Möglichkeit, die bisher manuell durchgeführte Reinigung, zu automatisieren. Der Schwerpunkt wird dabei auf die Automatisierung der sogenannten *Pinselftechnik* gelegt. Dazu wurde ein CNC-basierter Reinigungsroboter konzipiert, entwickelt, aufgebaut und anschließend verschiedenen Versuchsreihen unterzogen.

1.2 Der richtige Umgang mit optischen Komponenten

Bevor man sich überlegt, wie man optische Oberflächen bestmöglich reinigt, sollte untersucht werden, welche Umstände überhaupt zu einer Verschmutzung führen. Aus den gewonnenen Erkenntnissen lassen sich nützliche Maßnahmen zur Schmutzprävention ableiten, damit eine Reinigung erst gar nicht notwendig wird. Außerdem ist es wichtig, eine Rekontamination bereits sauberer Optiken zu verhindern. Mit jedem Reinigungsdurchgang, steigt die Gefahr, die meist sehr empfindlichen Oberflächen versehentlich zu beschädigen [6].

1.2.1 Arten von Verunreinigungen

Verunreinigungen optischer Oberflächen lassen sich grob in zwei Kategorien unterteilen. Einerseits treten sie in Form von diskreten Partikeln auf und andererseits als kontinuierlicher Schmutzfilm, welcher die Oberfläche überzieht. Es ist ebenso möglich, dass eine Mischform beider Typen vorliegt. Während Partikel dazu führen, dass das einfallende Licht gestreut wird, verursachen Schmutzfilme ungewollte Absorptionsbanden. [7] Diese Effekte wirken sich negativ auf die optische Performance der Komponenten aus, je nach Anwendung kommt es beispielsweise zu Einbußen der Bildqualität oder auch zu Transmissionsverlusten [5].

Wenn man von „Partikeln“ spricht meint man im Allgemeinen Staub und dieser wiederum, ist ein Sammelbegriff für unterschiedliche Arten von festen Teilchen in Gasen, die entweder organischer oder auch anorganischer Natur sein können [8]. Staub lässt sich außerdem anhand der Größe kategorisieren. Teilchen mit einem Durchmesser $> 10\mu m$ zählen zur Klasse der Grobstäube, alles darunter wird als Feinstaub bezeichnet [9]. Wenn in den folgenden Kapiteln von Staub gesprochen wird, bezieht sich das immer auf „Grobstaub“.

Ein praktisches Beispiel für anorganischen Staub wären Flusen, die sich von Textilien lösen oder Abriebe unterschiedlicher Materialien wie Gläser oder Metalle. Unter organischem Staub hingegen versteht man z.B. Hautschuppen, Haare oder auch Pollen die besonders in der Blütezeit vermehrt in der Luft zu finden sind. [10]

Verschmutzungen ohne definierte Abmessungen wie Fingerabdrücke, Verdampfungsrückstände diverser Flüssigkeiten oder Ausgasungen anderer Komponenten legen sich als Schicht auf der Oberfläche ab und fallen damit in die Kategorie der molekulare Kontamination (Schmutzfilme) [5].

1.2.2 Kontaminationsquellen

Fertigungsbedingte Verschmutzung

Fertigungsschritte wie Bohren, Drehen, Fräsen oder Schleifen von optischen Gläsern erzeugen Partikel, welche speziell in Kombination mit Schneidölen zu hartnäckigen Verschmutzungen führen können. Deshalb sollte schon zuvor darauf geachtet werden, dass man Öle verwendet, welche sich nach der Bearbeitung leicht entfernen lassen. [5]

Verschmutzungen während der Montage

Eine weitere, nicht zu unterschätzende Kontaminationsquelle entsteht bei der Montage von Optiken. Beispielsweise ist es eine gängige Methode, runde Optiken mittels *Vorschraubringen* in ihrer Fassung zu fixieren. Ist das Außengewinde des Vorschraubringes bzw. das Innengewinde der Optikfassung nicht einwandfrei sauber, kann es passieren, dass beim Fassen Abriebe entstehen, welche sich dann in Form von kleinen Metallspänen oder Eloxal-Rückständen auf der Optik ablegen und diese wieder verunreinigen.

Ein weiteres Verschmutzungsrisiko entsteht bei der Fixierung von Optiken mittels diverser Klebetechniken. Dabei kommen Harze, Zemente und Spezialkleber zum Einsatz, die bei unsauberem Arbeiten optisch relevante Flächen verschmutzen. In diesem Fall muss schnell reagiert werden, da sich diese im ausgehärteten Zustand nur sehr schwer wieder entfernen lassen und die Gefahr besteht, dass die Optik dadurch beschädigt wird [11].

Kontamination durch den Mensch

Die größte Kontaminationsgefahr für Optiken stellt der Mensch dar. Abgesehen davon, dass Menschen im Laufe des Tages selbst sehr viele organische Partikel freisetzen, schnappt man diese auch kontinuierlich auf, indem man sich bewegt. [5] Während dem Umgang mit Optiken können sich diese wieder lösen und optische Oberflächen verschmutzen.

Der Mensch verunreinigt Optiken aber nicht ausschließlich durch Partikel. Durch Husten oder Niesen, gelangen kleine Aerosolpartikel in die Umgebungsluft. Diese setzen sich auf der Optikoberfläche ab und hinterlassen dort kleine Schmutzfilme. Im Gegensatz zu Staub, lässt sich diese Art von Kontaminierung nicht einfach mit

Druckluft entfernen. In diesem Fall lässt sich die Verschmutzung nur mit Hilfe eines Lösungsmittels und einem geeigneten Reinigungsmedium von der Oberfläche „reiben“. Hierfür gibt es unterschiedliche Methoden, welche in Kapitel 1.3 auf Seite 7 näher beschrieben werden.

Kontinuierliche Schulungen im richtigen Umgang mit Optiken und eine geeignete Arbeitsausrüstung helfen dabei, menschenverursachte Verunreinigungen gering zu halten [5].

1.2.3 Die richtige Arbeitsumgebung

Sind Optiken neben mechanischen oder elektrischen Bauteilen erst mal zu einer funktionsfähigen Baugruppe zusammengesetzt, ist eine Reinigung in vielen Fällen nicht mehr möglich. Das bedeutet, die letzte Reinigung muss vor der Montage bzw. Integration stattfinden. Ab diesem Zeitpunkt ist es wichtig, die Baugruppe sauber zu halten und dafür ist eine definierte Arbeitsumgebung erforderlich. [5]

Um Rekontaminationen bereits sauberer Optiken zu vermeiden, ist es wichtig, dass bei der Gestaltung des Arbeitsplatzes bestimmte Punkte beachtet werden.

Tabelle 1 gibt einen kurzen Überblick darüber, auf was bei Optik-Arbeitsplätzen besonders zu achten ist.

Tabelle 1: Tipps für die Gestaltung eines Optik-Arbeitsplatzes

Ausstattung	Der Arbeitsplatz ist räumlich abgegrenzt (z.B. in Form einer Kabine) und die Innenwände sind mit einer mattschwarzen Abdeckung ausgekleidet. Die Arbeitsmatte sollte ebenfalls dunkel sein, das erlaubt eine besser Kontrolle der Optik und verhindert Streulicht. [12]
Beleuchtung	Die Innenraumbeleuchtung ist flexibel und die Beleuchtungsstärke liegt bei ca. 500 bis 600 lx. Als Lichtfarbe eignet sich ein neutrales Weiß. [12]
Werkzeuge	Werkzeuge sind außerhalb des Arbeitsbereiches zu lagern, damit die Arbeitsfläche möglichst frei bleibt. [12]
Verpackungsmaterial	Kartonagen, Papier oder ähnliches faserndes Verpackungsmaterial sind vom Arbeitsplatz fern zu halten, da sie den Arbeitsplatz kontaminieren. Spanerzeugende Arbeitsschritte dürfen nicht auf Optik-Arbeitsplätzen durchgeführt werden. [12]

Den für diesen Zweck optimalen Arbeitsplatz stellen sogenannte *Flow Boxen* dar. Unter einer *Flow Box* versteht man eine spezielle Kabine, welche die gewöhnliche, durch Partikel kontaminierte Umgebungsluft ansaugt und durch unterschiedlich feine Schwebstofffilter presst. Die dadurch erzeugte partikelarme Luft, fließt stromlinienförmig von Oberseite der Kabine nach unten und schafft so einen lokalen Reinraum, indem sie die Partikel in der Kabine nach außen befördert. [13]

Geht es um besonders heikle Optiken, so sind *Flow Boxen* alleine oftmals nicht mehr ausreichend. In diesem Fall kann auf einen vollwertigen Reinraum ausgewichen werden. Neben der bereits erläuterten laminaren Luftströmung, herrscht in Reinräumen zusätzlich ein kontinuierlicher Überdruck, der dafür sorgt, dass ein Partikeleintritt von außen nicht möglich ist [13]. Somit lassen sich noch höhere Reinheitsklassen erreichen. Ein weiterer Vorteil ist, dass sich Temperatur und Luftfeuchte auf einem konstanten Niveau halten lassen. Ein definiertes Raumklima ist wichtig, da optische Gläser oft sehr empfindlich auf Temperaturschwankungen reagieren [6]. Für eine bestmögliche Arbeitsumgebung lassen sich *Flow Box* und Reinraum auch kombinieren.

Das Arbeiten in Reinräumen erfordert, dass bestimmte Verhaltensregeln eingehalten werden. Dazu gehört z.B. das Tragen spezieller Arbeitskleidung welche bis auf wenige Ausnahmen alle Körperstellen bedeckt. Außerdem muss darauf geachtet werden, dass keine kontaminierten Werkzeuge oder Bauteile eingeschleust werden. Eine Verletzung der Regeln kann schnell dazu führen, dass die Partikelanzahl (lokal) auf ein Vielfaches des erlaubten Wertes ansteigt und der eigentliche Vorteil eines Reinraumes nicht mehr gegeben ist.

1.2.4 Persönliche Arbeitsausrüstung und Hilfsmittel

Neben den Anforderungen an die Arbeitsumgebung ist es ebenso wichtig, dass sich die persönliche Arbeitsausrüstung für den Umgang mit Optiken eignet.

Es ist besonders auf die Sauberkeit der Hände zu achten, vor allem, dass diese fett- und silikonfrei sind [12], denn Fingerabdrücke lassen sich oftmals nur schwer wieder entfernen. Generell gilt, dass eine Berührung optisch wirksamer Oberflächen mit der bloßen Haut in jedem Fall zu vermeiden ist. Das lässt sich leicht verhindern, indem Einwegfingerlinge bzw. Einweghandschuhe aus Baumwolle, Nitril oder puderfreiem Latex getragen werden. [6, 12]

Hantiert man mit besonders kleine Optiken, kann die Verwendung von Optikpinzetten von Vorteil sein. Diese sind typischerweise aus Kunststoff oder Bambusholz gefertigt, um zu verhindern, dass es zu unabsichtlichen Beschädigungen der Optik

kommt. Als Alternative dazu, bietet der Markt auch druckluftunterstützte Aufnahmewerkzeuge, sogenannte *Vakuumpinzetten*, an. [14]

Um Verunreinigungen besser zu erkennen, kann es sinnvoll sein, Hilfsmittel wie Vergrößerungsgläser oder Mikroskope einzusetzen. Einerseits helfen diese dabei Staub oder Fingerabdrücke besser zu erkennen, andererseits lässt sich mit ihnen auch der Reinigungserfolg kontrollieren. Außerdem helfen sie Verschmutzungen von Beschädigungen zu unterscheiden, denn über einen Kratzer zu wischen ist natürlich sinnlos und erhöht nur das Risiko, die Oberfläche noch weiter zu beschädigen. [15] Weiteres Zubehör, welches bei keinem Optik-Arbeitsplatz fehlen sollte, sind diverse Lichtquellen. Je nachdem welche Art der visuellen Kontrolle man durchführt, benötigt man entweder ein Durchlicht mit Diffusor, ein Auflicht oder einen flexiblen Faserlichtleiter, auch bekannt als *Schwanenhalsleuchte* [12]. Im Idealfall ist die Lichtstärke der Leuchtquelle regelbar.

1.2.5 Lagerung und Transport

Weitere wichtige Punkte betreffen die korrekte Lagerung und den Transport. Idealerweise transportiert man Optiken in luftdichten Behältnissen und bewahrt diese dort bis zu deren weiteren Verwendung auf.

Für weniger anspruchsvolle Komponenten, können einfache, verschließbare Kunststoffblister verwendet werden. Bei Komponenten mit empfindlicher Oberfläche empfiehlt es sich, diese zum Schutz zusätzlich noch einzeln zu verpacken. Dazu eignen sich Tüten aus *Pergaminpapier* oder spezielle „fusselfreie“ Optiktücher [6].

Halbfertige oder bereits vollständig montierte Baugruppen, lagert man am besten in Kunststoffboxen aus Polycarbonat, PTFE (Polytetrafluorethylen) oder PET-G (Glykol modifiziertes Polyethylenterephthalat). Boxen aus PP (Polypropylen) hingegen sollten nicht verwendet werden, da diese ausgasen und dadurch die beschichteten Optiken beeinflussen, indem sie die Oberfläche mit Ablagerungen kontaminieren. [16] Geringe Verunreinigungen reichen bereits aus, um laserinduzierte Schäden der Oberflächenvergütung zu verursachen, was speziell im Bereich von Hochleistungslasern zum Problem werden kann [17].

Die optimalen Lagerungsbedingungen erreicht man bei einer Temperatur zwischen 15 und 25 °C und einer relativen Luftfeuchte $< 30\%$ [16].

1.3 Manuelle Reinigungsmethoden

Im Zuge dieser Masterarbeit wurde auch recherchiert, welche Reinigungsmethoden dem aktuellen Stand der Technik entsprechen. Dabei stellte sich heraus, dass sich die Verfahren und Empfehlungen rund um die Pflege und Reinigung seit den letzten 400 Jahren nicht wesentlich verändert haben [18]. Ein optikspezifischer Standard hat sich dennoch nicht durchgesetzt, weder für die Reinigung noch für die Beurteilung der Sauberkeit von optischen Oberflächen [19].

„If it's not dirty, don't clean it!“

Zitiert aus [20]

Eingangs sei erwähnt, dass man Optiken nur dann reinigen sollte, wenn diese auch tatsächlich eine Verschmutzung aufweist, denn jeder Reinigungsprozess birgt auch die Gefahr, die Oberfläche versehentlich zu beschädigen [20].

Im Laufe der Zeit haben sich unterschiedliche Methoden etabliert, wobei sich nicht jede Art zu Reinigen für jede Optik eignet. Das bedeutet, es muss im Vorfeld anhand der Verschmutzungsart, des Verschmutzungsgrades, der Anzahl der zu reinigenden Optiken und der jeweiligen Vergütung abgeschätzt werden, welche Methode sich am besten eignet [6].

1.3.1 Reinigungsmittel

Neben der richtigen Reinigungsmethode, ist es auch wichtig, das richtige Reinigungsmittel zu verwenden. Während bei wasserlöslichen Verschmutzungen destilliertes Wasser ausreichend ist, lassen sich fettige Verunreinigungen, wie z.B. Fingerabdrücke, nur mit Hilfe von Lösungsmitteln entfernen [10].

Zu den am häufigsten verwendeten Lösungsmitteln zählen Aceton und Isopropylalkohol, auch bekannt als Isopropanol. Pures Aceton hat den Nachteil, dass es sehr schnell verdampft und deshalb manchmal nicht alle Verunreinigungen löst. Das Problem lässt sich umgehen, indem man es mit Methanol verdünnt. [21]

Isopropanol hingegen eignet sich für nahezu alle Optiken, außer jene mit Aluminiumbeschichtungen (z.B. Spiegel). Der Grund liegt darin, dass der Alkohol mit dem Aluminium eine Reaktion eingeht. [19] Weitere nicht kompatible Optik-Reinigungsmittel Kombinationen sind in Tabelle 2 auf Seite 8 angeführt.

Tabelle 2: Zu vermeidende Kombinationen bei der Reinigung von optischen Komponenten [19].

Reinigungsmittel, bzw. -methode	Beschichtung
Alkohole	Aluminiumbeschichtungen oder -substrate
Säuren wie HF oder HCl	Beschichtete Optiken, Optiken aus Zinksulfid und Zinkselenid, optische VIS- und UV-Gläser
MEC, MEK, TCE	Beschichtete Optiken
Alle	blanke Metallbeschichtungen wie Gold, Kupfer oder Aluminium
Ultraschallreinigung	optische Weichgläser aus z.B. CaF_2 , MgF_2 , SF_{11}

1.3.2 Druckluftreinigung

Partikel haben das Potential die Oberfläche während der Reinigung zu zerkratzen. Bevor man auf diverse Wisch- oder Pinseltechniken zurückgreift, ist es ratsam, den schlecht haftenden Schmutz von der Oberfläche zu blasen, ohne diese zu berühren. Dazu eignet sich die Erzeugung von Luftstößen mittels sogenannten *Blasebalg* oder Druckluft aus der Dose. [6, 20]

Das Unternehmen *Carl Zeiss*, weist im Onlineartikel „*Microscope Cleaning and Maintenance*“ darauf hin, dass bei Druckluftsprays die Gefahr besteht, dass extrem kalte Luft oder gefrierende Flüssigkeiten auf die Oberfläche treffen und diese irreparabel schädigen [22]. Gewöhnliche, durch einen Kompressor erzeugte Druckluft enthält meist viel Feuchtigkeit und ölige Rückstände. Sie eignet sich nur dann, wenn die Luft zuvor durch spezielle Druckluftfilter gereinigt und getrocknet wurde. [11]

Ein „Abpusten“ der Partikel mittels Atemluft, ist jedenfalls ungeeignet und führt eher dazu, dass die Oberfläche durch feine Speicheltröpfchen noch weiter kontaminiert wird. Bleibt das Abblasen mittels Druckluft erfolglos, so kann auf eine der folgenden Methoden zurückgegriffen werden.

1.3.3 Mechanische Abschrubverfahren

Bezüglich der Abschrubverfahren gibt es in der Literatur eine breite Übereinstimmung. Im Grunde unterscheidet man zwischen dem *Auftragen und Abziehen*, der *Pinseltechnik* und der *Wischtechnik* [6, 20, 21]. Welche Methode man anwendet, hängt

im Wesentlichen davon ab, wie hartnäckig der Schmutz an der Oberfläche haftet. Grundsätzlich sollte jedoch immer mit der für die Oberfläche am schonendsten Methode gestartet werden.

Auftragen und Abziehen

Diese Methode erlaubt es leichte Verunreinigungen zu entfernen und ist nur bei ungefasste Linsen und Spiegeln anwendbar [6].

Man legt die zu reinigende Optik zunächst auf ein sauberes, nicht scheuerndes Tuch. Spezielle Reinraum-Reinigungstücher eignen sich dafür besonders gut, da sich von diesen keine Gewebefasern lösen. Anschließend benetzt man ein zweites, ungefaltetes Tuch mit Aceton, Isopropylalkohol oder einer sonstigen, geeigneten Reinigungslösung. Das getränkte Tuch wird nun langsam und vorsichtig über die Oberfläche der Optik gezogen, bis das Lösungsmittel verdampft ist. [6, 14, 20]

Bei der Wahl des Lösungsmittel ist darauf zu achten, dass es sich für die vorliegende Optik eignet. Aceton beispielsweise, ist ein aggressives Lösungsmittel und darf deshalb keinesfalls bei Optiken aus Kunststoff angewendet werden [14]. Bei leichten Verschmutzungen reicht es oft aus, destilliertes Wasser zu verwenden [23].

Die Pinseltechnik

Bei dieser Reinigungstechnik ist ein zusätzliches Hilfsmittel in Form eines „Pinsels“ nötig. Dieser lässt sich aus einfachen Mitteln selbst herstellen.

Eine mögliche Variante ist es, Watte (100 % Baumwolle) auf einen Holzstiel aufzuwickeln, bis sich ein elliptischer Wattebausch bildet. Von gewöhnlichen, im Handel erhältlichen Wattestäbchen ist abzuraten, denn diese enthalten oft einen bestimmten Anteil aus synthetischen Fasern, welche die Oberfläche beschädigen können. [22] Des Weiteren lassen sich auch die bereits erwähnten Reinraumtücher als Pinsel umfunktionieren. Dafür werden diese mehrfach gefaltet, sodass die Länge der Faltkante in etwa die Optik-Oberfläche umspannt. Zum Halten des gefalteten Tuches kann z.B. eine Pinzette verwendet werden. [6]

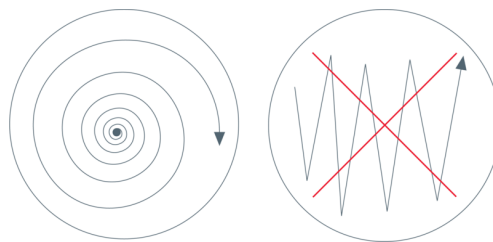


Abbildung 1: Richtige Reinigung nach dem Spiralmuster (links), und falsche Reinigung durch Zick-Zack-Bewegungen (rechts). Bildquelle: [24]

Die Reinigung erfolgt, indem man das Wattestäbchen bzw. den Linsentuchpinsel mit Lösungsmittel befeuchtet und in einer spiralförmigen Bewegung von der Mitte ausgehend nach außen wischt, siehe Abbildung 1. Die spiralförmige Bewegung ist dabei essentiell, denn durch sie werden die Unreinheiten gesammelt an den Rand der Optik gewischt. Eine Zick-Zack Bewegung hingegen, bewirkt nur eine Verteilung des Schmutzes. [22] Außerdem ist darauf zu achten, dass die Bewegung kontinuierlich ausgeführt wird, um Trocknungsspuren des Lösungsmittels zu vermeiden [20]. Für kleinere Optiken, lassen sich als Alternative zu den Selbstbau-Pinseln, auch Polyester-Tupfer mit langem Stiel verwenden.

Wischtechnik

Diese Technik wird bei besonders hartnäckigen Verschmutzungen angewendet. Im Grunde entspricht sie der Pinselmethode, mit dem Unterschied, dass das gefaltete Tuch zwischen die Finger geklemmt wird. Beim Abstreifen der Oberfläche sollte auf einen adäquaten, gleichmäßigen Druck geachtet werden. Die Methode eignet sich nicht zur Reinigung metallbeschichteter Oberflächen. [20, 6]

1.3.4 Reinigung mit Flüssigpolymeren

Ein weiterer Ansatz Optiken zu reinigen ist die Anwendung eines *Flüssigpolymers*, bei welchem im Unterschied zu den bereits beschriebenen Reinigungsmethoden weder Druckluft noch Lösungsmittel benötigt werden.

Die Technik dahinter wurde bereits in den 1980er Jahren entwickelt. Anstoß dafür war das Problem, dass bereits montierte und eingestellte Spiegel für die Reinigung ausgebaut werden mussten. Der Erfindung des *Flüssigpolymers* erlaubte es erstmals, Optiken an Ort und Stelle zu reinigen, ohne bereits eingestellte Optiken neu justieren zu müssen. [18]

Für die Reinigung wird das *Polymer* direkt auf die verschmutzte, optische Oberfläche aufgebracht, entweder durch Pinseln oder Sprühen. Nach wenigen Minuten härtet die Schicht zu einem dünnen Kunststofffilm aus, welcher den Schmutz an sich bindet. Das Polymer lässt sich im Anschluss einfach abziehen. [18]

Grundsätzlich eignet sich diese Technik für die meisten optischen Oberflächen, bei schlecht haftenden Beschichtungen wird von einer Anwendung jedoch abgeraten [25]. Durch seine zähe und flexible Eigenschaft, können Polymerfilme auch dazu verwendet werden, um bereits gereinigte Optiken bis zu deren Einbau zu schützen. Kratzer und Berührungen lassen sich damit während des Transportes effektiv verhindern. Im getrockneten Zustand schützt die Schicht außerdem vor Feuchtigkeit und Sulfidkontaminationen. [18, 25]

Der Reinigungserfolg dieser Methode wird auch durch Studien gestützt. In dem von Eric S. Bailey et al. veröffentlichtem Paper „*Increased Laser Damage Threshold by Protecting and Cleaning Optics Using First Contact Polymer Stripcoatings: Preliminary Data*“ wird bestätigt, dass die laserinduzierte Zerstörungsschwelle durch Reinigung mittels Polymerfilmen um bis zu 10 % angehoben wird [26].

1.4 Maschinelle Reinigungsmethoden

1.4.1 Ultraschallreinigung

Die Ultraschallreinigung ist eine Reinigungstechnik, bei welcher die *Kavitation* von Lösungsmitteln dazu genutzt wird, Oberflächen zu reinigen [27]. Das Reinigungsgerät besteht im Wesentlichen aus einer Schwingwanne und Ultraschallgeneratoren, die meist am Boden der Wanne angebracht sind. Die Generatoren erzeugen Schwingungen im Ultraschallbereich (20 kHz bis 500 kHz), welche über die Wanne in die Reinigungsflüssigkeit übertragen werden. Das bewirkt, dass in der Reinigungsflüssigkeit Unter- sowie Überdruckwellen entstehen. Der ständige Wechsel zwischen Unter- und Überdruck erzeugt kleine Kavitationsbläschen, welche solange anwachsen bis sie implodieren. Der durch die Implosion entstehende Wasserdruck löst die Verunreinigungen des Reinigungsgutes. [28]

Frequenz und Leistung des Ultraschalls sind variabel und lassen sich an das zu reinigende Gut und den Verschmutzungsgrad anpassen. Die Größe der Kavitationsbläschen steht in einem umgekehrt proportionalen Verhältnis zur Ultraschallfrequenz, während die Kavitationsintensität proportional zur Ultraschallleistung steigt [27].

Das Reinigungsergebnis hängt auch maßgeblich von der Wahl der richtigen Reinigungsflüssigkeit ab. Es ist wichtig zu wissen, um welche Art Verschmutzung es sich handelt, um auf Basis dessen, ein passendes Lösungsmittel einzusetzen. Außerdem ist zu beachten, dass die Reinigungslösung in regelmäßigen Abständen ersetzt werden sollte, da der Verschmutzungsgrad in der Lösung mit jedem Durchlauf steigt. Ultraschall entfernt Partikel mit einem Durchmesser von mehr als $25\ \mu\text{m}$ recht effektiv und eignet sich damit gut für Reinigungen während der Produktion von Optiken. Für optische Präzisionselemente hingegen, sollte die Reinigungstechnik bestenfalls zur Entfernung grober Verunreinigungen in Betracht gezogen werden und nicht als Endreinigungsmethode. [7]

1.4.2 Weitere maschinell unterstützte Reinigungsmethoden

Neben den eben beschriebenen Reinigungsverfahren, gibt es noch weitere Ansätze optische Oberflächen zu reinigen, siehe Tabelle 3. Diese Techniken sind nicht weit verbreitet, da ihr Einsatzbereich entweder sehr begrenzt oder die Durchführung sehr aufwändig ist.

Tabelle 3: Kurzbeschreibung weiterer gängiger Reinigungsmethoden

Methoden	Funktionsprinzip
CO_2 -Reinigung	Bei dieser Methode wird aus CO_2 erzeugter Schnee mit hoher Geschwindigkeit auf die Optikoberfläche geschleudert. Der Aufprall des Schnees bewirkt, dass Verunreinigungen von der Oberfläche geschleudert werden. Diese Methode eignet sich sowohl für Zwischen- und Endreinigung und wird meist bei schwer zugänglichen Oberflächen eingesetzt. [11]
Dampfreinigung	Die Dampfreinigung eignet sich für besonders empfindliche Oberflächen, bei welchen eine mechanische Reinigungsmethode nicht anwendbar ist. Lösungsmittel wie Alkohol oder Aceton werden zunächst in einem Bad bis zum Siedepunkt erhitzt. Im Anschluss wird die zu reinigende Oberfläche für eine kurze Zeit leicht gekippt über das Bad gehalten. Der durch das Erhitzen entstehende Dampf kondensiert an der deutlich kälteren optischen Oberfläche. Es bilden sich Kondensattropfen, welche den Schmutz von der Oberfläche spülen. Diese Technik erfordert erhöhte Sicherheitsmaßnahmen, da der erzeugte Lösungsmitteldampf schnell ein leicht entzündliches Niveau erreichen kann. [29]
Mikro-Vakuum Reinigung	Diese Technik basiert auf dem allgemein bekannten Staubsaugerprinzip. Eine Micro-Vakuumsonde wird nah an der verschmutzten Oberfläche platziert. Die durch die Düse eingesaugte Luft erzeugt so hohe Scherkräfte, dass sich Staub von der Oberfläche lösen und einsaugen lässt. Das Prinzip eignet sich nur für Partikelgrößen $> 50\mu m$ und ist sehr zeitaufwändig. [7]

1.5 Verifizierung der Sauberkeit

1.5.1 Quantitative Bestimmung von Verschmutzungen

Irving F. Stowers und Howard G. Patton schreiben in ihrem Paper „*Cleaning Optical Surfaces*“, dass sich die quantitative Bestimmung der Sauberkeit einer Oberfläche als schwierig erweist [7]. Es gibt jedoch drei Ansätze die dabei helfen sollen, den Reinheitsgrad zu bestimmen, siehe dazu Tabelle 4.

Tabelle 4: Vergleich von Sauberkeit auf Grundlage von Dicke, Gewicht und Partikelanzahl. In Anlehnung an [1].

Monoschicht	Eine Monoschicht, auch Atomlage oder Monogamer genannt, entspricht in etwa 10^{14} - 10^{15} Atomen pro cm^2 .
Gewicht	Ein $\mu\text{g}/\text{cm}^2$ entspricht fünf Kohlenstoff-Monoschichten.
Partikelanzahl	Ein Partikel ($\varnothing=10\ \mu\text{m}$) pro cm^2 ist einem $\frac{1}{20}$ eines Monolayers gleichzusetzen.

Theoretisch gelten Oberflächen als „extrem sauber“ wenn ihr Reinheitslevel kleiner einem Zehntel einer Monolayers entspricht. Aus praktischer Sicht lassen sich in Produktionsumgebungen Reinheiten von bis zu 10 Partikel pro cm^2 , mit einem Partikeldurchmesser $> 10\mu\text{m}$, erreichen. [7]

Um den Verschmutzungsgrad einer allgemeinen Oberfläche zu quantifizieren gibt es grundsätzlich zwei Methoden: Partikuläre Verschmutzung wie Staub, lässt sich einfach über die Zählmethode beziffern. Im Gegensatz dazu, sind Schmutzfilme nicht so einfach zu bestimmen. In diesem Fall setzt man auf eine NVR-Analyse (Non-volatile Residue) [30]. Dessen Konzept basiert auf der *Gravimetrischen Bestimmung* und entspricht im Grunde einer hochpräzisen Gewichtsmessung [31].

1.5.2 Klassifizierung der Oberflächenreinheit

Die Reinheit einer Oberfläche lässt sich anhand unterschiedlicher Normen und Standards klassifizieren.

Beispielsweise befasst sich Teil 9 der *ISO 14644* mit der Klassifizierung der *partikulären Oberflächenreinheit*. Der Anwendungsbereich umspannt alle festen Oberflächen von Reinräumen und deren zugehörige Reinraumbereiche. Dazu zählen auch Werkzeuge, Ausrüstungsgegenstände und *Produkte*. Die Einteilung erfolgt in acht SCP-Klassen (Surface Cleanliness by Particle Concentration) und basiert auf der

höchstzulässigen Partikelkonzentration pro Quadratmeter, wobei auch die Partikelgröße in die Klassifizierung miteinbezogen wird. [32] Die in der Norm beschriebenen Verunreinigungen beziehen sich ausschließlich auf Partikel, Schmutzfilme werden nicht behandelt.

Das amerikanische Gegenstück zur *ISO 14644-9:2022* ist der Industriestandard *IEST STD-CC1246E*, welcher die Produktoberflächensauberkeitsstufen für kommerzielle und nicht-kommerzielle Zwecke regelt. Im Gegensatz zur ISO-Norm, spezifiziert die *IEST STD-CC1246E* nicht nur die partikuläre Oberflächenreinheit, sondern auch nichtflüchtige Rückstände. Das partikelbasierte Reinheitslevel wird über die maximale Anzahl an Partikel pro 0.1 m^2 definiert und nimmt dabei ebenfalls Rücksicht auf die Partikelgröße. Für nichtflüchtige Rückstände wird je Level eine maximal erlaubte Masse pro 0.1 m^2 angegeben. [30]

Per Definition ließe sich der Reinheitsgrad optischer Oberflächen anhand beider Standards festlegen. Die Einheiten „pro m^2 “ bzw. „pro 0.1 m^2 “ lassen aber erahnen, dass eine Anwendung auf die meist deutlich kleineren Optikoberflächen nicht berücksichtigt wird.

Eine Norm, welche die Oberflächenreinheit für explizit optische Komponenten regelt, existiert bis dato nicht. Das bestätigt auch der namhafte Optikhersteller *asphericon* (Jena, Deutschland), siehe elektronische Anfrage in Anhang C.

1.5.3 Visuelle Sichtprüfung vor dem Einbau

Robert Schlack schreibt in seinem Buch „*The Proper Care Of Optics*“, dass das menschliche Auge durch sein stereo-visuelles System in der Lage ist, die Position und Höhe von Objekten zu erkennen. Es eignet sich deshalb sehr gut um Fehler und Verunreinigungen an optischen Oberflächen zu identifizieren. Diese Fähigkeit lässt sich weiter verbessern, indem man den Vergleich zwischen sauberen, fehlerfreien und verschmutzten oder defekten Optikoberflächen trainiert. Ein geschultes Auge erkennt selbst geringe Abweichungen in Kontrast, Farbe und Textur. [11]

Um das Auge bei der visuellen Kontrolle bestmöglich zu unterstützen, bedarf es einer Prüfungsumgebung wie in Kapitel 1.2.3, Seite 4 beschrieben. Der gezielte Einsatz unterschiedlicher Lichtquellen hilft dabei, Unreinheiten leichter zu erkennen. Auf diese Weise kann das Auge Partikel in der Größe von bis zu $10\text{ }\mu\text{m}$ auflösen [7]. Im Laufe der Zeit haben sich unterschiedliche Methoden zur Sichtprüfung etabliert. Diese unterscheiden sich hauptsächlich durch die Beleuchtungsart. Zu bedenken ist, dass die Ergebnisse visueller Sichtprüfungen immer an die subjektive Wahrnehmung des Prüfers geknüpft sind, unabhängig davon welche Methode man einsetzt [12].

Durchlichtkontrolle

Bei der Durchlichtkontrolle wird die zu prüfende Optik zwischen einer Lichtquelle und dem Auge platziert. Der typische Kontrollabstand zwischen Auge und Linse liegt zwischen 25 und 40 cm. Als Durchlichtbeleuchtung eignet sich eine LED-Lichtquelle mit Diffusorscheibe, wie in Abbildung 2 ersichtlich. [12]

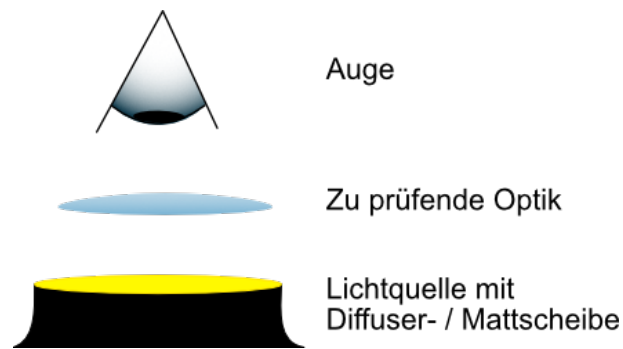


Abbildung 2: Prinzip der Durchlichtkontrolle. In Anlehnung an [12]

Streulichtkontrolle

Die Streulichtkontrolle erfolgt vor schwarzem Hintergrund. Die Lichtquelle ist so zu positionieren, dass das Licht von der gegenüberliegenden Seite in einem Winkel von mindestens 45° auf die Optik einstrahlt. [11] Durch ein leichtes Verkippen der Optik wird das einfallende Licht unterschiedlich gestreut, Verschmutzungen und Defekte lassen sich so gut erkennen.

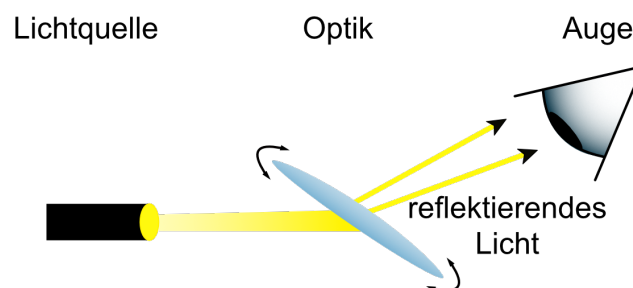


Abbildung 3: Schematische Darstellung der Streulichtkontrolle. In Anlehnung an [12]

Für diese Art von Prüfung eignet sich am besten stark gebündeltes Licht in Form einer Kaltlichtquelle mit flexiblem Lichtleiter, auch *Schwanenhals* genannt [12].

Auflichtkontrolle

Diese Methode eignet sich besonders gut, um Inhomogenitäten der Beschichtung und der Verkittung zu identifizieren. Als Lichtquelle kann entweder ein flexibler Lichtbalken, oder eine großflächige diffuse Lichtquelle eingesetzt werden. Auch bei dieser Technik wird die in der Zeichnung spezifizierte optische Oberfläche auf Sauberkeit und Beschädigungen wie Kratzer, Druckstellen oder dekorative Auffälligkeiten überprüft. [12]

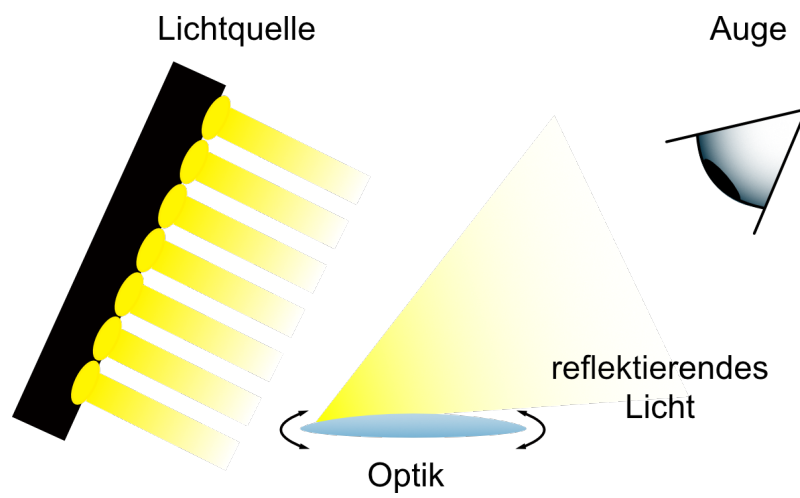


Abbildung 4: Beispielhafte Darstellung einer Oberflächeninspektion mit Hilfe von Auflicht, in Anlehnung an [12]

1.6 Einfluss von Verschmutzungen

Verschmutzte Oberflächen beeinflussen die optische Performance einer Komponente auf unterschiedliche Art und Weise. In welcher Form sie sich auswirken, hängt im Wesentlichen davon ab, ob die Kontamination als partikulärer Schmutz, oder als filmige Verunreinigungen vorliegt.

1.6.1 Verdunkelung und Streuung

Fällt Licht auf eine durch Partikel verunreinigte, optische Oberfläche, so kommt es einerseits zu Streuungseffekten und andererseits zu Abschattungen bzw. Transmissionsverlusten. Während für die Abschattung Größe und Anzahl der im optischen Pfad befindlichen Partikel entscheidend ist, hängt die Streuung vom Verhältnis zwischen Partikelgröße und Wellenlänge des einfallenden Lichtes ab. [5]

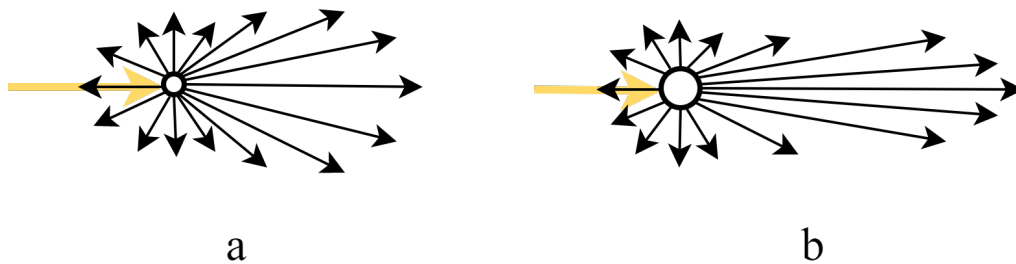


Abbildung 5: Typische durch *Mie-Streuung* erzeugte Muster an einem von links beleuchteten a) kleineren und b) etwas größeren Partikel. In Anlehnung an [33]

Geht man von einem Partikeldurchmesser $>10\ \mu\text{m}$ aus, so ist die Wellenlänge von sichtbarem Licht im Vergleich deutlich kleiner. In diesem Fall lässt sich die resultierende Streuung mithilfe der *Mie-Theorie* berechnen [33]. Abbildung 5 zeigt das Muster der *Mie-Streuung* an Partikeln unterschiedlicher Größe.

Es sei darauf hingewiesen, dass das Phänomen der Streuung den optischen Pfad nicht nur innerhalb einer Optik beeinflusst. Das erzeugte Streulicht kann sich auch außerhalb der betroffenen Komponente auf andere Teile des optischen Systems auswirken. [5]

In der Spektroskopie beispielsweise, kann Streulicht zu Problemen führen, da das zugrundeliegende Prinzip auf der Messung von Lichtintensitäten beruht. Ein zu hoher Grad an Streulicht beeinflusst die Messung. [34]

In der Mikroskopie äußern sich verschmutzte Optiken in Form von verringerter Bildschärfe und verringertem Kontrast. Nicht jede Verunreinigung wirkt sich da-

bei gleich stark auf das Bild aus. Ausschlaggebend ist, welche optische Komponente betroffen ist. Grundsätzlich gilt, dass der Einfluss einer Verunreinigung auf das Bild zu nimmt, je näher sie sich am zu untersuchenden Objekt befindet. Es ist deshalb besonders bei Frontlinsen der Objektive auf ausreichend Sauberkeit zu achten. Bei Digital-Mikroskopen kommt hinzu, dass auch die Oberfläche des Sensors sowie dessen Schutzglas starken Einfluss auf die Bildgebung haben. [10]

1.6.2 Absorptionsbanden

Verunreinigungen in Form eines Schmutzfilmes führen zu unerwünschten Absorptionsbanden und können insbesondere bei Lasersystemen zu Problemen führen [7]. Das einfallende Licht wird je nach Wellenlänge mehr oder weniger vom Schmutzfilm absorbiert und erzeugt dort eine lokale Erwärmung. Wenn die Energieabsorption des Schmutzfilmes hoch genug ist, kann es zu thermisch-bedingten Beschädigungen der Optik kommen. [5, 35]

$$I(d) = I_0 e^{-\alpha \cdot d} \quad (1)$$

Mathematisch lässt sich die Absorption des Lichtes mit Hilfe des *Lambert-Beer'schen Gesetzes* (1) abschätzen. Nichtlinearitäten und Streuungseffekte werden dabei nicht berücksichtigt. Der lineare Absorptionskoeffizienten α ist von der Wellenlänge des Lasers und der Konzentration der absorbierenden Schicht abhängig. Die durch den Laser zugeführte Lichtintensität entspricht I_0 . Jener Lichtanteil, den der Schmutzfilm mit der Dicke d passiert, tritt als Lichtintensität $I(d)$ wieder aus und trifft auf die Oberfläche der Optik. Je höher die Differenz $I_0 - I(d)$, desto höher die Gefahr, dass es zu einer irreversiblen Schädigung der Optik kommt. [35]

Ein in diesem Zusammenhang oftmals erwähnter Begriff ist die sogenannte *Laserzerstörungsschwelle*. Sie ist eine wichtige Kenngröße im Bereich der Laseroptiken und gibt an, welcher maximalen Laserleistung pro Fläche eine optische Komponente ausgesetzt werden kann, ohne dass diese zerstört wird [36]. Je höher der Verschmutzungsgrad, desto niedriger die Laserzerstörungsschwelle.

Liegt der Schmutzfilm in Form eines Tröpfchens vor, so kann dieser zusätzlich als partikuläre Verunreinigung gesehen werden, welche wie bereits erwähnt zu Streuungseffekten führt [5].

1.7 Ziel und Anforderungen

Ziel dieser Masterarbeit ist die Entwicklung eines Reinigungssystems, welches im Unternehmen WILD GmbH in den Montageprozess optischer Baugruppen eingesetzt werden soll.

Bisher erfolgte die Reinigung von Optiken ausschließlich manuell, anhand der in Kapitel 1.3.3 beschriebenen *Pinselftechnik*. Wird dabei zu viel Kraft aufgewendet, kann es passieren, dass die Oberflächenvergütung beschädigt wird, was in Folge die optische Performance der Komponente beeinträchtigt.

Das zu entwickelnde Reinigungssystem soll gewährleisten, dass die eingelegte Optik, über ihre gesamte Oberfläche hinweg, mit einem möglichst konstanten Druck gereinigt wird. Das Hauptaugenmerk liegt dabei in der Reinigung von exklusiv runden Optiken (vorwiegend Linsen) unterschiedlicher Größe und Krümmung. Der Reinigungsablauf soll soweit automatisiert werden, dass die verschmutzten Optiken nur noch in den Roboter eingelegt werden müssen.

1.7.1 Anforderungen an die Mechanik

Das grundsätzliche Reinigungsprinzip der *Pinselftechnik*, soll beibehalten aber automatisiert werden. Von der Automatisierung ausgenommen sind das Einlegen, das Wenden und das Entnehmen der zu reinigenden Optik.

Aufnahmeeinheit

Die Aufnahmeeinheit ist so zu gestalten, dass eine große Bandbreite an unterschiedlichen runden Optiken eingespannt werden kann. Bezüglich Form und Durchmesser des Reinigungsgutes sind folgende Bedingungen zu erfüllen:

Tabelle 5: Anforderungen an die Aufnahmeeinheit

Wert Maß	Min. in mm	Max.m in mm
Optikdurchmesser	15	60
Krümmungsradius	23	305

Lösungsmiteleinheit

Die Zufuhr des Lösungsmittels soll ebenfalls durch den Reinigungsroboter erfolgen. Da bei der Reinigung größtenteils Aceton als Lösungsmittel eingesetzt wird, ist darauf zu achten, dass ausschließlich beständige Materialien zur Anwendung kommen.

Aufnahme der Reinigungsstäbchen

Als „Pinsel“ sollen spezielle *Polyestertupfer* der Firma Texwipe (Kernersville, USA) zum Einsatz kommen. Da diese von Zeit zu Zeit zu wechseln sind, ist darauf zu achten, dass die Pinselaufnahme so konstruiert ist, dass ein Austausch unkompliziert und ohne viel Zeitaufwand erfolgen kann.

Energieeffizienz und Nachhaltigkeit

Im Hinblick auf die aktuell vorherrschende Energiekrise ist das System möglichst energieeffizient auszulegen. Des Weiteren soll der derzeitige, hohe Verbrauch an Lösungsmitteln reduziert werden.

1.7.2 Funktionale Anforderungen

Optikprofile

Das Reinigungssystem soll so flexibel sein, dass sich neben gewöhnlichen konvex- bzw. konkav-gekrümmten Optikoberflächen auch abgestufte Profile oder plane Flächen reinigen lassen. Der Druck des Reinigungsstäbchens auf die optische Oberfläche darf sich dadurch nicht wesentlich verändern.

Betriebsart

Das Reinigungssystem soll grundsätzlich als Standalone-Gerät betrieben werden. Optional soll es aber auch möglich sein, den Roboter über netzwerkfähige Geräte wie PC, Notebook oder Tablet zu bedienen, um beispielsweise grundlegende Einstellungen zu ändern.

Es soll außerdem möglich sein, die Reinigungseinstellungen an die jeweilige Optik anzupassen. Das Reinigungsprofil einer Linse umfasst mindestens folgende einstellbare Parameter:

- Rotationsgeschwindigkeit der Optikaufnahme
- Abgegebene Lösungsmittelmenge
- Kraft (bzw. Druck), mit welcher das Reinigungsstäbchen auf die Oberfläche der Optik drückt
- Anzahl der Reinigungsdurchläufe

Die angeführten Parameter müssen sich in Form eines Reinigungsprofiles speichern lassen. AnwenderInnen soll es möglich sein, Profile anzulegen, zu löschen und zu ändern.

2 Methoden

2.1 Konzepterstellung

2.1.1 Herangehensweise

Die grundlegende Aufgabe des Roboters besteht darin, den manuellen Reinigungsprozess und im weiteren Sinne die Bewegungen der MitarbeiterInnen nachzuempfinden. Für die Ausarbeitung eines Konzeptes war es zunächst erforderlich, den Ablauf zu begutachten, weshalb das Projekt damit gestartet wurde, den manuellen Reinigungsprozess vor Ort (Fa. WILD) zu untersuchen. Dazu wurden die einzelnen Arbeitsschritte der MitarbeiterInnen während der Reinigung genau beobachtet und analysiert. Die daraus gewonnenen Informationen wurden genutzt, um die Aufgabenstellung zu präzisieren und verschiedene Lösungsansätze auszuarbeiten.

2.1.2 Mechanisches Basissystem

Die Analyse des Reinigungsablaufes ergab, dass für die Nachahmung der in Kapitel 1.3.3 beschriebenen *Pinselmethode*, mindestens ein zwei-achsiges System notwendig ist. Als Basis dafür, wurde ein CNC-basiertes (Computerized Numerical Control) Positionierungssystem in Betracht gezogen, welches sich flexibel einsetzen lässt und zudem als DIY-Bausatz (Do It Yourself) erhältlich ist.

Die Positioniergenauigkeit solcher DIY-Bausätze ist zwar nicht mit jener von professionellen Systemen vergleichbar, aber für den geplanten Einsatz und insbesondere für den Bau eines Prototypen ausreichend.

2.1.3 CAD Konstruktion

Zur besseren Veranschaulichung, wurde aus der Projektidee ein dreidimensionales Modell des Reinigungsroboters erstellt, siehe Abbildung 6 auf Seite 22. Eine digitale Version der Projektidee bietet mehrere Vorteile. Die räumliche Darstellung hilft beispielsweise bei Design-Diskussionen. Außerdem lassen sich aus den 3D-Modellen, ohne viel Aufwand 2D-Zeichnungen für die anschließende Herstellung der Fertigungsteile ableiten. Bei nahezu allen gängigen CAD-Anwendungen (Computer Aided Design) ist es zudem möglich, die Bauteilmodelle für den 3D-Druck, in Form von STL-Dateien (Standard Triangulation Language), zu exportieren.

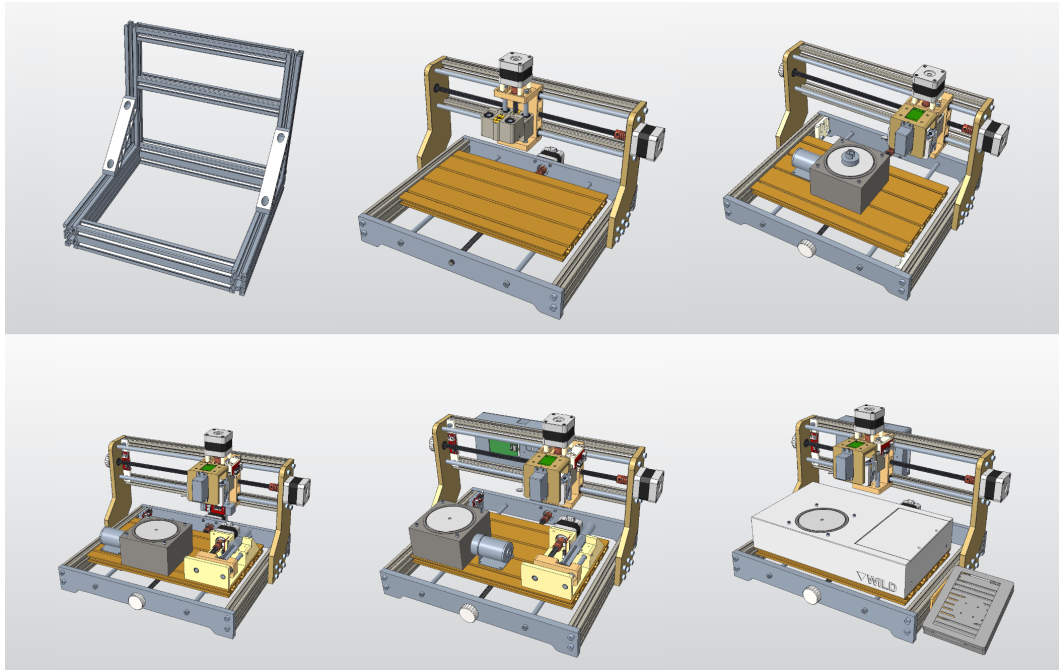


Abbildung 6: Verschiedene Phasen während der Konzeptentwicklung

Für die Entwicklung des Konzeptes wurde die CAD-Software *Creo* (PTC Inc., Boston, USA) herangezogen. Ein hilfreiches Feature von *Creo* ist, dass auch Bewegungen simuliert werden können. Damit konnten Probleme wie z.B. Kollisionen von Bauteilen bereits vor der Fertigung bzw. dem Druck erkannt und behoben werden. Für Normteile wie Schrauben, Muttern, Stifte usw. wurde auf frei zugängliche Bibliotheken des Unternehmens Trace Parts GmbH (Amberg, Deutschland) zurückgegriffen. *Creo* erlaubt es, neben seinen eigenen Dateiformaten auch Modelle anderer CAD-Software-Hersteller zu importieren und in das bestehende Projekt einzubauen.

2.2 Werkstoffe und Fertigungstechniken

2.2.1 Mechanische Komponenten

Bei der Herstellung der Fertigungsteile wurde größtenteils auf *Rapid Prototyping* in Form von 3D-Druck gesetzt. Der große Vorteil des 3D-Drucks liegt darin, dass Bauteile kurzerhand und kostengünstig gefertigt werden können, was speziell für den Bau von Prototypen interessant ist.

Gedruckt wurde hauptsächlich mit dem Material *Onyx* (Markforged, Watertown, USA). Die chemische Zusammensetzung von *Onyx* ermöglicht es, dass die gedruckten Teile auch gegenüber starker Lösungsmittel wie Aceton resistent sind [37]. Für die weniger anspruchsvollen Bauteile wurden Filamente aus *Polylactid Acid* (PLA) und *Acrylnitril-Butadien-Styrol-Copolymer* (ABS) verwendet. Um die Druckqualität konstant zu halten, kamen drei unterschiedliche Drucker zum Einsatz welche zuvor auf jeweils ein Filament optimiert wurden, siehe dazu Tabelle 6.

Tabelle 6: Auflistung aller 3D-Drucker, welche für den Bau des Reinigungsroboters verwendet wurden. Bei den Geräten *Mark Two* und dem *EL-102* handelt es sich im Gegensatz zum *Ender3 S1* um professionelle Industriedrucker.

Drucker	Hersteller	Filament	Druckraum in mm
Mark Two	Markforged	Onyx	320 x 132 x 154
Ender 3 S1	Creality	PLA	220 x 220 x 270
EL-102	EVO-TECH	ABS	500 x 400 x 510

Trotz der gut entwickelten 3D-Druck-Technologie und den teils hervorragenden Materialeigenschaften der Filamente, eignet sich *Rapid Prototyping* nicht für jedes Bauteil. In solchen Fällen wurde auf die klassischen Fertigungsverfahren in Kombination mit höherwertigen Materialien (Metalle) zurückgegriffen.

2.2.2 Elektronische Komponenten

Breakout Boards

Neben den mechanischen Bauteilen, benötigt das Reinigungssystem auch einige elektronische Komponenten. Beispielsweise kommen Module wie Schrittmotortreiber, ADC (Analog-Digital Wandler) oder DC-DC Wandler in Form eines *Breakout Boards* zum Einsatz.

Bei *Breakout Boards* sind die, für die Funktion zugrunde liegenden *ICs* (Integrierten Schaltkreise), bereits auf kleinen Platinen verlötet und alle wichtigen Anschlüsse als Steckverbindungen herausgeführt. Das erleichtert die Entwicklung des Gesamtsystems und macht die Module im Fall eines Defektes leicht austauschbar. Außerdem lässt sich das Reinigungssystem durch Anwendung von *Breakout Boards* zukünftig auch leichter vervielfältigen.

Ein Nachteil von *Breakout Boards* ist jedoch, dass sie nicht sonderlich flexibel sind. Für die Konfiguration eines *ICs* ist oftmals die externe Beschaltung ausschlaggebend und eben jene, ist bei *Breakout Boards* nur sehr eingeschränkt bis gar nicht änderbar. Kapitel 2.5.4 auf Seite 44 beschreibt einen konkreten Fall, bei dem es nötig war, die Platine zu manipulieren.

Logiklevel

Bei der Wahl der Elektronikmodule war neben den funktionalen Anforderungen darauf zu achten, dass diese im Hinblick auf die Logik-Technologie mit der Steuerungsplatine kompatibel sind. Der für die Systemsteuerung eingesetzte Einplatinencomputer *Raspberry Pi* (Raspberry Pi Ltd., Cambridge, England), basiert auf einer 3.3 V Logik und ist gegenüber einer 5 V Logik nicht tolerant [38]. Das bedeutet, dass jegliche direkt angeschlossene Peripherie ebenfalls auf einer 3.3 V Logik basieren muss. Werden die GPIO-Pins (General Purpose Input/Output) des Raspberry Pi mit mehr als 3.3 V beschalten, besteht die Gefahr einer irreversiblen Schädigung.

Kabelverbindungen

Die Verkabelung des Roboters basiert zum größten Teil auf Klemm- und Steckverbindungen, das erleichtert etwaige Änderungen oder Reparaturen des Systems. Da sowohl bei Steuerungsplatine, als auch bei den *Breakout Boards* Stiftheihen im 2.54 mm Raster verlötet sind, wurden Flachbandkabel in Kombination mit den weit verbreiteten *Dupont-Steckverbindern* eingesetzt. Für die Anschlüsse von Schrittmotoren und Endschalter kamen Stecker und Buchsen der JST-RE Serie zum Einsatz.

2.2.3 Normteile

Für alle bisher nicht angesprochenen mechanischen Bauteile wie Schrauben, Muttern, Passstifte, Gleitlager usw. wurde auf Normteile unterschiedlicher Hersteller zurückgegriffen.

2.3 Vorversuche

Vor dem eigentlich Aufbau des Reinigungsroboters wurden die in Betracht gezogenen Sensoren und Aktoren auf ihre Tauglichkeit geprüft. Das Ziel war es herauszufinden, ob sich die Komponenten für eine Implementierung in das Reinigungssystem eignen. Es wurde außerdem ermittelt, wie sich die elektronischen Komponenten softwaretechnisch ansteuern bzw. auslesen lassen. Folgende Versuche wurden ausgeführt:

- Aufbau einer seriellen Kommunikation zwischen PC und CNC-Controller über UART (Universal Asynchronous Receiver / Transmitter)
- Einlesen von Endschalterzuständen über *Polling*- und *Interrupts*
- Indirekte und direkte Ansteuerung eines Servomotors über Pulsweitenmodulation (PWM)
- Ansteuerung und Auslesen eines PWM basierenden Nivellierungssensors
- Kraftmessung mittels Wägezelle in Verbindung mit einem ADC-*Breakout Board*

Als Steuerung wurde das weit verbreitete Microcontroller-Entwicklungsboard *Arduino UNO R3* (Arduino S.r.l., Monza, Italien) verwendet. Eine ausführliche Beschreibung der durchgeführten Testreihe ist in der Seminararbeit mit dem Titel „*Ansteuerung unterschiedlicher Sensoren und Aktoren über einen Microcontroller*“ nachlesbar [39].

2.4 Systemüberblick

Das Reinigungssystem lässt sich im Wesentlichen in ein CNC-Positionierungssystem, einer zentralen Steuerung, einer Anzeigeeinheit, einem Drehteller mit Optikaufnahme, einer Lösungsmittelfördereinheit und dem Reinigungskopf mit Abtastfunktion einteilen. Tabelle 7 enthält eine kurze Funktionsbeschreibung aller genannten Einheiten. Auf Einzelheiten wird in den folgenden Kapiteln näher eingegangen.

Tabelle 7: Kurzbeschreibung der sechs Einheiten des Reinigungsroboters

Einheit	Kurzbeschreibung
CNC-Positionierungssystem	Das Positionierungssystem bildet die Basis des Reinigungsroboters. Alle weiteren Einheiten bauen auf ihr auf. Die Hauptfunktion besteht darin, den Reinigungskopf im dreidimensionalen Raum zu positionieren und damit die Bewegungen der in Punkt 1.3.3 beschriebenen <i>Pinselmethode</i> nachzuahmen. Die Positionierung erfolgt über drei Schrittmotor-betriebene Trapezspindeln.
Zentrale Steuerung	Wie der Bezeichnung zu entnehmen ist, koordiniert diese Einheit den gesamten Reinigungsablauf durch Ansteuerung der unterschiedlichen Aktoren, so wie dem Auslesen von Sensoren.
Bedien- und Anzeigeeinheit	Die Einheit ermöglicht die Interaktion mit dem Anwender in Form eines touchfähigen LCD-Monitors.
Drehtellereinheit	Die Drehtellereinheit rotiert die zu reinigende Optik über einen Gleichstromgetriebemotor. Die Geschwindigkeit ist variabel und wird über das CNC-Positioniersystem geregelt.
Lösungsmittelfördereinheit	Sie besteht aus einer Spritzenpumpe und einem Schlauchsystem, über welches das Lösungsmittel zum Reinigungsstäbchen transportiert wird.
Reinigungskopf	Diese multifunktionale Einheit hat mehrere Aufgaben: Zum einen dient sie als Aufnahme für das Reinigungsstäbchen. Zum anderen wird mittels integrierter Wägezelle die Kraft (bzw. der Druck) gemessen, mit der das Stäbchen auf die Optik drückt. Mit dem verbauten Nivellierungssensor <i>BLTouch</i> ist es möglich das Oberflächenprofil der Optik abzutasten.

Alle Einheiten des Reinigungsroboters sind in Abbildung 7 dargestellt. Im Normalfall ist die Pumpeinheit des Lösungsmittelfördersystems (5) durch eine Klappe verdeckt, zur besseren Verdeutlichung wurde diese im CAD-Modell ausgeblendet.

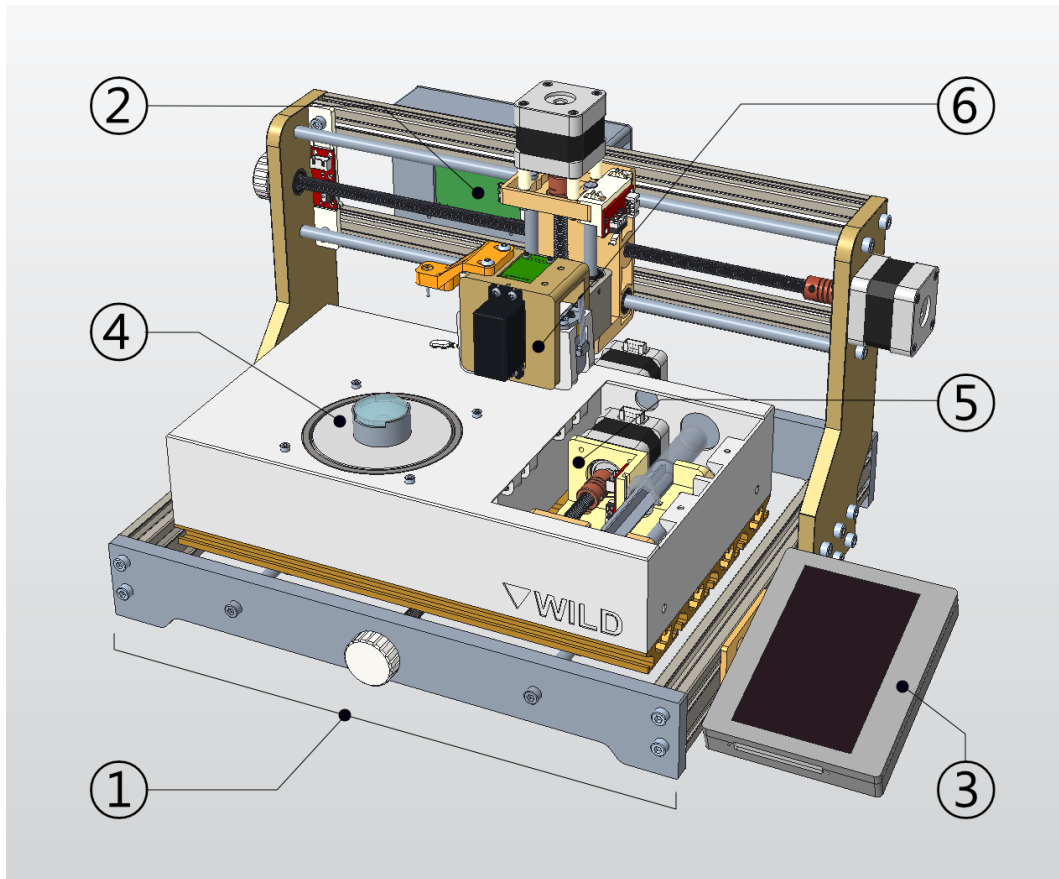


Abbildung 7: CAD-Modell des Reinigungsroboters in seiner aktuellsten Version, Stand Januar 2023.

- (1) CNC-Positionierungssystem (Grundaufbau)
- (2) Zentrale Steuerungseinheit
- (3) Bedien- und Anzeigeeinheit
- (4) Drehtellereinheit
- (5) Lösungsmittelleinheit
- (6) Reinigungskopf

In Abbildung 8, Seite 28 wird verdeutlicht, wie die einzelnen Systemeinheiten miteinander verknüpft sind. Richtung und Inhalt des Informationsflusses sind in Form von Pfeilen visualisiert. Die beiden gestrichelten Linien zwischen Steuerung und PC kennzeichnen, dass es sich um eine optionale Verbindung handelt.

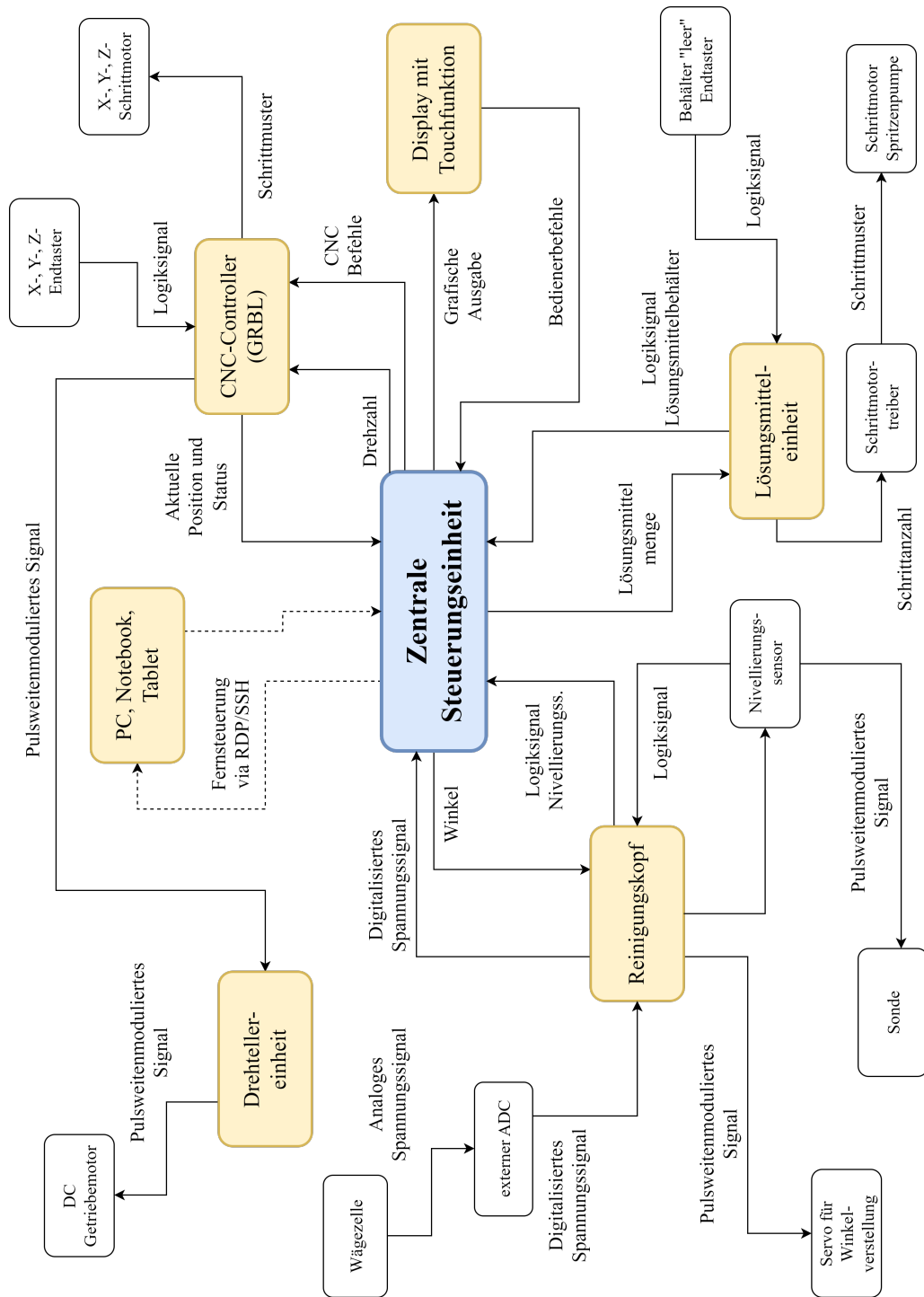


Abbildung 8: Blockdiagramm des Reinigungssystems mit Visualisierung des Informationsflusses

2.5 Hardware

2.5.1 Zentrale Steuerungseinheit

Wie bereits erwähnt, basiert die zentrale Steuerung des Reinigungsroboters auf einem *Raspberry Pi* der vierten Generation, kurz RPI4. Der große Vorteil dieses Single-Board-Computers ist, dass er die einfache Bedienbarkeit eines Desktop PCs mit der Hardwarenähe von Microcontrollern verbindet. Diese Kombination macht den RPI4 zu einer prädestinierten Steuerung für das gegenwärtige Projekt.

I/O Pins

Der auf einer ARM-Architektur basierende RPI4 bietet mit 40 GPIO-Pins (nicht alle davon sind frei konfigurierbar) [38], genügend Ein- und Ausgänge, um alle im System eingesetzten Sensoren und Aktoren zu bedienen.

Tabelle 8: Konfiguration des 40-poligen RPI4 Pin-Headers. Farblich hinterlegte Pins dienen der Versorgung bzw. sind für andere Funktionen reserviert (grün).

Aktor/Sensor	Funktion	Pin		Funktion	Aktor/Sensor
	3.3V	1	2	5V	BLTouch Versorgung
	GPIO 2	3	4	5V	
	GPIO 3	5	6	GND	Masseverbindung
	GPIO 4	7	8	GPIO 14	
	GND	9	10	GPIO 15	
Schrittmotor Step	GPIO 17	11	12	GPIO 18	
Schrittmotor Enable	GPIO 27	13	14	GND	BLTouch Ground
Schrittmotor Direction	GPIO 22	15	16	GPIO 23	Servo PWM
	3.3V	17	18	GPIO 24	BLTouch PWM
	GPIO 10	19	20	GND	
	GPIO 9	21	22	GPIO 25	
	GPIO 11	23	24	GPIO 28	
BLTouch Endstop	GND	25	26	GPIO 7	
BLTouch Endstop	GPIO 0	27	28	GPIO 1	
	GPIO 5	29	30	GND	
	GPIO 6	31	32	GPIO 12	
	GPIO 13	33	34	GND	HX711 Ground
	GPIO 19	35	36	GPIO 16	HX711 Clock
Schrittmotor Endstop	GPIO 26	37	38	GPIO 20	HX711 Data
Schrittmotor Endstop	GND	39	40	GPIO 21	

Die Pins sind multifunktional, das heißt, sie lassen sich bei Bedarf auch als SPI- (Serial Peripheral Interface), I²C- (Inter-Integrated Circuit) oder UART-Schnittstelle nutzen, um damit zusätzliche Hardware anzusprechen [38].

Ein Nachteil des RPI4 ist, dass er im Unterschied zu den meisten Microcontrollern keine analogen Eingänge besitzt. Das bedeutet, dass für die Erfassung analoger Signale ein externer ADC notwendig ist. Im konkreten Anwendungsfall wurde dazu auf einen HX711 (AVIA Semiconductor Ltd., Xiamen, China) zurückgegriffen, welcher das digitalisierte Sensorsignal über ein serielles Zweidraht-Protokoll an den RPI4 sendet.

Die Steuerung muss unter Anderem auch in der Lage sein, ein PWM-Signal zu generieren. Es wird zur Ansteuerung des Servomotors benötigt, mit welchem der Winkel des Reinigungskopfes verstellt wird. Mit dem RPI4 lassen sich PWM-Signale sowohl software-, als auch hardwaretechnisch generieren und an den entsprechenden Pins ausgeben [38].

USB-Schnittstellen

Zur Bedienung von *Plug and Play* Geräten stehen dem RPI4 zwei USB 3.0 und zwei USB 2.0 Schnittstellen zur Verfügung [38]. Über sie erfolgt z.B. die Kommunikation mit dem CNC-Controller des Positionierungssystems. Die USB-Schnittstellen werden außerdem dazu verwendet, um das Display zu versorgen und die Eingaben des Touchdisplays zu erfassen.

Grafik

Ein *Raspberry Pi* besitzt im Gegensatz zu Microcontrollern eine GPU (Graphics Processing Unit) [38]. Die GUI (Graphical User Interface) des Roboters lässt sich damit ohne Weiteres über HDMI (High Definition Multimedia Interface) auf einem externen Display anzeigen.

Konnektivität

Für die Anbindung an TCP/IP-Netzwerke besitzt der RPI4 sowohl eine kabelgebundene 1GbE-Ethernet Schnittstelle, als auch eine Wireless-LAN (Local Area Network) Netzwerkkarte nach dem IEEE 802.11ac Standard [38]. Die Netzwerkanbindung des Gerätes erlaubt es den AnwenderInnen, den Roboter über ein externes Geräte wie Notebooks oder PCs zu bedienen. Diese Funktionalität soll vorwiegend dazu eingesetzt werden um grundlegende Einstellungen des Reinigungsroboters zu ändern.

Betriebssystem

Auf einem *Raspberry Pi* lassen sich unterschiedliche Linux Distributionen aufsetzen. Da das Reinigungssystem über eine Benutzeroberfläche in Verbindung mit einem Touchdisplay bedient werden soll, war bei der Wahl des Betriebssystems darauf zu achten, dass dieses auch eine Desktop Umgebung mitbringt. Installiert wurde das auf Linux *Debian* basierende Betriebssystem *Raspberry Pi OS* in der Version 11 und dem GUI Toolkit GTK+3.

2.5.2 3-Achsen-Positionierungssystem

Die Realisierung des 3-Achsen-Positionierungssystems basiert auf einem open-source Hardwarebausatz. Der Bausatz entspricht eigentlich einer Graviermaschine, dessen Positionierungssystem für dieses Projekt zweckentfremdet wurde. Eingesetzt wurde das Set *Genmitsu 3018 Pro* (SaintSmart, Las Vegas, USA), welches alle nötigen mechanischen Komponenten, sowie CNC-Steuerung und Schrittmotoren inkludiert. Da diese Art Bausatz für den Heimwerkerbereich gedacht ist, sind Genauigkeit und Präzision nicht mit professionellen Positionierungssystemen vergleichbar. Die mechanische Auflösung wird vom Hersteller nicht spezifiziert, liegt aber basierend auf eigenen Berechnungen bei theoretisch 0.0025 mm, siehe dazu Berechnung (2) auf Seite 35. Da für den geplanten Einsatzzweck eine Auflösung im Bereich $\frac{1}{10}$ mm genügt, ist das System für die Positionierung des Reinigungskopfes mehr als ausreichend.

Mechanischer Grundaufbau

Den starren Grundaufbau bilden 20x20 mm breite Doppel-Aluminiumprofile welche mit Kunstharzplatten verschraubt sind. Diese Kombination sorgt für ausreichende Stabilität, welche für den weiteren Aufbau des Roboters nötig ist. Die Aluminiumprofile haben zudem den Vorteil, dass sie sehr flexibel betreffend der Montage weiterer Bauteile sind. So lassen sich die Endschalter beispielsweise, an beliebigen Stellen entlang der Profilverlängerung befestigen.

Die Grundfläche des Y-Schlittens ist als Aluminiumprofilplatte ausgeführt und bietet mit 300x180 mm ausreichend Platz, um darauf sowohl die Drehteller- als auch die Lösungsmittelfördereinheit unterzubringen.

Führungstechnik

Die Bewegungen entlang der drei Achsen erfolgt über die linear geführten X-, Y- und Z-Schlitten. Für die Positionierung der Schlitten werden Trapezgewindespindeln mit einer Steigung von 2 mm eingesetzt. Der Antrieb dieser erfolgt über Schrittmotoren. Für möglichst spielfreie Bewegungen kommen sogenannte *Anti-Backlash* Muttern

zum Einsatz. Darunter versteht man eine zweiteilige Spindelmutter, welche mit Hilfe einer Druckfeder vorgespannt werden, um das Spiel zwischen Gewindespindel und Mutter zu unterdrücken, siehe Abbildung 9, Seite 32.

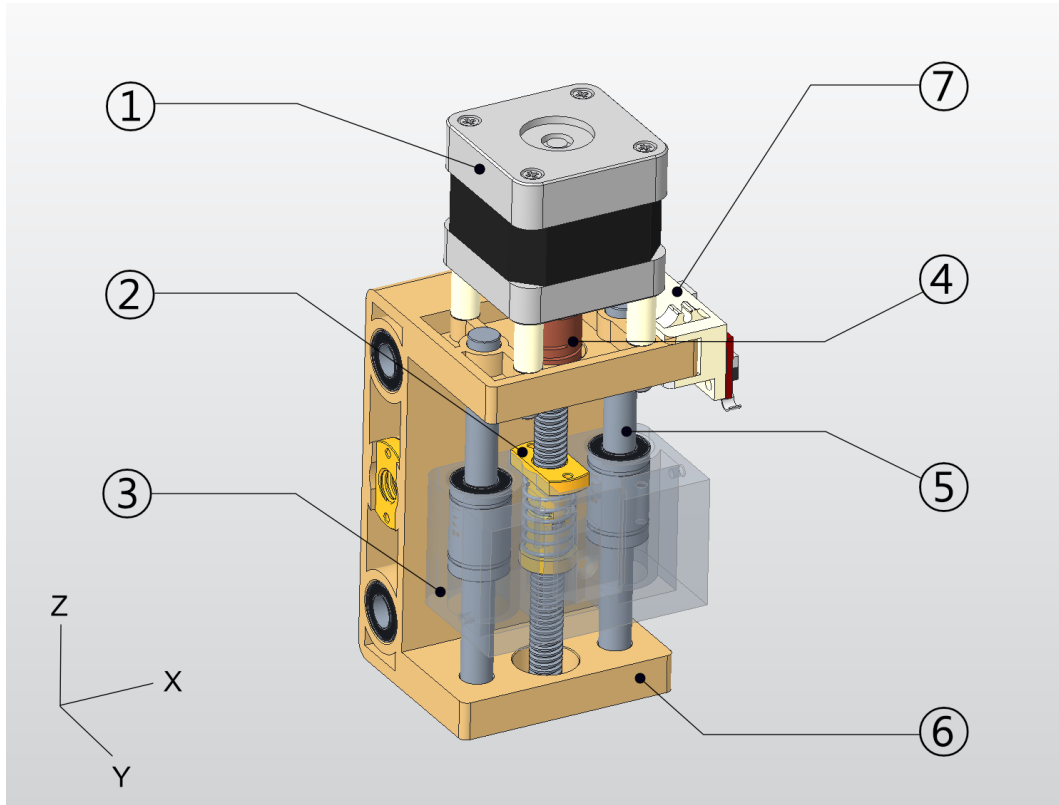


Abbildung 9: Veranschaulichung der Führungstechnik und des Anti-Backlash-Systems anhand des XZ-Schlittens. Die Federkraft eliminiert das Axialspiel zwischen Spindel und Mutter.

- (1) Schrittmotor
- (2) Anti-Backlash-System bestehend aus zwei Muttern und einer Druckfeder
- (3) Z-Schlitten mit Schwalbenschwanzaufnahme
- (4) Flexible Kupplung
- (5) Führungsstangen mit Linearkugellager
- (6) Originaler X-Schlitten
- (7) Adapter für Z-Endschalter-Platine

Die Verbindung zwischen Antriebswelle des Schrittmotors und der Trapezgewindespindel ist mit einer flexiblen Kupplung realisiert. Dadurch lässt sich ein eventueller axialer Versatz zwischen Spindel und Motorwelle geringfügig ausgleichen, was mit einer starren Kupplung nicht möglich wäre.

CNC-Controller Hardware

Die Aufgabe des CNC-Controllers ist es, die GCODE Befehle der zentralen Steuereinheit entgegenzunehmen und in Bewegungen umzusetzen. Das betrifft sowohl die Positionierung des Reinigungskopfes über die drei Schrittmotoren, als auch die Rotation des Drehtellers über den DC-Getriebemotor.

Der CNC-Controller basiert auf einem *ATmega328P* (Microchip, Chandler, USA) [40] Microcontroller. Die erforderlichen Treiberbausteine sind auf der Controller-Platine verlötet und die maximalen Spulenströme sind bereits auf die Schrittmotoren abgestimmt. Das bedeutet, dass sich die Motoren ohne zusätzliche Endstufe, also direkt über den Controller, betreiben lassen. Es ist zudem möglich bis zu sechs Endschalter (zwei je Achse) an den Controller anzuschließen.

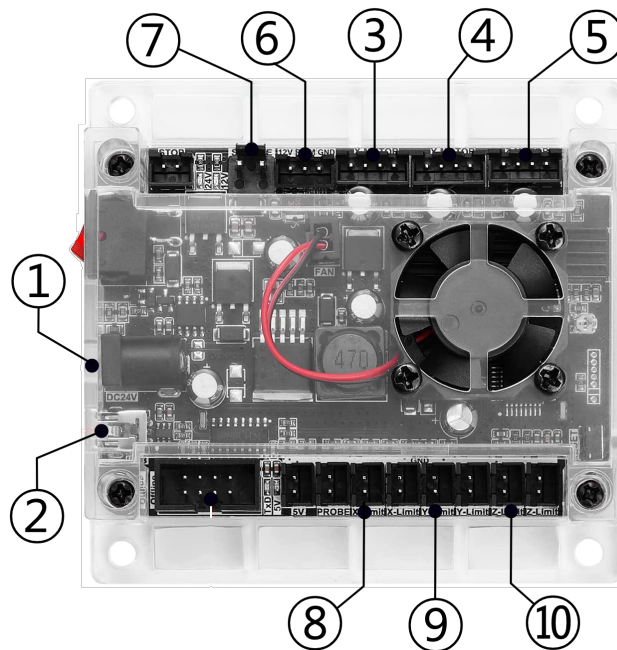


Abbildung 10: Anschlussplan des CNC-Controllerplatine, in Anlehnung an [41]

- (1) Versorgungsanschluss für ein 24 V Netzteil
- (2) USB-Anschluss für die Kommunikation mit der zentralen Steuerung
- (3) Schrittmotor X-Achse
- (4) Schrittmotor Y-Achse
- (5) Schrittmotor Z-Achse
- (6) Anschluss für Step-Down Konverter
- (7) PWM-Ausgang für DC-Getriebemotor
- (8) Endschalter X-Achse
- (9) Endschalter Y-Achse
- (10) Endschalter Z-Achse

Die Controller-Platine verfügt des Weiteren über einen dreipoligen Versorgungsanschluss (6) der ursprünglich dazu gedacht ist, einen Gravurlaser über PWM zu betreiben. Die 12 V Ausgangsspannung des Anschlusses werden in diesem Fall dazu verwendet, weitere Peripherie des Reinigungsroboters zu versorgen. Details dazu werden in Kapitel 2.5.3 erläutert.

Endschalter

Die Endschalter sind kurz vor dem mechanischen Anschlag der jeweiligen Achse montiert und haben zwei Aufgaben: Einerseits verhindern sie, dass der mechanischen Anschlag erreicht wird und andererseits dienen sie als Referenz.

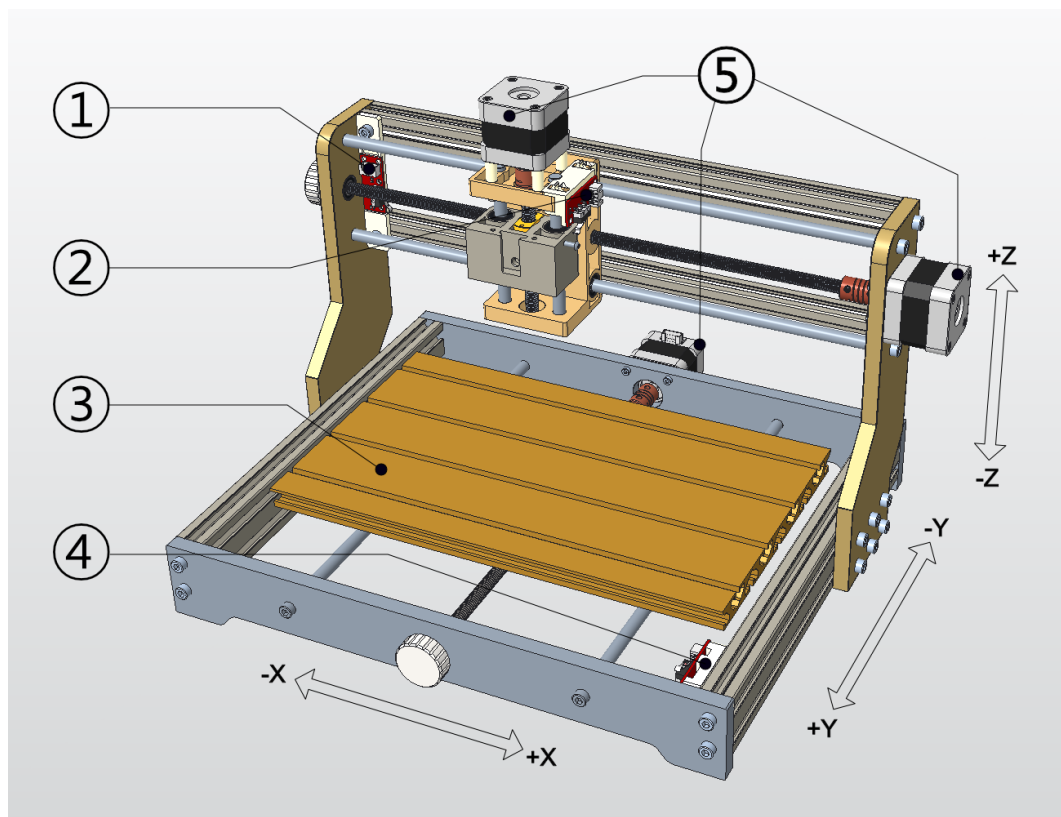


Abbildung 11: Mit Endschalter modifiziertes Positionierungssystem

- (1) Endschalter der X-Achse
- (2) Endschalter der Z-Achse
- (3) Aluminiumprofilplatte (Z-Schlitten)
- (4) Endschalter Y-Achse
- (5) X-, Y- und Z-Schrittmotor

Eine Referenz ist deshalb wichtig, da es sich bei der Schrittmotorsteuerung um einen offenen Regelkreis handelt. Das bedeutet, dem CNC-Controller wird nicht rückgemeldet in welcher absoluten Position sich die Schlitten befinden. Die Positionsbestimmung erfolgt ausschließlich anhand der gefahrenen Schritte. Für eine absolute Positionierung der Schlitten ist deshalb die Definition des Maschinenursprungs nötig. Das erfordert zumindest einen Endschalter je Achse. Die jeweils zweite Extremposition lässt sich über den maximalen Verfahrweg der jeweiligen Achse definieren, siehe Tabelle 10, Seite 37.

Mechanische Auflösung

Bei den Antriebsmotoren der Achsen handelt es sich um gewöhnliche NEMA17 (National Electrical Manufacturers Association) bipolar Schrittmotoren. Die Standardschrittweite der Motoren liegt bei 1.8° . Für eine volle Umdrehung sind somit 200 Schritte notwendig [42]. Um die Positionsgenauigkeit zu erhöhen, werden die Motoren im *Viertel-Schritt-Modus* betrieben. Das bedeutet, dass die ursprünglichen 200 Schritte/Umdrehung auf 800 Schritte/Umdrehung „aufgeblasen“ werden.

$$d_{mech} = \frac{\text{Spindelsteigung}}{\text{Schritte pro Umdrehung}} = \frac{2 \frac{\text{mm}}{\text{Umdrehung}}}{800 \frac{\text{Schritte}}{\text{Umdrehung}}} = 0.0025 \frac{\text{mm}}{\text{Schritt}} \quad (2)$$

Berücksichtigt man die Spindelsteigung von 2 mm pro Umdrehung, so ergibt sich laut Berechnung (2) eine theoretische mechanische Auflösung von $d_{mech}=0.0025$ mm. Dieser Wert ist aber auf Grund der Fertigungstoleranzen anderer mechanischer Komponenten praktisch nicht erreichbar.

CNC-Controller Firmware

Bei der Firmware des CNC-Controllers handelt es sich um das open-source Softwareprojekt *GRBL* in der Version 1.1f. Sie nimmt Instruktionen in Form eines Zeichenstrings über die UART-Schnittstelle entgegen und verarbeitet diese weiter. Für den Verbindungsaufbau sind die in Tabelle 9 ersichtlichen Parameter erforderlich.

Tabelle 9: Parameter für den Verbindungsaufbau mit dem GRBL-basierten CNC-Controller [43].

Parameter	Wert
Baudrate	115200
Datenbits	8
Stopbits	1
Parität	Keine

Damit die Zeichenkette vom CNC-Controller interpretiert werden kann, muss diese einem bestimmten Format entsprechen. Jeder vom RPI4 gesendete String ist deshalb mit den beiden Steuerzeichen *CR* (Carriage Return bzw. '\r') und *LF* (Linefeed bzw. '\n') zu terminieren:

`<Befehl in Form einer Zeichenkette>\r\n`

Wird in der Zeichenkette ein anwendbarer Befehl erkannt, so antwortet der Controller mit einem „ok“ bzw. den angeforderten Daten. Nicht erkannte Befehle werden ohne Rückmeldung verworfen.

Über das eben beschriebene Befehlsformat lassen sich nicht nur Instruktionen in Form von GCODEs ausführen, sondern auch Informationen über den Gerätestatus auslesen und Grundeinstellungen ändern.

Anpassung der Grundeinstellungen

Letzteres ist vor der erstmaligen Inbetriebnahme durchzuführen, da einige Einstellungen an die Bedürfnisse des Reinigungsroboters anzupassen sind. Dazu zählt neben der Aktivierung des Homingzyklus und der Software-Endschalter auch die Invertierung der X-Achse sowie die Hinterlegung der maximalen Verfahrswege und Geschwindigkeiten aller Achsen. Die aktuell gespeicherten Grundeinstellungen lassen sich über den Befehl '\$\$' auslesen [43]. Der Controller antwortet darauf mit einer Liste an Einstellungen sowie den zugehörigen Werten.

Tabelle 10: Auszug aller geänderten Grundeinstellungen der GRBL-Firmware. Bei allen nicht angeführten Parametern wurde der Standardwert beibehalten.

Parameter	Wert	Beschreibung
\$20	1	Aktiviert die Software-Endschalter
\$22	1	Aktivierung des Homing-Zyklus
\$23	1	Invertiert die Richtung der X-Achse
\$100	800	Setzt die Schritte/mm für die X-Achse
\$101	800	Setzt die Schritte/mm für die Y-Achse
\$102	800	Setzt die Schritte/mm für die Z-Achse
\$110	1500	Max. Vorschubgeschwindigkeit der X-Achse in mm/min
\$111	1500	Max. Vorschubgeschwindigkeit der Y-Achse in mm/min
\$112	600	Max. Vorschubgeschwindigkeit der Z-Achse in mm/min
\$130	200	Max. Verfahrweg des X-Schlittens in mm
\$131	75	Max. Verfahrweg des Y-Schlittens in mm
\$132	36	Max. Verfahrweg des Z-Schlittens in mm

Der maximale Verfahrweg des Y-Schlittens lässt sich z.B. überschreiben, indem man den Befehl '\$131=<Wert>' an den Controller sendet. Alle Änderungen der Grundeinstellungen werden in den nichtflüchtigen EEPROM (Electrically Erasable Programmable Read-Only Memory) des Controllers geschrieben und sind daher nur einmalig durchzuführen.

GCODE Instruktionen

Für eine Bewegung der Schlitten, müssen Instruktionen in Form von GCODEs an den Controller gesendet werden. Ein Instruktion sieht beispielsweise wie folgt aus:

```
1: G91
2: G01 X10 Y-7.5 Z-1.73 F500
3: G90
4: G00 X1 Y1 Z1
5: M03 S100
6: M05
```

Im obigen Beispiel wird mit 'G91' zuerst der Positionierungsmodus auf *relativ* gestellt [43]. Die darauf folgende Zeile lässt die Schlitten 10 mm in X-Richtung, -7.5 mm in der Y-Richtung und -1.73 mm in Z-Richtung fahren, unabhängig davon, in

welcher Position sie sich gerade befinden. Mit dem Parameter 'F' lässt sich der Vorschub definieren [43], im aktuellen Beispiel wird dieser auf 500 mm/min eingestellt. G-Parameter kennzeichnen den Bewegungsmodus. 'G01' entspricht dabei einer linearen Bewegung, während mit 'G00' im Eilgang zwischen zwei Punkten verfahren wird [43]. In Zeile 3 und Zeile 4 wird auf die *absolute* Positionierung gewechselt und der Punkt (1/1/1) mm angefahren.

Absolutpositionierungen erfordern, dass zuvor ein *Homing-Zyklus* durchgeführt wurde. Unter dem Homing-Zyklus versteht man eine Referenzfahrt, bei welcher die Endschalter aller Achsen abgefahren werden. Die Referenzfahrt muss bei jedem Neustart des CNC-Controllers durchgeführt werden.

In Zeile 5 und 6 wird mittels Parameter 'M' der Antriebsmotor des Drehtellers aktiviert bzw. deaktiviert. 'S100' steht dabei für die Drehzahl des Motors in min^{-1} . Das Instruktionsformat ist sehr flexibel. Der Controller unterscheidet nicht zwischen Groß- und Kleinschreibung und die Reihenfolge der Koordinaten ist beliebig wählbar. Wird kein F-Parameter mit übergeben, so wird der letzte Wert herangezogen. Bei negativen Werten lässt sich das Minus auch vor der Achsenbezeichnung setzen.

Echtzeitbefehle

Weitere, für den Reinigungsroboter essenzielle Befehle, sind die Zeichen '?', '!', '~' und '\030'. Es handelt sich dabei um Echtzeitbefehle, die den CNC-Controller veranlassen, sofort darauf zu reagieren [43].

Sendet man z.B. ein '?', so antwortet der CNC-Controller mit seinem aktuellen Status (*Idle*, *Run*, *Alarm*,...) und seiner gegenwärtigen Maschinenposition in Form von XYZ-Koordinaten [43]. Die zentrale Steuerung fragt Status und Koordinaten alle 20 ms ab. Das kurze Intervall ist nötig, da die Steuerung zu jedem Zeitpunkt wissen muss, wo sich die Schlitten befinden bzw., ob ein Problem mit dem Positionierungssystem vorliegt. Das ist vor allem wichtig, wenn es um die Abtastung der Optikoberflächen geht.

Ein '!' pausiert den aktuellen Positionierungsvorgang [43] indem alle Schrittmotoren gestoppt werden. Bei diesem Vorgang geht die aktuelle Position der Schlitten nicht verloren. Zu beachten ist, dass der Befehl keinen Einfluss auf die Bewegung des Drehtellers hat - dieser rotiert ungeachtet davon weiter. Der Befehl '~' setzt den Positionierungsvorgang wieder fort [43].

Unter Umständen ist es notwendig ein *Softreset* des CNC-Controller durchzuführen. Dazu muss das Steuerzeichen 'STRG+X' an den Controller gesendet werden [43]. Das ist z.B. notwendig wenn der Controller die Position eines Schlittens verliert.

Fixpositionen

Grundsätzlich erfolgt die Verstellung der Schlitten über relative Positionierung. Dazu sind fixe Maschinenpunkte nötig, auf welche sich die relativen Bewegungen beziehen. In den Grundeinstellungen der Robotersoftware wurden dafür drei Fixpositionen in Form von absoluten Koordinaten hinterlegt.

Tabelle 11: Koordinaten der Fixpositionen

Achse Position	X in mm	Y in mm	Z in mm
Beladeposition	152	0	0
Startposition	152	-69	0
Abtastposition	62	-67	0

Beladeposition: Diese Position wird angefahren, wenn eine Optik in den Reinigungsroboter eingelegt bzw. wieder entnommen werden soll. Der Y-Schlitten positioniert sich dabei in der Position ' $Y=0$ ', in welcher die Optikaufnahme am besten zugänglich ist, siehe Abbildung 12.

Startposition: In dieser Position befindet sich der Kopf des Reinigungstäbchens (Polyester-Pad) zentral über der Optik. Die Position dient als Bezugspunkt für die, während der Reinigung ausgeführten, relativen Bewegungen von X- und Z-Schlitten.

Abtastposition: Die Abtastposition dient als Startpunkt der Profilaufnahme, die Sensorsonde und die optische Achse stimmen in dieser Position überein.

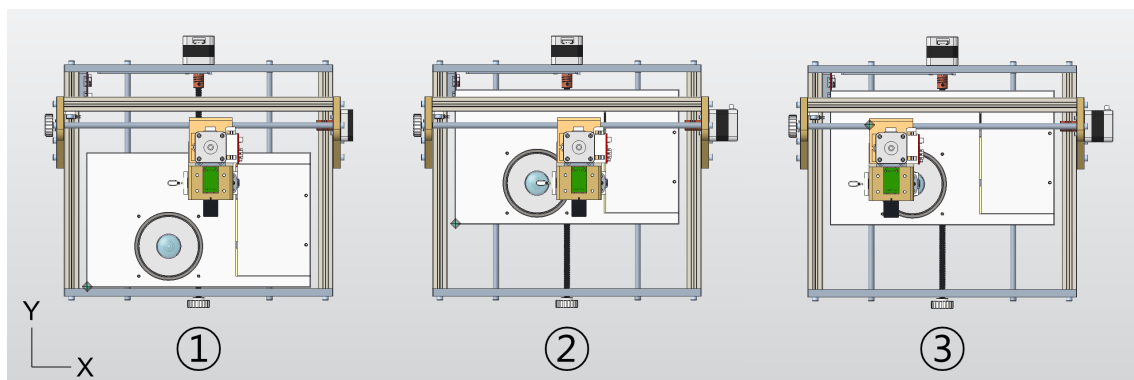


Abbildung 12: Darstellung von Belade- (1), Start- (2) und Abtastposition (3)

2.5.3 Drehtellereinheit

Aus mechanischer Sicht stellte die Konstruktion der Drehtellereinheit die größte Herausforderungen des Projektes dar. Zum einen musste bei der Entwicklung berücksichtigt werden, dass der Z-Schlitten, auf welchem der Drehteller befestigt wird, nur eine begrenzte Fläche bietet. Auf der anderen Seite war darauf zu achten, dass sich die, durch die drehenden Teile entstehende Geräuschentwicklung, in Grenzen hält.

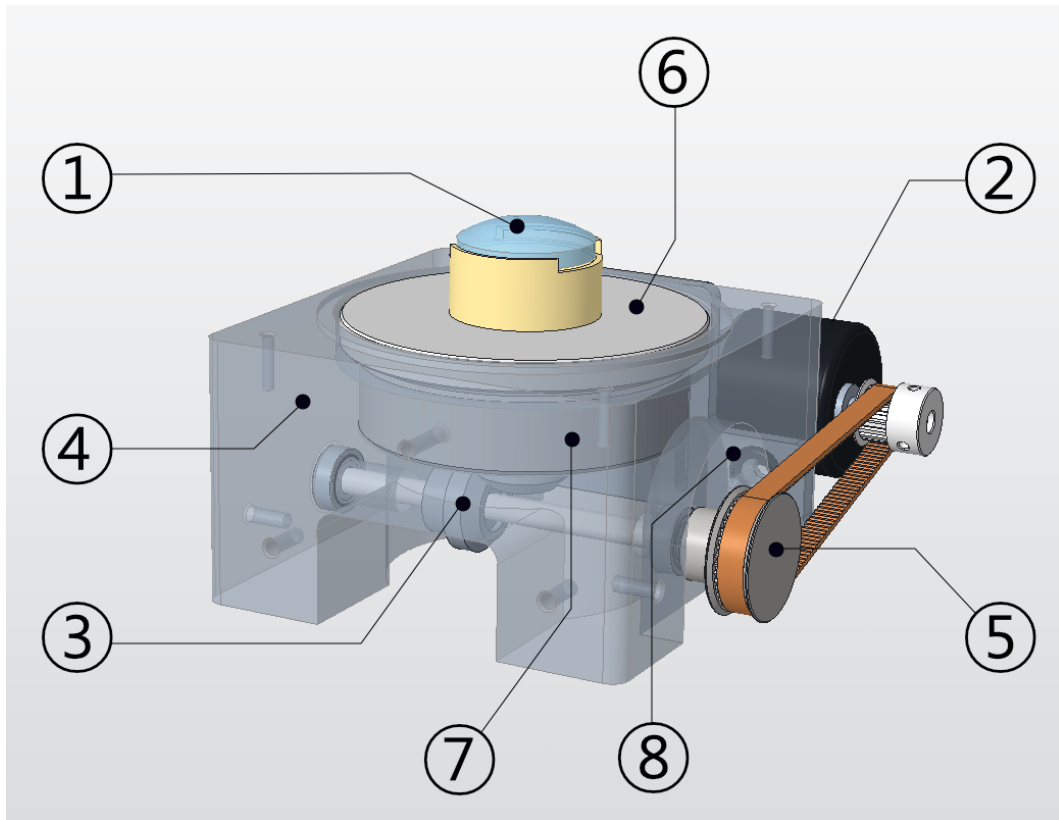


Abbildung 13: Mechanik der Drehtellereinheit. Zur besseren Veranschaulichung des 90 ° Kegelgetriebes, ist der Getriebeblock leicht transparent dargestellt.

- (1) Optikaufnahme mit eingesetzter Linse
- (2) DC-Getriebemotor mit Synchronscheibe GT2-16T
- (3) Getriebewelle mit geklemmten Kegelrad
- (4) Getriebeblock
- (5) Synchronscheibe GT2-40T
- (6) Drehteller mit verschraubtem Kegelrad
- (7) Hauptlager
- (8) Gummibuffer

Mechanik

Der Getriebeblock bildet die Basis der Drehtellereinheit. In ihm sitzt das radiale Hauptlager, über welches der Drehteller gelagert ist. Die Differenz zwischen Außendurchmesser des Hauptlagers und dem Außendurchmesser des Drehtellers, wurde bewusst klein gewählt, denn dadurch rotiert der Drehteller ruhiger. Das massive Rillenkugellager sorgt zudem für mehr Gewicht, welches den im Verhältnis leichten, aus *Onyx* gedruckten Getriebeblock stabilisiert.

Wie in Abbildung 13, Seite 40 zu sehen, wird der Antrieb zweimal untersetzt. Der Grund liegt einerseits darin, die Drehzahl etwas zu reduzieren und damit gleichzeitig das Drehmoment zu erhöhen. Andererseits wird die Baugruppe dadurch klein und kompakt zu gehalten.

Das Kegelgetriebe bezweckt die 90° Umsetzung der Drehbewegung. In diesem Fall wird weder über- noch untersetzt. Die beiden Kegelräder sind aus POM (Polyoxymethylen) gefertigt. Dieser Werkstoff wurde gewählt, um die Laufgeräusche besonders bei Drehzahlen $> 200 \text{ min}^{-1}$ etwas zu reduzieren. Für den Zahnriementrieb zwischen Motorwelle und Welle des Kegelgetriebes kommen Synchronscheiben und Riemen des GT2-Standards zum Einsatz. Untersetzt wird von 16 auf 40 Zähne.

Neben dem umgesetzten Konzept, wurde auch ein direkter Antrieb des Drehtellers bzw. eine direkte Kopplung der Antriebswelle des Motors mit der Getriebewelle in Betracht gezogen. Beide Varianten waren aber aufgrund einer zu hohen Geräuschentwicklung bzw. Platzmangels am Y-Schlitten nicht zielführend.

Optikaufnahme

Aufgrund der Vielfalt von Optikoberflächen und -durchmesser, ist eine universelle Aufnahme nicht sinnvoll. Die bessere Lösung sind in diesem Fall an die Optik angepasste Adapter, welche sich über eine Schraubverbindung an den Drehteller befestigen lassen. Die Adapter sind so konstruiert, dass die Optik über eine Pressverbindung in der Aufnahme gehalten wird. Das macht den Wechsel zwischen den Optiken gleichen Typs einfacher.

Drehzahlregelung

Der Antrieb des Drehtellers erfolgt mittels Gleichstrom-Getriebemotor, welcher sich mit einer Spannung von bis zu 15 V speisen lässt und dabei eine Leistung von max. 19.8 W umsetzen kann [44].

Die Regelung der Drehzahl passiert über den CNC-Controller, welcher eine pulsweitenmodulierte Rechteckspannung erzeugt und am Anschluss (7) (siehe Abbildung 10, Seite 33) ausgibt. Durch Variation des Verhältnisses zwischen Ein- und Auszeit

der Spannung kommt es zu unterschiedlichen Spannungsmittelwerten [45] an den Motorklemmen. Die Drehzahl des Motors wiederum verhält sich proportional zum angelegten Spannungsmittelwert [46].

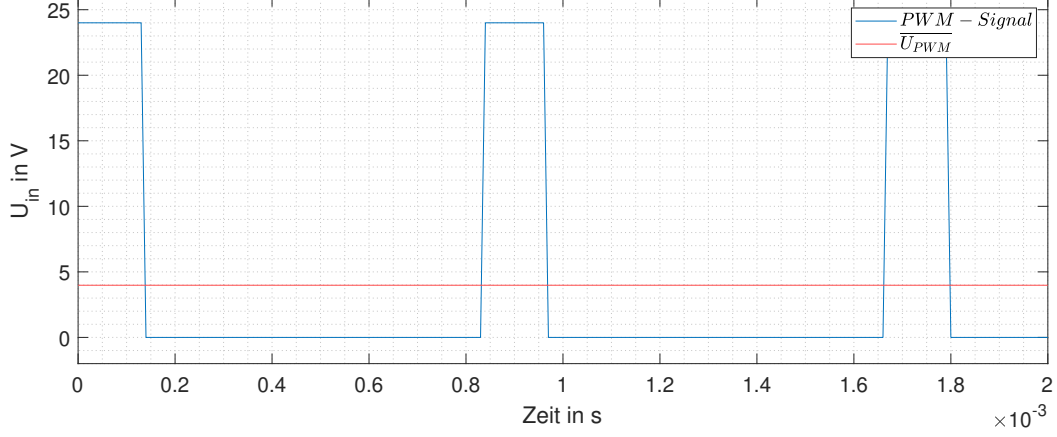


Abbildung 14: Beispiel für ein durch den CNC-Controller erzeugtes PWM-Signal.

Das folgende Rechenbeispiel bezieht sich auf das Signal aus Abbildung 14 und erläutert den Zusammenhang zwischen Pulsweite und Drehzahl des Motors bei einer Versorgungsspannung von $U_{in}=24\text{ V}$, einer PWM-Frequenz von $1204\frac{1}{s}$ und einer Einzeit von $t_{ein}=137.6\text{ }\mu s$:

$$\begin{aligned}\overline{U_{PWM}} &= U_{in} \cdot \frac{t_{ein}}{t_{ein} + t_{aus}} = U_{in} \cdot t_{ein} \cdot f_{PWM} = \\ &= 24V \cdot 137.6 \cdot 10^{-6}s \cdot 1204\frac{1}{s} = 3.98V\end{aligned}\quad (3)$$

Laut Datenblatt [44] erreicht der Getriebemotor bei einer Versorgungsspannung von 15 V eine Drehzahl von $1648\frac{1}{min}$. Die Angabe bezieht sich dabei auf die bereits im Verhältnis von 1:10 unteretzte Drehzahl des Planetengetriebes. Über diesen Zusammenhang k lässt sich die theoretische Drehzahl an der Getriebewelle n_{GW} wie folgt berechnen:

$$n_{GW} = k \cdot \overline{U_{PWM}} = \frac{1648\frac{1}{min}}{15V} \cdot 3.98V = 437.27\frac{1}{min}\quad (4)$$

Für die Ermittlung der Drehteller-Drehzahl muss nun noch die Untersetzung des Zahnriemenantriebes von $z_{GW}=16$ auf $z_{DT}=40$ Zähne über folgendes Verhältnis berücksichtigt werden:

$$\frac{z_{GW}}{z_{DT}} = \frac{n_{DT}}{n_{GW}} \quad (5)$$

$$n_{DT} = n_{GW} \cdot \frac{z_{GW}}{z_{DT}} = 437.27 \frac{1}{min} \cdot \frac{16}{40} = 174.91 \frac{1}{min}$$

Das Ergebnis aus (3) bis (5) zeigt, dass bei einer Pulsweite von $137.6 \mu s$ eine theoretische Rotationsgeschwindigkeit von 174.91 min^{-1} zu erwarten ist.

2.5.4 Reinigungskopf

Diese Einheit stellt die komplexeste Baugruppe des Reinigungssystems dar, da sie unterschiedliche Funktionen vereint. Über die verbaute Wägezelle, wird der Druck gemessen welcher auf die Optikoberfläche einwirkt. Ein Servomotor ermöglicht es, den Winkel des Reinigungskopfes an die Krümmung der Optik anzupassen und ein integrierter Nivellierungssensor tastet die Oberfläche der eingelegten Optik ab und erstellt daraus ein Oberflächenprofil.

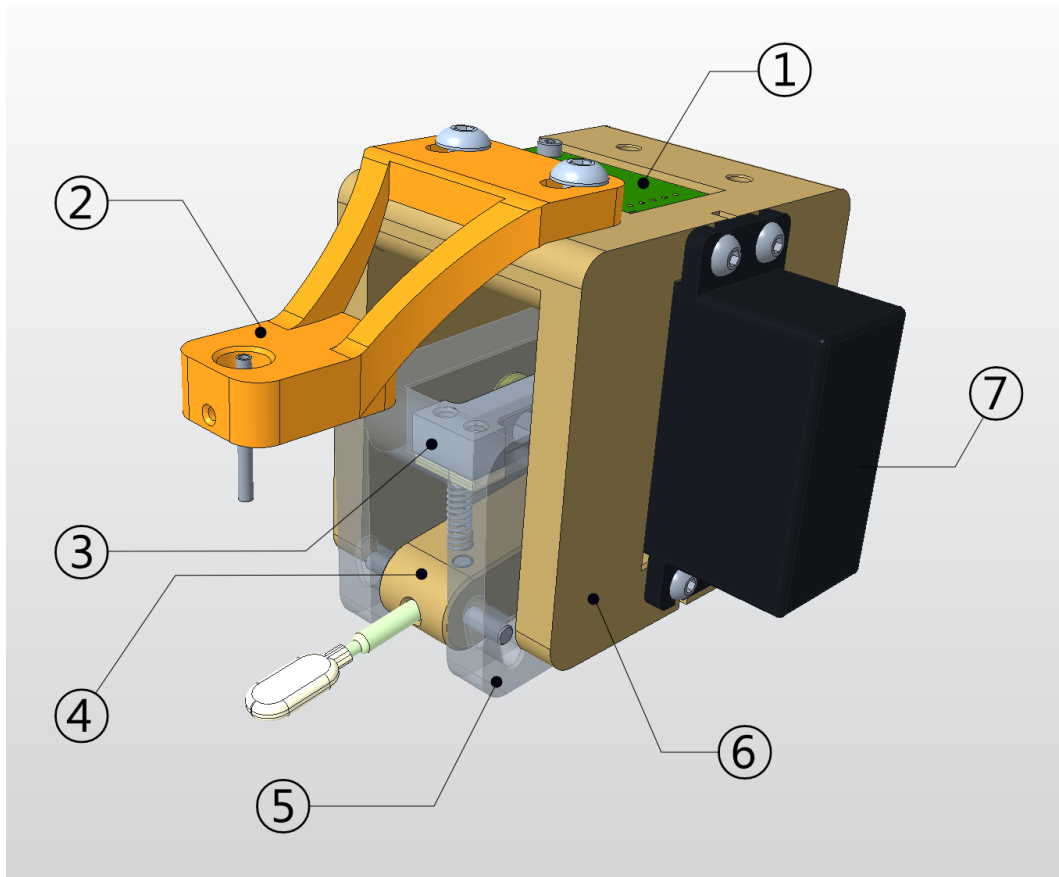


Abbildung 15: Mechanischer Aufbau des Reinigungskopfes. Der in dieser Ansicht unsichtbare Nivellierungssensor ist auf der rechten Seite des Sensorhalters montiert.

- (1) HX711 Breakout Board
- (2) Halterung mit Lösungsmitteldüse
- (3) Druckfedergepufferte Wägezelle
- (4) Gelagerte Aufnahme mit eingesetztem Reinigungsstäbchen
- (5) Sensorhalter mit oberem und unterem Anschlag zur Begrenzung der Aufnahme
- (6) Kopfbasis mit Schwalbenschwanzaufnahme (nicht sichtbar)
- (7) Servomotor

Kraftmessung

Die Kraftmessung erfolgt mithilfe einer Druckfeder-gepufferten Wägezelle, welche für eine maximale Belastung von 200 g ausgelegt ist. Durch die kontinuierliche Erfassung der Messwerte, lässt sich der zuvor in der GUI eingestellte Anpressdruck, konstant halten. Das soll einerseits eine gleichbleibende Reinigungsqualität gewährleisten und andererseits verhindern, dass es zu Beschädigungen der empfindlichen Beschichtungen kommt.

Die eingesetzte Wägezelle besteht aus insgesamt vier DMS (Dehnmessstreifen), welche zu einer *Wheatstone'sche Brückenschaltung* zusammengeschaltet sind. Das analoge Ausgangssignal der Messbrücke wird über einen externen 24-Bit ADC (HX711) verstärkt und digitalisiert. Die darauf folgende Auswertung des digitalen Spannungssignales erfolgt durch die zentrale Steuereinheit. Abbildung 16 zeigt den Zusammenhang der eben genannten Komponenten.

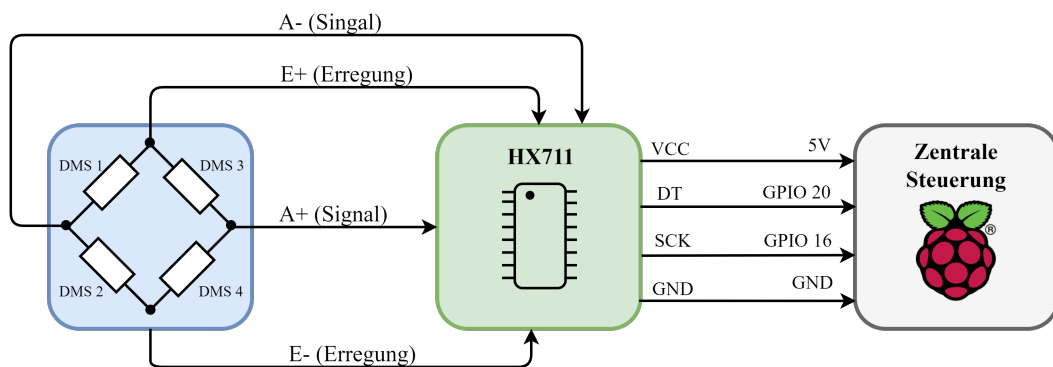


Abbildung 16: Blockdiagramm der Kraftmessung mittels Wägezelle. Die Erregungsdrähte der Brücke sind mit E+ und E- gekennzeichnet. Über die Anschlüsse A+ und A- wird ein analoge Spannungssignal im Millivolt-Bereich abgenommen. Die Übertragung des digitalisierten Signales erfolgt über einen Zweidraht-Bus (DT und SCK).

Der HX711 unterstützt zwei unterschiedliche Abtastraten. Erfordert die Anwendung genaue Messergebnisse, so liefert der IC max. 10 SPS (Samples Per Second). Für eine höhere Samplingrate muss die externe Beschaltung geändert werden. Konkret ist der Logiklevel von Pin15 (RATE) von 'LOW' auf 'HIGH' zu ziehen. Mit dieser Konfiguration ist, unter Einbußen der Genauigkeit, eine Abtastfrequenz von bis zu 80 SPS möglich. [47]

Im vorliegenden Anwendungsfall liegt die Priorität in einer schnelleren Abtaste. Da der verwendete Typ des HX711 *Breakout Boards* standardmäßig auf 10 SPS eingestellt ist, war eine Modifikation der Hardware nötig. Dazu wurde die bestehende Pin15 $\leftrightarrow$ GND Leiterbahn auf der Platine gekappt und eine Verbindung zwischen Pin15 und dem Versorgungspin V_{CC} (Voltage at the Common Collector) gelegt.

Berechnung des Maximaldrucks

Wie in Abbildung 15, Seite 44 ersichtlich, ist die Verstellung der Aufnahme durch einen unteren, sowie einem oberen mechanischen Anschlag begrenzt. Diese Maßnahme dient dem Schutz der Wägezelle, da diese bei Überschreiten der maximalen Belastbarkeit irreversibel überdehnt wird.

Befindet sich die Aufnahme am oberen Anschlag, entspricht die auf den Sensor einwirkende Kraft einer Belastung von 140.5 g. Diese Masse stellt auch gleichzeitig die maximale Belastung dar, welche auf die Oberfläche einer Optik einwirken kann. Aufgrund der Druckfeder, wirkt auch im unbelasteten Zustand des Reinigungskopfes permanent eine Masse von 32.69 g auf die Wägezelle. Berücksichtigt man die Fläche des Polyester-Pads mit 14x6.8 mm (Reinigungsstäbchen), so ergibt sich folgender maximaler Druck, welcher auf die Optik einwirken kann:

$$\begin{aligned}
 p &= \frac{F}{A} = \frac{m \cdot G}{l_{pad} \cdot b_{pad}} = \frac{140.5 \cdot 10^{-3} kg \cdot 9.81 \frac{m}{s^2}}{14 \cdot 10^{-3} m \cdot 6.8 \cdot 10^{-3} m} = \\
 &= 14.478 \cdot 10^3 \frac{N}{m^2} = 1.4478 \frac{N}{cm^2}
 \end{aligned} \tag{6}$$

Druckreinigungsprozess

Der ideale Reinigungsdruck hängt im Grunde vom Verschmutzungsgrad und der Oberflächenbeschichtung ab. Um diese beiden Faktoren zu berücksichtigen, lässt sich der Reinigungsdruck für jede Optik individuell festlegen. Die Herausforderung dabei ist, dass der eingestellte Druck auch bei nicht-planen Flächen konstant gehalten werden muss. Für die Umsetzung ist also eine Regelung des Drucks erforderlich. Der Druck wird aufgebaut, indem der Roboter den kompletten Reinigungskopf über die Verstellung des Z-Schlittens senkt. Das Polyester-Pad wird dadurch mehr und mehr auf die rotierende Optik gepresst. Ist ein zuvor definierte Druck-Schwellwert erreicht, stoppt die Z-Verstellung. Anschließend bewegt der Roboter den Reinigungskopf in +0.05 mm Schritten entlang der X-Achse. Handelt es sich bei der eingelegten Optik um eine konvexe Oberfläche, wird nach jeder X-Verstellung geprüft, ob der Druck unter die eingestellten Schwelle abgefallen ist. Ist dies der Fall, so wird der

Kopf zusätzlich um -0.05 mm gesenkt, um den Druck wieder aufzubauen. Dieser Ablauf wiederholt sich so lange, bis der Roboter den gesamten Optikradius abgefahren ist.

Im Gegensatz dazu, wird bei konkav-gekrümmten Optiken permanent ermittelt, ob der aktuelle Druck unter dem eingestellten Schwellwert liegt und ggf. nachgeregelt, indem der Reinigungskopf um +0.05 mm angehoben wird.

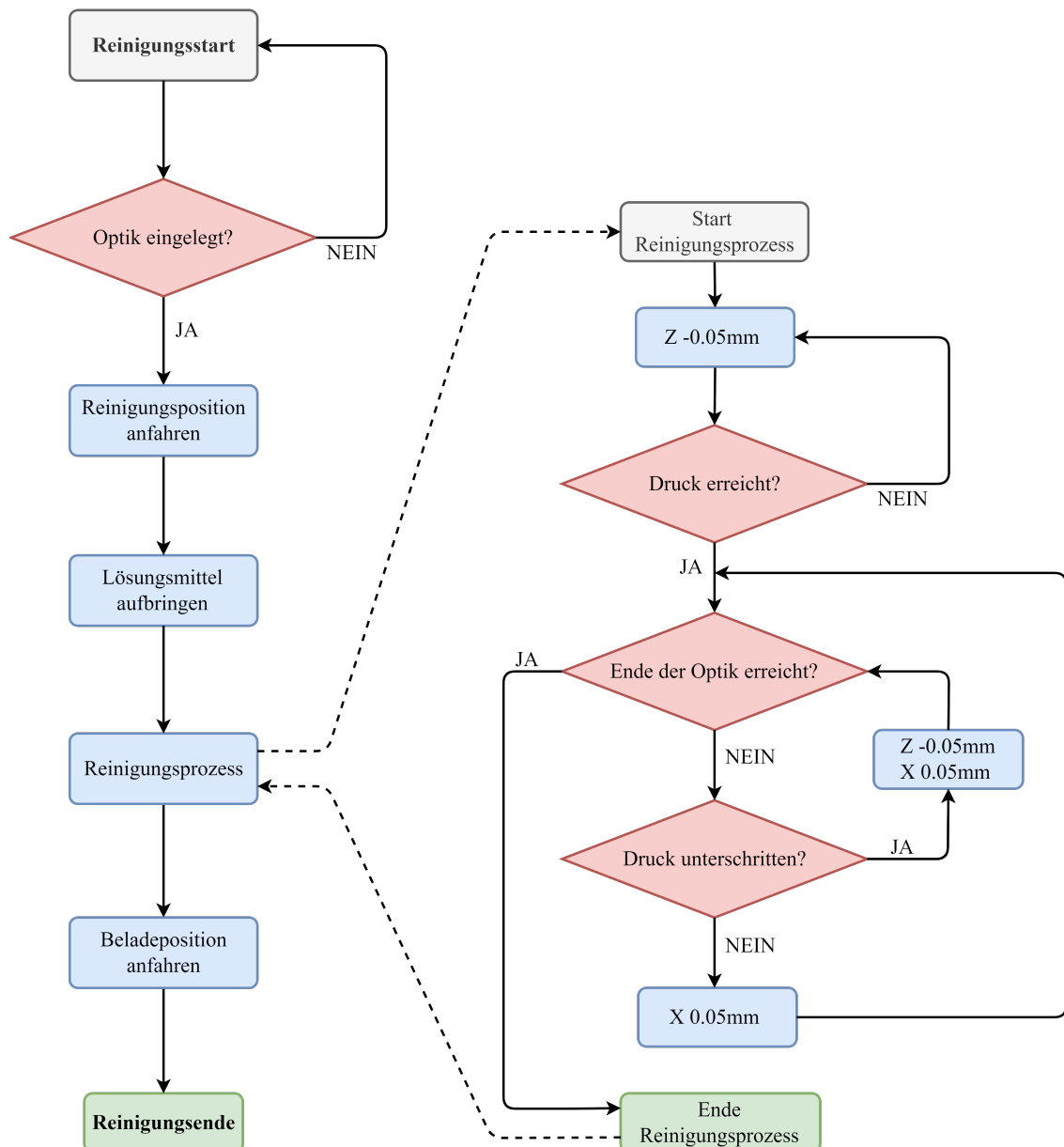


Abbildung 17: Das linke Flussdiagramm zeigt den groben Programmbau während einer Reinigung. Rechts wird der detaillierte Reinigungsprozess, ferner die Druckregelung einer konvexen Optik dargestellt.

Kopfverstellung

Die unterschiedlich gekrümmten Optikoberflächen erfordern es, dass sich der Reinigungskopf an diese anpassen lässt. Der Sensorhalter ist deshalb drehbar gelagert und kann in einem Winkelbereich von -20° bis 30° positioniert werden. Die Verstellung des Winkels erfolgt über einen gewöhnlichen Modellbau-Servomotor. Damit sich der Winkel auch unter Belastung nicht ändert, muss das Haltemoment des Servos größer als das Drehmoment sein, welches über das Reinigungsstäbchen aufgebracht wird. Der verbaute Servomotor *SPT4412LV* (SPT-Servo, Shantou, China) erreicht bei einer Versorgung mit 4.8 V ein maximales Haltemoment von 10 kgf·cm [48]. Der auf die Motorwelle wirkende Hebel (Drehachse bis Spitze des Polyester-Pads) hat eine Länge von 6.5 cm und die maximal aufbringbare Masse (begrenzt durch den obereren Anschlag am Sensorhalters) liegt bei 140.5 g. Daraus ergibt sich bei Vernachlässigung der Schwerkraft folgende Momenten-Ungleichung:

$$M_{Servo} > M_{Sensorhalter} \quad (7)$$

$$M_{Servo} > m_{max} \cdot l_{Hebel}$$

$$10kgf \cdot cm > 140.5 \cdot 10^{-3}kgf \cdot 6.5cm$$

$$10kgf \cdot cm > 0.913kgf \cdot cm$$

Aus Ungleichung (7) geht hervor, dass das Haltemoment des Servomotors im Vergleich zum maximal zu erwartenden Drehmoment in etwa um den Faktor 10 höher liegt. Daraus folgt, dass sich der ausgewählte Servomotor kräftetechnisch für die vorgesehene Anwendung eignet.

Die elektrische Ansteuerung erfolgt ähnlich wie beim DC-Getriebemotor über ein PWM-Signal (siehe Kapitel 2.5.3, Drehzahlregelung), mit dem Unterschied, dass Pulsweite und Periodendauer des Signales vom Hersteller vorgegeben sind. Die notwendige PWM-Frequenz wird mit 50 Hz angegeben, was einer Periodendauer von 20 ms entspricht [48]. Durch Variation der Pulsweite, kann der Stellwinkel des Servos und damit die Position des Sensorhalters geändert werden.

Um eine mechanische Kollision zwischen Sensorhalter und der Kopfbasis zu vermeiden, wurde eine obere sowie eine untere Maximalposition definiert. Alle Positionen außerhalb dieses Bereiches werden softwaremäßig ignoriert, um Beschädigungen des Reinigungsroboters zu vermeiden.

Tabelle 12: Zusammenhang zwischen Pulsweite und Winkel des Sensorhalters. Die Ermittlung der Winkel erfolgte über das 3D-Modell des Roboters.

Bezeichnung	Pulsweite t_{ein} in μs	Winkel in $^{\circ}$
Maximalposition	1900	30
Benetzposition	1880	28.8
Normalposition	1540	0
Minimalposition	1300	-20

Wie in Tabelle 12 ersichtlich, gibt es neben den der Minimal- und Maximalposition noch zwei weitere wichtige Servostellungen: die Normalposition und die Benetzposition. Die Normalposition entspricht jener Lage, bei welcher der Sensorhalter parallel zur Kopfbasis ausgerichtet ist. Sie wird immer dann eingenommen, wenn ein Reinigungsvorgang abgeschlossen ist, bzw. der Roboter initialisiert wird. In dieser Position steht das Reinigungsstäbchen in einem Winkel von 15° zur horizontalen Ebene.

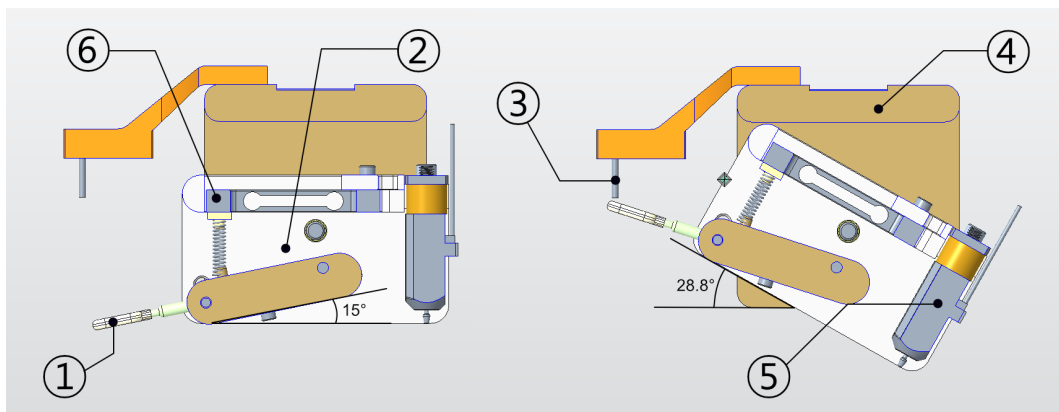


Abbildung 18: Schnitt des Reinigungskopfes in Normalposition (links) und während dem Benetzen des Polyester-Pads (rechts)

- (1) Reinigungsstäbchen mit Polyester-Pad
- (2) Sensorhalter
- (3) Düse
- (4) Kopfbasis
- (5) Nivellierungssensor
- (6) Wägezelle

Die Benetzposition wird eingenommen, um das Polyester-Pad des Reinigungsstäbchens mit Lösungsmittel zu befeuchten. Der Abstand zwischen Düse und Polyester-Pad

wurde so gewählt, dass diese sich gerade nicht berühren. Damit wird verhindert, dass sich unter der Düse ein Tropfen bildet, dessen Oberflächenspannung das Einziehen des Lösungsmittels in das Polyester-Pad erschwert.

XZ-Korrektur der Startposition

Jede, von der Normalposition abweichende Stellung des Reinigungskopfes führt dazu, dass sich auch die absolute X- und Z-Koordinate des Polyester-Pads ändern. Das hat zur Folge, dass das Pad während der Reinigung nicht zentral auf die Oberfläche der Optik aufsetzt, deshalb ist eine Korrektur der Startposition erforderlich.

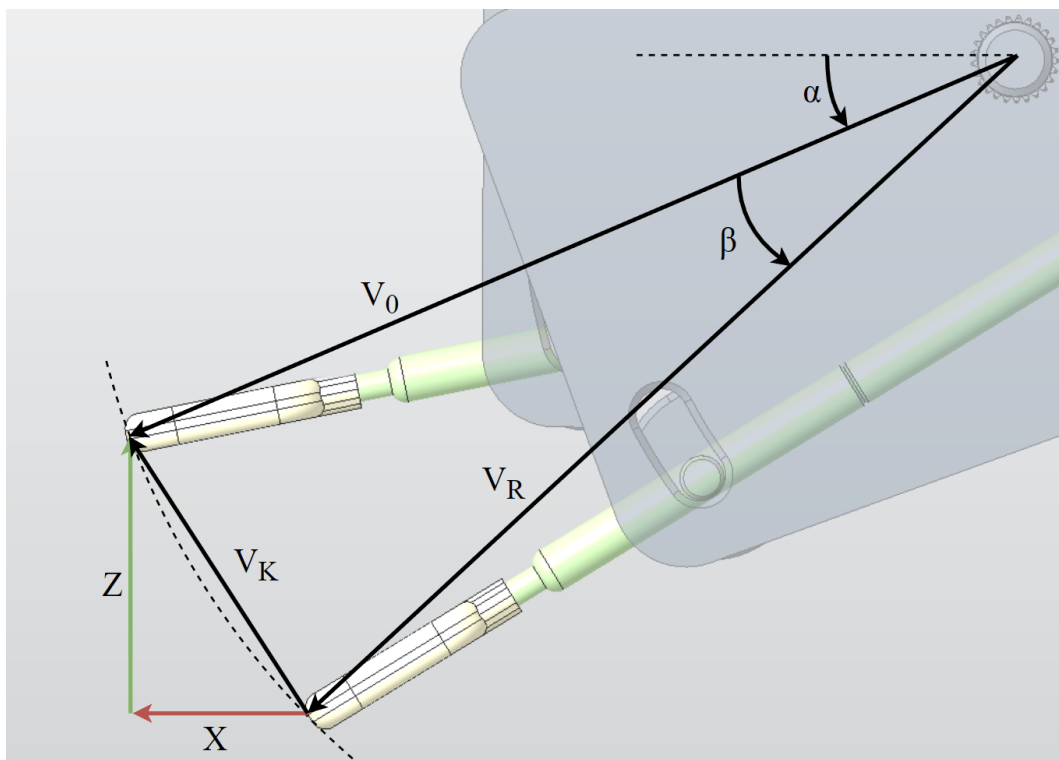


Abbildung 19: Geometrische Darstellung der Korrektur. Befindet sich der Reinigungskopf in der Normalposition, lässt sich die Lage des Pads über den Normalposition-Vektor V_0 bestimmen. Der Vektor V_R beschreibt die Position des Pads bei geändertem Reinigungswinkel. V_K entspricht dem Korrekturvektor mit seinen Projektionen X und Z.

Der Winkel $\alpha=22.7^\circ$ und die Beträge der beiden Vektoren ($|V_0| = |V_R| = 66 \text{ mm}$) sind durch die mechanische Konstruktion vorgegeben und können somit über das 3D-Modell bestimmt werden. Der Winkel β ist variabel und entspricht dem eingestellten Reinigungswinkel.

Im folgenden Berechnungsbeispiel wird für den Reinigungsvektor ein Winkel $\beta=20^\circ$ ($\hat{=}$ Minimalposition) angenommen:

$$\begin{aligned}
\vec{V}_0 &= |V_0| \cdot e^{j(180^\circ+\alpha)} = 66mm \cdot e^{j(180^\circ+22.7^\circ)} \\
&= 66 \cdot [\cos(202.7^\circ) + j \cdot \sin(202.7^\circ)] \\
&= -(60.888 + j25.470)mm \\
\vec{V}_R &= |V_R| \cdot e^{j(180^\circ+\alpha+\beta)} = 66mm \cdot e^{j(180^\circ+22.7^\circ+20^\circ)} \\
&= 66 \cdot [\cos(222.7^\circ) + j \cdot \sin(222.7^\circ)] \\
&= -(48.504 + j44.759)mm
\end{aligned} \tag{8}$$

Um den X- und Z-Korrekturwert zu erhalten, muss der Differenzvektor V_K berechnet und anschließend in seine Komponenten aufgeteilt werden.

$$\begin{aligned}
\vec{V}_K &= \vec{V}_0 - \vec{V}_R \\
&= X + jZ \\
&= -(60.888 + j25.470)mm - (-(48.504 + j44.759)mm) \\
&= (-12.384 + j19.289)mm
\end{aligned} \tag{9}$$

Die Ergebnissen in (8) und (9) zeigen, dass bei einer Verstellung des Reinigungswinkels um 20° , die Startposition des Reinigungskopfes um $X=-12.384$ mm und $Z=19.289$ mm korrigiert werden muss. Die Korrekturen erfolgen über das CNC-System, indem X- und Z-Werte, der in den Grundeinstellungen hinterlegten Startposition, als Offset dazu addiert werden.

Reinigung über Oberflächenprofile

Das Reinigungssystem bietet neben der eben beschriebenen Druckreinigung noch zwei alternative Methoden an: die Profil- und die Radiusreinigung.

Die Idee hinter der Profilreinigung ist, dem Roboter die Oberfläche einer Optik „anzutrainieren“, indem er diese abtastet und die so erfassten Koordinaten in Form eines Oberflächenprofils ablegt. Bei der anschließenden Reinigung werden die gespeicherten Koordinaten abgefahren. Zu Beginn des Reinigungsprozesses unterscheidet sich die Profilreinigung nur unwesentlich von der Druckreinigung. Nach dem Einlesen des

Oberflächenprofils wird der Kopf auch hier solange abgesenkt, bis der eingestellte Soll-Druck erreicht ist. Ab diesem Zeitpunkt ist im Gegensatz zum Druckreinigungsprozess keine permanente Drucküberwachung mehr erforderlich, denn die Oberfläche wird anhand der gespeicherten Koordinaten Punkt für Punkt abgefahren. Abbildung 20 veranschaulicht die Profilreinigung in Form eines Flussdiagramms.

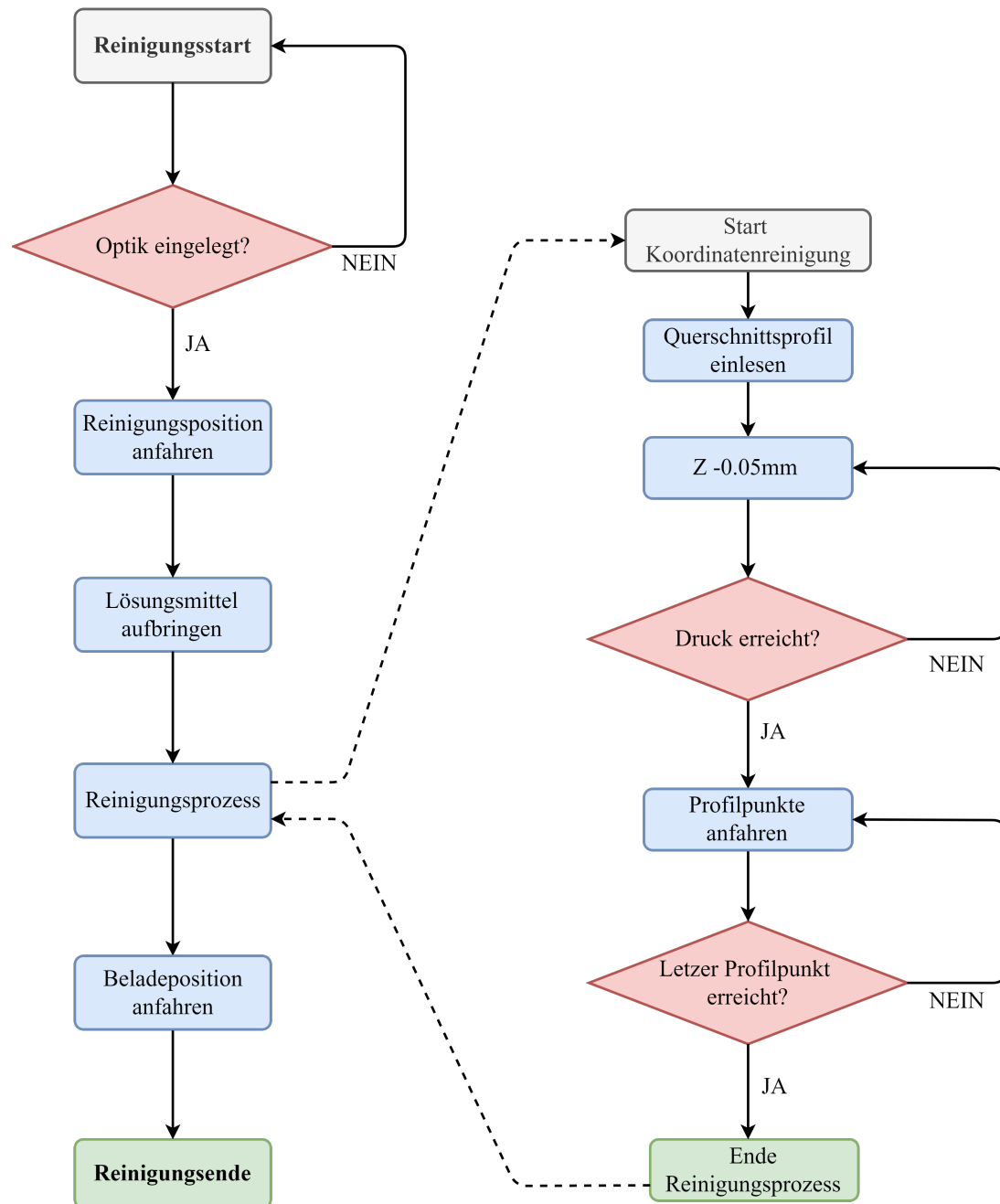


Abbildung 20: Linkes Flussdiagramm zeigt den kompletten Reinigungsvorgang. Die rechte Seite zeigt den Ablauf des Reinigungsprozesse über die zuvor gespeicherten Koordinatenpunkte.

Radiusreinigung

Die Radiusreinigung stellt einen Sonderfall der Profilreinigung dar. Die Robotersoftware berechnet den Krümmungsradius der Optikoberfläche anhand der gespeicherten Koordinaten. Das erfordert, dass zuvor mindestens drei Punkte abgetastet wurden. Idealerweise wählt man dafür den Startpunkt, den Endpunkt sowie einen dazwischenliegenden Punkt aus, siehe Abbildung 21.

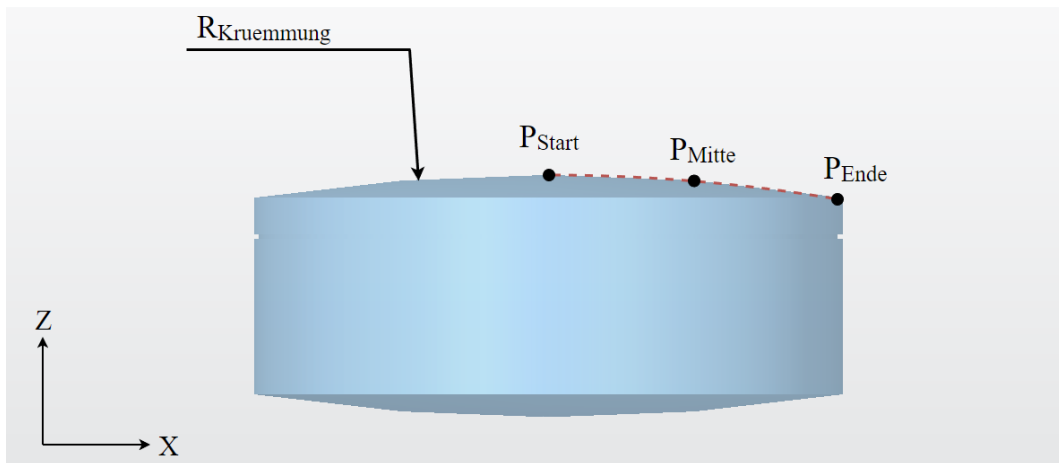


Abbildung 21: Achromat mit eingezeichneten Abtastpunkten zu Berechnung des Krümmungsradius. Die rot-strichlierte Linie kennzeichnet den beabsichtigten Verfahrensweg des Reinigungskopfes.

Während der Reinigung interpoliert das CNC-Positionierungssystem, ausgehend von der aktuellen Position P_{Start} , den Verfahrensweg unter Angabe von Krümmungsradius $R_{Kruemmung}$ und Endpunkt P_{Ende} . Die zentrale Steuerung sendet dazu einen der folgenden Befehle an den CNC-Controller:

```
1: G02 X(P_ENDE) Z(P_ENDE) R(KRÜMMUNG)
2: G03 X(P_ENDE) Z(P_ENDE) R(KRÜMMUNG)
```

Der GCODE Parameter 'G02' in der ersten Zeile bewirkt eine Kreisinterpolation im Uhrzeigersinn, während mit 'G03' eine Interpolation gegen den Uhrzeigersinn erreicht wird [49], welche beispielsweise bei der Reinigung von konkaven Linsen erforderlich ist. Die Z-Koordinate des Startpunktes ist variabel und ergibt sich, gleich wie bei Profil- und Druckreinigung, durch Erreichen des zuvor eingestellten Soll-Drucks, welcher durch Absenken des Z-Schlittens langsam aufgebaut wird.

Ermittlung des Oberflächenprofils

Zur Erfassung des Oberflächenprofils wurden zwei unterschiedliche Sensoren in Betracht gezogen: ein kapazitiver Näherungsschalter und ein Nivellierungssensor. Der wesentliche Unterschied dieser beiden Sensoren besteht darin, dass der Näherungsschalter berührungslos arbeitet, während der Nivellierungssensor erst auslöst sobald die Oberfläche berührt wird. Damit auch Optiken mit großen Krümmungsradien zuverlässig erfasst werden, muss der eingesetzte Sensor eine Genauigkeit von zumindest 0.1 mm aufweisen.

Kapazitiver Näherungsschalter.

Diese Art Sensor beruht auf der Verstimmung eines RC-Oszillators. Gelangt ein Objekt (metallisch oder nicht-metallisch) in die aktive Zone des Sensors, so ändert sich die Kapazität des RC-Schwingkreises und damit seine Amplitude. Wird eine bestimmte Schwelle erreicht, kippt ein nachgeschalteter Schmitt-Trigger den Ausgangszustand des Sensors. [49]

Durch schrittweises Absenken des Reinigungskopfes nähert sich der Näherungsschalter der Optik und löst aus, sobald ein zuvor justierter Schaltabstand unterschritten wird. Im gleichen Zug stoppt die zentrale Steuereinheit alle Bewegungen und erfasst die aktuelle Z- und X-Koordinate. Dieser Vorgang wird an beliebig vielen Punkten entlang der X-Achse wiederholt. Aus den gespeicherten Koordinatenpunkten lässt sich ein XZ-Profil der Oberfläche erstellen. Für diesen Ansatz wurde der Näherungsschalter *LJC18A3-B-Z/BX* (Heschen Electric Co. Ltd, Foshan, China) eingesetzt.

Nivellierungssensor

Der ursprüngliche Zweck von Nivellierungssensoren liegt darin, das Druckbett von 3D-Druckern abzutasten. Mit der daraus gewonnen Höhenkarte lassen sich etwaige Flächenunebenheiten des Betts softwaremäßig kompensieren.

Das Prinzip kann aber auch dazu genutzt werden, um das bereits erwähnte, zweidimensionale Oberflächenprofil einer Optik zu erstellen. Eingesetzt wurde der Sensor *BLTouch Smart V3.0* (Antclabs Inc., Seoul, Korea). Abbildung 22 auf Seite 55 zeigt wie der Sensor mit der zentrale Steuerung verbunden werden muss, um damit ein Profil abzutasten.

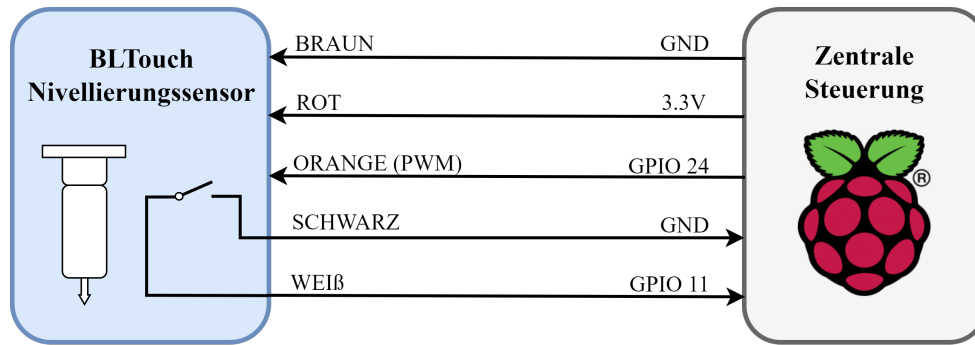


Abbildung 22: Anschlussdiagramm zwischen Nivellierungssensor und dem Pinheader des *Raspberry Pi*. Der eingezeichnete Endschalter (NO, Normally Open) schließt, sobald die Sonde eine Oberfläche berührt.

Der Nivellierungssensor besteht im Grunde aus einem Microcontroller, einem Hall-sensor einer Magnetspule und aus einer Sonde. Die Sonde ist außerdem einseitig permanent magnetisiert und lässt sich so über die Magnetspule heben und senken, ähnlich einem *Solenoid*. Trifft die Sonde auf eine Oberfläche, wird sie leicht angehoben und triggert damit den internen Hall-Sensor. Der Microcontroller reagiert darauf, indem er die Magnetspule bestromt und die Sonde sofort wieder anhebt. Die Steuerung der Sonde erfolgt über eine Pulsweitenmodulation. Der Microcontroller wertet dazu das durch die zentrale Steuerung generierte PWM-Signal, ferner dessen Pulsweite, aus und reagiert darauf mit unterschiedlichen Aktionen. [50]

Tabelle 13: Auszug der für das Reinigungssystem relevanten PWM-Signale. Der sensorinterne Microcontroller toleriert Abweichungen von $\pm 20 \mu s$. [50]

Anweisung	Pulsweite in μs
Selbsttest, Sonde wird 10x aus- und einfahren)	1780
Sonde einfahren	1475
Sonde ausfahren	650

Abtastprozess mittels Nivellierungssensor

Für die Profilerfassung nimmt das CNC-Positionierungssystem zunächst die Abtastposition ein. In dieser Position stimmen optische Achse und Sondenachse überein. Durch Senden des entsprechenden PWM-Signales (siehe Tabelle 13, Seite 55), wird die Sonde ausgefahren. Das CNC-Positionierungssystem senkt den Reinigungskopf nun schrittweise ab, bis die Sonde auf die Oberfläche trifft und wie bereits beschrieben, wieder eingezogen wird. Gleichzeitig wird der zentralen Steuerung signalisiert, dass die Oberfläche erreicht wurde. Die aktuellen X- und Z-Koordinaten werden gespeichert und das CNC-Positionierungssystem hebt den Reinigungskopf wieder an. Dieser Vorgang wiederholt sich, bis der Radius der Optik vollständig abgetastet wurde.

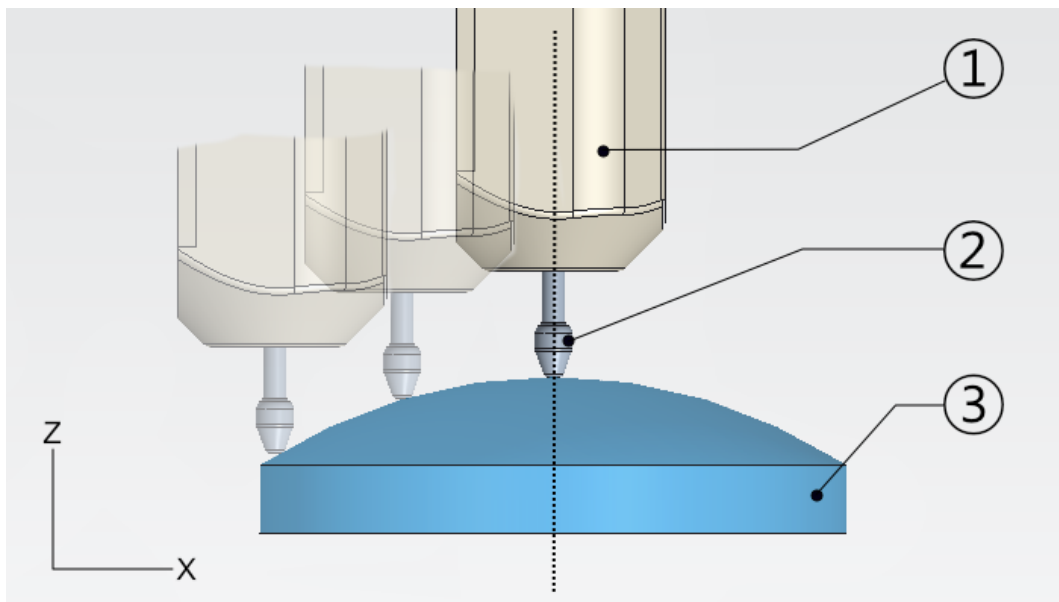


Abbildung 23: Visualisierung des Abtastvorgangs mittels Nivellierungssensor entlang der Oberfläche einer Linse

- (1) Sensorgehäuse
- (2) Kunststoffspitze der Sonde
- (3) konvexe Linse

Da die Optik während der Reinigung rotiert, ist es ausreichend nur eine Hälfte abzutasten. Je kleiner der Schrittweite entlang der X-Achse gewählt wird, desto länger dauert zwar der Abtastprozess, aber desto höher ist dafür die Auflösung. Ein Oberflächenprofil besteht aus mindestens drei Abtastpunkten und die Anzahl lässt sich über die Software des Roboters einstellen.

2.5.5 Lösungsmittelfördereinheit

Im folgenden Kapitel wird erläutert, mit welcher Technik das Lösungsmittel automatisiert in den Reinigungsvorgang eingebunden wird. Prinzipiell kann die Einheit in eine Pumpe mit Lösungsmittelbehälter und einem Schlauchsystem samt Düse unterteilen werden. Aus Platzgründen ist die Pumpe am Y-Schlitten des Roboters untergebracht. Das hat zur Folge, dass das Lösungsmittel über ein Schlauchsystem zum Reinigungskopf transportiert werden muss. Über die dort befestigte Düse, wird es in einstellbaren Mengen auf das Polyester-Pad des Reinigungsstäbchens abgegeben.

Spritzenpumpe

Die Förderung des Lösungsmittels erfolgt über eine Spritzenpumpe. Das Prinzip ist einfach und eignet sich gut für die in diesem Fall nicht-kontinuierliche Förderung von Flüssigkeiten.

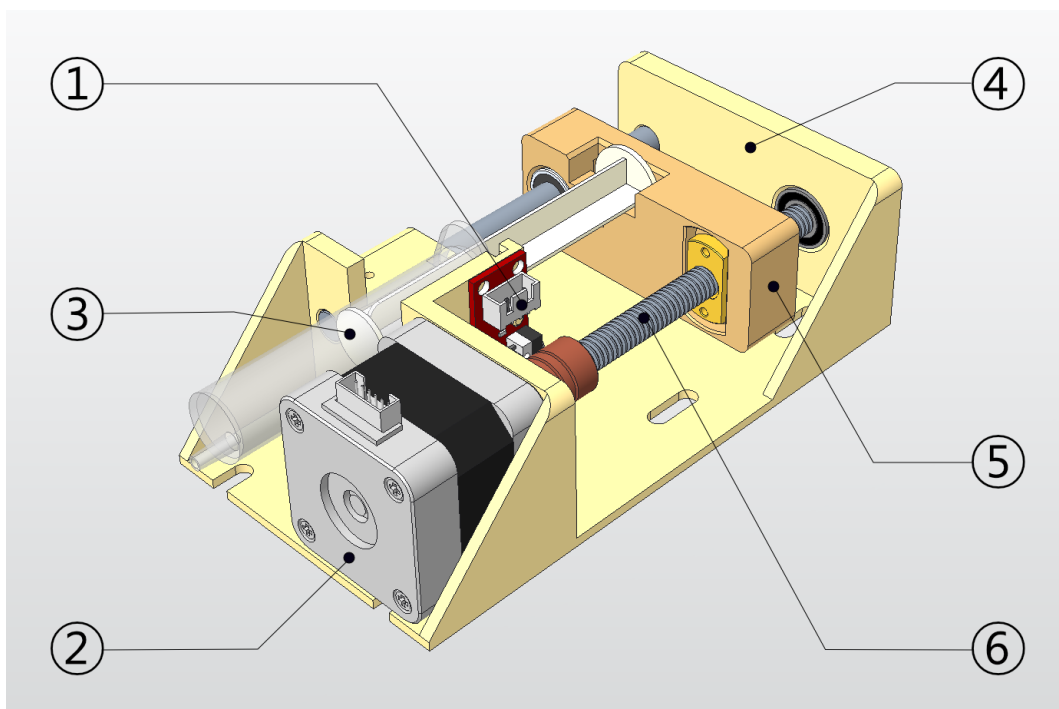


Abbildung 24: Schrittmotorbetriebene Pumpeneinheit mit Einwegspritze als Lösungsmittelbehälter

- (1) Endschalterplatine
- (2) NEMA17 Schrittmotor
- (3) Einwegspritze
- (4) Basis
- (5) Kolbenschieber
- (6) T8-Trapezgewindespindel mit 2 mm Steigung

Für die Kraftübertragung auf den Spritzenkolben wurde am Konzept des CNC-Positionierungssystems festgehalten: Eine Trapezgewindespindel überführt die Rotationsbewegung des Antriebsmotors in eine translatorische Bewegung. Der mit dem Spritzenkolben verbundene Kolbenschieber, lässt sich damit entlang einer Achse bewegen und drückt so das Lösungsmittel aus der Spritze. Der Antrieb erfolgt über einen bipolaren Schrittmotor, der im Microstepping-Mode betrieben wird. Dadurch lässt sich die Abgabemenge noch feiner dosieren. Als Lösungsmittelbehälter werden gewöhnliche Einwegspritzen mit einem Fassungsvermögen von 12 ml verwendet. Über den an der Basis befestigten Endschalter wird verhindert, dass der Kolbenschieber auf Anschlag fährt. Löst dieser aus, wird dem/der AnwenderIn mitgeteilt, dass die Spritze neu aufgezogen werden muss.

Schrittmotoransteuerung

Im Gegensatz zum CNC-Positionierungssystem, wird der Schrittmotor der Spritzenpumpe nicht über den CNC-Controller angesteuert, sondern über die zentrale Steuerung. Aus leistungstechnischen Gründen lässt sich dieser nicht direkt über den RPI4 betrieben. Das bedeutet, dass zusätzlich ein Schrittmotortreiber notwendig ist. Zur Anwendung kommt der Treiberbaustein *DRV8825* (Texas Instruments, Dallas, USA) in Kombination mit einem *Breakout Board*.

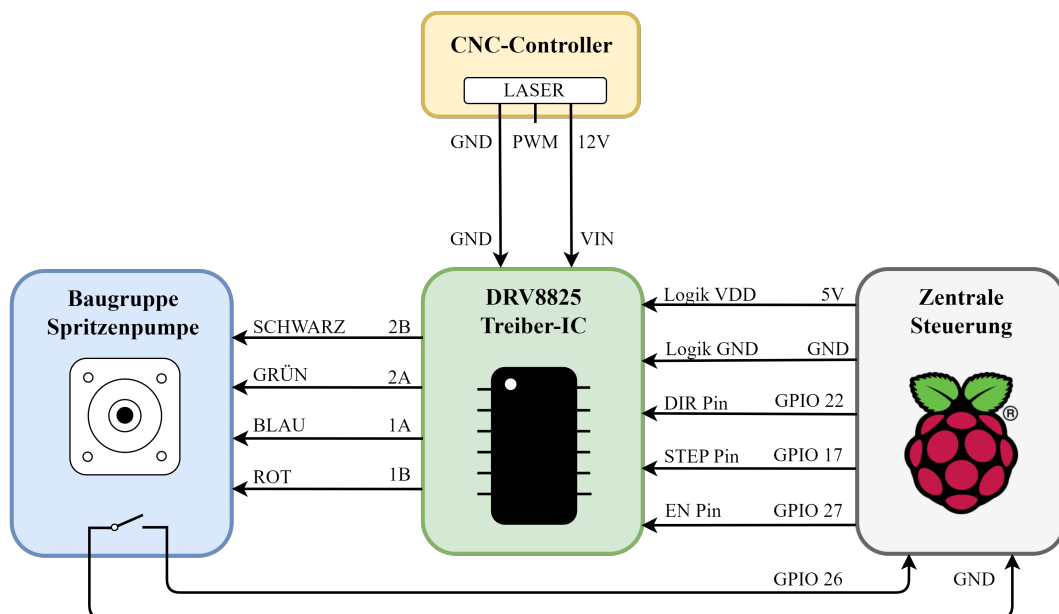


Abbildung 25: Blockdiagramm der Schrittmotorsteuerung. Für die Versorgung der Schrittmotoren wurden die 12V des nicht benötigten Laserausgang des CNC-Controllers abgegriffen.

Der eingesetzte Treiberbaustein erlaubt es, den Schrittmotor auch im *Microstepping-Mode* zu betreiben. Die Konfiguration erfolgt hardwaremäßig über die am *Breakout Board* befindlichen DIP-Schalter (Dual In-Line Package). Der Treiber ist in der feinsten Schrittauflösung konfiguriert. Das bedeutet, dass ein Schritt in weitere 32 Schritte unterteilt wird.

Tabelle 14: Erforderliche Schalterstellungen der DIP-Switches um den Schrittmotor im 1/32 Microstepping-Mode zu betreiben [51].

Pin am IC	Schalter am Board	Stellung
MODE0	MS1	ON
MODE1	MS2	ON
MODE2	MS3	ON

Als Antrieb wird der Bipolar-Schrittmotor *17HS4401S* (Handson Technology) eingesetzt, welcher laut Datenblatt [42] mit einem maximalen Strom von 1.7A pro Phase belastet werden darf.

Um Schäden an den Motorwicklungen zu vermeiden, lässt sich der maximale Strom über die Referenzspannung des Treiber-ICs begrenzen. Der Zusammenhang zwischen Referenzspannung $V_{(xREF)}$ und der Strombegrenzung I_{CHOP} ist wie folgt definiert [51]:

$$I_{CHOP} = \frac{V_{(xREF)}}{5 \cdot R_{ISENSE}} \quad (10)$$

Wie in Berechnungsvorschrift (10) ersichtlich, ist der maximale Strom nicht nur von der Referenzspannung, sondern auch vom verbauten Messwiderstand R_{ISENSE} abhängig. Die am *Breakout Board* verlöteten Messwiderstände, weisen einen Wert von 0.1Ω auf. Daraus ergibt sich, dass die Referenzspannung auf folgenden Wert eingestellt werden muss:

$$\begin{aligned} V_{(xREF)} &= I_{CHOP} \cdot 5 \cdot R_{ISENSE} = \\ &= 1.7A \cdot 5 \cdot 0.1\Omega = 0.85V \end{aligned} \quad (11)$$

Die Spannung kann über das Trimpotentiometer am *Breakout Board* sowohl eingestellt als auch gemessen werden. Alle in diesem Zusammenhang notwendigen Messungen wurden mit einem *Fluke 175 True RMS* Handmultimeter durchgeführt.

Schlauchsystem

Das Schlauchsystem führt von der Spritzenspitze bis zur Düse am Reinigungskopf und wird auf dem Weg dahin mehrmals um bis zu 180° gebogen. Dem entsprechend hoch sind auch die Materialanforderungen an den Schlauch. Er muss einerseits flexibel genug sein, um ihn verlegen zu können und andererseits widerstandsfähig, um den permanenten Bewegungen der Schlitten während der Reinigung standzuhalten.

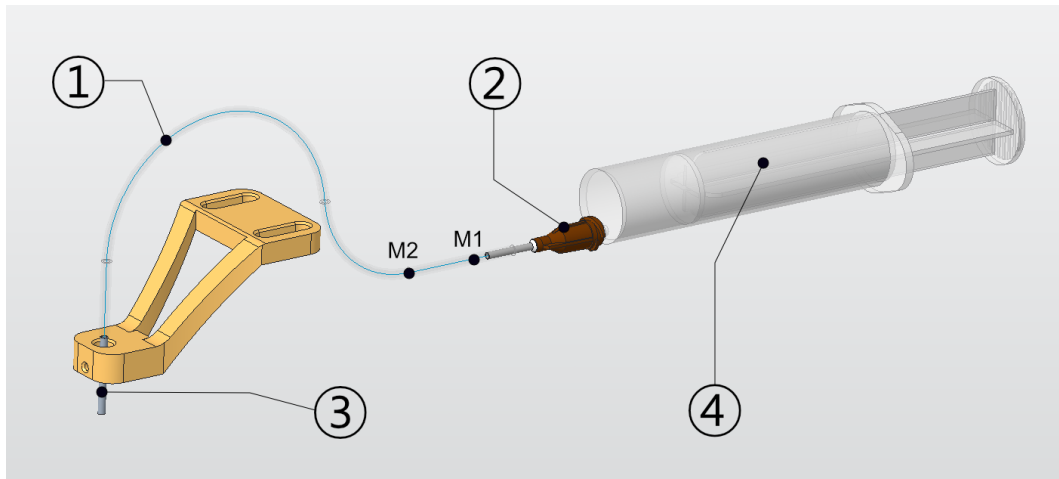


Abbildung 26: Schlauchsystem des Reinigungsroboters für die Förderung verschiedener Lösungsmittel. Zur besseren Veranschaulichung ist der Schlauch gekürzt dargestellt. Die symbolhaft eingezeichneten Markierungen M1 und M2 dienen der Bestimmung der Fördermenge.

- (1) Schlauch
- (2) Industrie-Kanüle mit *Luer-Lock* Außengewinde
- (3) Düse
- (4) Einwegspritze

Außerdem ist eine Beständigkeit gegenüber unterschiedlicher Lösungsmittel wie beispielsweise Aceton notwendig. Pumpenseitig ist der Schlauch mit einer Dosiernadel (Industrie-Kanüle) verbunden, welche sich leicht an der Spritzenspitze an- und abstecken lässt. Auf der Seite des Reinigungskopfes endet der Schlauch direkt an der Düse. Die Düse ist im Grunde eine Dosiernadel bei welcher der Gewindekörper *Luer-Lock* entfernt wurde.

Berechnung der Abgabemenge pro Schritt

Die Abgabemenge pro Mikroschritt wurde über die Differenz des Flüssigkeitsstandes im Schlauch bestimmt. Markierung M1 kennzeichnet den Flüssigkeitsstand vor dem Pumpvorgang und Markierung M2 danach. Wird der Schrittmotor um 400 Mikroschritte gedreht, ergibt sich zwischen den beiden Markierungen eine Längendifferenz

von $\Delta M = 51$ mm. Da auch der Schlauchinnendurchmesser mit $d_i = 1.6$ mm bekannt ist, lässt sich die Fördermenge V_{400} über die Zylindervolumen-Formel berechnen:

$$\begin{aligned} V_{400} &= \pi \cdot \left(\frac{d_i}{2}\right)^2 \cdot \Delta M = \pi \cdot \left(\frac{1.6\text{mm}}{2}\right)^2 \cdot 51\text{mm} \\ &= 102.54\text{mm}^3 = 102.54\mu\text{l} \end{aligned} \quad (12)$$

Aus dem Ergebnis in (12) kann anschließend die Fördermenge pro Mikroschritt (mstep) bestimmt werden.

$$\frac{V_{400}}{400 \text{ msteps}} = \frac{102.54\mu\text{l}}{400 \text{ msteps}} = 256.35 \frac{\text{nl}}{\text{mstep}} \quad (13)$$

Schlauchinhalt und Speichervolumen

Die Gesamtlänge des verlegten Schlauches liegt in etwa bei $l_{\text{Schlauch}} = 1000$ mm, daraus ergibt sich, dass ein gefüllter Schlauch ca. 2 ml an Lösungsmittel enthält.

$$\begin{aligned} V_{\text{Schlauch}} &= \pi \cdot \left(\frac{d_i}{2}\right)^2 \cdot l_{\text{Schlauch}} = \pi \cdot \left(\frac{1.6\text{mm}}{2}\right)^2 \cdot 1000\text{mm} \\ &= 2010.62\mu\text{l} \end{aligned} \quad (14)$$

Mit welcher Lösungsmittelmenge das Polyester-Pad des Reinigungsstäbchen benetzt wird, hängt von der Fläche der zu reinigenden Optik ab. Geht man von einem Schnitt von $50 \mu\text{l}$ je Reinigung aus, so lassen sich mit der vollen 10 ml Spritze 200 Reinigungen durchführen.

2.5.6 Bedien- und Anzeigeeinheit

Die Bedien- und Anzeigeeinheit dient zur Interaktion mit dem Anwender und besteht im Wesentlichen aus einem Display, einem belüfteten Gehäuse und einer Halterung, über welche die Einheit am Roboter fixiert ist.

Dank der Touch-Funktion des Displays, sind im Grunde keine weiteren Eingabegeräte wie Maus oder Tastatur erforderlich. Bei Bedarf lassen sich diese aber trotzdem an den RPI4 anstecken und verwenden. Neben einer Bedienung mit den Fingern lässt sich der Roboter wahlweise auch über einen kapazitiven Eingabestift steuern. Die Bildübertragung erfolgt über HDMI und die für den Betrieb benötigte Versorgungsspannung, wird über eine der USB-Schnittstellen der zentralen Steuerung bereit gestellt. Über dieselbe USB-Verbindung erfolgt auch der Datenaustausch zwischen Touch-Matrix und zentraler Steuerung.



Abbildung 27: Bedien- und Anzeigeeinheit des Roboters

- (1) Display mit integriert kapazitiver Touch-Matrix
- (2) Halterung zur Befestigung an der rechten Aluminiumprofilschiene
- (3) Unteres, durch Schlitze belüftetes Gehäuse
- (4) Oberes Gehäuseteil

Bei dem verbauten Touch-Display handelt sich um das Modell *RC050* (Elecrow, Shenzhen, China). Es misst 5 Zoll in der Diagonale und schafft eine Bildschirmauflösung von 800x480 px [52].

Wie in Abbildung 27 zu sehen, ist das Display hochkant verbaut. Die Drehung um 90° hat zwei Gründe: Einerseits ergeben sich dadurch kompaktere Abmessungen des Gesamtsystems und andererseits wäre ein Anschluss der Kabel bei horizontaler Ausrichtung nicht möglich.

2.5.7 Elektrische Versorgung und Leistungsverbrauch

Alle am System angeschlossenen, elektrischen Komponenten werden über das Netzteil des CNC-Controllers gespeist. Da diese mit unterschiedlichen Spannung arbeiten, ist es notwendig, die Ausgangsspannung des Netztes in weitere Spannungsebenen zu unterteilen. Das passiert in erster Linie über das CNC-Controller-Board, welches standardmäßig über einen 24 V-, einen 12 V- und einen 5 V-Ausgang verfügt. Damit es zu keiner Überlastung des 5 V-Ausgangs kommt, werden die 12 V des ungenutzten Laseranschlusses über einen DC/DC-Wandler in 5 V konvertiert um damit die zentrale Steuerung zu betreiben, siehe Abbildung 28.

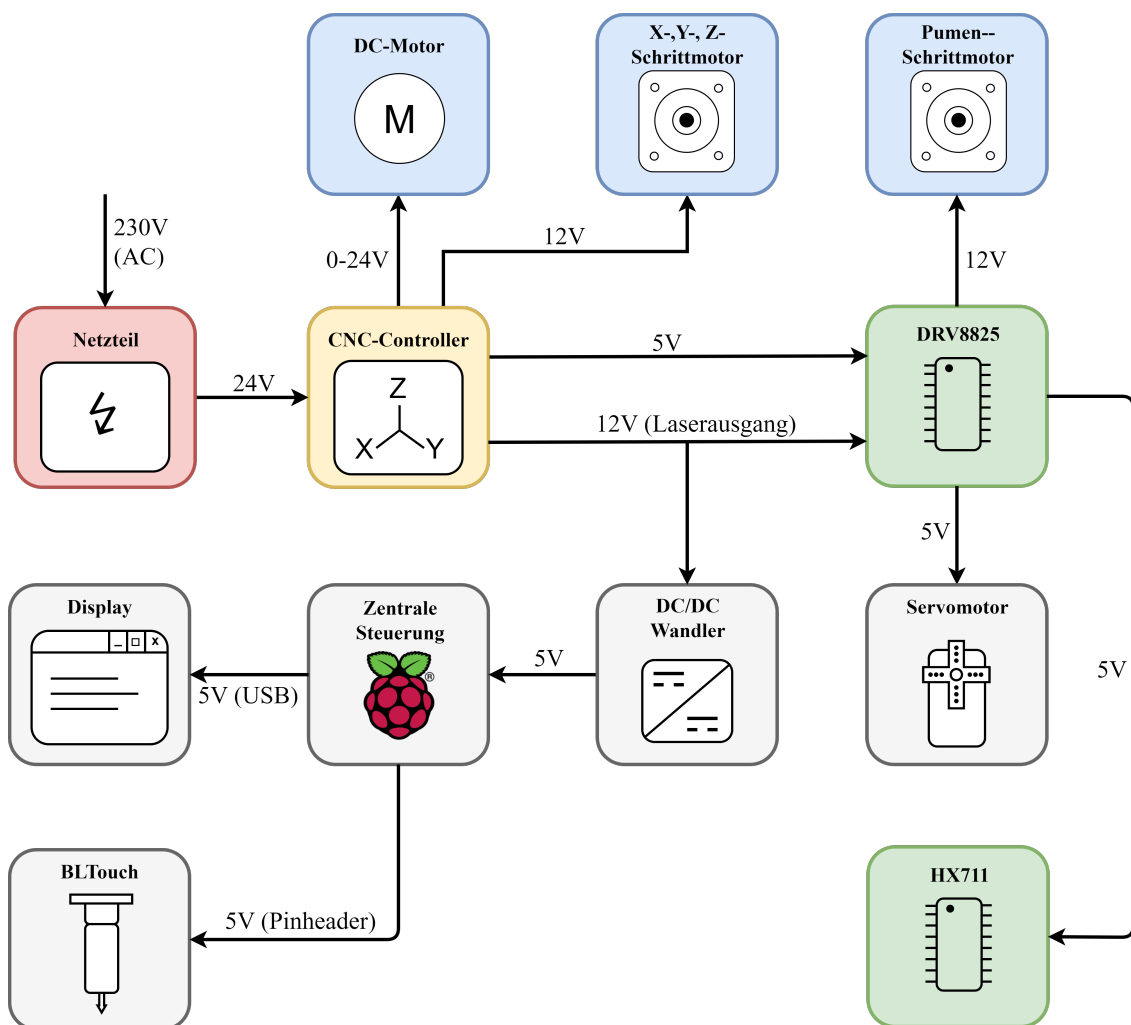


Abbildung 28: Aufbau der elektrischen Versorgung des Reinigungsroboters

Die maximal mögliche Ausgangsleistung des Netztes liegt laut Typenschild bei 96 W. Bei gewöhnlichem Gebrauch benötigt der Roboter eine durchschnittliche Leistung von 12 W.

2.6 Software

Die Steuerungssoftware des Roboters ist in der Programmiersprache *Python* entwickelt, da es für *Python* zahlreiche open-source Bibliotheken zur Ansteuerung der an den *Raspberry Pi 4* angeschlossenen Hardware gibt. Der angewendete Programmierstil orientiert sich am *PEP 8* (Python Enhancement Proposal 8) Coding-Standard.

Das auf der zentralen Steuerung aufgesetzte Betriebssystem *Raspberry PI OS*, in der Version 11, inkludiert bereits den erforderlichen Python-Interpreter. Für Entwicklung und Ausführung der Software mussten allerdings weitere Pakete installiert und Änderungen an den Systemeinstellungen vorgenommen werden, siehe dazu Kapitel 2.6.1.

2.6.1 Vorbereitungen

Konfiguration des Betriebssystems

Die grundlegende Konfiguration der zentralen Steuerung lässt sich über die Konsole mit dem Befehl '`$ sudo raspi-config`' ändern [53]. Beispielsweise muss hier die standardmäßig deaktivierte SSH-Schnittstelle (Secure Socket Shell) aktiviert werden um einen Remote-Zugriff auf den Raspberry Pi zu ermöglichen.

Da der Roboter auch über andere Geräte wie PC, Notebook oder Tablet bedient werden soll, ist es zudem notwendig, dass ein RDP-Tool (Remote Desktop Protokoll) installiert ist. Das erforderliche Linux-Paket heißt XRDP und kann über '`$ sudo apt install xrdp`' nachgerüstet werden.

Installation zusätzlicher Python-Pakete

Einige der im Quellcode eingebundenen Python-Pakete sind standardmäßig vorinstalliert, andere wiederum, müssen über den Paketmanager PIP (Packet Installer for Python) bzw. APT (Advanced Packaging Tool) nachinstallieren werden.

Damit sich die Robotersoftware in seiner finalen Version starten lässt, sind alle in Tabelle 15, Seite 65 aufgelisteten Pakete notwendig. Die Installation erfolgt über die Konsole des RPI4 und kann unter Einhaltung folgender Syntax durchgeführt werden:

```
$ sudo pip3 install <Paketname>
```

Eine Ausnahme stellen die Pakete HX711 und PySide2 dar. Diese sind nicht über PyPI (Python Package Index), der Quelle auf die PIP zurückgreift verfügbar. In diesem Fall erfolgt die Installation über APT.

```
$ sudo apt install <Paketname>
```

Zu beachten ist, dass unter Linux manche Befehle Root-Rechte erfordern. Befehle mit dem Präfix '`sudo`' deuten darauf hin, dass sie als *Superuser*, also mit der höchsten Berechtigungsstufe ausgeführt werden.

Tabelle 15: In der Robotersoftware verwendete Python-Pakete, welche nicht standardmäßig installiert sind

Paketname	Version	Beschreibung
pigpio	1.78	Das Paket ermöglicht es, PWM Signale zu erzeugen und an einem der GPIO-Pins auszugeben.
RPi.gpio	0.7.0	'RPi.gpio' erlaubt den Zugriff auf die GPIO-Pins. Es wird verwendet um den Endschalter der Spritzenpumpe einzulesen und ist außerdem Voraussetzung für das Schrittmotor Modul ' <code>rpimotorlib</code> '.
pyserial	3.5b0	Das Paket ' <code>pyserial</code> ' erlaubt den Zugriff auf die seriellen Ports (USB) des Raspberry Pi und wird benötigt um eine Verbindung mit dem CNC-Controller aufzubauen.
rpimotorlib	3.1	' <code>pyserial</code> ' wird verwendet, um den Schrittmotor der Lösungsmittelfördereinheit anzusteuern.
matplotlib	3.5.2	Das Paket ' <code>matplotlib</code> ' wird in Kombination mit der grafischen Benutzeroberfläche verwendet. Mit ihm lassen sich Diagramme im Matlab-Stil erstellen und animieren. Die Software benötigt es, um den Druckverlauf sowie das Optikprofil grafisch darzustellen.
netifaces	0.11.0	Wird verwendet, um den Hostnamen und die aktuelle IP-Adresse (Internet Protocol) der Netzwerkadapter für den Fernzugriff auszulesen.
HX711	1.0.0	Damit lässt sich der externe ADC, im weiteren Sinne die auf die verbaute Wägezelle einwirkende Kraft, auslesen.
PySide2 (core)	5.15.2-1	' <code>PySide2</code> ' ermöglicht es, das mit dem <i>Qt-Designer</i> (The Qt Company, Espoo, Finnland) erstellte GUI-Design mit den einzelnen Methoden der Software zu verknüpfen und kann somit als Schnittstelle zwischen Frontend und Backend der Software gesehen werden.

2.6.2 Entwicklungswerkzeuge

Programmierungsumgebung

Theoretisch wäre die Entwicklung der Robotersoftware auch direkt am RPI4 möglich gewesen. Der zu Beginn eingesetzte und in *Raspberry Pi OS* bereits vorinstallierte Python-Editor *Thonny* besitzt jedoch nur eingeschränkte Funktionalität und eignet sich deshalb im besten Fall für kurze Skripte bzw. kleine Software-Projekte. Für Software größeren Umfangs, wie jene des Roboters, ist es sinnvoller, eine umfassendere IDE (Integrated Development Environment) zu verwenden.

Aufgrund der begrenzten Performance lassen sich am RPI4 keine leistungsintensiven IDEs ausführen. Deshalb war es notwendig, auf eine Remote-Entwicklung über SSH auszuweichen. Dabei verbindet sich die am Entwicklungs-System (z.B. Windows PC) installierte IDE über das TCP/IP Protokoll mit dem Ziel-System, in diesem Fall dem RPI4. Der Quellcode wird zwar am entfernten PC geschrieben, Ausführung und Speicherung erfolgt jedoch am Ziel-System.

Als IDE wurde die Software *Visual Studio Code* (Microsoft, Redmond, USA) eingesetzt. Für diese Programmierungsumgebung gibt es etliche *Add-Ons*, welche die Entwicklung und das Debuggen von Python-Projekten erleichtern. Außerdem unterstützt *Visual Studio Code* die bereits erwähnte Remote-Entwicklung über SSH.

X-Window System

Ein Umstand, welcher sich durch die Remote-Entwicklung ergibt ist, dass die grafische Ausgabe am Ziel-System erfolgt, in diesem Fall am Display des Roboters. Das ist zwar grundsätzlich erwünscht, da der Roboter hauptsächlich als Standalone-Gerät betrieben wird, aber gerade in der Entwicklungsphase ist es oft hilfreich, die GUI am Entwicklungs-PC auszugeben. Dies lässt sich durch Installation eines X11-Servers (X-Window System) realisieren. Abbildung 29 veranschaulicht das Prinzip grafisch.

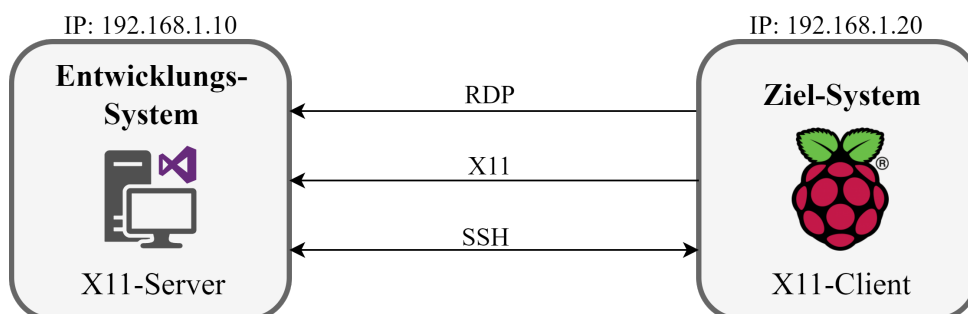


Abbildung 29: Darstellung aller auf TCP/IP basierenden Protokolle zwischen Entwicklungs- und Ziel-System

Das Entwicklungs-System wird somit zum „Netzwerk-Monitor“, auf welchem X11-Clients ihre grafischen Ausgaben weiterleiten können. Mit folgendem Befehl kann zwischen lokaler und entfernter Anzeige gewechselt werden:

```
$ sudo export DISPLAY=192.168.1.10:0.0 #Ausgabe am Dev-PC
$ sudo export DISPLAY=localhost:0.0    #Lokales Touch-Display
```

Qt-Designer

Die grafische Oberfläche der Robotersoftware wurde mit dem Programm *Qt-Designer* erstellt. Steuerelemente wie Buttons, Textfelder, Checkboxes usw. sind bereits vorgefertigt und lassen sich via Drag & Drop im Fenster platzieren.

Das Programm speichert das designte Fenster als UI-Datei ab, welches direkt über den Python-Code eingelesen und anschließend über das bereits erwähnte 'PySide2' Paket mit den einzelnen Methoden der Software verknüpft werden kann. UI-Dateien basieren auf der Sprache XML (Extensible Markup Language), kleinere Anpassungen des Designs lassen sich deshalb auch ohne den *Qt-Designer* durchführen.

2.6.3 Programmstruktur

Alle für die Software relevanten Dateien sind im Ordner '*\linsenfischer*' zu finden, welcher im Verzeichnis '*\home\wild\Python*' abgelegt ist.

Tabelle 16: Auflistung der unterschiedlichen Dateiformate des Softwareprojektes und deren relative Pfade.

Typ	Suffix	Beschreibung und Speicherort
Config-Datei	*.cfg	Es existiert eine Konfigurationsdatei welche im Ordner ' <i>./einstellungen</i> ' abgelegt ist. In ihr sind alle grundlegenden Einstellungen hinterlegt.
Profil-Dateien	*.profil	Profil-Dateien beinhalten spezifische Informationen über die zu reinigende Optik und werden standardmäßig im Ordner ' <i>./profile</i> ' abgelegt.
Modul-Dateien	*.py	Sie sind im Root-Verzeichnis des Projektordners zu finden und beinhalten den Quellcode der Software.
Design-Dateien	*.ui	Design-Dateien beschreiben den Aufbau der GUI im XML-Format und sind ebenfalls im Root-Verzeichnis abgelegt.
Bild-Dateien	*.png	Alle Logos und Icons die in der GUI verwendet, sind im Ordner ' <i>./logos</i> ' zu finden.

Grundkonfiguration

In der Datei `'einstellungen.cfg'` sind alle wesentlichen Grundeinstellungen der Software hinterlegt. Beispielsweise enthält sie die USB-Verbindungseinstellungen für den CNC-Controller, diverse Konstanten und Umrechnungsfaktoren, die GPIO-Pin-Konfigurationen, die Fixpositionen des CNC-Positionierungssystems und vieles mehr. Der Inhalt ist nicht codiert und kann über jeden gewöhnlichen Texteditor verändert werden. Während des Programmstarts wird die Datei zeilenweise eingelesen und ausgewertet. Jede Zeile entspricht dabei einer Einstellung und muss folgendes Format aufweisen:

Attribut:Wert

Kommentare sind mit einem `'#'` gekennzeichnet und werden ebenso wie Leerzeilen verworfen. Die Grundkonfiguration lässt sich zwar über ein eigenes Menü in der Software anzeigen, Änderungen können jedoch aus Sicherheitsgründen nur in der Datei selbst vorgenommen werden. Damit diese übernommen werden, ist ein Neustart der Robotersoftware notwendig.

Optikprofile

Das Format eines Optikprofils ähnelt dem der Konfigurationsdatei. Die Informationen werden nach demselben Prinzip gespeichert und ausgelesen. Ein Optikprofil besteht im Wesentlichen aus drei Abschnitten: den Allgemeinen Daten, den Reinigungsparametern und den XZ-Profilkoordinaten.

Die allgemeinen Daten bestehen aus Informationen wie z.B. Artikelnummer, Kunde oder Krümmungstyp. Im Abschnitt Reinigungsparameter sind Lösungsmittelmenge, maximaler Druck, Optikradius, Drehzahl usw. festgelegt. Der Abschnitt XZ-Profilkoordinaten ist optional und nur dann vorhanden, wenn die Optik vor dem Speichern des Profils abgetastet wurde.

Programmmodule

Für eine bessere Übersicht, ist die Software in neun einzelne Module eingeteilt. Die Methoden sind so zusammengefasst, dass sie immer einer Hardware bzw. einer Softwareeinheit zuordenbar sind. So enthält beispielsweise das Modul `'bltouch.py'` alle nötigen Funktionen, die für die Ansteuerung und das Auslesen des Nivellierungssensors notwendig sind.

Die in Tabelle 17 (Seite 69) erwähnte Datei `'main.py'` stellt das Hauptmodul dar und ist damit jene Datei, die dem Python-Interpreter übergeben werden muss, um die Software des Roboters zu starten.

Tabelle 17: Auflistung aller Python-Module die für die Robotersoftware erstellt wurden

Modulname	Kurzbeschreibung
bltouch.py	Das Modul ermöglicht das Heben und Senken der Sonde über PWM und verarbeitet die Zustandsänderung des Endschalters, welcher bei Berührung einer Oberfläche auslöst. Außerdem steuert es den gesamten Abtastvorgang während der Aufzeichnung eines Oberflächenprofils.
controller.py	Über dieses Modul erfolgt die serielle Kommunikation mit dem CNC-Controller. Das umfasst neben dem Aufbau und der Initialisierung der Schnittstelle, auch das Senden von Befehlen sowie das Empfangen des Gerätestatus.
hilfsfunktionen.py	Diese Datei beinhaltet eine Sammlung an kleinen Hilfsfunktionen. Beispielsweise wird in ihr der empfangene Gerätestatus zerlegt um daraus die aktuellen Koordinaten und den Zustand des Gerätes zu extrahieren. Zudem enthält das Modul unterschiedliche Funktionen, über welche Umwandlungen und Korrekturen durchgeführt werden.
main.py	Als zentrales Modul verknüpft sie die Elemente der GUI mit den einzelnen Methoden und Funktionen, erfasst Benutzereingaben, koordiniert die Reinigungsprozesse und managed die Anzeige des Haupt- sowie des Einstellungen-Fensters.
profil.py	Das Modul wird aufgerufen, wenn das Profil einer Optik gespeichert bzw. gelesen wird. Ein Optikprofil besteht immer aus allgemeinen Daten, spezifischen Reinigungsparametern und den optionalen Profilkordinaten. Bei den implementierten Methoden geht es im Wesentlichen um Dateimanipulation.
schrittmotor.py	In der ' <code>schrittmotor.py</code> ' erfolgt die Ansteuerung des Schrittmotors der Spritzenpumpe über die GPIO-Pins des RPI4.
sensor.py	Das Modul wird benötigt, um den Kraftsensor (Wägezelle) zu initialisieren und auszulesen.
servo.py	Über die ' <code>servo.py</code> ' erfolgt die Verstellung des Reinigungskopfes. Wie in der ' <code>bltouch.py</code> ', wird auch hier ein PWM-Signal erzeugt und an einem der GPIO-Pins ausgegeben.

Benutzeroberfläche

Das Design der Benutzeroberfläche ist über die beiden Dateien 'hauptfenster.ui' und 'einstellungen.ui' definiert. Die Datei 'hauptfenster.ui' wird beim Start des Programms eingelesen und erzeugt daraus das Hauptfenster.

Die Bildschirmauflösung des Touch-Displays ist mit 800x480 px begrenzt. Damit trotzdem alle notwendigen Steuerelemente (Buttons, Textfelder, Diagramme usw.) Platz finden, sind die Inhalte kategorisiert und auf die vier Registerkarten *Profile*, *Reinigung*, *Manuelle Kontrolle* und *Aufzeichnung* aufgeteilt. Für eine gleichmäßige Ausrichtung der einzelnen Elemente kommen Vertikal-, Horizontal- und Grid-Layouts in Kombination mit Platzhaltern zum Einsatz. Die Größe des Hauptfensters ist an das Display angepasst und lässt sich nicht ändern. Eine dynamische Größenanpassung der einzelnen Steuerelemente ist deshalb nicht notwendig.

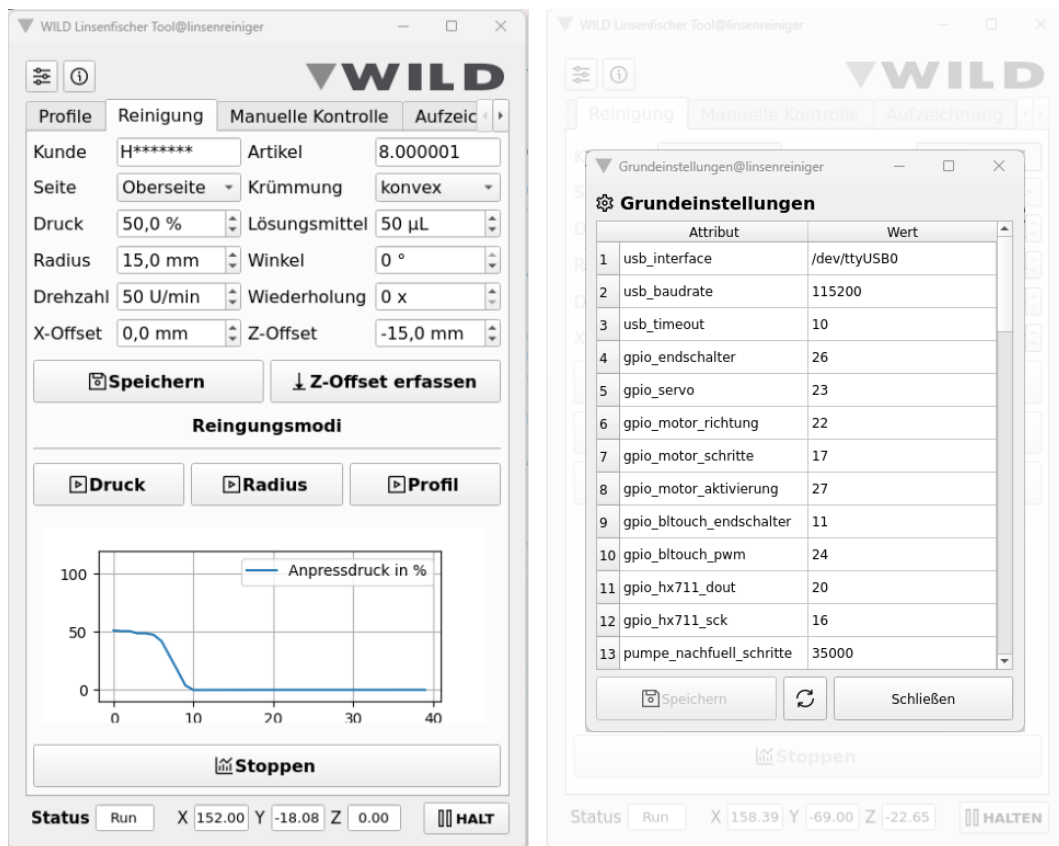


Abbildung 30: Auf der linken Seite ist das Hauptfenster des Programms zu sehen. Die rechte Seite zeigt das geöffnete Einstellungen-Fenster.

Über die Datei 'einstellungen.ui' wird das Fenster mit den Grundeinstellungen aufgebaut. In ihm werden die aus der 'einstellungen.cfg' eingelesenen Attribute und Werte in Form einer Tabelle ausgegeben.

Zur besseren Verdeutlichung der Funktion, ist jeder Button zusätzlich mit einem Icon versehen. Alle dafür notwendigen Bilddateien sind im Ordner '`\logos`' abgelegt und die Verknüpfung mit der Design-Datei erfolgt unter der Angabe des relativen Speicherpfades.

2.6.4 Programmstart

Damit das Programm nicht über die Konsole gestartet werden muss, ist am Desktop des Raspberry Pi die Verknüpfung '`linsenfischer.dekstop`' abgelegt. Wird diese geöffnet, startet im Hintergrund das ausführbare Shell-Skript '`run.sh`', in welchem dem Python-Interpreter die Datei '`main.py`' aus dem Root-Verzeichnis des Softwareprojekts übergeben wird. Für einen manuellen Start des Programms, muss zuerst in das Arbeitsverzeichnis gewechselt werden:

```
$ cd ~/Python/linsenfischer      #Wechselt das Programmverzeichnis
$ python main.py                 #Führt die Robotersoftware aus
```

2.6.5 Interrupts

Bei der Erstellung von Optikprofilen spielt die Reaktionszeit der zentralen Steuerung eine wesentliche Rolle. Zur Erfassung der Z-Koordinate, senkt die Steuerung den Z-Schlitten mittels Schleife in 0.01 mm Schritten ab. Trifft die Sonde des Sensors auf die Oberfläche, wechselt der Logikpegel am GPIO-Pin 11 von HIGH auf LOW und stoppt dadurch die Absenkung. Reagiert die Steuerung zu langsam, wird die tatsächliche Z-Koordinate mit jedem weiteren Schleifendurchlauf um 0.01 mm verfälscht.

Aus diesem Grund wird die Pegeländerung am GPIO-Pin über *Interrupts* erfasst. Bei *Interrupts* wird im Gegensatz zum *Polling* sofort auf eine Änderung reagiert, in dem das aktuelle Programm unterbrochen wird [54]. Das spart Zeit und verhindert dadurch falsche Ergebnisse bei der Abtastung des Profils.

2.6.6 Multithreading

Während einer Reinigung ist es erforderlich, dass die zentrale Steuerung gewisse Aufgaben gleichzeitig ausführt. Darunter fällt z.B. die permanente Abfrage der aktuellen Position, das Auslesen des Kraftsensors oder auch die Positionierung der Schlitten. Damit die zeitliche Abfolge des Reinigungsprozesses möglichst unbeeinflusst bleibt, werden *Threads* eingesetzt. Python stellt dafür ein eigenes Threading-Modul zur Verfügung, welches es ermöglicht, Aufgaben zumindest quasi-parallel abzuarbeiten. Neben den eben genannten Tasks, wird auch die Live-Anzeige des Druckverlaufs über Threads abgewickelt. Eine Echtzeitanzeige während der Reinigung wäre ohne

Threading nicht möglich, da die permanente Aktualisierung des Diagramms relativ viel Zeit in Anspruch nimmt. Das würde den ohnehin zeitkritischen Reinigungsprozess stark verlangsamen und zu unvorhersehbaren zeitlichen Konflikten führen.

Im gegenwärtigen Softwareprojekt werden neben dem Main-Thread, welcher unter Anderem die Aktualisierung der Benutzeroberfläche verwaltet, gleichzeitig bis zu vier weitere Worker-Threads ausgeführt.

Das Prinzip *Multithreading* löst zwar etwaige Timing-Probleme, führt aber dafür zu neuen Herausforderungen, auf welche in den Kapiteln *Lock-Objekte* und *Signals* auf Seite 73 näher eingegangen wird.

2.6.7 Lock-Objekte

Bei der Anwendung von *Multithreading* kann es passieren, dass unter gewissen Umständen mehrere Threads gleichzeitig auf ein und dieselbe Ressource zugreifen. Im gegenwärtigen Fall betrifft das die serielle Schnittstelle (USB), über welche mit dem CNC-Controller kommuniziert wird, siehe Abbildung 31.

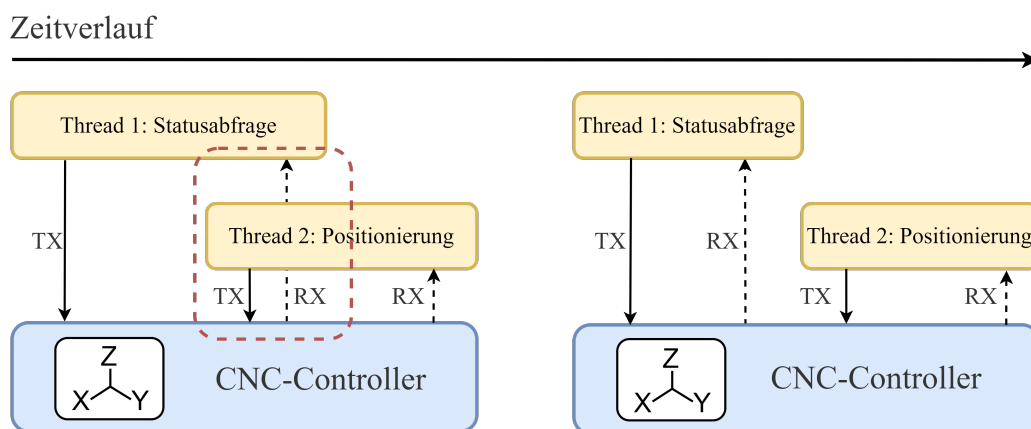


Abbildung 31: Die linke Seite visualisiert die Situation, welche durch *Multithreading* auftreten kann. Thread 2 versucht Daten an den CNC-Controller zu senden obwohl die USB-Schnittstelle noch durch Thread 1 belegt ist. Auf der rechten Seite erfolgt der Zugriff nacheinander und somit ohne Überschneidungen.

Um einen Mehrfachzugriff zu verhindern, bietet *Python* die Möglichkeit kritische Programmabschnitte über ein *Lock-Objekt* zu sperren [55]. Im konkreten Fall sperrt Thread 1 den Zugriff auf die serielle Schnittstelle, bis die angeforderte Information der Statusabfrage empfangen werden. Erst dann wird der Codeabschnitt für Thread 2 wieder freigegeben, welcher einen Positionierungsbefehl an den Controller sendet.

2.6.8 Signals

Eine weitere Eigenschaft, welche bei der Anwendung von Multithreading beachtet werden muss ist, dass die direkte Manipulation von Objekten der Benutzeroberfläche (z.B. Labels) ausschließlich über den erzeugenden, in diesem Fall dem Main-Thread, möglich ist.

In einem konkreten Anwendungsfall (siehe dazu Abbildung 32) soll dem Anwender über ein Popup mitgeteilt werden, dass die Reinigung abgeschlossen ist und die Optik entnommen werden kann. Da der Reinigungsprozess ebenfalls über einen Thread abgewickelt wird, lässt sich das Popup, wie bereits erwähnt, nicht über diesen erzeugen. Erzeugt man das Popup-Fenster direkt über den Main-Thread, erscheint es auf Grund der Parallelisierung bereits zum Start der Reinigung, und damit zu früh.

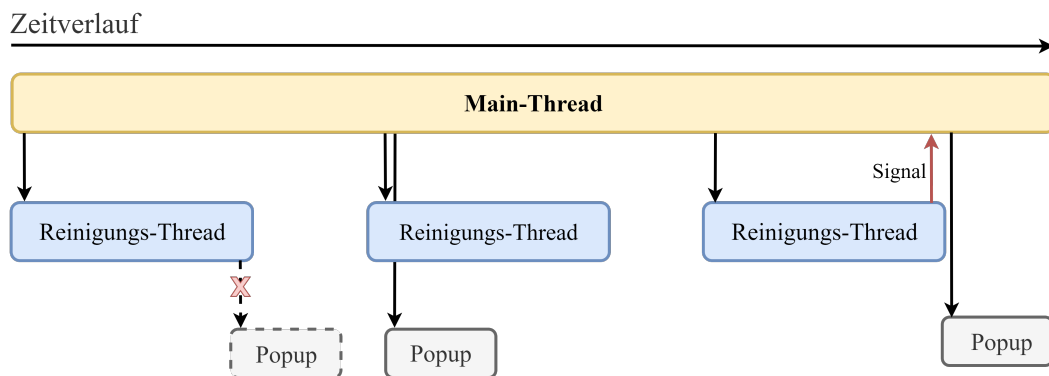


Abbildung 32: Visualisierung der Situation anhand eines Blockdiagrammes

Links: Popup lässt sich nicht erstellen

Mitte: Popup wird zum falschen Zeitpunkt erstellt

Rechts: Popup wird zum korrekten Zeitpunkt erstellt

Damit das Popup im richtigen Moment geöffnet wird, wurden *Signals* verwendet. Über *Signals* lässt sich eine Kommunikation zwischen Main-Thread und Reinigungs-Thread aufbauen. Der Reinigungs-Thread teilt dem Main-Thread mit, zu welchem Zeitpunkt Änderungen der Benutzeroberfläche durchgeführt werden sollen. Das Prinzip kommt auch bei der Abtastung eines Oberflächenprofils zur Anwendung. In diesem Fall signalisiert der Abtast-Thread dem Main-Thread nach jedem abgetasteten Punkt, dass der Fortschrittsbalken (Reiter *Aufzeichnung*) aktualisiert werden muss.

2.6.9 Übersicht der Programmfunktionen

Abbildung 33 auf Seite 74 zeigt eine Übersicht der einzelnen Programmfunktionen der Robotersoftware.

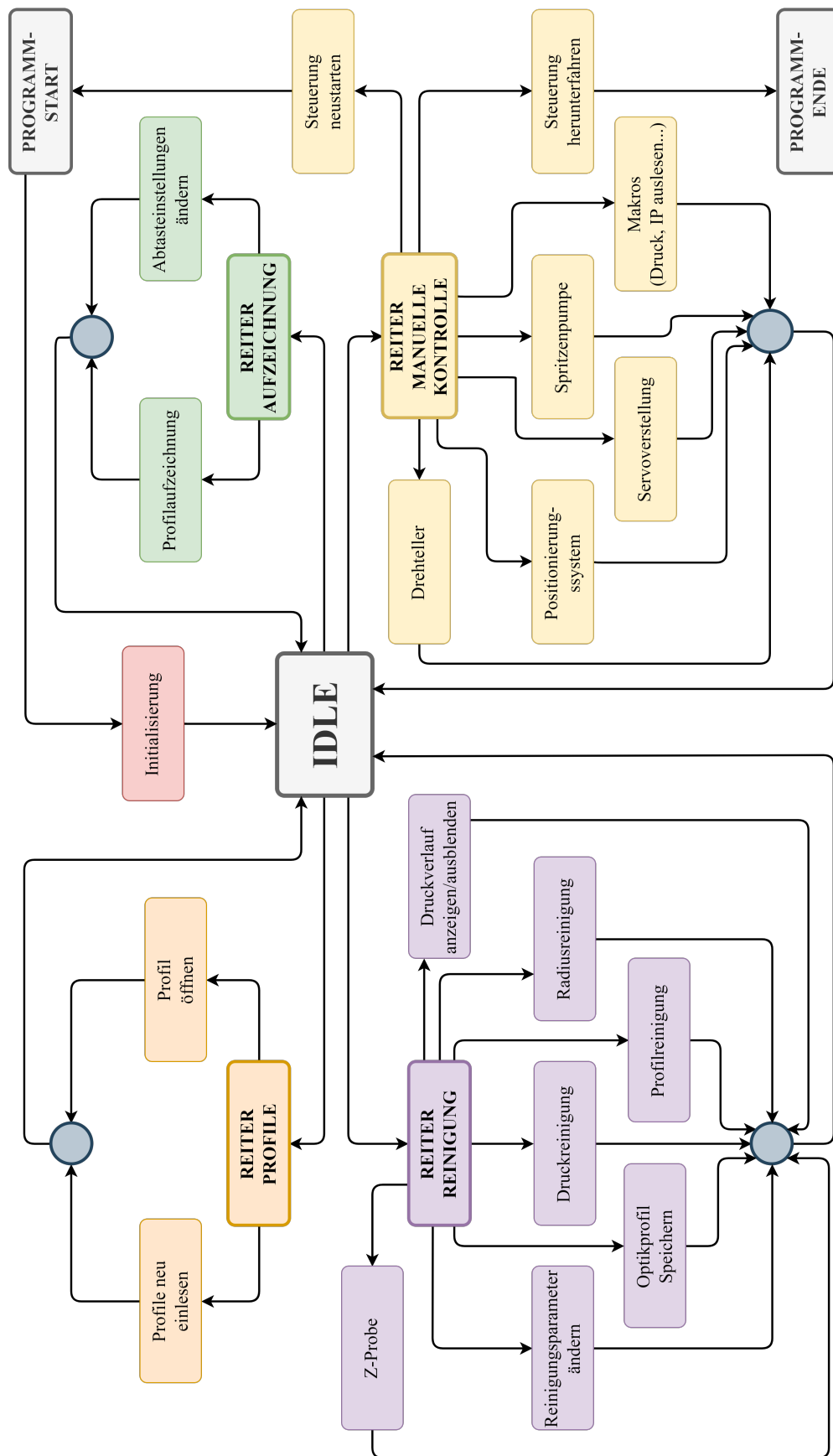


Abbildung 33: Nach Registerkarte kategorisierte Übersicht der einzelnen Funktionen der Robotersoftware

2.6.10 Dokumentation

Die Dokumentation der Software basiert auf *DocStrings* und Inline-Kommentaren. *DocStrings* sind Kommentare die auch über mehrere Zeilen reichen können und mit drei einfachen ('''') oder doppelten (""") Anführungszeichen beginnen, als auch enden.

Im Gegensatz zu einzeiligen Kommentaren, welche in Python mit einem *Hashtag* ('#') initiiert werden, lassen sich *DocStrings* mit geeigneten Tools wie *Sphinx* auslesen, um daraus automatisiert eine Dokumentation des Softwareprojekts zu erstellen. Voraussetzung ist, dass ein bestimmtes Format eingehalten wird. Ein typisches *DocString*-Kommentar einer Funktion sieht wie folgt aus:

```
"""Berechnet die an den CNC-Controller zu sendende Drehzahl
    über die lineare Regression.

Args:
    gewuenschte_drehzahl (int): Drehzahl in 1/min.
    steigung (float): Steigung der Geradengleichung.
    offset (float): Offset der Geradengleichung.

Returns:
    float: Gibt die korrigierte Drehzahl in 1/min zurück.
"""
```

Obiges Beispiel entspricht dem *DocString*-Stil von Google, zu erkennen an den Schlüsselwörtern 'Args' und 'Returns'. Sphinx durchsucht den gesamten Quellcode nach *DocStrings*, sammelt die darin befindlichen Informationen und generiert daraus die Dokumentation, welche sich im Anschluss in unterschiedlichen Formaten wie z.B. *HTML* (Hypertext Markup Language) oder *PDF* (Portable Document Format) ausgegeben lässt. [56]

2.7 Versuchsreihen

Nach der ersten Inbetriebnahme wurde der Roboter unterschiedlichen Versuchsreihen unterzogen. Die daraus gewonnenen Ergebnisse dienen einerseits dazu, Präzision und Linearität der verbauten Sensoren zu spezifizieren und andererseits um den Reinigungserfolg zu evaluieren.

2.7.1 Abtastsensoren

Präzision bei Einzelpunktabtastung

Bei der Profilabtastung wurde zunächst für beide Sensoren (Nivellierungs- und kapazitiver Näherungssensor) ermittelt, wie hoch die punktuelle Abtastpräzision ist. Dazu wurden drei Punkte entlang der Oberfläche eines Achromates ($\varnothing = 30\text{ mm}$) jeweils 200 Mal abgetastet um daraus Messunsicherheit und Mittelwert zu bestimmen. Die Absenkung des Z-Schlitten erfolgte mit einer Auflösung von 0.01 mm .

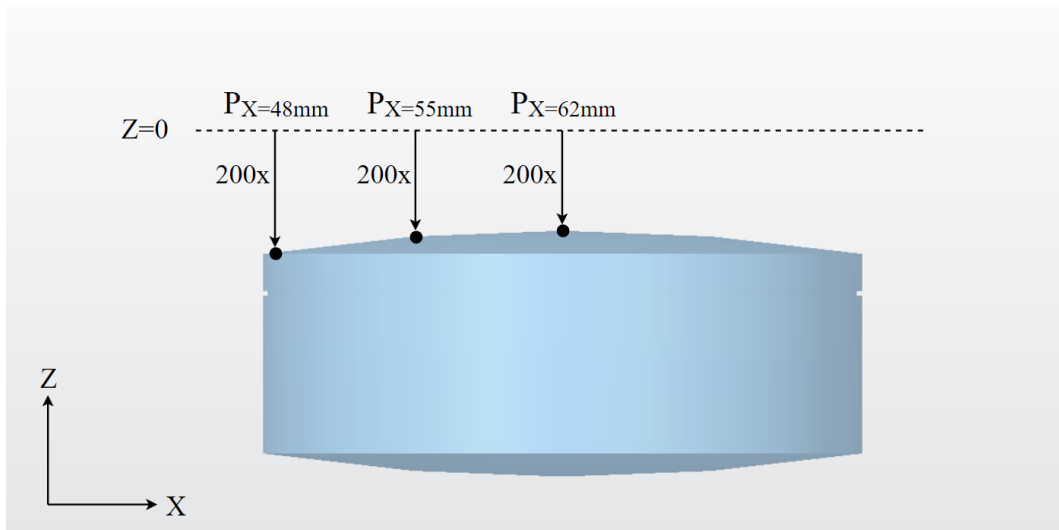


Abbildung 34: Achromat mit eingezeichneten Abtastpunkten

Präzision eines Oberflächenprofils

Die beiden Sensoren wurden im Anschluss daran noch einer weiteren Testreihe unterzogen um herauszufinden, ob sich die Messwerte auch nach einem zwischenzeitlichen Verschiebung in der X-Achse reproduzieren lassen.

Dazu wurden jeweils 10 Messpunkte (im Abstand von $d=1.61\text{ mm}$) entlang der Oberfläche erfasst um daraus ein Profil zu erstellen. Jeder der 10 Messpunkte entspricht dem Median aus fünf Einzelpunktabtastungen. Die Absenkung des Z-Schlitten erfolgte, wie schon bei der Einzelpunktabtastung, mit einer Auflösung von 0.01 mm .

Abbildung 35 veranschaulicht das Vorgehen anhand einer Grafik. Insgesamt wurden je Sensor drei Oberflächenprofile aufgenommen und verglichen.

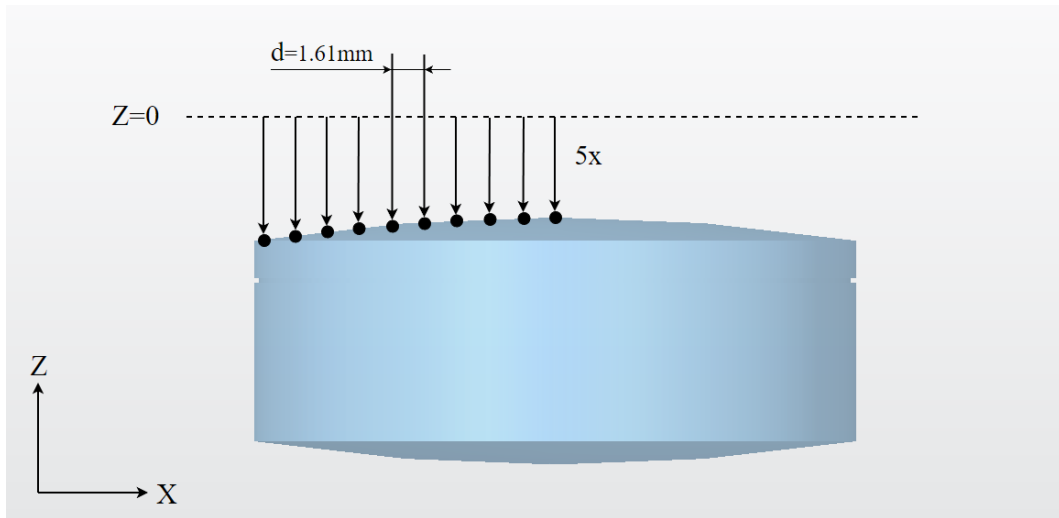


Abbildung 35: Darstellung der Abtastpunkte bei Aufnahme eines Oberflächenprofils eines Achromates

Auf Basis der Ergebnisse von Einzelpunkt- und Profilabtastung wurde entschieden, welcher der beiden Sensoren sich besser für den Einsatz im Reinigungsroboter eignet.

2.7.2 Wägezelle

Da die Druck- bzw. Kraftmessung einen essentiellen Teil aller drei Reinigungsmodi darstellt, wurde auch die Wägezelle zwei Versuchen unterzogen, um daraus Präzision und Linearität des Sensors zu ermitteln.

Präzision des Kraftsensors

Begonnen wurde damit, das Verhalten im Dauerbetrieb zu untersuchen, um mögliche Tendenzen zu identifizieren und die Präzision zu bestimmen. Dafür wurde der Reinigungskopf für den Druckaufbau zuerst auf $Z=-22\text{ mm}$ abgesenkt und unmittelbar danach wieder angehoben, um die Wägezelle wieder zu entlasten. Dieser Vorgang wurde unter Einhaltung einer 2.5 sekundigen Pausenzeit 200 Mal durchgeführt. Im Anschluss daran wurde derselbe Versuch noch mit einer Pausenzeit von 5 s bzw. 7.5 s wiederholt. Eine Visualisierung des Versuches ist in Abbildung 36 auf Seite 78 zu sehen.

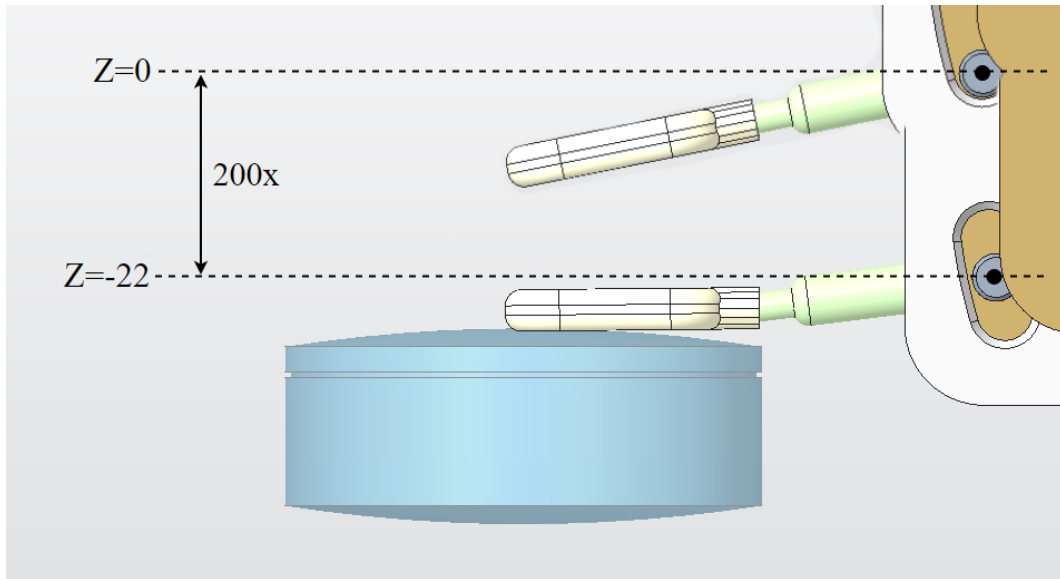


Abbildung 36: Methode zur Ermittlung der Präzision des Kraftsensors. Die Abbildung soll außerdem verdeutlichen, dass sich das Pad aufgrund der Flexibilität des Schaftes an die gekrümmte Oberfläche der Optik anpasst.

Linearität

Eine weitere, untersuchenswerte Eigenschaft einer Wägezelle ist die Linearität. Zur Ermittlung dieser, wurde der Reinigungskopf in Zehntel- Millimeter Schritten von $Z=-17.0$ mm auf $Z=-26$ mm abgesenkt und dabei der Druck bzw. die auf das Pad wirkende Kraft, nach jedem Schritt gemessen und gespeichert.

2.7.3 Ermittlung der tatsächlichen Drehzahl

Die Rotationsgeschwindigkeit des Drehtellers lässt sich, wie bereits erwähnt, über den CNC-Controller steuern. Da die an der Motorwelle abgegebene Drehzahl sowohl über das Planetengetriebe, als auch über den Riemenantrieb übersetzt wird, stimmt die eingestellte Drehzahl nicht mit der tatsächlichen Rotationsgeschwindigkeit des Drehtellers überein. Außerdem generiert der CNC-Controller ein unsauberes PWM-Signal, was die Abweichung noch zusätzlich erhöht. Aus diesem Grund wurde die tatsächliche Rotationsgeschwindigkeit n_t gemessen und der eingestellten Drehzahl n_e gegenübergestellt. Durch Linearisierung des so ermittelten Drehzahl-Drehzahl Verlaufes, konnten die Koeffizienten p_1 und p_2 für die Korrektur bestimmt und in die Grundeinstellungen der Robotersoftware eingepflegt werden. Der Zusammenhang zwischen den Drehzahlen lässt sich über die umgeformte Geradengleichung (15) beschreiben.

$$n_e = \frac{n_t - p_2}{p_1} \quad (15)$$

Damit der Drehteller letztendlich die gewünschte Drehzahl n_t erreicht, muss die korrigierte Drehzahl n_e an den Controller gesendet werden.

Für die Messung der Drehzahl wurde das Linienfolger Breakout Board *KY-033* verwendet, welches in Abbildung 37 mit (1) gekennzeichnet ist. Der Linienfolger besteht im Grunde aus einer IR-Empfangs- und einer IR-Sendediode und kann je nach Intensität des reflektierten Infrarotlichts zwischen hellen und dunklen Flächen unterscheiden. Überschreitet das empfangene Infrarotlicht einen bestimmten Schwellwert, so ändert sich der Logikpegel am Ausgangspin des Sensors.

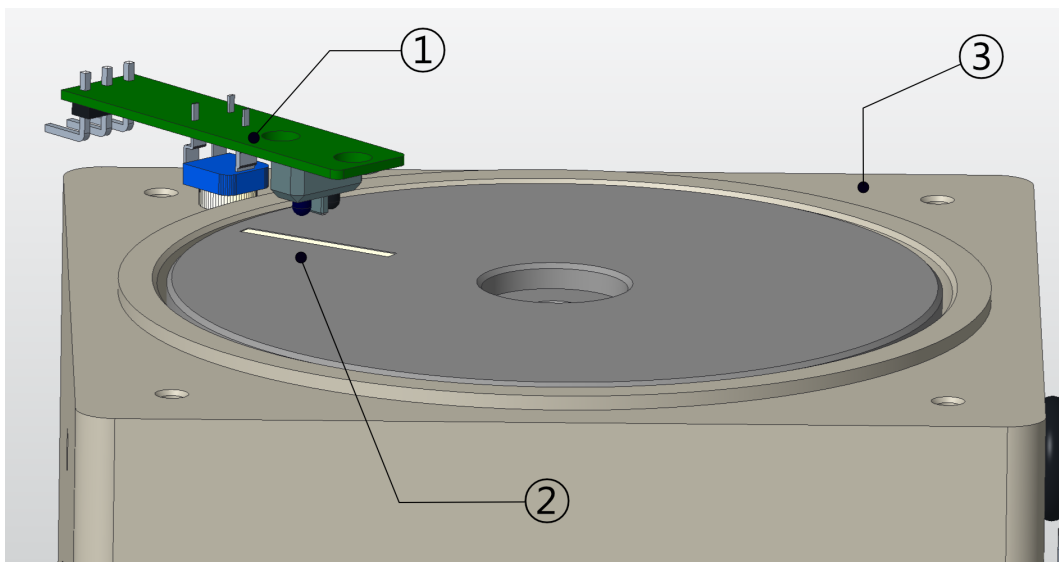


Abbildung 37: Drehzahlmessung mittels Sensorplatine KY-033

- (1) Sensorplatine mit Sende- und Empfangsdiode
- (2) Drehteller mit Markierung
- (3) Getriebeblock

Durch die Rotation des Drehtellers, ändert sich der Pegel in regelmäßigen Abständen. Über die verstrichene Zeit zwischen zwei Pegeländerungen kann auf die Umdrehungen pro Minute zurückgerechnet werden. Für ein genaueres Ergebnis wurde je Drehzahl der Median aus 20 Zeitmessungen gebildet. Die Erfassung erfolgte mit Hilfe des Microcontrollers *Arduino UNO R3*.

2.7.4 Testaufbau zur Evaluierung der Reinigung

Für die Evaluierung der Reinigung wurden die Oberflächen zweier unterschiedlicher Optiken untersucht, welche in Abbildung 38 ersichtlich sind. Geprüft wurde, wie effizient sich Staubablagerungen, fett- und schweiß-basierte Fingerabdrücke sowie Speicheltröpfchen entfernen lassen, da diese Arten von Verschmutzungen in der Praxis am häufigsten auftreten.



Abbildung 38: Links ein Achromat aus einem optischen Kohärenztomografen (Augenheilkunde) und rechts eine sphärische Linse mit Zentrierbohrung

Partikuläre Verunreinigungen

Was die partikulären Verunreinigungen betrifft, so wurde der Reinigungserfolg quantifiziert, indem man die Partikel vor und nach der Reinigung gezählt und der Größe nach klassifiziert hat. Für jede Größenklasse wurde im Anschluss die Reinigungseffizienz RE separat bestimmt. Das Vorgehen lehnt an die *Klassifizierung der Abreinigbarkeit* an, welche im Buch „*Reinraumtechnik*“ von Lothar Gail et al. [57] beschrieben wird.

Formel (16) zeigt, wie sich die Reinigungseffizienz für Partikel $\geq 10\mu m$ bestimmen lässt. Die Variablen n_{davor} und n_{danach} entsprechen der Partikelanzahl vor bzw. nach dem Reinigungsvorgang. Das Ergebnis der Berechnung wird in Prozent ausgegeben.

$$RE_{\geq 10\mu m} = 100 \cdot \left(1 - \frac{n_{danach, 10\mu m}}{n_{davor, 10\mu m}} \right) \quad (16)$$

Abbildung 39 auf Seite 81 zeigt, dass die Aufnahmen bearbeitet werden müssen, damit sich Größe und Anzahl der Partikel per Software bestimmen lassen. Es ist außerdem notwendig, den Maßstab der Aufnahme an den Maßstab der Software anzugleichen.

Damit sich die meist sehr unterschiedlich geformten Partikel bezüglich ihrer Größe vergleichen lassen, wurde jeweils der *Feret*-Durchmesser (Äquivalentdurchmesser) bestimmt. Alle durchgeführten Bildaufbereitungen und Analysen wurden mit Hilfe der Bildbearbeitungssoftware *ImageJ* (National Institutes of Health, Bethesda, USA) durchgeführt.

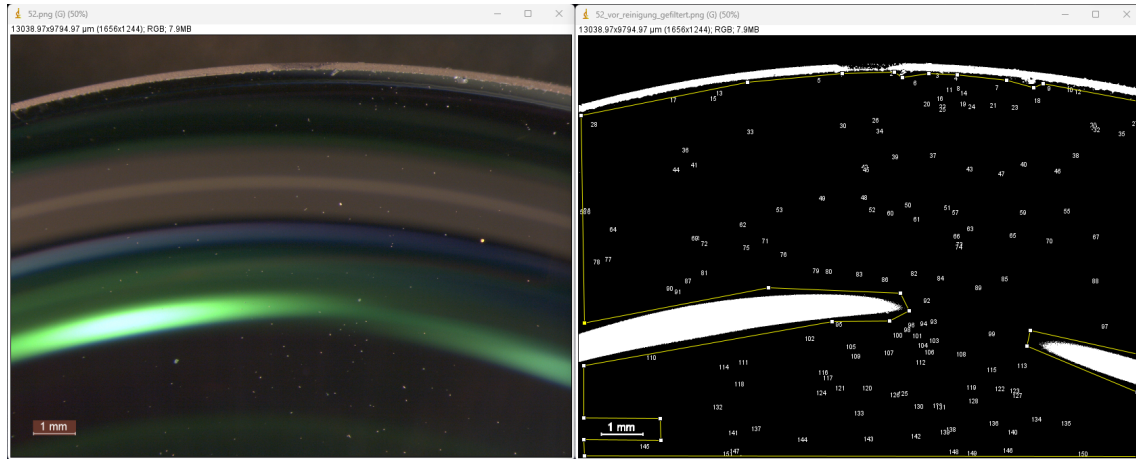


Abbildung 39: Die linke Seite zeigt die originale Aufnahme einer verschmutzten Optik. Auf der rechten Seite sieht man die bearbeitete Version, in welcher alle erkannten Partikel mit einer ID-Nummer gekennzeichnet sind.

Filmige Verschmutzungen

Im Gegensatz zu den partikulären Verschmutzungen ist das Verfahren zur Quantifizierung von NVRs wesentlich aufwändiger und verlangt, wie bereits in Kapitel 1.5 erwähnt, hochgenaue Geräte zur Gewichtsbestimmung. In dieser Arbeit wurden die Reinigungsergebnisse für filmige Verunreinigungen (Fingerabdrücke, Speicheltropfen) durch visuelle Interpretation bestimmt.

Messaufbau

Abbildung 40 auf Seite 82 zeigt das Setup, mit welchem die Optikoberflächen untersucht wurden. Zur Anwendung kam das Digital-Mikroskop *M80* und der Bildsensor *MC190 HD* (Leica, Wetzlar, Deutschland) in Verbindung mit der Software *Leica Application Suite*.

Damit sowohl vor als auch nach der Reinigung derselbe Bereich der Oberfläche betrachtet wird, wurde die Optikkante mit einem Permanentmarker markiert. Als Lösungsmittel kam hauptsächlich Aceton in verschiedenen Mengen zum Einsatz.

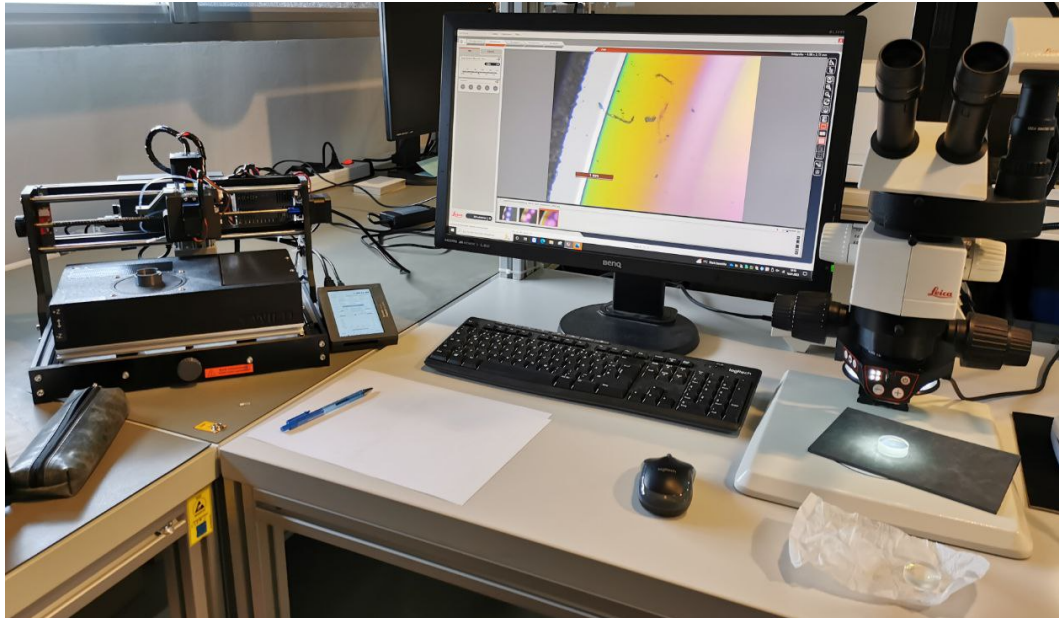


Abbildung 40: Die linke Seite des Bildes zeigt den Reinigungsroboter, rechts ist das Digital-Mikroskop zur Begutachtung der Optikoberflächen zu sehen.

2.7.5 Auswertung und Darstellung der Ergebnisse

Die aus den Messergebnissen berechneten Mittelwerte und Standardabweichungen wurden mit der Software *Matlab* (Mathworks Inc, Natick, USA) bestimmt. *Matlab* wurde außerdem dazu verwendet, die Ergebnisse in unterschiedlichen Plots darzustellen.

3 Ergebnisse

3.1 Profilabtastung

3.1.1 Punktuelle Wiederholgenauigkeit

Abbildung 41 zeigt die Ergebnisse der Einzelpunktabtastung mittels Nivellierungssensor an drei unterschiedlichen Punkten des Achromates. Die Ergebnisse in Tabelle 18 basieren auf jeweils 200 Messwerten.

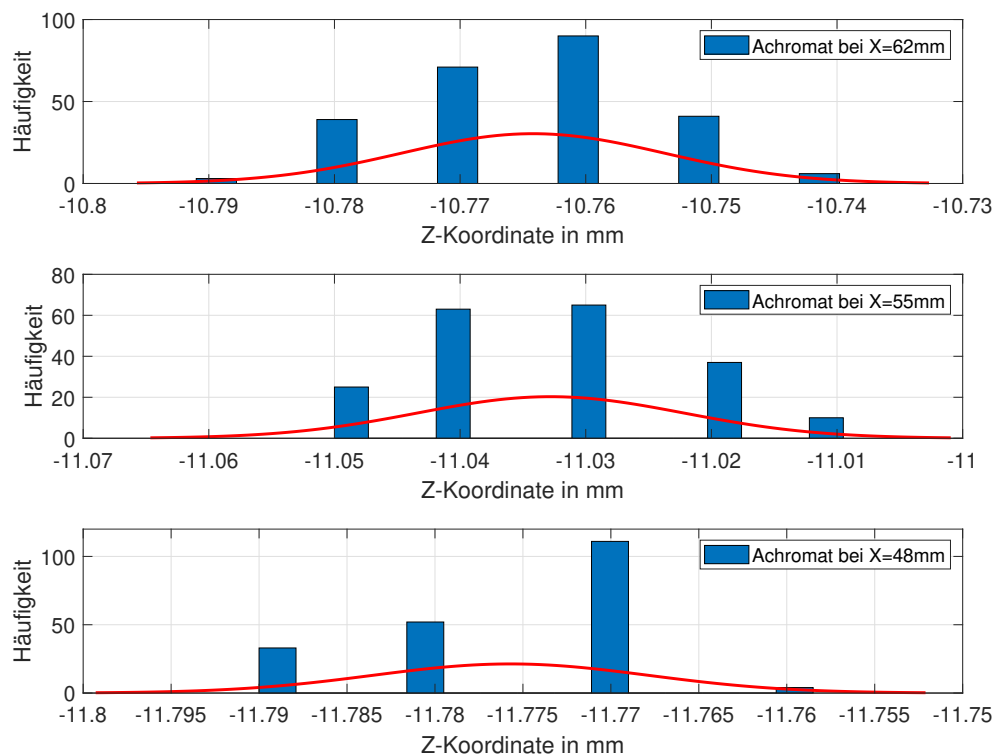


Abbildung 41: Histogramme der Einzelpunktabtastung bei den Punkten X=62mm (oben), X=55mm (mittig) und X=48mm unten

Tabelle 18: Statistische Kennzahlen des Nivellierungssensors.

Punkt	X-Position	Mittelwert μ	Standardabweichung σ
1	62 mm	-10.7642 mm	10.5 μm
2	55 mm	-11.0328 mm	10.6 μm
3	48 mm	-11.7757 mm	7.9 μm

Analog zum Histogramm des Nivellierungssensors, sind in Abbildung 42 die Messwertverteilungen des kapazitiven Näherungsschalters abgebildet. Die statistischen Kennzahlen in Tabelle 19 zeigen, dass der Näherungsschalter im Vergleich zum Nivellierungssensor eine fast 5-fach höhere Standardabweichung aufweist.

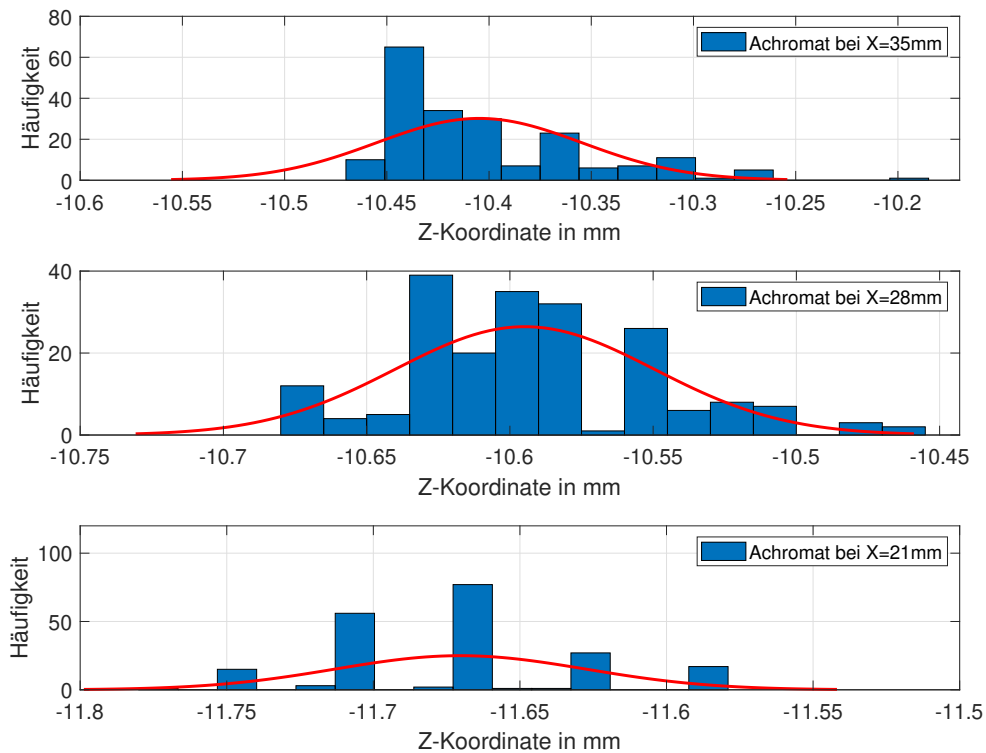


Abbildung 42: Ergebnisse der Einzelpunktabtastung mittels kapazitivem Näherungsschalter an wiederum drei unterschiedlichen Oberflächenpunkten des Achromates

Tabelle 19: Mittelwerte und Standardabweichungen des kapazitiven Näherungsschalters.

Punkt	X-Position	Mittelwert μ	Standardabweichung σ
1	35 mm	-10.4048 mm	50.2 μm
2	28 mm	-10.5948 mm	45.3 μm
3	21 mm	-11.6706 mm	42.8 μm

3.1.2 Wiederholgenauigkeit der Profilabtastung

In Abbildung 43 werden die Ergebnisse der Profilabtastung dargestellt. Die höhere Präzision des Nivellierungssensors spiegelt sich auch bei den aufgenommenen Profilen wieder. Auffallend ist, dass die Ergebnisse im direkten Vergleich stark voneinander abweichen, siehe „Direkter Vergleich“.

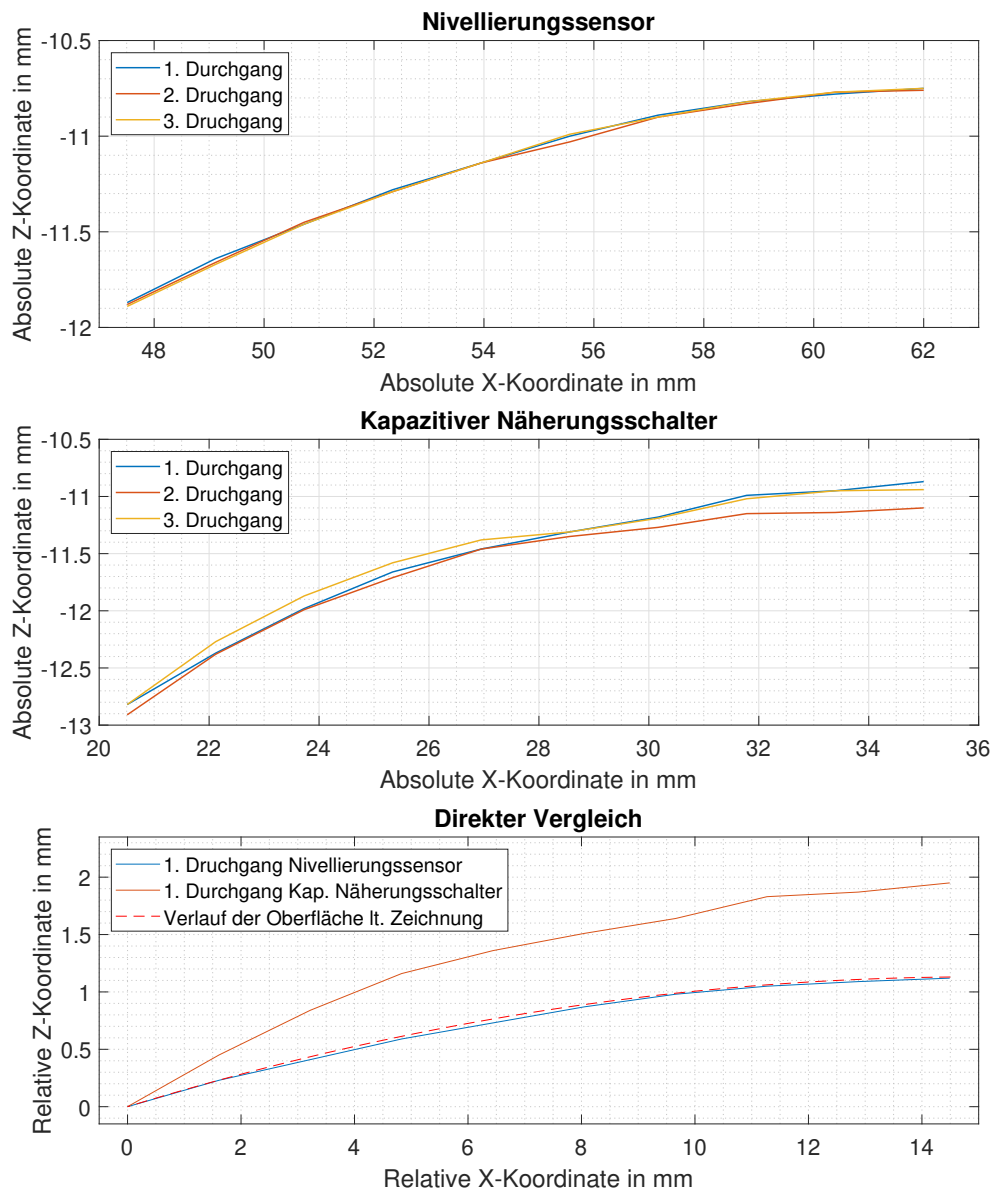


Abbildung 43: Ergebnisse bei mehrfacher Profilabtastung desselben Achromates. Bei der Interpretation der Diagramme ist zu beachten, das X- und Z-Achse einen unterschiedlichen Maßstab aufweisen.

3.2 Druckmessung

Präzision der Wägezelle

Die Auswertungen in Abbildung 44 zeigen, dass die Pausenzeiten zwischen den einzelnen Messungen einen Einfluss auf den tendenziellen Verlauf des Druckwertes haben. Je höher die Pause, desto weniger Streuung ist zu erwarten, siehe Tabelle 20.

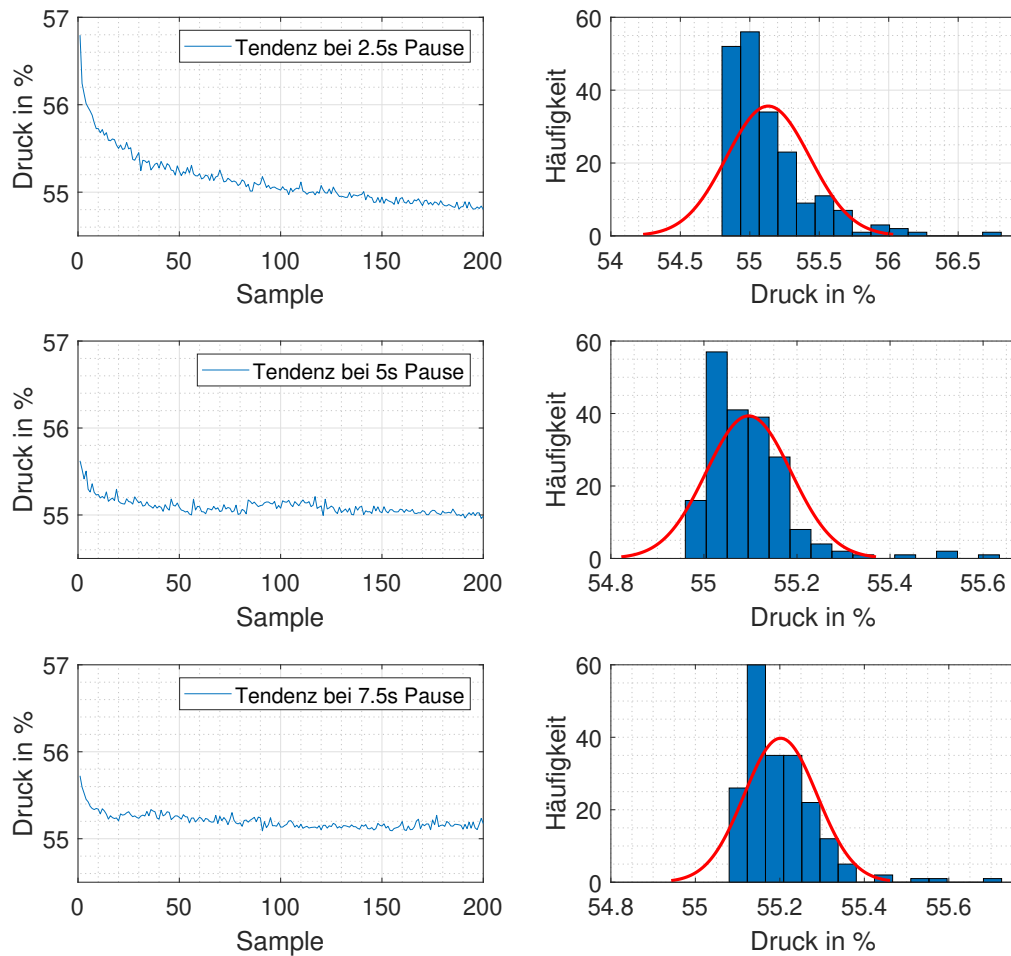


Abbildung 44: Präzision der Wägezelle bei Absenkung auf Z=-22mm

Tabelle 20: Statistische Auswertung der Messergebnisse des Kraftsensors.

Serie	Messabstand	Mittelwert μ	Standardabweichung σ
1	2.5 s	55.1323 %	0.3001 %
2	5.0 s	55.0962 %	0.0912 %
3	7.5 s	55.2018%	0.0863 %

Linearität der Wägezelle

Das Diagramm in Abbildung 45 vergleicht den durch Messungen erfassten realen Druckverlauf mit der durch *Matlab* gefitteten Idealkurve, dessen Parameter in Tabelle 21 zu finden sind. Der reale Verlauf ist annähernd linear und weicht von der Idealkurve hauptsächlich bei geringer Belastung der Wägezelle ab.

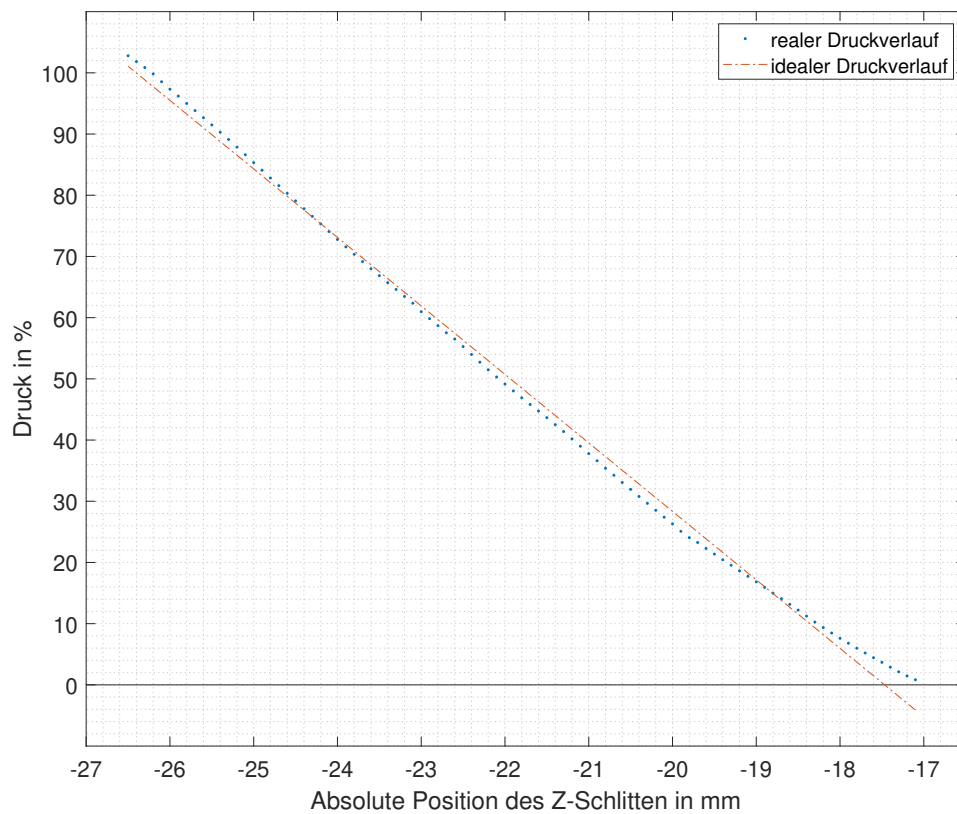


Abbildung 45: Druckverlauf bei Absenkung des Z-Schlittens von -17.0 mm auf -26.5 mm in einem Intervall von 0.1 mm. Die gefittete Idealkurve folgt der Geradengleichung $y = k \cdot x + d$.

Tabelle 21: Parameter der gefitteten Idealkurve.

Parameter	Wert
k	-11.1919
d	-195.4960

Druckverläufe der unterschiedlichen Reinigungsmodi

In Abbildung 46 sind die Ergebnisse der Druckmessung während einer Reinigung abgebildet. Für alle drei Modi wurde dasselbe Achromat herangezogen. Die blauen Verläufe stellen den optimalen Verlauf bei korrekt eingesetztem Reinigungsstäbchen bzw. nach Optimierung der Reinigungsalgorithmen dar. Negativbeispiele (fehlende Optimierung, falsch eingesetztes Stäbchen) sind als oranger Verlauf dargestellt.

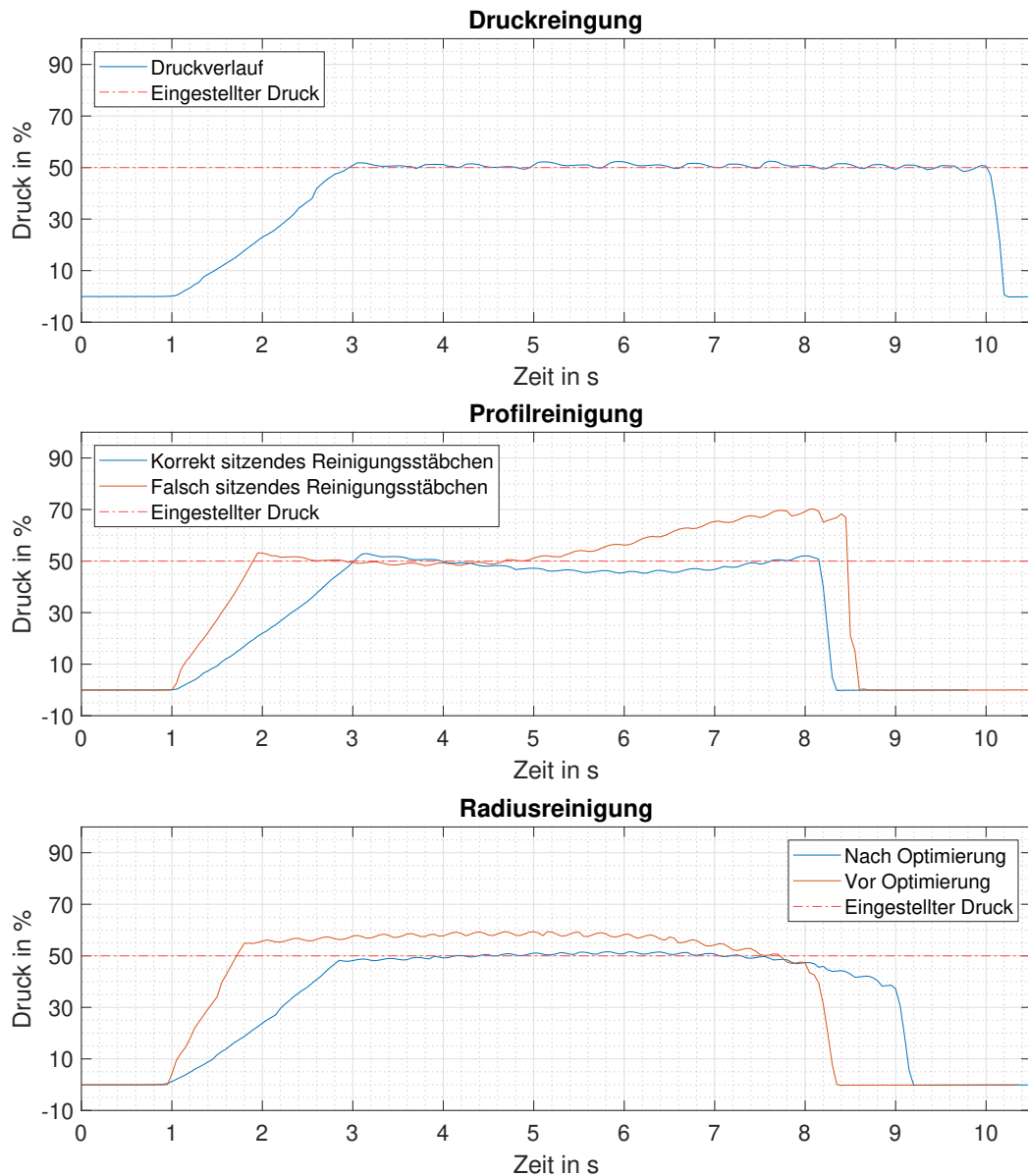


Abbildung 46: Vergleich der Druckverläufe bei unterschiedlichem Reinigungsmodus. Die Druckreinigung braucht mit über 10 Sekunden deutlich am längsten.

3.3 Drehteller

3.3.1 PWM-Signal

Der in Abbildung 47 ersichtliche Signalverlauf zeigt, dass der CNC-Controller kein perfektes PWM-Signal ausgibt. Die Spannung steigt schon vor der eigentlichen Ein-zeit des Signales auf ca. 6.5 V und verursacht damit einen Anstieg der mittleren Spannung $\overline{U_{PWM}}$. Bei einer Impulsbreite von $t_{ein}=137.6 \mu s$ sollte laut Berechnungsvorschrift (3) auf Seite 42, theoretisch eine mittlere Spannung von $\overline{U_{PWM}} = 3.96 \text{ V}$ ausgegeben werden, tatsächlich liegt diese aber bei 7.6 V.

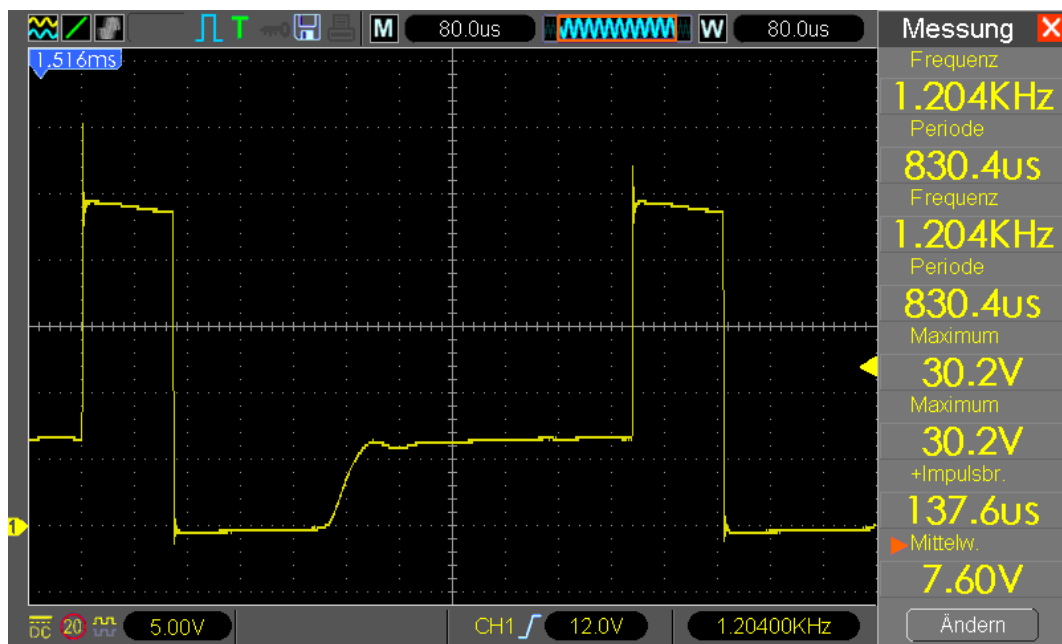


Abbildung 47: Reales PWM-Signal, welches vom der CNC-Controller zur Ansteuerung des DC-Getriebemotors generiert wird.

3.3.2 Linearität der Drehzahl

In Abbildung 48 ist der Zusammenhang zwischen eingestellter und gemessener Drehzahl ersichtlich. Die orange, strich-punktierte Gerade entspricht dem idealen Verlauf, welcher mit Hilfe der lineare Regression erstellt wurde.

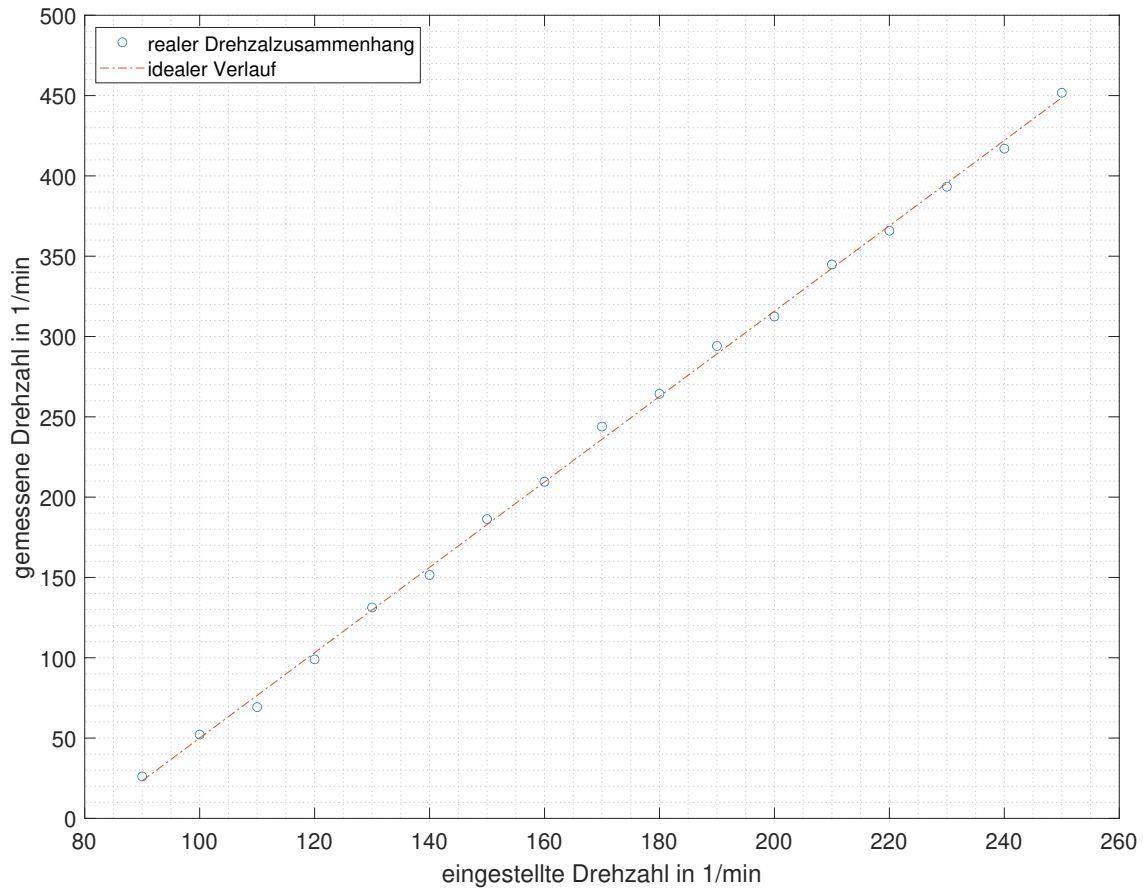


Abbildung 48: Ergebnisse der Drehzahlmessung

Rotationen unter 25 min^{-1} konnten nicht erfasst werden, da bei niederen Drehzahlen die mittlere Spannung des PWM-Signales $\overline{U_{PWM}}$ zu gering ist, um das Losbrechmoment des DC-Getriebemotors zu überwinden.

Tabelle 22: Polynomkoeffizienten der Linearisierung 1. Grades

Koeffizient	Wert
p_1	2.6581
p_2	-215.8110

Mit Hilfe der Polynomkoeffizienten in Tabelle 22 lässt sich die Drehzahl korrigieren, welche an den CNC-Controller gesendet werden muss, um den Drehteller mit der eingestellten Drehzahl rotieren zu lassen.

3.4 Reinigungsergebnisse

3.4.1 Partikuläre Kontaminationen

In Abbildung 49 sieht man die sphärische Linse vor und nach der Reinigung durch den Roboter. Das Farbenspiel ergibt sich aufgrund der Oberflächenvergütung der Optik, in diesem Fall eine Anti-Reflexbeschichtung für eine Wellenlänge von 1550 nm.

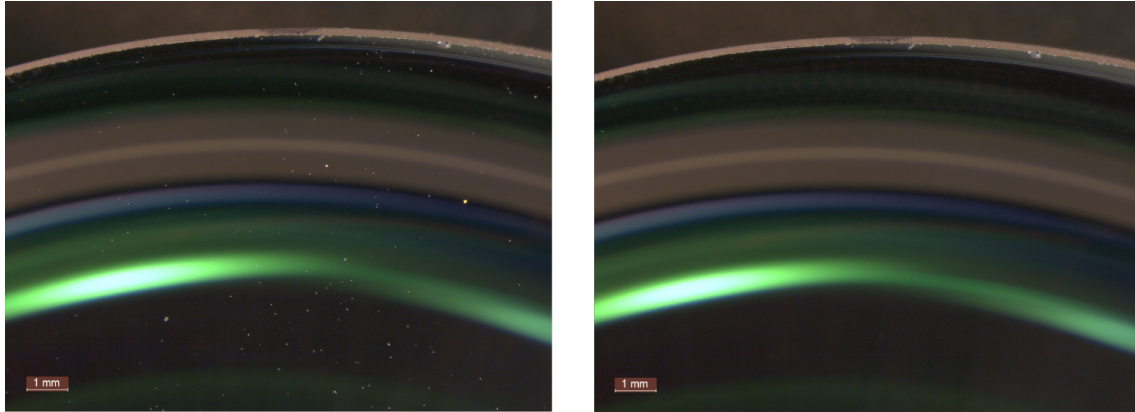


Abbildung 49: Das linke Bild zeigt die verunreinigte sphärische Linse direkt nach der Entnahme aus der Schutzverpackung. Die erfassten Äquivalentdurchmesser (D_{eq}) der Partikel liegen zwischen 11.13 und 120.19 μm . Rechts zu sehen, dieselbe Linse nach der Reinigung mittels Roboter. Die Reinigung wurde mit einem Druck von 50 %, einer Lösungsmittelmenge von 50 μl und einer Rotation von 120 min^{-1} durchgeführt.

Tabelle 23 fasst die Ergebnisse der gezählten Partikel sowie die berechnete Effizienz der Reinigung zusammen, sortiert nach auftretender Partikelgröße.

Tabelle 23: Ergebnisse der Reinigung der sphärischen Linse

D_{eq} in μm	Partikel- konzentration vor Reinigung	Partikel- konzentration nach Reinigung	Reinigungs- effizienz in %
≥ 10	74	2	97.3
≥ 20	59	0	100
≥ 30	51	0	100
≥ 40	35	1	97.1
≥ 50	9	0	100
≥ 60	2	0	100
≥ 70	3	0	100
≥ 80	3	0	100

Die Fotos in Abbildung 50 zeigen einen weiteren Vorher-Nachher Vergleich einer durch Staub und Flusen verunreinigten Optik. Die violetten, orangen und gelben Farben sind wieder auf die Reflexion der Oberflächenbeschichtung zurückzuführen. Bei genauerem Hinsehen, lassen sich zudem auch viele graue Flecken erkennen die scheinbar die ganze sichtbare Oberfläche überziehen. Diese sind auf die Schaumstoffunterlage zurückzuführen und deshalb nicht als Verschmutzung zu interpretieren.

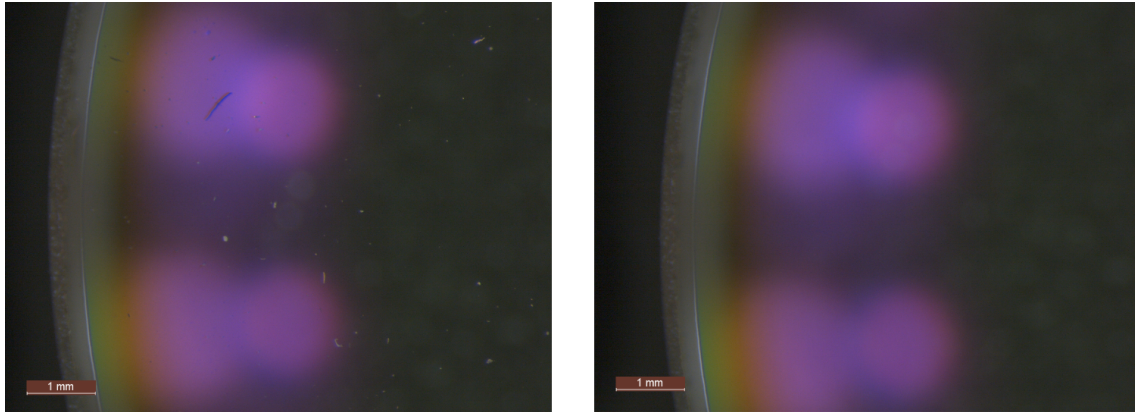


Abbildung 50: Links die stark verschmutzte Oberfläche des Achromates und rechts die gleiche Oberfläche nach der Reinigung. Die erfassten Äquivalentdurchmesser D_{eq} der Partikel liegen zwischen 15.75 und 540.15 μm . Die Reinigung wurde mit 50 % Druck, 50 μl Aceton und einer Drehzahl von 120 min^{-1} durchgeführt.

Die Ergebnisse in Tabelle 24 zeigen, dass alle partikulären Verunreinigungen entfernt werden konnten. Partikeldurchmesser $< 10 \mu m$ wurde nicht erfasst.

Tabelle 24: Ergebnisse der Reinigung des Achromates

D_{eq} in μm	Partikel- konzentration vor Reinigung	Partikel- konzentration nach Reinigung	Reinigungs- effizienz in %
≥ 10	61	0	100
≥ 20	43	0	100
≥ 30	19	0	100
≥ 40	3	0	100
≥ 50	4	0	100
≥ 60	7	0	100
≥ 70	3	0	100
≥ 80	9	0	100

3.4.2 Fettige Schmutzfilme

Die Gegenüberstellungen in Abbildung 51 und 52 zeigen die Reinigungsergebnisse beim Versuch einen fettigen Fingerabdruck zu entfernen. Der verschmutzte Bereich ist in diesem Beispiel sehr großflächig, wie es in der Praxis nur sehr selten der Fall ist.

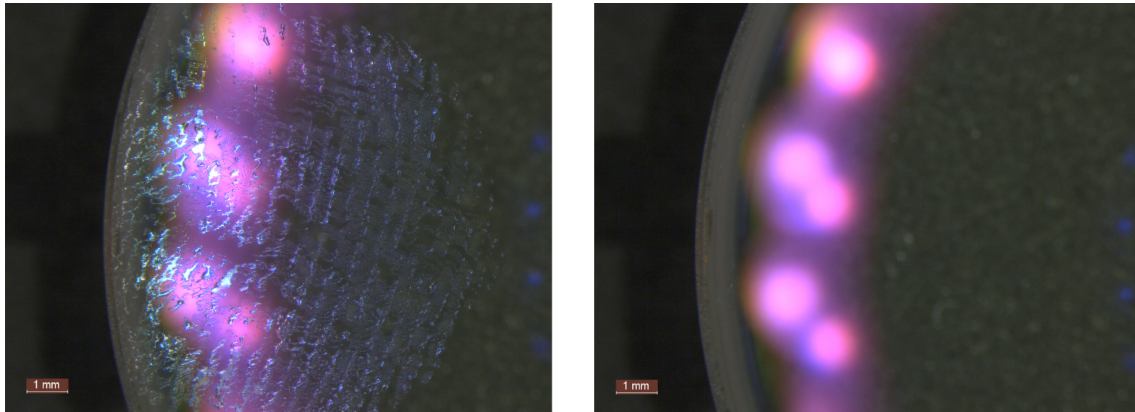


Abbildung 51: Links ist das durch einen fettigen Fingerabdruck verunreinigte Achromat zu sehen. Rechts: Dasselbe Achromat nach der Reinigung. Dieses Ergebnis wurde nach einmaligem Durchlauf mit 66 % Druck, 80 μl Lösungsmittel und einer Rotation von 100 min^{-1} erreicht.

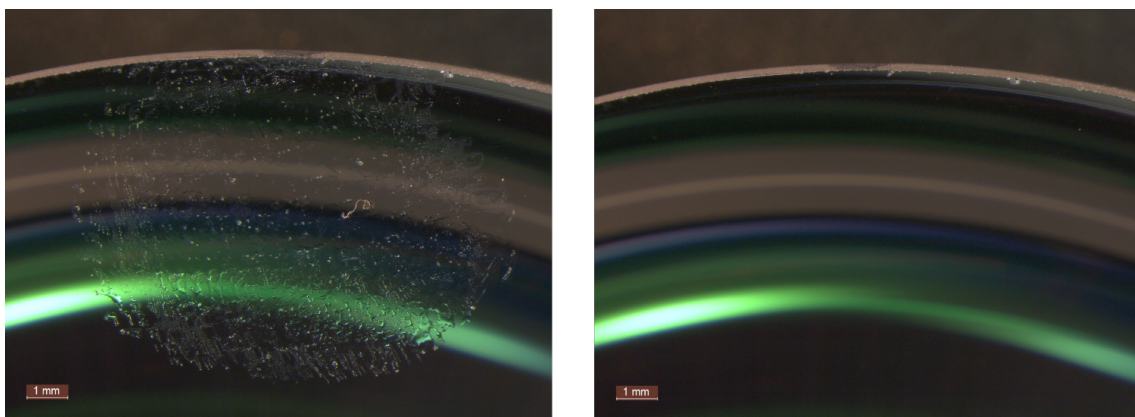


Abbildung 52: Links: Fettiger Fingerabdruck auf der Unterseite der sphärischen Optik (plane Fläche); Rechts: Reinigungsergebnis bei einmaligem Durchlauf unter einem Druck von 40 %, einer Lösungsmittelmenge von 60 μl , einem Verstellwinkel von -12° und einer Rotation von 100 min^{-1} .

3.4.3 Verunreinigung durch Schweißfinger

Die Fingerabdrücke in Abbildung 53 und 54, linkes Bild, wurden direkt nach dem Ausziehen der Schutzhandschuhe (aus Nitril) auf die Oberfläche gedrückt. Im Gegensatz zu den in Kapitel 3.4.2 abgebildeten Fingerabdrücken basieren diese auf Schweiß. Es sind zudem auch vereinzelte Partikel zu erkennen bei welchen angenommen wird, dass es sich um kleine Salzkristalle handelt. In beiden Fällen ist zu erkennen, dass ein Reinigungsdurchgang nicht ausreichend ist, um die Verschmutzungen wieder vollständig zu entfernen.

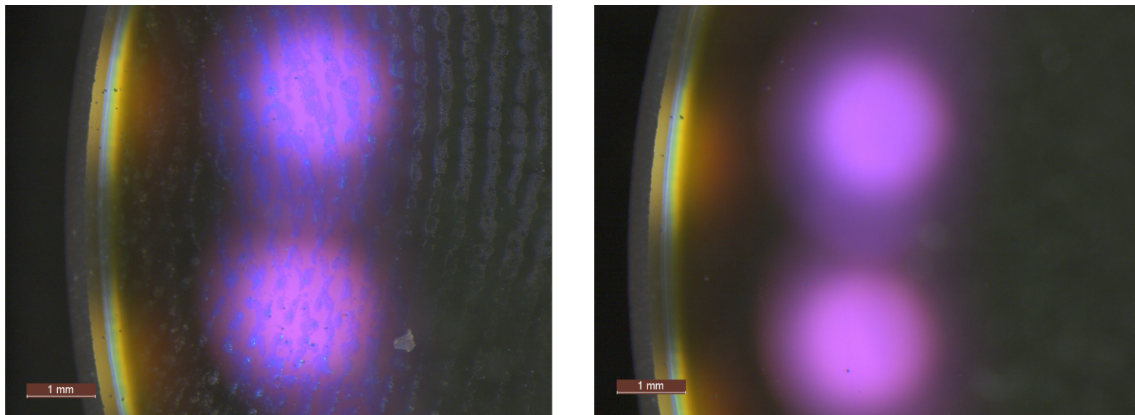


Abbildung 53: Im linken Bild ist eine Mischverunreinigung aus Partikeln, einem schweiß-basierten Fingerabdruck und einer Hautschuppe zu sehen. Die rechte Seite zeigt das Ergebnis nach einer Reinigung mit 85 % Druck, 90 μl Lösungsmittel und einer Rotationsgeschwindigkeit von 100 min^{-1} .

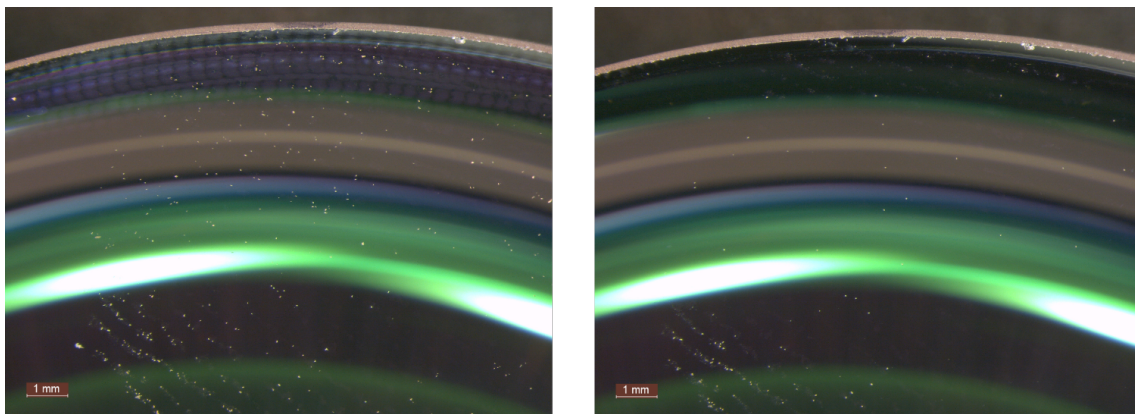


Abbildung 54: Erneuter Versuch einen eingetrockneten schweiß-basierten Fingerabdruck zu reinigen - in diesem Fall auf der sphärischen Linse. Das rechte Foto zeigt das Ergebnis nach der Reinigung mit 50 % Druck, 50 μl Lösungsmittel und einer Rotationsgeschwindigkeit von 100 min^{-1} .

3.4.4 Speicheltröpfchen

Die Aufnahmen in Abbildung 55 zeigen eine Verunreinigung in Form von Speicheltröpfchen in Kombination mit Partikeln. Die konzentrischen Kreise im rechten oberen Foto treten auf, wenn ein zu hohe Rotationsgeschwindigkeit des Drehtellers gewählt wird.

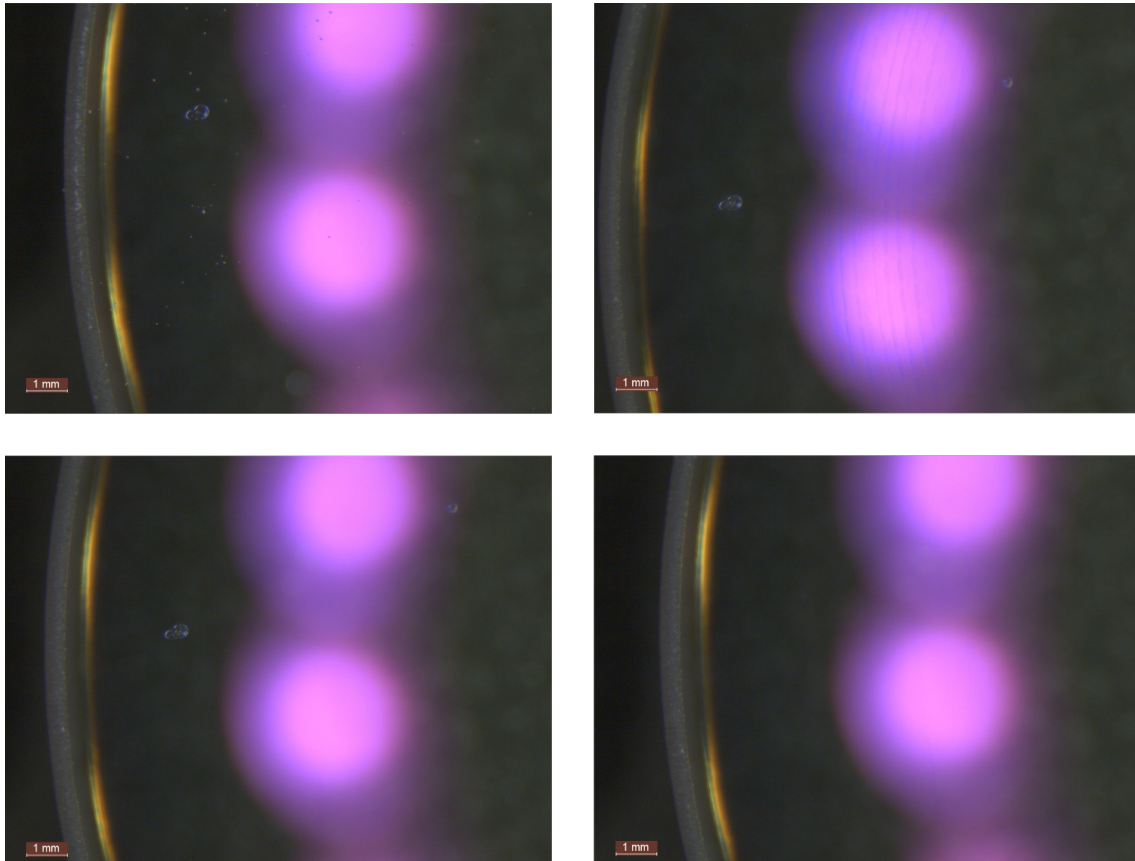


Abbildung 55: Versuch ein durch Speicheltröpfchen und Partikel verunreinigtes Achromat zu reinigen. Oberes linkes Bild zeigt die Ausgangssituation. Rechts daneben das Ergebnis nach dem ersten Reinigungsdurchgang bei 50 % Druck, 50 μl Lösungsmittel und einer Rotationsgeschwindigkeit von 280 min^{-1}

Betrachtet man Abbildung 55 genauer, so erkennt man, dass zwar alle Partikel entfernt werden konnten, die beiden Speicheltröpfchen aber nach wie vor vorhanden sind. Auch eine im Anschluss durchgeführte Reinigung brachte keinen Erfolg, siehe dazu Bild links unten. Die rechte untere Aufnahme zeigt das Reinigungsergebnis nach fünfmaliger manueller Reinigung. In diesem Fall wurde als „Lösungsmittel“ anstatt von Aceton, die an der Optikoberfläche beschlagene Atemluft verwendet.

3.4.5 Vergleich zwischen manueller und automatischer Reinigung

Die vier Aufnahmen in Abbildung 56 zeigen den direkten Vergleich zwischen manueller und automatischer Reinigung des Achromates. Der obere Vorher-Nachher-Vergleich ist der automatisierten Reinigung mittels Roboter zuzuordnen, wobei die linke Abbildung vor und die rechte Abbildung nach der Reinigung aufgenommen wurde. Die untere Gegenüberstellung zeigt das Ergebnis der manuellen Reinigung.

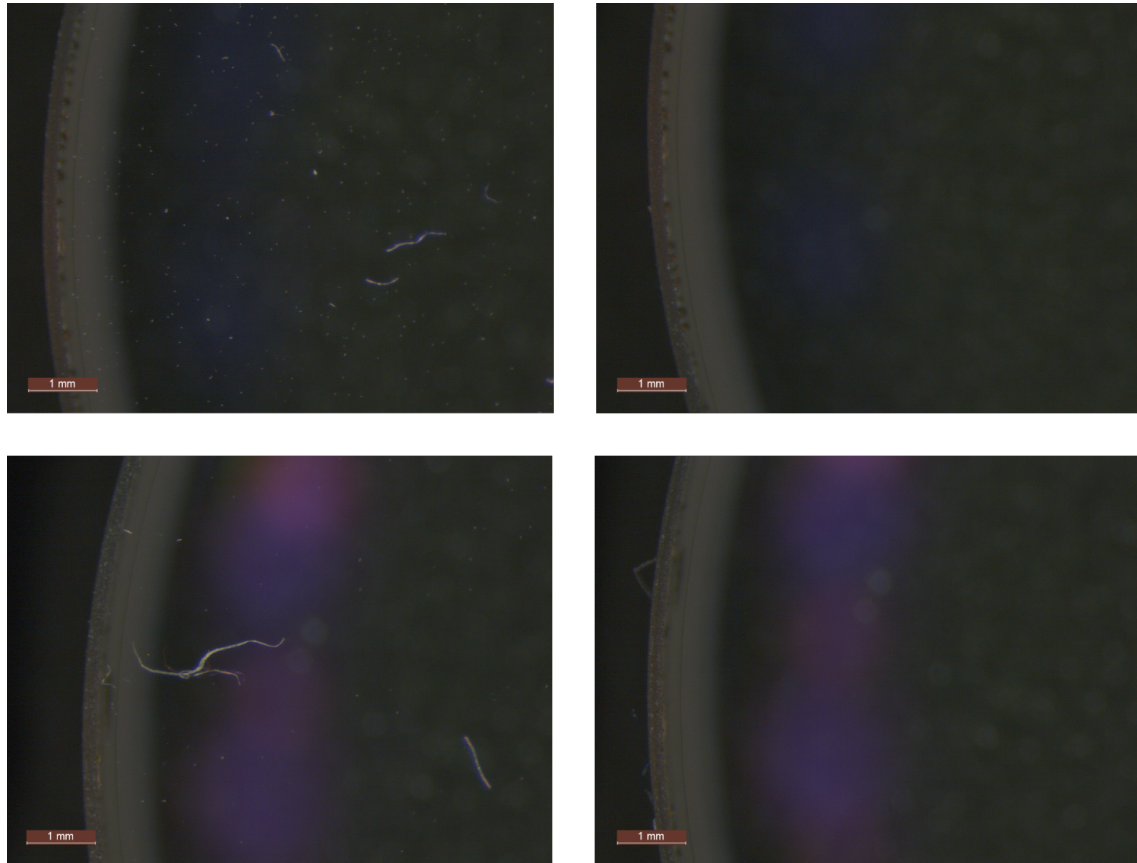


Abbildung 56: Direkter Vergleich zwischen manueller (unten) und automatischer (oben) Reinigung. Das Ergebnis der automatischen Reinigung wurde mit 50 % Druck, 50 μl Lösungsmittel und einer Rotationsgeschwindigkeit von 100 min^{-1} erreicht.

Sowohl nach der manuellen, als auch nach der automatischen Reinigung sind (visuell betrachtet) keine partikulären Rückstände mehr erkennbar. Die hellgrauen Flecken sind wiederum auf die Schaumstoffunterlage zurückzuführen.

4 Diskussion

4.1 CNC-Positionierungssystem

Die durchgeführten Reinigungsversuche haben gezeigt, dass das eingesetzte CNC-Positionierungssystem seinen Zweck im Reinigungsroboter erfüllt und dass die Positionierungen der einzelnen Schlitten mit ausreichender Genauigkeit durchgeführt werden. Ein kleiner Nachteil ergibt sich durch den offenen Regelkreis zwischen CNC-Controller und Schrittmotoren. Die gegenwärtige Position der Schlitten wird zwar im 20 ms Takt an die zentrale Steuerung rückgemeldet, die Koordinaten entsprechen jedoch nicht zwangsläufig der tatsächlichen Position. Das liegt daran, dass die Positionsrückmeldung nicht, wie bei professionellen CNC-Systemen, über die an den Motorwellen sitzenden *Encoder* erfolgt, sondern softwaremäßig berechnet wird.

Aus praktischer Sicht führt das nur dann zu einem Problem, wenn die Trapezgewindespindeln nach dem Initialisierungsvorgang unbeabsichtigt und unwissend per Hand verstellt werden.

4.2 Steuereinheit und Software

Der als zentrale Steuerung agierende Einplatinencomputer *Raspberry Pi 4* ist leistungsstark genug, um die in Python geschriebene Robotersoftware inklusive Benutzeroberfläche flüssig auszuführen.

Auch die an den Pinheader angeschlossene Peripherie lässt sich grundsätzlich problemlos ansteuern bzw. auslesen. Es konnte jedoch beobachtet werden, dass der externe ADC *HX711* in seltenen Fällen falsche Druckwerte ausgibt. Diese werden allerdings von der verwendeten *HX711*-Bibliothek erkannt und gekennzeichnet. So lassen sich Ausreißer ohne viel Aufwand per Software herausfiltern und verwerfen. Da der Sensor mit 20 Hz ausgelesen wird, bleibt der Reinigungsablauf davon unbeeinflusst.

Die angeschlossenen Sensoren und Aktoren belegen 17 der 40 GPIO-Pins. Das bedeutet, dass die Steuerung hardwaretechnisch noch genügend Potential zur Weiterentwicklung des Reinigungssystems bietet. Beispielsweise könnte die Drehtellereinheit um einen optischen Sensor erweitert werden, der vor jeder Reinigung prüft, ob eine Optik in das Gerät eingelegt wurde. Ein weiteres sinnvolles Feature wäre, den Roboter um einen Luftblasendetektor zu erweitern um Trockenreinigungen zu verhindern.

4.3 Lösungsmittelfördereinheit

Die als Spritzenpumpe realisierte Lösungsmittelleinheit funktioniert sehr zuverlässig und ist im Unterschied zu der ebenfalls in Betracht gezogenen *Peristaltikpumpe* wartungsarmer, da der Schlauch in diesem Fall kein Verbrauchsmaterial darstellt. Die Kombination aus Trapezgewindespindel mit 2 mm Steigung und dem 1/32 Microstepping-Mode des Schrittmotors erlaubt eine theoretische Dosiergenauigkeit von 236.35 nl/mstep, was bei einer typischen Abgabemenge von 50 μl mehr als ausreichend ist.

Das Fördersystem hat aber auch Nachteile, folgende zwei Punkte sollten deshalb berücksichtigt werden: Wird z.B. über mehrere Tage nicht gefördert, beginnt das im Schlauch befindliche Lösungsmittel langsam zu verdampfen. Darum ist es ratsam die Düse vor einer längeren Pause (z.B. Wochenende) luftdicht zu verschließen, um ein Ausdampfen zu verhindern. Aus sicherheitstechnischer Sicht, stellt das verflüchtigte Lösungsmittel kein Problem für die Umgebung dar, da die Mengen im Nanoliter-Bereich liegen und über einen langen Zeitraum entweichen. Die Gefahr besteht eher darin, dass nach einer längeren Stehzeit des Gerätes in den ersten Reinigungsdurchgängen kein Lösungsmittel auf das Polyester-Pad getropft wird und der Roboter die Optik trockenreinigt. Der fehlende Schlauchinhalt lässt sich deshalb im Bedarfsfall über die Software manuell nachfördern, siehe dazu Bedienungsanleitung Kapitel 5.2.4.

Das System ist zudem auch empfindlich gegenüber Lufteinschlüssen im Schlauch, welche vor allem beim Wiederbefüllen der Spritze entstehen können. Lufteinschlüsse führen dazu, dass die Abgabemenge nicht mit den eingestellten Werten übereinstimmen, deshalb ist darauf zu achten, dass das Schlauchsystem möglichst frei von Luftbläschen bleibt.

4.4 Reinigungskopf

4.4.1 Messmethoden der Profilaufzeichnung

Vergleicht man die beiden getesteten Sensoren hinsichtlich ihrer Messmethode, so hat der kapazitive Näherungsschalter den Vorteil, dass der Abstand berührungslos erfasst wird. Es besteht somit kein Risiko die Beschichtungen zu beschädigen. Im Gegensatz dazu, basiert die Messung mittels Nivellierungssensor auf Oberflächenkontakt. Es hat sich herausgestellt, dass Berührungen kein Problem darstellen, da die Sonde ein geringes Eigengewicht aufweist und aus Kunststoff gefertigt ist. Auch nach vielfacher Abtastung der selben Optik konnte keine auf den Sonde zurückzuführende Beschädigung festgestellt werden.

4.4.2 Präzision der Einzelpunktabtastung

Die Ergebnisse in Tabelle 18, Seite 83 und Tabelle 19, Seite 84 zeigen, dass bei einer wiederholten Abtastung eines einzelnen Punktes beide Sensoren brauchbare Ergebnisse liefern, der Nivellierungssensor aber deutlich präziser ist.

Die maximale Standardabweichung des kapazitiven Sensors liegt bei $50.2\ \mu\text{m}$, was für die vorliegende Anwendung ausreichend wäre. Mit dem Nivellierungssensor *BLTouch* konnten im besten Durchgang sogar Standardabweichungen unter $10\ \mu\text{m}$ erreicht werden. Im Hinblick auf die Präzision eignen sich also beide Sensoren.

Bei der Interpretation der Ergebnisse ist zu beachten, dass auch Faktoren wie Auflösung der Schrittmotoren oder Toleranzen der mechanischen Teile (z.B. der Trapezgewindespindel) eine Rolle spielen. Die erfassten Messwerte beschreiben genauer betrachtet die Präzision des Sensors im Zusammenspiel mit dem CNC-System.

4.4.3 Erfassung von Oberflächenprofilen

Der Vergleich der erfassten Oberflächenprofile in Abbildung 43 (Seite 85) bestätigt, dass sich mit dem Nivellierungssensor präzisere Ergebnisse erzielen lassen. Im direkten Vergleich der beiden Sensoren zeigt sich außerdem, dass die beiden erstellten Profilverläufe stark von einander abweichen. Bezieht man die tatsächlichen Abmaße der Optik mit ein, so wird klar, dass die Abweichung eindeutig auf den kapazitiven Näherungsschalters zurückzuführen ist.

Eine mögliche Erklärung für den nicht konstanten Schaltabstand ist, dass das erzeugte elektrische Feld, auf welchem die Messung basiert [49], je nach Abtastpunkt in einem unterschiedlichen Winkel auf die Oberfläche der Optik trifft und dadurch unterschiedlich stark gedämpft wird. Angesichts diesen Umstandes und der ohnehin höheren Präzision des Nivellierungssensors, stellt der kapazitive Näherungsschalter keine Option mehr dar.

4.4.4 Präzision und Linearität der Druckmessung

Die grafische Auswertung der Messergebnisse in Abbildung 44 auf Seite 86 brachte zum Vorschein, dass der Druck je nach Pausenzeit zwischen den Messungen mehr oder weniger abnimmt. Speziell bei kurzen Pausenzeiten ist eine deutliche Tendenz erkennbar. Dieses Verhalten ist aber nicht, wie zunächst vermutet, auf die Wägezelle selbst zurückzuführen. Der Grund dafür ist, dass sich die Steifigkeit des aus Polypropylen gefertigten Kunststoffschäfts (Reinigungsstäbchen) bei kurzzeitigen Wiederholungen, etwas verringert. Im ungünstigsten Fall (Pausenzeit = 2.5 s), lag die durchschnittliche Abweichung zum Mittelwert des Drucks bei 0,3%.

Da ein Reinigungsvorgang deutlich mehr als 2.5 s in Anspruch nimmt, siehe dazu Abbildung 46 auf Seite 88, ist keine wesentliche Beeinflussung auf die Druckmessung und im weiteren Sinne auf das Reinigungsergebnis zu erwarten.

Was die Linearität der Wägezelle betrifft, so sind keine großen Auffälligkeiten beobachtbar. Die Abbildung 45, Seite 87 zeigt zwar geringe Abweichungen zum gefitteten Idealverlauf, für die gegenwärtige Anwendung sind diese aber vernachlässigbar. Eine nachträgliche Linearisierung ist somit nicht erforderlich.

4.4.5 Druckverläufe der Reinigungsmodi

Wie in Abbildung 46 auf Seite 88 ersichtlich, weichen die Druckverläufe (blaue Kurven) der drei Reinigungsmodi geringfügig von einander ab. Betrachtet man die maximale Abweichung zum Sollwert, so liefert die Druckreinigung die besten Ergebnisse. Das lässt sich damit begründen, dass bei dieser Reinigungsmethode der Druck auf die Oberfläche permanent nachgeregelt wird.

Gleichzeitig ist aber zu erkennen, dass der Verlauf im Vergleich zur Profil- und Radiusreinigung eine etwas höhere Welligkeit aufweist. Eine gewisse Welligkeit des Signals ist grundsätzlich bei allen Verläufen beobachtbar und auf eine „eiernde“ Bewegung der Optik zurückzuführen, welche entsteht, wenn diese leicht verkippt in die Aufnahme eingelegt wird. Bei der Druckreinigung kommt hinzu, dass durch die Regelung ein ebenfalls leicht welliger Verlauf entsteht. Die Kombination beider Effekte führt zu der etwas höheren Wellenamplitude.

Was die Profil- und Radiusreinigung betrifft, so sind unterschiedliche Faktoren ausschlaggebend, um einen möglichst konstanten Druckverlauf zu erzielen. Den größten Einfluss hat die Position des Reinigungsstäbchens. Rastet die Kerbe des Schaftes nicht korrekt in die Aufnahme ein, führt der daraus resultierende Versatz in X- und Z-Achse zu einem versetzten Startpunkt und folglich zu einem ungleichmäßigen Druckverlauf.

Abgesehen davon, bewirken eine zu grobe Schrittauflösung sowie eine zu hohe Absenkgeschwindigkeit ein Überschwingen des Druckwertes beim Absenken des Reinigungskopfes. Im Gegensatz zur Druckreinigung, wird eine zu hohe Abweichung des Drucks bei der Profil- und Radiusreinigung nicht kompensiert. Ein dadurch entstehender Fehler würde sich anschließend über das gesamte Profil ziehen. Der Algorithmus beider Reinigungsmodi wurde deshalb dahingehend optimiert, um ein Überschwingen zu minimieren.

Abbildung 46 auf Seite 88 verbildlicht die eben erwähnte Situation. Bei einem falsch eingesetzten Reinigungsstäbchen weicht der Druckverlauf um bis zu 20 % vom Sollwert ab (siehe oranger Verlauf bei Profilreinigung).

Zusammenfassend kann gesagt werden, dass sich die drei Modi bezüglich der Druckverläufe nicht wesentlich unterscheiden. Der Reinigungsprozess ist bei Radius- und Profilreinigung zwar kürzer, dafür hat die Druckreinigung den Vorteil, dass sie gegenüber schlecht eingesetzter Reinigungsstäbchen unempfindlicher ist.

4.5 Drehtellereinheit

4.5.1 Kraftübertragung und Platzbedarf

Kräftetechnisch erfüllt die entwickelte Drehtellereinheit die Anforderungen. Auch wenn das Polyester-Pad mit 100 % des möglichen Drucks auf die Optik gepresst wird, bleibt die Drehzahl konstant. Das liegt einerseits am leistungsstarken DC-Getriebemotor und andererseits an der zusätzlichen Untersetzung durch das Riemenge triebe. Die kompakte Bauweise ermöglicht es zudem, die gesamte Einheit trotz des geringen Platzes am Y-Schlitten unterzubringen.

4.5.2 Drehzahl und Geräuscentwicklung

Das die Rotationsgeschwindigkeit des Drehtellers nicht mit der an den CNC-Controller gesendeten Drehzahl übereinstimmt, war aufgrund der doppelten Untersetzung zu erwarten. Zusätzlich wird die Drehzahl noch durch das unsaubere PWM-Signal beeinflusst, welches vom CNC-Controller ausgegeben wird (siehe dazu den Verlauf der Spannung in Abbildung 47 auf Seite 89). Die Gegenüberstellung von gemessener und tatsächlicher Drehzahl in Abbildung 48 auf Seite 90 zeigt, dass aber ein nahezu linearer Zusammenhang besteht, über welchen sich die an den CNC-Controller zu sendende Drehzahl bestimmen lässt.

Verbesserungsbedarf besteht bezüglich der Geräuscentwicklung bei Drehzahlen ab 250 min^{-1} . Diese entstehen zum einen aufgrund der doppelten Untersetzung und andererseits aufgrund der schlechten Qualität der verbauten Kegelräder, welche das 90° Getriebe bilden. Es konnte herausgefunden werden, dass die Geräuscentwicklung auf den mangelhaften Rundlauf der Kegelräder zurückzuführen ist.

4.6 Reinigungserfolg

4.6.1 Partikuläre Verunreinigung

Fasst man die Ergebnisse der Reinigungsversuche aus Punkt 3.4.1 zusammen, so zeigt sich, dass sich partikuläre Verschmutzungen wie Staubkörnchen oder Fussel durch den Roboter sehr gut entfernen lassen. Die Reinigungseffizienz liegt in beiden Fällen über 97 %, siehe dazu Tabelle 23 und 24 auf Seite 91 und 92.

Dabei gilt zu beachten, dass nur Partikel $\geq 10 \mu m$ betrachtet wurden. Über die Anzahl an kleineren Partikeln kann keine Aussage getroffen werden. Aus der Tatsache heraus, dass Optiken für gewöhnlich nur mit dem bloßen Auge auf Verunreinigungen untersucht werden [11] und somit Partikel $\leq 10 \mu m$ ohnehin nicht erfasst werden [7], lässt sich schließen, dass das Reinigungsergebnis ausreichend ist.

4.6.2 Schmutzfilme

Fettige Fingerabdrücke

Durch eine visuelle Interpretation der Vorher-Nachher-Bilder lässt sich die Aussage treffen, dass der Reinigungsroboter in der Lage ist fettige Fingerabdrücke restlos zu entfernen. Zu beachten ist, dass bei großflächigen, fettigen Verunreinigungen das Reinigungsstäbchen nach jedem Durchlauf gewechselt bzw. gedreht werden sollte, da es sonst bei den darauf folgenden Reinigungen zu einem Verschmieren der Oberfläche kommen kann.

Fingerabdrücke auf Schweiß-Basis

Die Reinigungsversuche zeigten, dass sich Fingerabdrücke auf Schweißbasis nur sehr schwer entfernen lassen. An diesem Punkt sei erwähnt, dass sich diese Art von Verschmutzung auch mit der manuellen *Pinselmethode* nicht ohne Weiteres reinigen lässt. Der Grund liegt darin, dass sich das verwendete Aceton für diesen Typ von Verunreinigung nicht eignet. Das einzig wirksame „Lösungsmittel“ war in diesem Fall die an der Optikoberfläche beschlagene Atemluft, welche im produktiven Einsatz des Roboters mit DI-Wasser (deionisiertes Wasser) ersetzt werden sollte.

Verunreinigungen durch Speichel

Diese Art Verschmutzung stellt die größte Herausforderung dar, sowohl für die Reinigung mittels Roboter als auch für die manuelle *Pinseltechnik*. Der Grund liegt einmal mehr darin, dass sich Speichel mit dem eingesetzten Lösungsmittel (Aceton) nur schwer, bis gar nicht entfernen lässt. In diesem Fall muss wiederum auf DI-Wasser zurückgegriffen werden [11].

4.7 Schlussfolgerung

Das Ziel dieser Masterarbeit war die Entwicklung eines Reinigungssystems, welches die bisher manuell durchgeführte Reinigung von optischen Komponenten mittels *Pinseltechnik* automatisiert. Die manuelle Entfernung von Verschmutzungen durch MitarbeiterInnen, wird oft mit unterschiedlicher Genauigkeit durchgeführt und der auf die Optikoberfläche ausgeübte Druck variiert von MitarbeiterIn zu MitarbeiterIn. Eine zu hohe Kraft- bzw. Druckeinwirkung kann in manchen Fällen sogar zu visuell erkennbaren, irreversiblen Beschädigungen der Oberflächenvergütung führen. Der entwickelte, halbautomatische Reinigungsroboter regelt den Druck, welcher auf die zu reinigenden Optik einwirkt und hält ihn so konstant. Dadurch lassen sich Beschädigungen während dem Reinigungsvorgang verhindert. Es wurden drei unterschiedliche Reinigungsmodi implementiert, um eine möglichst hohe Bandbreite an unterschiedlichen Optiken abzudecken. Nach keinem der im Anschluss durchgeführten Reinigungsversuche konnte eine visuell erkennbare Schädigung der Optik festgestellt werden.

Die in Kapitel 1.7 festgelegten funktionalen und mechanischen Anforderungen an das Gerät, konnten vollständig erfüllt werden. Die Kombination zwischen CNC-Positionierungssystem und Nivellierungssensor ermöglicht Profilabtastungen mit einer Messunsicherheit von maximal $10.6\ \mu\text{m}$, der Druck auf die Optik lässt sich theoretisch bis auf 0.3 % regeln und mittels implementierter Spritzenpumpe kann das aufzubringende Lösungsmittel in μl -Schritten dosiert werden. Der Roboter fungiert grundsätzlich als eigenständiges Gerät, kann aber auch über andere, netzwerkfähige Geräte wie Notebook, PC oder Tablet gesteuert werden, was vor allem beim Erstellen und Speichern von neuen Optikprofilen von Vorteil sein kann.

Durch verschiedene Tests, welche mit Hilfe von visueller Interpretation und der *Klassifizierung der Abreinigbarkeit* bewertet wurden, konnte festgestellt werden, dass sich mit Hilfe des Reinigungsroboters partikuläre Verschmutzungen sowie fettige Fingerabdrücke unter Zuhilfenahme von Aceton nahezu restlos entfernen lassen. Schweißbasierte Fingerabdrücke und Speicheltröpfchen hingegen stellen das Reinigungssystem noch vor eine Herausforderung. Durch eine Erweiterung der Fördereinheit um zusätzliche Lösungsmittel, könnten sich zukünftig auch diese Art von Verschmutzungen beseitigen lassen.

Anhand eines direkten Vergleiches zwischen manueller und automatisierter Reinigung konnte gezeigt werden, dass sich mit der Anwendung des Reinigungsroboters mindestens gleich gute Reinigungsergebnisse erzielen lassen.

5 Bedienungsanleitung

5.1 Grundlegendes

5.1.1 Aufbau der Benutzeroberfläche

Die grafische Benutzeroberfläche gliedert sich in einen statischen Kopf- und Fußbereich sowie einem dynamischen Mittelbereich, dessen Inhalt auf vier Registerkarten aufgeteilt ist.

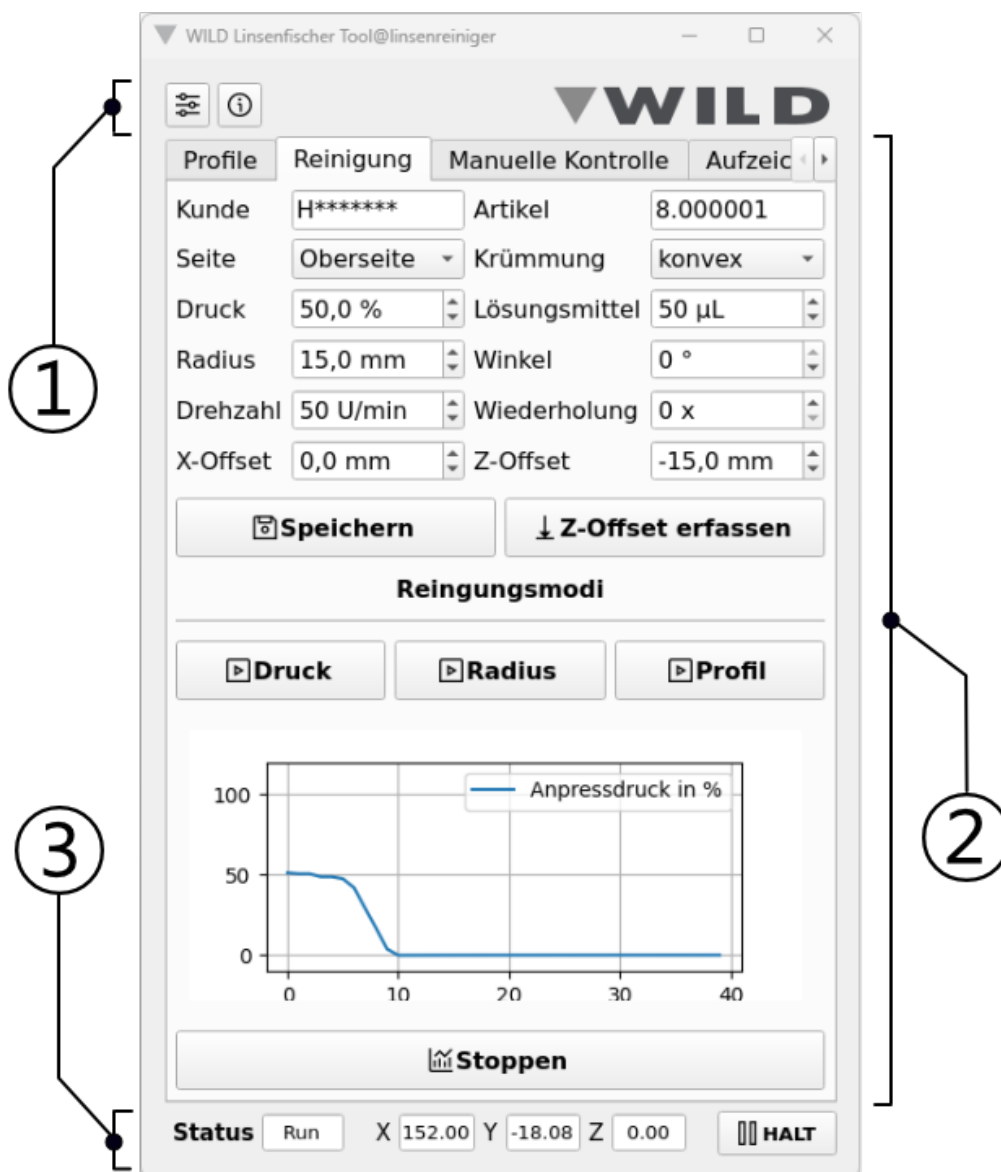


Abbildung 57: Grundlegender Aufbau der Benutzeroberfläche, (1) Kopfbereich, (2) Mittelbereich und (3) Fußbereich

Kopfbereich

Der Kopfbereich beinhaltet allgemeine Information über Softwareversion und Einstellungen, welche über die Schaltflächen **Einstellungen** (Schieberegler Symbol) und **i** abrufbar sind.

Mittelbereich

Über den Mittelbereich lässt sich der Roboter bedienen. Die Software startet in der Registerkarte *Profile*. Die Funktionalität dieser und aller weiteren Registerkarten wird in Kapitel 5.2 näher erläutert.

Fußbereich

Der Fußbereich gibt Auskunft über den aktuellen Status des Gerätes und die gegenwärtige Absolutposition von X-, Y- und Z-Schlitten. Mit dem Button **Halt** lassen sich im Bedarfsfall Bewegungen des Gerätes ohne Positionsverlust pausieren.

5.1.2 In- und Außerbetriebnahme

Für die Inbetriebnahme des Roboters sind folgende Schritte durchzuführen:

1. Das Gerät mit dem Netzteil verbinden, über den Netzschalter (roter Kippschalter auf der rechten Seite) einschalten und den Boot-Vorgang abwarten.
2. Über die Desktop-Verknüpfung **START** die Software des Roboters starten. Während dem Start wird die Hardware initialisiert.
3. Der Roboter ist bereit, sobald der Status im Fußbereich von **Run** auf **Idle** wechselt.

Wird das Gerät nicht mehr benötigt kann es auf zwei verschiedene Arten ausgeschaltet werden:

1. In der Benutzeroberfläche in den Reiter *Manuelle Kontrolle* wechseln und dort im Abschnitt *Makros* auf **Ausschalten** klicken.
2. Die Software mit Klick auf die Schaltfläche **X** in der Titelleiste des Softwarefensters beenden. Anschließend kann die zentrale Steuerung manuell über **Start** > **Abmelden** > **Herunterfahren** beendet werden.

Um Beschädigungen der SD-Karte zu vermeiden, darf der Netzschalter erst betätigt werden, wenn die Steuerung vollständig heruntergefahren ist. Dafür muss mindestens fünf Sekunden gewartet werden.

5.2 Funktionen

5.2.1 Anlegen eines neuen Optikprofils

Für ein optimales Reinigungsergebnis müssen zuerst die Reinigungsparameter ermittelt werden. Diese unterscheiden sich von Optik zu Optik und sind zuvor anhand von Reinigungsversuchen zu ermitteln. Der Software bietet folgende Einstellungsmöglichkeiten:

Tabelle 25: In der Software einstellbare Reinigungsparameter

Parameter	Einheit	Kurzbeschreibung
Druck	%	Maximaler Druck, mit welchem das Polyester-Pad auf die Oberfläche der Optik drückt; ein Wert von 100 % entspricht einem Druck von $1.45 \frac{N}{cm^2}$
Lösungsmittel	nl	Menge an Lösungsmittel welches auf das Polyester-Pad aufgebracht wird
Radius	mm	Radius der eingelegten Optik
Winkel	°	Winkel des Reinigungskopfes (muss nur bei konkavgekrümmten Oberflächen verstellt werden, der Einstellbereich liegt zwischen 0 und -20 °)
Drehzahl	$\frac{1}{min}$	Drehzahl mit welcher die Drehtellereinheit rotiert; die Mindestdrehzahl liegt bei $40 \frac{1}{min}$
Wiederholungen	x	Anzahl der Wiederholungen (sollte grundsätzlich bei '0x' belassen und nur bei hartnäckigen Verschmutzungen erhöht werden)
X-Offset	mm	Gibt den Abstand zwischen optischer Achse und Reinigungsstartpunkt in X-Richtung an; wird nur bei speziellen Optiken benötigt
Z-Offset	mm	Gibt den Abstand zwischen Z=0 und Reinigungsstartpunkt in Z-Richtung an

Für spezielle Optiken, z.B. Linsen mit Zentrierbohrungen, ist es unter Umständen notwendig die X-Koordinate des Reinigungsstartpunktes zu versetzen. Das lässt sich durch Erhöhung des X-Offsets erreichen. Der Z-Offset hingegen legt fest, wie weit der Roboter den Reinigungskopf im Eilvorgang senkt, bevor er sich in kleinen Schritten an die Oberfläche heran tastet.

Je höher der Z-Offset ist, desto schneller ist der initiale Absenkvorgang. Der Wert lässt sich auch automatisch ermitteln, indem der Button Z-Offset erfassen im Reiter *Reinigung* gedrückt wird. X- und Z-Offset können unabhängig voneinander eingestellt werden.

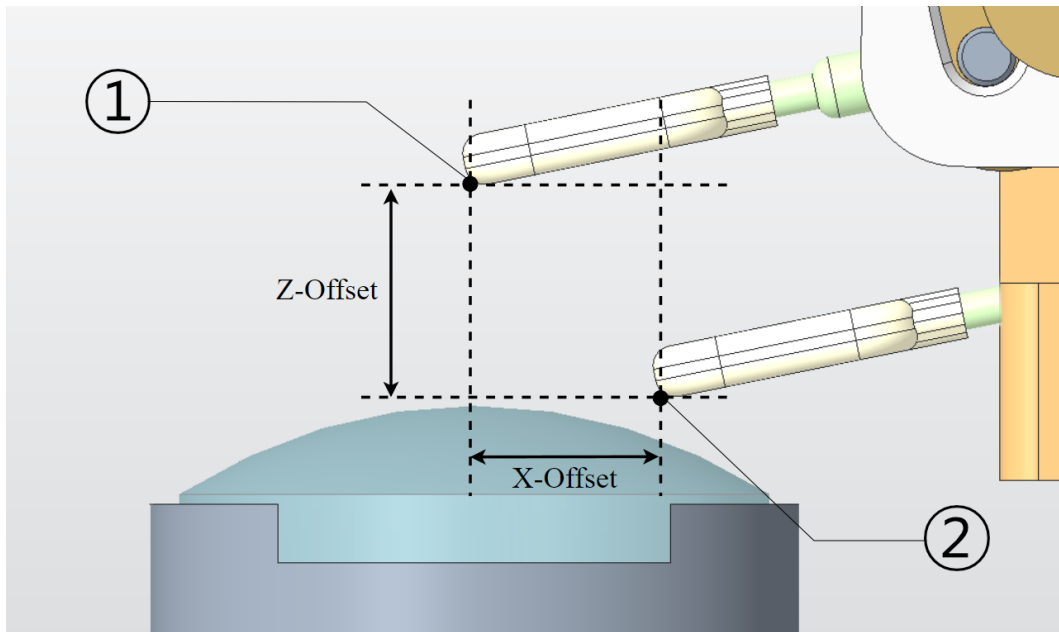


Abbildung 58: Bedeutung der beiden Parameter X- und Z-Offset. (1) Startpunkt der Reinigung, (2) In X- und Z-Richtung versetzter Startpunkt

Das Profil einer Optik besteht neben den Reinigungsparametern auch aus einem allgemeinen Teil, dessen Information die Reinigung nicht direkt beeinflussen. Tabelle 26 fasst diese kurz zusammen.

Tabelle 26: Allgemeine Eigenschaften eines Optikprofils.

Feld	Kurzbeschreibung
Kunde	Diese Eigenschaft hat keinen Einfluss auf die Reinigung. Sie ist als zusätzliche Information zu sehen und hilft bei der Auswahl des Profils im Profilbrowser.
Artikelnummer	Firmeninterne Artikelnummer welche zur eindeutigen Identifikation eines Profils im Profilbrowser dient
Seite	Mit dieser Eigenschaft wird zwischen 'Oberseite' und 'Unterseite' der Optik unterschieden.
Krümmung	Als Krümmung kann wahlweise zwischen 'konvex', 'konkav' und 'planar' gewählt werden. Diese Einstellung wird nur bei der Druckreinigung berücksichtigt.

Sind Reinigungsparameter und allgemeine Angaben getätigt, kann das Profil über den Button Speichern der Profildatenbank hinzugefügt werden. Dabei wird eine Profil-Datei im Ordner

`'home\wild\Python\linsenfischer\profile\'`

abgelegt. Sind Änderungen des Optikprofils nötig, so können diese auch direkt in den Profildateien durchgeführt werden.

5.2.2 Erstellung eines Optikprofils

Optional ist es möglich, ein XZ-Oberflächenprofil der Optik zu erstellen. Das XZ-Oberflächenprofil bietet eine weitere Möglichkeit der Reinigung und basiert im Gegensatz zur Druckreinigung nicht auf der Ausregelung des Drucks. Bei dieser Methode werden die erfassten Koordinaten verwendet, um das Polyester-Pad entlang der Oberfläche zu bewegen. Ein Oberflächenprofil muss nur einmal erfasst werden und kann anschließend über das Optikprofil mitgespeichert werden.

Tabelle 27: Einstellungsmöglichkeiten bei der Erfassung eines Oberflächenprofils.

Feld	Einheit	Kurzbeschreibung
Samples	x	Gibt an, wie viele Punkte entlang des angegebenen Radius abgetastet werden; Die Auflösung lässt sich verfeinern, indem die Anzahl der Samples erhöht wird.
Radius	mm	Radius der eingelegten Optik; Ein Sicherheitsfaktor von 1 mm wird automatisch durch die Software abgezogen.
X-Offset	mm	Gibt den Abstand zwischen optischer Achse und Reinigungsstartpunkt in X-Richtung an; Wird nur bei speziellen Optiken benötigt.
Z-Offset	mm	Gibt den Abstand zwischen Z=0 und Reinigungsstartpunkt in Z-Richtung an

Je mehr Samples eingestellt werden, desto länger dauert der Abtastvorgang. Der Fortschritt ist in dem *Ladebalken* im unteren Abschnitt des Fensters ersichtlich. Die Ergebnisse werden anschließend in einem Diagramm visualisiert.

Abtastvorgang

1. In den Reiter *Aufzeichnung* wechseln und die Parameter ändern.
2. Abtastung mit Klick auf Koordinaten aufzeichnen starten, der Roboter wechselt in die Beladeposition.
3. Optik einlegen und die Aufforderung mit OK bestätigen. Die Aufzeichnung startet, indem der Roboter in die Abtastposition fährt. Den anschließenden Abtastprozess abwarten.
4. Nach erfolgreicher Profilaufnahme, wechselt der Roboter wieder in die Beladeposition. Optik wieder entnehmen und das Popup mit OK schließen.

5.2.3 Durchführung einer Reinigung

Grundsätzlich stehen drei verschiedene Reinigungsmodi zur Auswahl, die Druck-, die Profil- und die Radiusreinigung. Die letzten beiden lassen sich nur durchführen, wenn zuvor ein Oberflächenprofil mit mindestens drei Abtastpunkten erstellt und gespeichert wurde.

Druckreinigung

1. Im Reiter *Profile* das gewünschte Profil suchen und markieren. Mit dem Button Profil laden lässt sich die Auswahl öffnen. Das Programm wechselt automatisch in den Reiter *Reinigung*.
2. Bei Bedarf die Reinigungsparameter anpassen.
3. Mit Klick auf Druckverlauf anzeigen lässt sich das Live-Diagramm aktivieren (optional).
4. Reinigung mit Klick auf Druckreinigung starten, der Roboter nimmt die Beladeposition ein.
5. Linse einlegen und in der Software mit OK bestätigen. Abwarten, bis die Reinigung durchgeführt wurde und der Roboter wieder in die Beladeposition wechselt.
6. Die Optik entnehmen und das Popup wieder mit OK bestätigen.

Profil- und Radiusreinigung

Der Ablauf von Profil- und Radiusreinigung unterscheidet sich nur unwesentlich von der Druckreinigung. Der einzige Unterschied liegt in Punkt 4, bei welchem in diesem Fall der Button Profilreinigung bzw. Radiusreinigung zu wählen ist.

5.2.4 Manuelle Kontrolle des Roboters

Über den Reiter *Manuelle Kontrolle* ist es möglich die einzelnen Hardwareeinheiten des Roboters manuell zu steuern. Das betrifft das CNC-Positionierungssystem, die Winkelverstellung des Reinigungskopfes, den Drehteller sowie die Spritzenpumpe. Außerdem sind im Abschnitt *Makros* weitere nützliche Funktion zu finden.

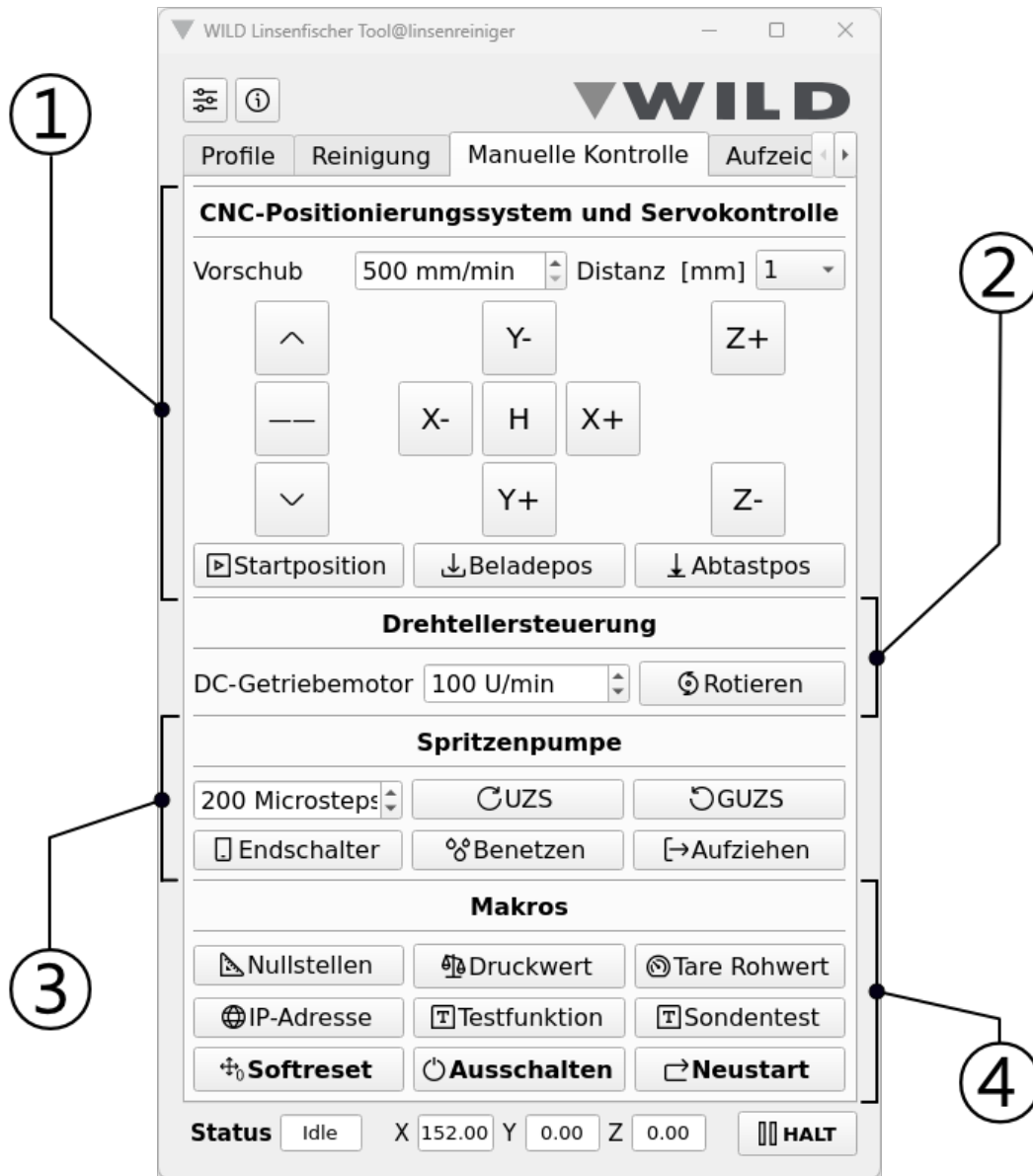


Abbildung 59: Grafische Oberfläche des Reiters *Manuelle Kontrolle*

- (1) CNC-Positionierungssystem und Servokontrolle
- (2) Steuerung des Drehtellermotors
- (3) Manuelle Kontrolle der Spritzenpumpe
- (4) Weitere Funktionen

CNC-Positionierungssystem und Servokontrolle

In diesem Abschnitt lassen sich X-, Y- und Z-Schlitten unter Angabe von Distanz und des Vorschub manuell verstellen. Ein manuelles Verfahren der Schlitten ist für gewöhnlich nur im Fehlerfall notwendig.

±X, Y, Z	Positioniert die Schlitten entlang der der jeweiligen Achse
H	Fährt die Schlitten in den Maschinennullpunkt
Startpos	Fährt den Roboter in die Reinigungsposition
Beladepos	Fährt den Roboter in die Be- und Entladeposition
Abtastpos	Fährt den Roboter in die Abtastposition
∧	Verstellt den Reinigungskopf um 2° nach oben
--	Grundposition des Reinigungskopfes
∨	Verstellt den Reinigungskopf um 2° nach unten

Drehtellersteuerung

In diesem Bereich ist es möglich den Drehteller mit unterschiedlichen Geschwindigkeiten rotieren zu lassen. Die Drehzahl lässt sich mittels Spin-Box einstellen. Eine Drehzahländerung wird erst übernommen, wenn der Motor aus- und wieder eingeschaltet wird. Ein- und Ausschalten geschieht über den Toggle-Button bzw. .

Spritzenpumpe

Im Abschnitt *Spritzenpumpe* lässt sich die Lösungsmittelfördereinheit manuell steuern. Dazu stehen fünf Funktionen zur Verfügung:

UZS	Dreht den Pumpenmotor um die eingestellte Schrittzahl im Uhrzeigersinn
GUZS	Dreht den Pumpenmotor um die eingestellte Schrittzahl gegen den Uhrzeigersinn
Benetzen	Benetzt das Polyester-Pad mit Lösungsmittel
Aufziehen	Zieht die Spritze auf
Endschalter	Gibt den Status des Endschalters der Spritzenpumpe aus

Makros

Der Abschnitt *Makros* beinhaltet eine Sammlung an funktionalen und informativen Funktionen. Rückgabewerte werden über ein Popup eingeblendet.

Nullstellen	Setzt den aktuellen Wert des Kraftsensors auf Null
Druckwert	Gibt den aktuellen Druckwert in % aus
Benetzen	Gibt den Tare-Rohwert aus
IP-Adresse	Zeigt die aktuelle IP-Adresse und den Hostnamen an
Testfunktion	Reservierter Button zum Debuggen der Software
Sonde	Toggle-Button der die Sonde aus- bzw einfährt
Softreset	Setzt den CNC-Controller zurück, z.B. bei Schrittverlusten notwendig
Ausschalten	Beendet die Software und fährt die zentrale Steuerung herunter
Neustart	Beendet die Software und startet die zentrale Steuerung neu

5.2.5 Anzeige & Änderung der Basiseinstellungen

Damit Änderungen grundlegender Einstellungen nicht direkt im Quellcode durchgeführt werden müssen, gibt es eine Konfigurationsdatei, welche alle Grundeinstellungen des Reinigungsroboters enthält. Die dort festgelegten Werte lassen sich zwar in der GUI über die Schaltfläche anzeigen, aber aus Sicherheitsgründen nicht ändern. Eine Änderung ist ausschließlich in der Konfigurationsdatei selbst durchführbar, welche unter folgendem Pfad zu finden ist:

```
'\home\wild\Python\linsenfischer\einstellungen\einstellungen.cfg'
```

Jede Änderung ist mit Bedacht durchzuführen, da sie das Verhalten des Roboters beeinflusst bzw. dessen Funktion beeinträchtigen kann. Eine Änderung darf nur durch geschultes Personal durchgeführt werden, da sonst unter Umständen eine Beschädigungen der Hardware droht.

5.3 Instandhaltung

5.3.1 Tausch des Reinigungsstäbchens

Die eingesetzten Reinigungsstäbchen (Polyester-Pads) müssen von Zeit zu Zeit getauscht werden. Das Intervall hängt vom Verschmutzungsgrad und dem Kontaminationstyp der zu reinigenden Optiken ab und ist deshalb nicht fix vorgegeben. Als Richtlinie können die empirisch ermittelten Werte in Tabelle 28 herangezogen werden.

Tabelle 28: Maximale Anzahl empfohlener Reinigungsdurchgänge je Padseite.

Typ Grad	lose Partikel	haftende Partikel	Schmutzfilme
niedrig	15	10	5
mittel	12	5	3
hoch	8	5	2

Der Wechsel erfolgt indem das alte Stäbchen aus der Aufnahme gezogen wird. Beim Einsetzen des neuen Stäbchens ist darauf zu achten, dass die Kerbe des grünen Kunststoffschafte am Passstift einrastet. Die richtige Position des Reinigungsstäbchens ist essentiell für den Reinigungsprozess.

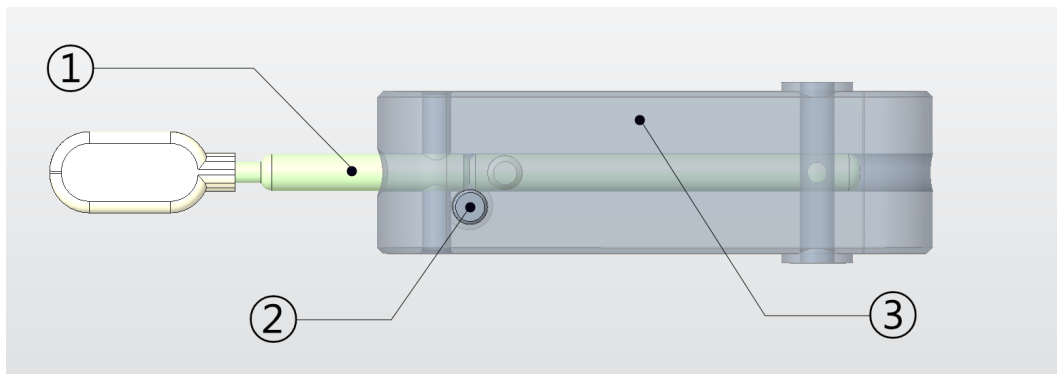


Abbildung 60: Darstellung des Kerbe-Stift Klicksystems

- (1) Schaft mit Kerbe
- (2) Passstift
- (3) Aufnahme für Reinigungsstäbchen

Als Reinigungsstäbchen eignen sich ausschließlich die *Polyester-Tupfer* TX761K des Herstellers Texwipe. Es ist zu beachten, dass die Stäbchen vor dem Einsetzen auf eine Schaftlänge von ca. 24 mm gekürzt werden müssen.

5.3.2 Befüllen der Lösungsmittelspritze

Der Roboter prüft vor jeder Reinigung, ob sich noch genug Lösungsmittel in der Spritze befindet. Ist der Inhalt aufgebraucht, wird der Anwender über ein Popup darüber informiert.

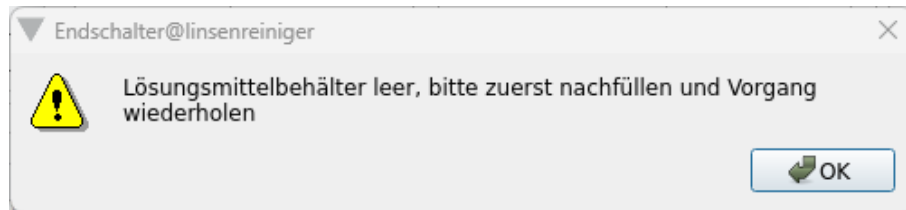


Abbildung 61: Popup bei leerem Lösungsmittelbehälter

Ein erneutes Auffüllen der Lösungsmittelspritze lässt sich auf zwei unterschiedliche Arten durchführen. Bei beiden Varianten ist zwingend darauf zu achten, dass eine persönliche Schutzausrüstung (Schutzbrille etc.) getragen wird, da ein ungeschütztes Hantieren mit Lösungsmitteln gesundheitliche Schäden zur Folge haben kann.

Variante 1

1. Die beiden Schrauben auf der Oberseite des Y-Schlittens lösen und den Wartungsdeckel öffnen, um zur Spritzenpumpe zu gelangen.
2. Die Spritze aus der Pumpe entnehmen und vollständig aufziehen, um das restliche Lösungsmittel aus dem Schlauch zu entfernen. Das ist notwendig, da sonst das Lösungsmittel beim Abstecken der Spritze aus dem Schlauch rinnt.
3. Spritze vom Schlauchsystem abstecken und einen Hilfsbehälter mit genügend Lösungsmittel befüllen.
4. Die Spritzenspitze in den Lösungsmittelbehälter eintauchen und vollständig (bis auf die 12 ml Markierung) aufziehen. Etwaige Lufteinschlüsse vorsichtig herausdrücken.
5. Spritze wieder mit dem Schlauchsystem verbinden und ca. 2 ml per Hand herausdrücken, um den Schlauch wieder zu befüllen.
6. In der Roboter-Software, in den Reiter *Manuelle Kontrolle* wechseln und dort im Abschnitt *Spritzenpumpe* den Button **Aufziehen** betätigen.

7. Anschließend kann die auf ca. 10 ml aufgezogene Spritze wieder eingelegt werden. Den Wartungsdeckel wieder schließen und verschrauben.

Die zweite Variante ist wesentlich einfacher, birgt aber die Gefahr, dass während dem Aufziehen der Spritze, Luft in das Schlauchsystem gelangt. Ist dies der Fall, muss die eingeschlossen Luft über Variante 1 wieder entfernt werden.

Variante 2

1. Einen temporären Behälter vorbereiten und mit ausreichend Lösungsmittel befüllen.
2. Die Gewindeschraube am Düsenhalter lösen und die Düse vorsichtig aus der Halterung ziehen.
3. Anschließend die Düse tief genug in den mit Lösungsmittel gefüllten Behälter tauchen.
4. In der Roboter-Software, in den Reiter *Manuelle Kontrolle* wechseln und dort im Abschnitt *Spritzenpumpe* den Button Aufziehen anklicken. Dabei darauf achten, dass beim Aufziehen keine Luft mit eingesaugt wird.
5. Düse aus dem Lösungsmittelbehälter entfernen und wieder am Düsenhalter festschrauben.

5.4 Fehlerbehebung

5.4.1 CNC-Controller reagiert nicht mehr

Sollte der CNC-Controller, oder das Programm nicht mehr reagieren, muss der Roboter neugestartet werden. Dazu folgende Schritte durchführen:

1. X- und Y-Schlitten über die Rändelschrauben auf den Maschinennullpunkt zurück bewegen.
2. Im Reiter *Manuelle Kontrolle*, Abschnitt *Makros* den Button Neustarten betätigen.
3. Reagiert auch die Software nicht mehr, kann die zentrale Steuerung manuell über Start > Abmelden > Neustarten neugestartet werden.

5.4.2 Tare-Wert außerhalb der Toleranz

Ist das Polyester-Pad während der Initialisierung des Kraftsensors belastet, wird eine Fehlermeldung ausgegeben. Zur Behebung sind folgende Schritte durchzuführen:

1. X- und Y-Schlitten über die Rändelschrauben soweit verstellen, dass das Polyester-Pad entlastet ist.
2. In den Reiter *Manuelle Kontrolle* wechseln und dort im Abschnitt *Makros* den Button Nullstellen betätigen.
3. War die Nullstellung erfolgreich, Popup mit Klick auf OK bestätigen.
4. Zur Kontrolle über den Button Druckwert den aktuellen Druckwert auslesen. Nach erfolgreicher Nullstellung wird ein Druckwert um '0%' ausgegeben.

5.4.3 Vorgehen bei Schrittverlusten

Wird eine der Spindeln nach Initialisierung des Gerätes beabsichtigt oder auch unbeabsichtigt per Hand verdreht, geht die Position der Schrittmotoren verloren. Ein anschließendes Verfahren der Schlitten kann zur Beschädigung der Hardware führen, da die, in der CNC-Steuerung hinterlegte Position nicht mehr mit den tatsächlichen Koordinaten übereinstimmt. In dieser Situation sind folgende Punkte durchzuführen:

1. X- und Y-Schlitten über die Rändelschrauben per Hand auf den Maschinennullpunkt zurückbewegen.
2. Im Reiter *Manuelle Kontrolle* im Abschnitt *Makros* den Button Soft-Reset betätigen.

5.4.4 Alarm-Status beheben

Befindet sich der Roboter nach dem Einschalten im Status '**Alarm**', deutet das auf eine fehlgeschlagene Referenzfahrt hin. Zur Behebung des Problems, ist vor dem Start der Software zu überprüfen, dass keiner der drei Schlitten einen Endschalter betätigt.

Ein weiteres Indiz für eine fehlgeschlagene Referenzfahrt ist, dass sich die Schlitten über den Reiter *Manuelle Kontrolle* nicht verschieben lassen.

Literatur

- [1] A. Thoß, D. Eidemüller, *et al.*, “Biomedizinische Optik und Augenoptik: Photonik in den Lebenswissenschaften,” in *Clusterreport Optik & Photonik*, p. 96, Wirtschaftsförderung Brandenburg, September 2018.
- [2] Wild Gruppe, “Medizinische Lasertechnik,” Oktober 2018. https://www.wild.at/wp-content/uploads/2018/10/WILD-Gruppe_Market-Focus_Medizinische-Lasertechnik_web.pdf.
- [3] Wild Gruppe, “Lasertechnik: Laseranwendungen für höchste Ansprüche,” November 2020. <https://www.wild.at/wp-content/uploads/2020/11/trm-broschuere.pdf>.
- [4] Wild Gruppe, “Testen, testen, testen.,” September 2020. <https://www.wild.at/news/pcr-test-blutgasanalyse-ivd/>, letzter Zugriff: 20.01.2023.
- [5] J.-Y. Plessier, M. Henrist, *et al.*, “A New Guideline on Contamination Control for Space Optical Instruments,” vol. 467, pp. 129 – 134, Januar 2001.
- [6] A. Polinski, “Optiken richtig reinigen: Ein Leitfaden zur Handhabung und Reinigung von Optiken,” Februar 2018. <https://www.asphericon.com/blog/detail/optiken-richtig-reinigen>, letzter Zugriff: 13.10.2022.
- [7] I. F. Stowers and H. G. Patton, “Cleaning Optical Surfaces,” in *Optical Coatings: Applications and Utilization II* (G. W. DeBell and D. H. Harrison, eds.), vol. 0140, pp. 16 – 31, International Society for Optics and Photonics, SPIE, September 1978.
- [8] R. Trierweiler, *Staub: Natürliche Quellen und Mengen*. Essentials, Springer Fachmedien Wiesbaden, 2020.
- [9] I. Fernandes, “Staub ist nicht gleich Staub,” März 2020. https://www.planet-wissen.de/gesellschaft/sauberkeit/staub_feine_partikel_mit_grosser_wirkung/index.html, letzter Zugriff: 14.10.2022.
- [10] M. Zölffel, “The Clean Microscope,” Septmeber 2008. https://hcbi.fas.harvard.edu/files/hcbidoug/files/50-10025_the_clean_microscope_e.pdf.
- [11] R. Schalck, *The Proper Care of Optics: Cleaning, Handling, Storage, and Shipping*. SPIE Digital Library, SPIE Press, 2013.

- [12] P. Weinzerl, “Arbeitsanweisung AA-0003: Montage und Prüfung von Optik-Baugruppen,” April 2021. (unveröffentlicht).
- [13] Spetec, “Reinraumtechnik: Portable Reinraumtechnik,” Dezember 2021. https://www.spetec.de/downloads/Reinraumtechnik_Prospekt.pdf.
- [14] Edmund Optics, “Reinigung von Optiken: Tipps für eine sachgemäße Pflege,” <https://www.edmundoptics.de/knowledge-center/application-notes/optics/cleaning-optics/>, letzter Zugriff: 17.10.2022.
- [15] R. Paschotta, “Cleaning of Optics,” https://www.rp-photonics.com/cleaning_of_optics.html, letzter Zugriff: 18.10.2022.
- [16] Floriane Teofanescu, “Cleaning Optics: Choosing the Best Method,” https://www.photonics.com/Articles/Cleaning_Optics_Choosing_the_Best_Method/a32199, letzter Zugriff: 17.10.2022.
- [17] G. Guéhenneux *et al.*, “Impact of outgassing organic contamination on laser-induced damage threshold of optics: effect of laser conditioning,” in *Laser-Induced Damage in Optical Materials: 2005* (G. J. Exarhos, A. H. Guenther, *et al.*, eds.), vol. 5991, p. 59910F, International Society for Optics and Photonics, SPIE, 2006.
- [18] D. Hamilton and J. Hamilton, “Optics restoration - First Contact Liquid Polymer Restores Sensitive Optical Surfaces to >As New< Condition,” pp. 32 – 34, Februar 2012.
- [19] E. Kubacki, “The dirt on cleaning optics,” *Photonics spectra*, vol. 40, pp. 56 – 60, März 2006.
- [20] Newport, “Technical Note: How to Clean Optics,” <https://www.newport.com/n/how-to-clean-optics>, letzter Zugriff: 17.10.2022.
- [21] Sunex, “Cleaning Procedure for Optics,” <https://sunex.com/2020/01/31/cleaning-procedure-for-optics/>, letzter Zugriff: 21.10.2022.
- [22] R. Rottenfusser, E. E. Wilson, *et al.*, “Microscope Cleaning and Maintenance,” <https://zeiss-campus.magnet.fsu.edu/print/basics/care-print.html>, letzter Zugriff: 19.10.2022.
- [23] I. Cordero, “How to care for and clean optical surfaces,” *Community Eye Health Journal*, vol. 23, Dezember 2010.

- [24] J. Millington, “*How to Clean Microscope Optics*,” <https://opti-tech.ca/how-to-clean-microscope-optics/>, letzter Zugriff: 18.10.2022.
- [25] P. Jackson and J. Hamilton, “Cleaning and Protecting Large Mirrors Using a Polymer Solution,” April 2008. <https://www.mobilityengineeringtech.com/component/content/article/adt/pub/briefs/photonics/4809>, letzter Zugriff: 18.10.2022.
- [26] E. S. Bailey, N. Confer, *et al.*, “Increased Laser Damage Threshold by Protecting and Cleaning Optics Using First Contact Polymer Stripcoatings: Preliminary Data,” *Laser-Induced Damage in Optical Materials*, vol. 7132, 2008.
- [27] F. Fuchs, “19 - Ultrasonic cleaning and washing of surfaces,” in *Power Ultrasonics*, pp. 577–609, Elsevier Ltd, 2015.
- [28] L. Azar, “Cavitation in Ultrasonic Cleaning and Cell Disruption,” *Controlled Environments*, pp. 14–17, Februar 2009.
- [29] J. L. Lizon and S. Deiries, “Cleaning of extremely sensitive optical surfaces,” in *Advances in Optical and Mechanical Technologies for Telescopes and Instrumentation*, vol. 9151, pp. 1793–1800, SPIE, 2014.
- [30] R. Kohli, “Chapter 1 - Sources and Generation of Surface Contaminants and Their Impact,” in *Developments in Surface Contamination and Cleaning* (R. Kohli and K. Mittal, eds.), pp. 1–49, Oxford: William Andrew Publishing, 2015.
- [31] Mettler Toledo, “Test auf nicht flüchtige Rückstände: Gravimetrische Bestimmung des NVR-Gehalts,” https://www.mt.com/at/de/home/applications/Laboratory_weighing/non-volatile-residue-test-gravimetric-determination.html, letzter Zugriff: 14.01.2022.
- [32] ISO 14644-9:2022, “Cleanrooms and associated controlled environments — Part 9: Assessment of surface cleanliness for particle concentration,” Norm, International Organization for Standardization, Geneva, CH, November 2022.
- [33] PhysicsOpenLab, “Light Scattering,” Juli 2019. <https://physicsopenlab.org/2019/07/10/light-scattering/>, letzter Zugriff: 15.01.2023.
- [34] D. Katsir, “Stray Light Reduction in Industrial Optics,” März 2021. <https://acktar.com/stray-light-reduction/>, letzter Zugriff: 03.11.2022.

- [35] M. Damm, “Entwicklung eines automatisierten Messplatzes zur Bestimmung ultrakurzpuls-laserinduzierter Zerstörschwellen optischer Komponenten,” Master’s thesis, Friedrich-Schiller-Universität Jena, Mai 2009.
- [36] Edmund Optics, “Hintergrundinformationen zu optischen Spezifikationen: Laserzerstörschwelle,” <https://www.edmundoptics.de/knowledge-center/application-notes/optics/understanding-optical-specifications/>, letzter Zugriff: 03.11.2022.
- [37] Markforged, “Onyx Chemical Resistivity,” Mai 2018. <https://static.markforged.com/downloads/ChemicalResistivitySlick.pdf>.
- [38] Raspberry Pi, “Datasheet: Raspberry Pi 4 Model B,” Datenblatt, Juni 2019. <https://datasheets.raspberrypi.com/rpi4/raspberry-pi-4-datasheet.pdf>, Release 1.
- [39] M. Natali, “Ansteuerung unterschiedlicher Sensoren und Aktoren über einen Microcontroller,” Seminararbeit, September 2022. (unveröffentlicht).
- [40] SaintSmart, *Benutzerhandbuch: SainSmart Genmitsu Controller Board*, Juni 2020. http://s3.amazonaws.com/s3.image.smart/download/101-60-284/Controller_Board_Benutzerhandbuch-German-V1.0-20200612.pdf.
- [41] SaintSmart, “GRBL Controller Board for Genmitsu CNC Router 3018, 3018-PRO, 1810-RPO,” <https://www.sainsmart.com/collections/genmitsu-cnc-replacement-upgrade-parts/products/controller-board-for-genmitsu-cnc-router-3018-3018-pro-1810-rpo>, letzter Zugriff: 22.01.2023.
- [42] Handson Technology, “Data Specs: 17HS4401S Stepper Motor,” Datenblatt, Juni 2017. <https://www.handsontec.com/dataspecs/17HS4401S.pdf>.
- [43] T. Friebe *et al.*, “Grbl Wiki,” Oktober 2021. <https://github.com/gnea/grbl/wiki>, letzter Zugriff: 22.01.2023, Revision 26.
- [44] RS Pro, “Datasheet: 12 V dc, 1000 gcm, Brushed DC Geared Motor, Output Speed 1318 rpm,” Datenblatt, März 2020. <https://docs.rs-online.com/d935/A700000007082005.pdf>.
- [45] K. Tkotz *et al.*, *Fachkunde Elektrotechnik*. Haan-Gruiten: Verlag Europa-Lehrmittel, 23 ed., 2002.

- [46] M. Fatma and M. Hamid, “PWM speed control of dc permanent magnet motor using a PIC18F4550 microcontroller,” *IOP Conference Series: Materials Science and Engineering*, vol. 602, p. 012017, September 2019.
- [47] AVIA Semiconductor, “HX711: 24-Bit Analog-to-Digital Converter (ADC) for Weigh Scales,” Datenblatt, Juni 2009. https://cdn.sparkfun.com/datasheets/Sensors/ForceFlex/hx711_english.pdf.
- [48] SPT, “SPT4412LV-210,” Datenblatt, Juni 2021. <http://www.spt-servo.com/Product/1568293017.html>, letzter Zugriff: 29.11.2022.
- [49] G. Häberle *et al.*, *Tabellenbuch Mechatronik: Tabellen, Formeln, Normenanwendung*. Haan, Deutschland: Europa-Lehrmittel, 4 ed., Januar 2006.
- [50] Antclabs, “BLTouch: Auto Bed Leveling Sensor for 3D Printers,” <https://www.antclabs.com/bltouch-v3>, letzter Zugriff: 04.12.2022.
- [51] Texas Instruments, “DRV8825 Stepper Motor Controller IC,” Datenblatt, Juli 2014. <https://www.ti.com/lit/ds/symlink/drv8825.pdf>, Revision F.
- [52] Elecrow, “RC050 5 inch HDMI 800 x 480 Capacitive Touch LCD Display,” https://www.elecrow.com/wiki/index.php?title=RC050_5_inch_HDMI_800_x_480_Capacitive_Touch_LCD_Display_for_Raspberry_Pi/_PC/_SONY_PS4, letzter Zugriff: 06.12.2022.
- [53] J. Calvert, A. Scheller, *et al.*, “Raspberry Pi Documentation: The raspi-config Tool,” April 2022. <https://www.raspberrypi.com/documentation/computers/configuration.html>, letzter Zugriff: 22.01.2023.
- [54] K. Huang, “Robotics and Embedded Systems: Interrupts vs. Polling,” Technische Universität München, Oktober 2015. <http://archive.www6.in.tum.de/www6/pub/Main/TeachingWs2015HSCDLegoCar/4.3-IO-Interrupts.pdf>.
- [55] Python Software Foundation, “*Thread-based parallelism: Lock Objects*,” <https://docs.python.org/3/library/threading.html?highlight=locks#lock-objects>, letzter Zugriff: 23.01.2023.
- [56] Sphinx developers, “SPHINX Python Documentation Generator.” <https://www.sphinx-doc.org/en/master/index.html>, letzter Zugriff: 09.01.2023.
- [57] L. Gail and U. Gommel, *Reinraumtechnik*. VDI-Buch, Berlin, Heidelberg: Springer Berlin / Heidelberg, 4 ed., 2018.

A Stücklisten

Tabelle 29: Auflistung aller mechanischen und elektrischen Zukaufteile

Bauteil	Hersteller	Beschreibung	Stk.
Getriebemotor	RS Pro	12 V (DC), 1000 g·cm, 1318 min ⁻¹	1
CNC Fräse	Saint Smart	Genmitsu 3018-PRO	1
Servo	SPT	SPT4412LV, 6V, 12 kg.cm	1
Vorspannmutter	POLISI3D	T8x2mm	1
Steuerung	Raspberry PI	Raspberry PI 4 8 GB	1
Endschalter	WJMY	3-Polig, 19.3x14.4x2.7 cm	4
Silikonschlauch	Antrella	Innen-Ø 1.6 mm, Außen-Ø 3 mm	1
ADC	Angeek	HX711 Breakoutboard, 24Bit	1
Wägezelle	Sourcing Map	200g, 4.7x1.2x0.63 cm	1
Kegelrad	RS Pro	POM, 1:1, 20 Zähne, Bohrung 6 mm	2
Trapezspindel	Sienoc	T8 2 mm Steigung, 100 mm, mit Kupfermutter	1
Treiberboard	DollaTek	Expansionboard, 42x42x15 mm	1
Treiber	AZDelivery	DVR8825	1
Schrittmotor	Usongshine	NEMA17, 1.5 A, 1.8 °	1
Display	Elecrow	Kapazitives Touchdisplay, 5", 800x480 Pixel, HDMI	1
DC-DC Wandler	Yizhet	LM2596, 3.0-40 V zu 1.5-35 V	1
Touchsensor	Antclabs	BLTouch, Smart V3.1	1
USB-Kabel	Goobay	Micro-USB auf USB-A Kabel 0.6 m	1
HDMI-Kabel	Goobay	HDMI auf micro HDMI Kabel 0.5 m	1
USB-Kabel	Delock	Mini-USB auf USB-A Kabel 0.5 m	1
Einwegspritze	Norm-Ject	Luer exzentrisch, 10/12 ml	1
Dosiernadel	Gonano	Luer-Lock, ID 1.37 mm	2
USB-C Stecker	JSER	USB 3.1, schwarzes Gehäuse	1

Tabelle 30: Auflistung aller mechanischen Fertigungsteile

Bauteil	Datei	Material	Stk.
Z-Schlitten	z_schlitten_v2.prt	Onyx	1
Befestigung Z-Schlitten	befestigung.prt	PLA	1
Rändelschraube	raendelschraube.prt	PLA	2
Steuerungsgehäuse	halterung_rpi4.prt	PLA	1
Steuerungsgehäuse Deckel	halterung_rpi4_deckel.prt	PLA	1
Adapter	adapter.prt	Onyx	1
Kopf	kopf_v2.prt	Onyx	1
Aufnahme	pad_halter.prt	Onyx	1
Halterung für Feder	federhalter.prt	Onyx	1
Halterung Düse	duesen_halter.prt	PLA	1
Endschalteradapter X	adapter_endschalter_x.prt	Onyx	1
Endschalteradapter Y	adapter_endschalter_y.prt	Onyx	1
Endschalteradapter Z	adapter_endschalter_z_v2.prt	Onyx	1
Displaygehäuse Basis	display_gehaeuse_unten.prt	PLA	1
Displaygehäuse Deckel	display_gehaeuse_oben.prt	PLA	1
Displaygehäuse Deckel	display_gehaeuse_oben.prt	PLA	1
Displayhalterung	display_halter.prt	ABS	1
Deckel Y-Schlitten	haube_deckel.prt	Onyx	1
Drehteller Getriebeblock	karussell_getriebeblock.prt	Onyx	1
Drehteller	drehteller.prt	Onyx	1
Halterung DC-Motor	halterung_dcmotoerv2.prt	Onyx	1
Synchrone Scheibe T40	gt2_t40.prt	PLA	1
Befestigungswinkel Drehteller	karussell_winkel_rechts.prt	PLA	3
Befestigungsschiene Drehteller	karussell_winkel_links.prt	PLA	1
Befestigungswinkel Abdeckung	gehaeuse_winkel_schmal.prt	PLA	1
Befestigungswinkel Abdeckung	gehaeuse_winkel_schmal.prt	PLA	1
Spritzenpumpe Basis	boden.prt	PLA	1
Kolbenschieber	kolbenschieber.prt	PLA	1
Welle 6 mm	-	NiRoSta	1

Tabelle 31: Auflistung aller verbauten Normteile außer Schrauben, Muttern und Stifte

Bauteil	Hersteller	Beschreibung	Stk.
Rillenkugellager	SKF	6207-Z/C3, Innen-Ø 35 mm, Außen-Ø 72 mm, Breite 17 mm	1
Rillenkugellager	GRW Germany	SS696 2Z, Innen-Ø 6 mm, Außen-Ø 15 mm, Breite 5 mm	3
Linearlager	Sourcing Map	LM8UU, Innen-Ø 8 mm, Außen-Ø 15 mm, Länge 17 mm	3
Linearlager	Sourcing Map	LM10UU, Innen-Ø 10 mm, Außen-Ø 19 mm, Länge 29 mm	4
Gleitlager	Iglidur	GFM-0506-06, Innen-Ø 5 mm, Außen-Ø 6 mm, Länge 5 mm	1
Gleitlager	Iglidur	GFM-0405-06, Innen-Ø 4 mm, Außen-Ø 5 mm, Länge 6 mm	1
Gleitlager	Iglidur	GSM-0405-04, Innen-Ø 4 mm, Außen-Ø 5 mm, Länge 4 mm	1
Synchrone Scheibe	SeeRaph	GT2, 16 Zähne, 6 mm Bohrung, Aluminium	1
Zahnriemen	Turmberg3D	GT2, 320 mm Länge, 6 mm Breite	1
Gewindeeinsatz	Bossard	Außengewinde M5, Innengewinde M3, Länge 4 mm	7
Sicherungsscheibe	Bossard	BN 810 für 6 mm Welle	1

B Verwendete Geräte und Software

Tabelle 32: Für die Entwicklung des Roboters verwendete Messgeräte

Gerät	Hersteller	Modell	Seriennummer
Handmultimeter	Fluke	175 True RMS	54520442
Oszilloskop	Voltcraft	DSO1062D	T 1G/012 009902
Digitales Kraftmessgerät	Alluris	FMI-100B5	05688
Mikroskop	Leica	M80	-
Kameraaufsatz	Leica	MC190 HD	1120133289

Tabelle 33: Für die Entwicklung des Roboters verwendete Software

Name	Hersteller	Kategorie	Version
Creo	PTC	CAD	7.0.0.0
Matlab	MathWorks	Mathematik	R2020a
Affinity Photo	Serif	Bildbearbeitung	1.10.6.1655
Visual Studio Code	Microsoft	IDE	1.74.2
Thonny	Raspberry Pi	Softwareentwicklung	3.3.14
GIT	-	Softwareentwicklung	2.39.0
MobaXterm	Mobatek	Terminal, X11	22.1
ImageJ	National Institute of Health	Bildbearbeitung	1.53t
Leica Application Suite	Leica Microsystems	Bildbearbeitung	4.13.0
Arduino IDE	Arduino	Softwareentwicklung	2.0.3
Qt-Designer	Qt Company	GUI Design	5.11.1
Sphinx	-	Softwareentwicklung	5.3.0
Python Interpreter	Python Software Fd.	Softwareentwicklung	3.9.2

C Korrespondenzen



Manuel Natali <manuel.natali@gmail.com>

... / [Kontakt Webseite asphericon] // Verschmutzungsgrad von Optiken

Stefan Klinzing <s.klinzing@asphericon.com>
An: manuel.natali@gmail.com

6. Januar 2023 um 11:16

Hallo Herr Natali,
vielen Dank für Ihre Nachricht!

Ich habe Ihre Frage bzgl. "Verschmutzungsgrad von Optiken" mit unserer Qualitätssicherung besprochen und kann Ihnen nun Folgendes mitteilen:

"... speziell für Optiken gibt es eine solche Norm nicht, wohl aber die allgemeine DIN EN ISO 14644-9, die sich mit Verschmutzungen durch Partikel beschäftigt, in diese auch Optiken fallen. Auch die VDA 19 sei hier genannt. Was filmische Verunreinigungen anbelangt, existieren derzeit noch keine allgemeinen Normen."

Ich hoffe, diese Informationen helfen Ihnen weiter und wünsche Ihnen viel Erfolg für Ihre Diplomarbeit.

Freundliche Grüße

--

Stefan Klinzing
Vertrieb / Sales
E-Mail: s.klinzing@asphericon.com