
BACHELORARBEIT

Herr
Shyngys Muratbek, Ing.

**Entwicklung eines automatisierten
Systems zur Datenerfassung und
Sortierung von Mikrovibrationssensoren**

Mittweida, Mai 2023

Fakultät Ingenieurwissenschaften

BACHELORARBEIT

Entwicklung eines automatisierten Systems zur Datenerfassung und Sortierung von Mikrovibrationssensoren

Autor:

Shyngys Muratbek

Studiengang:

Elektrotechnik Automation - Industrie 4.0

Seminargruppe:

EA19wA-b

Erstprüfer:

Prof. Dr. rer. nat. Dieter Braun

Zweitprüfer:

Prof. Dr.-Ing. Renè Pleul

Einreichung:

Mittweida, 20.05.2023

Verteidigung/Bewertung:

Mittweida, 24.05.2023

Faculty of **Engineering Sciences**

BACHELOR THESIS

Development of an automated system for data acquisition and sorting of micro vibration sensors

Author:

Shyngys Muratbek

Course of Study:

Electrotechnology Automation - Industry 4.0

Seminar Group:

EA19wA-b

First Examiner:

Prof. Dr. rer. nat. Dieter Braun

Second Examiner:

Prof. Dr.-Ing. Renè Pleul

Submission:

Mittweida, 20.05.2023

Defense/Evaluation:

Mittweida, 24.05.2023

Bibliografische Beschreibung:

Muratbek, Shyngys:

Entwicklung eines automatisierten Systems zur Datenerfassung und Sortierung von Mikrovibrations-sensoren. – 2023. – 42 S.

Mittweida, Hochschule Mittweida – University of Applied Sciences, Fakultät Ingenieurwissenschaften, Bachelorarbeit, 2023.

Referat:

Die vorliegende Bachelorarbeit widmet sich bedeutenden Aspekten im Bereich des Ingenieurwe-sens. Im Fokus steht die Untersuchung des Verhaltens von Mikrovibrationssensors. Zudem wird ein Prototyp eines automatisierten Auswertesystems für diese Sensoren konzipiert und umgesetzt. Die erfolgreiche Durchführung dieser Arbeit erfordert nicht nur theoretische Kenntnisse, sondern auch praktische Fähigkeiten und den Einsatz von Tools wie Bash-, C- und Python-Programmierung sowie PCB- und CAD-Design.

Inhaltsverzeichnis

Inhaltsverzeichnis	I
Abbildungsverzeichnis	III
Tabellenverzeichnis	V
Nomenklatur	VII
Danksagung	IX
1 Einleitung	1
2 Theoretische Grundlagen	3
2.1 Grundlagen der Messtechnik	3
2.2 Funktionsweise des Gleichstrommotors	5
2.3 Funktionsweise des Schrittmotors	6
2.4 Funktionsweise von Kugelschaltern und Mikrovibrationssensoren	10
3 Untersuchung des Verhaltens der Mikrovibrationssensoren	13
3.1 Ziel der Untersuchung	13
3.2 Bekannte Messdaten	13
3.3 Versuchsaufbau	16
3.4 Datenaufnahme	17
3.5 Ergebnisse	18
4 Entwurf eines multifunktionalen Moduls	23
4.1 Konzept	23
4.2 Wendelförderer	24
4.3 Multifunktionales Modul	25
4.3.1 Messmanipulator	25
4.3.2 Mikrocontroller	27
4.3.2.1 Auswahl eines Mikrocontrollers	27
4.3.2.2 Algorithmus	28
5 Implementation eines multifunktionalen Moduls	31
5.1 Aufbau eines Prototyps	31
5.2 Elektrische Schaltungen	32
5.3 Softwaretechnische Lösungen	35
5.4 Testmethodik	36
6 Fazit und Ausblick	41
Anhang	43
A Implementation der Funktionen in der Programmiersprache C	43
B Bash-Befehle	47

Literaturverzeichnis	49
Eidesstattliche Erklärung	51

Abbildungsverzeichnis

2.1	Kennlinie eines 3-Bit-AD-Wandlers [12]	4
2.2	Schrittmotoren mit Dauermagnet-Rotor. a) unipolar; b) bipolar. Die Elektromagnete A – B und C – D sind gegeneinander in einem Winkel von 90° angeordnet. [16]	7
2.3	Vollschrittbetrieb [16]	7
2.4	Halbschrittbetrieb [16]	7
2.5	Das Drehmoment in Abhängigkeit von Bauart und Schrittfrequenz [10]	8
2.6	Schrittmotorsteuerung mit der Direktsteuerung durch Mikrocontroller [10]	9
2.7	Funktionsweise eines Kugelsensors [3]	11
2.8	Merkmale des Sensors MVS0409 [11]	12
2.9	Anordnung der Sensoren im Wafer [3]	12
3.1	Kundenapplikation	13
3.2	Messanlage EIFuTe [3]	14
3.3	Erläuterung des Testkonzepts - Ausrichtung des Wafers im Raum	14
3.4	ESB der Sensoren des Wafers	14
3.5	Beurteilungsmatrix [3]	15
3.6	Optimiertes elektronisches Auswertekonzept für Testsysteme [3]	15
3.7	Gemittelte Pulszahl und On_{Ratio} über 20.000 Entwicklungssensoren	16
3.8	Versuchsaufbau	16
3.9	ESB Messschaltung	16
3.10	Ansprechverhalten eines hochwertigen Sensors (links) und eines minderwertigen Sensors (rechts)	17
3.11	Sektor 1 - 4, ξ in Abhängigkeit von t_h	19
3.12	Sektor 5 - 8, ξ in Abhängigkeit von t_h	20
3.13	Sektor 1 - 4, ξ in Abhängigkeit von n_p	20
3.14	Sektor 5 - 8, ξ in Abhängigkeit von n_p	21
3.15	Beziehung zwischen ξ und den Parameter \bar{t}_h und \bar{n}_p	21
4.1	Visualisierung des Konzeptes mittel CAD-Modellierung	23
4.2	CAD-Modell eines Wendelförderers	24
4.3	Konzept eines multifunktionalen Moduls als CAD-Modell	25
4.4	CAD-Modelle eines Messmanipulators	26
4.5	Technische Zeichnung einer würfelförmigen Messstation	26
4.6	Flussdiagramm des entwickelten Algorithmus	28
5.1	Messmanipulator	32
5.2	multifunktionales Modul	32
5.3	Messschaltung	32
5.4	Hilfeschaltung	32
5.5	Elektrische Schaltung zur Steuerung der Pumpe	33
5.6	Pinbelegung von ULN2003 [8]	34
5.7	funktionales Blockschaltdiagramm [8]	34
5.8	Schaltungsentwurf mittels EAGLE	34
5.9	Visualisierung der rohen Messdaten des Oszilloskops	38
5.10	Filterung der Messdaten des Oszilloskops und Visualisierung der Messdaten des Mikrocontrollers	39

5.11 Filterung der Messdaten des Oszilloskops und Visualisierung der Messdaten des Mikro-controllers	39
5.12 Visualisierung des diskretisierten Messsignals	39

Tabellenverzeichnis

2.1 Klassifizierung von Messsignalen bezüglich der Signalformen [12]	3
2.2 Die Erregungszustände der Schrittmotoren mit Dauermagnetrotor [10]	10
3.1 Gruppen und Filtern	18
3.2 Datenrepräsentation Gruppen C, D, E	18

Nomenklatur

I	Strom
M	Drehmoment
On_{Ratio}	prozentuales Verhältnis von Hightime zur Gesamtzeit der Messung
R_H	Widerstandswert des MVS im oberen Schaltpunkt (206Ω)
R_L	Widerstandswert des MVS im unteren Schaltpunkt (100Ω)
\vec{B}	Magnetfeld
\vec{F}	Kraft
\vec{l}	Länge des Leiters
\bar{n}_p	Mittelwert der Anzahl der Impulse pro Messung über acht Sektoren
\bar{t}_h	Mittelwert der Hightime über acht Sektoren
μC	Mikrocontroller
ξ	Güte eines MVS in Prozent
n_m	Anzahl der Messungen
n_o	Anzahl der qualitativ minderwertigen Messergebnissen
n_p	Anzahl der Impulse pro Messung
r	Korrelationskoeffizient
t_h	Hightime
ADW	Analog-Digital-Wandler
MVS	Mikrovibrationssensor

Danksagung

An dieser Stelle möchte ich mich bei all denjenigen bedanken, die mich während der Anfertigung dieser Bachelorarbeit unterstützt und motiviert haben.

Zuerst gebühren mein Dank Prof. Dr.-Ing. René Pleul und Prof. Dr. Dieter Braun die meine Bachelorarbeit betreut und begutachtet haben. Für die hilfreichen Anregungen und die konstruktive Kritik bei der Erstellung dieser Arbeit möchte ich mich herzlich bedanken.

Ich bedanke mich bei dem Projektleiter in der B&B-Gruppe Herrn Sören Gast, für die organisatorische Unterstützung meiner Forschung.

1 Einleitung

Die Fertigung von **Mikrovibrationssensor (MVS)** stellt einen komplexen mehrstufigen Prozess dar, der von zahlreichen Einflussfaktoren geprägt ist und eine bedeutende Rolle für die Qualität des Endprodukts spielt. Infolgedessen besteht stets die Möglichkeit, dass während der Herstellung Fehlerereignisse auftreten können. Im Falle einer Reklamation seitens der Kunden über fehlerhaftes Verhalten müssen Stichproben des bestellten Auftrags von MVS überprüft werden. Derzeit erfolgt diese Überprüfung manuell durch die Spezialisten am Standort Mittweida der B&B Gruppe. Eine Automatisierung dieses Prozesses könnte jedoch zu einer Verbesserung der Geschwindigkeit und Effektivität führen. Um diesen Prozess automatisieren zu können, ist ein klares Verständnis der Eigenschaften von MVS und des Herstellungsprozesses unerlässlich.

Folglich zielt diese Bachelorarbeit darauf ab, das Verhalten fehlerhafter MVS anhand einer realen Kundenapplikation zu untersuchen, um ausreichend Informationen für die Konzeption und den Aufbau eines Prototyps zur Auswertung fehlerhafter Sensoren zu sammeln. Im Rahmen dieser Bachelorarbeit stellt das Konzept einen integralen Bestandteil dar und wird gezielt auf die Ausstattung des Unternehmens zugeschnitten. Der Entwurf wird als Prototyp realisiert, um eine praktische Überprüfung der konzipierten Ideen durchzuführen.

Konkret sind folgende Fragen zu klären:

- Wie lässt sich eine qualitative Unterscheidung zwischen hochwertigen und minderwertigen MVS vornehmen?
- Ist es möglich, auf Basis der vorhandenen Messdaten eine Klassifizierung der MVS vorzunehmen?
- Wie kann der Auswertungsprozess effizient automatisiert werden?
- Welche Herausforderungen gilt es beim Aufbau des Prototyps zu überwinden und wie können diese erfolgreich bewältigt werden?

Die vorliegende Bachelorarbeit ist in der Zusammenarbeit mit einem Partnerunternehmen – B&B-Gruppe, entstanden. Dieses Unternehmen verfügt über eine umfassende Expertise im Bereich der Leiterplattenfertigung und bietet eine breite Produktpalette an.

2 Theoretische Grundlagen

2.1 Grundlagen der Messtechnik

Die heutige Messtechnik nutzt vorwiegend elektrische Signale, jedoch finden auch mechanische, pneumatische und hydraulische Signale Anwendung, insbesondere im Kontext von technischen Regelsystemen. Eine vollständige Charakterisierung von Messsignalen erfordert jedoch mehr als diese bloße Unterscheidung nach Signalart. Zumindest eine Unterscheidung des Wertevorrats von einem oder mehreren Informationsparametern (analog oder diskret) sowie der zeitlichen Erscheinungsform des Signals (kontinuierlich oder diskontinuierlich) ist notwendig. Die Tabelle 2.1 stellt die Klassifizierung von Messsignalen bezüglich der Signalformen dar. Im Kontext der folgenden Kapitel ist es von

Tabelle 2.1: Klassifizierung von Messsignalen bezüglich der Signalformen [12]

lfd. Nr	Signalcharakteristik	Erläuterung	Vorteil	Nachteil
1.1	Analog	Informationsparameter kann theoretisch beliebig viele Werte innerhalb seines Wertebereichs annehmen	Proportionale Abbildung zwischen Messsignal und Informationsparameter	einfach zu stören, z. B. durch externe Störsignale, Rauschen Temperaturdrift usw.
1.2	Diskret	Informationsparameter kann nur endlich viele Werte innerhalb seines Wertebereiches annehmen	Störeinflüsse machen sich erst nach Überschreiten von Grenzwerten bemerkbar	Bei der Abbildung analoger Messwerte auf einen diskreten Informationsparameter entsteht ein Informationsverlust
2.1	Kontinuierlich	Informationsparameter kann zu jedem beliebigem Zeitpunkt seinen Wert ändern	Jederzeit ist der zeitliche Verlauf von Messwerten verfolgbar	Störungen können jederzeit wirken, Informationsmenge ist oft unnötig groß
2.2	Diskontinuierlich	Informationsparameter kann nur zu diskreten Zeitpunkten seinen Wert ändern	Störungen zwischen den Zeitpunkten der Parameteränderungen können sich nicht auswirken	Informationen stehen nur zu diskreten Zeitpunkten zur Verfügung
3.1	Determiniert	Determiniertheit des Informationsparameters	Information mit einmaliger Messung gewinnbar	Information kann durch Störung unbrauchbar werden
3.2	Stochastisch	Informationsparameter repräsentiert eine stochastische Größe	Störungen machen sich entsprechend ihrer Verteilung nur stark reduziert bemerkbar, sie werden über die Messzeit integriert	Information ist erst mit mehrmaligen Messungen zu gewinnen, das erfordert einen großen Zeitbedarf

Bedeutung zu erwähnen, dass der Fokus dieser Arbeit hauptsächlich auf der Verwendung des Mikrocontrollers zur Messung und Verarbeitung eines deterministischen, kontinuierlichen Analogsignals liegt.

In der heutigen Zeit der digital arbeitenden Messtechnik und rechnergestützter Systeme wird eine immer größere Nachfrage nach digitalisierten Eingangsinformationen gestellt. Da die meisten Messgrößen in analoger Form vorliegen, ist eine Analog-Digital-Wandlung notwendig, um digitale Messinformationen bereitzustellen. Ein analoges Signal ist durch einen unendlich großen Wertevorrat gekennzeichnet, der im Rechner aufgrund seiner endlichen Speicherstellen nicht darstellbar ist. Daher muss der Wertevorrat auf eine endliche Menge beschränkt werden, um digitalisiert werden zu können. Die Analog-Digital-Wandlung ermöglicht diese Beschränkung, indem das analoge Signal in ein diskretes Signal umgewandelt wird. Dabei wird das kontinuierliche Signal in diskrete Stufen unterteilt, um es digital darstellbar zu machen. Aufgrund der zentralen Rolle, die die Analog-Digital-Wandlung in der modernen Messtechnik einnimmt, ist es unabdingbar, diesem Thema besondere Aufmerksamkeit zu schenken. Die Analog-Digital-Wandlung, kurz AD-Wandlung, bezeichnet dabei den Vorgang der Umwandlung eines analogen Signals in ein digitales Signal. Unter einem analogen Signal versteht man ein kontinuierliches Signal, das eine unendliche Anzahl an möglichen Werten annehmen kann, während ein digitales Signal aus einer endlichen Anzahl von diskreten Werten besteht. Als Begriffsbestimmung für die Analog-Digital-Wandlung gilt die Definition:

„Der unendliche Wertevorrat der analogen Größe wird auf einen endlichen Wertevorrat von Teilbereichen (Quanten) abgebildet.“ [12]

In Abbildung 2.1 wird der grundlegende Gedanke der Analog-Digital-Wandlung verdeutlicht. Die dargestellte gestufte (Treppen-)Kurve repräsentiert die reale Übertragungskurve eines 3-Bit-Analog-Digital-Wandlers, welcher im Folgenden als ADW abgekürzt wird. Bei dieser Darstellung ist die Konstanz des digitalen Ausgangssignals erkennbar, solange sich das analoge Eingangssignal innerhalb des Intervalls von 1 LSB¹ ändert. Erst bei Überschreiten dieses Intervalls wird eine Änderung am ADW-Ausgang nachweisbar. Für einen technisch realisierten ADW ist die Genauigkeit

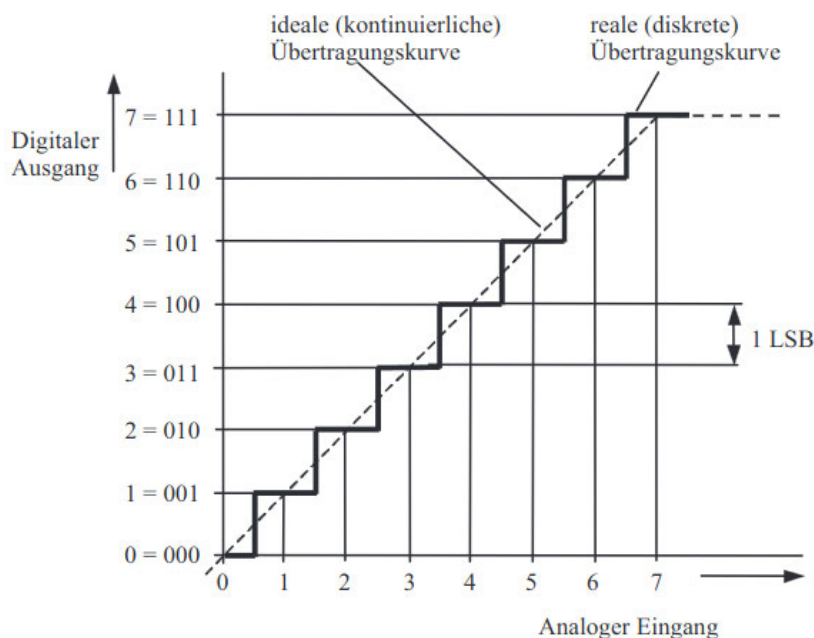


Abbildung 2.1: Kennlinie eines 3-Bit-AD-Wandlers [12]

der Referenzspannung entscheidend für seine erreichbare Auflösung. Die gestrichelt gezeichnete Übertragungskurve in Abbildung 2.1 stellt den Übergang zu unendlich kleinen Quanten dar und entspricht wiederum dem Übertragungsverhalten eines reinen Analogsystems, also einem System

¹least significant bit = kleinstwertigstes Bit, sinngemäß: kleinstes unterscheidbares Inkrement

mit einem unendlichen Wertevorrat für die Ausgangsgröße. Da die Quantisierungsfehler der ADW-Ausgangssignale aufgrund der endlichen Schrittweiten der gestuften Übertragungskurve jedoch nicht verschwinden, ist die Wiedergabe des analogen Signals durch die ADW-Ausgangssignale nicht perfekt.

Bei einer Untersuchung der Analog-Digital-Wandlung hinsichtlich ihres zeitlichen Verhaltens zeigt sich, dass neben der Diskretisierung der Amplitude auch das zeitliche Verhalten des Ausgangssignals diskret ist. Dies resultiert aus der endlichen Reaktionszeit, die jeder ADW benötigt, um auf ein analoges Eingangssignal mit einem digitalen Ausgangssignal zu reagieren. Daher werden nur eine endliche Anzahl von Proben (Samples) aus dem kontinuierlichen Eingangssignal entnommen. Diese Zeitdiskretisierung muss jedoch nicht zwangsläufig mit einem Informationsverlust einhergehen. Das Shannonsche Abtasttheorem gibt Aufschluss darüber, wie oft eine Sinusschwingung abgetastet werden muss, um aus dem digitalisierten Signal wieder regeneriert werden zu können [14]. Wenn f die Frequenz der zu digitalisierenden Sinusschwingung und f_a die Abtastfrequenz für die Sample-Entnahme ist, gilt nach Shannon:

$$f_a > 2 \cdot f \quad (2.1)$$

Korrelation ist ein wichtiges statistisches Konzept, das häufig in der Messtechnik verwendet wird, um Zusammenhänge zwischen zwei oder mehr Variablen zu untersuchen. Die Korrelation ist ein Maß dafür, wie eng die Beziehung zwischen den Variablen ist. Es gibt zwei Arten von Korrelation: positive Korrelation und negative Korrelation. Eine positive Korrelation bedeutet, dass zwei Variablen tendenziell gemeinsam ansteigen oder abfallen, während eine negative Korrelation bedeutet, dass eine Variable tendenziell steigt, während die andere fällt. Die Korrelation wird üblicherweise durch den Korrelationskoeffizienten gemessen. Der **Korrelationskoeffizient** (r) ist ein Wert zwischen -1 und 1, wobei -1 eine perfekte negative Korrelation, 1 eine perfekte positive Korrelation und 0 eine vollständige Unabhängigkeit zwischen den Variablen bedeutet. Der Korrelationskoeffizient wird wie folgt berechnet [13]:

$$r_{xy} = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^n (x_i - \bar{x})^2 \sum_{i=1}^n (y_i - \bar{y})^2}} \quad (2.2)$$

wobei x und y die beiden zu korrelierenden Variablen sind, n die Anzahl der Beobachtungen ist, und \bar{x} und \bar{y} die Mittelwerte von x und y sind.

2.2 Funktionsweise des Gleichstrommotors

Gleichstrommotoren nutzen für ihre Funktionsweise die Prinzipien der Induktion sowie der Lorentzkraft. Diese beiden physikalischen Phänomene bilden die Grundlage für die Arbeitsweise dieser Motoren. Wenn ein Stück eines Leiters, der von einem Strom I in Richtung \vec{l} durchflossen wird, in ein Magnetfeld \vec{B} gebracht wird, entsteht eine Kraft \vec{F} . Diese Kraft wird als Lorentzkraft bezeichnet.

$$\vec{F} = I \cdot (\vec{l} \times \vec{B}) \quad (2.3)$$

Das Prinzip der elektromagnetischen Induktion besagt, dass eine Leiterschleife, durch die ein magnetischer Fluss Φ durch die Fläche A fließt, eine induzierte Spannung U_{ind} erzeugt, wenn sich der Fluss im Laufe der Zeit t ändert und das Integral von $d\vec{s}$ durch $d\vec{A}$ eine rechts wendige Orientierung aufweist. Diese Änderung des magnetischen Flusses kann durch eine Änderung des äußeren Magnetfeldes oder durch Bewegung der Schleife im Magnetfeld hervorgerufen werden. Die induzierte

Spannung U_{ind} ist proportional zur Änderungsrate des magnetischen Flusses Φ mit der Zeit t und zur Größe der Fläche A der Schleife. [2]

$$U_{ind} = -\frac{d\Phi}{dt} \quad \text{mit} \quad \Phi = \int_A \vec{B}dA \quad \text{und} \quad U_{ind} = \oint_{\partial A} \vec{E}d\vec{s} \quad (2.4)$$

Wenn die Leiterschleife geschlossen ist, führt die induzierte Spannung U_{ind} zu einem induzierten Strom I_{ind} . Wenn jedoch eine externe Spannungsquelle an die Leiterschleife angeschlossen wird, die einen Strom I entgegen der induzierten Spannung U_{ind} treibt, wandelt die Lorentzkraft die von der Spannungsquelle gelieferte elektrische Energie in mechanische Energie um. Wenn die Leiterschleife in der „toten Zone“ platziert wird, erreicht der magnetische Fluss Φ bezüglich des Drehwinkels ein Extremum und die Lorentzkraft verschwindet.

In Gleichstrommotoren wird die Leiterschleife entlang einer Welle um einen Anker geführt und über Kommutatorlamellen kontaktiert. Obwohl Gleichstrommotoren prinzipiell Wechselstrommaschinen mit einer durch die Kommutierung entstehenden internen Wechselrichtung sind, erzeugt ein positiver Strom I immer ein positives Drehmoment M . Diese Motoren haben meist mehrere Windungen pro Ankerwicklung und zur Vermeidung von Stillstand in der toten Zone auch mehrere gegeneinander versetzte Ankerwicklungen und Kommutatorlamellen.

Es gibt viele verschiedene Ausführungen von Motoren, die je nach Anwendungsbereich unterschiedliche Leistungen erbringen. Diese können im Bereich von wenigen Milliwatt bis zu mehreren Megawatt liegen und haben unterschiedlich ausgelegte Ankerführungen, Polzahlen und Bürsten.

2.3 Funktionsweise des Schrittmotors

Schrittmotoren besitzen einen mehrpoligen Stator und einen wicklungslosen Rotor. Durch entsprechendes Ansteuern der Statorwicklungen führt der Rotor einen Schritt aus, was einer Drehbewegung um einen bestimmten Winkel entspricht. Der Stator besteht aus mehreren Wicklungen und der Rotor ist entweder aus einem weichmagnetischen Material gefertigt oder als Dauermagnet ausgeführt. Die Anzahl der magnetischen Pole des Rotors bestimmt den Schrittwinkel bzw. die Auflösung. Ein typischer Schrittwinkel beträgt $1,8^\circ$ (200 Schritte/Umdrehung). Die Rotoren von Schrittmotoren haben üblicherweise 50 oder 100 Pole. Um die nächstgelegenen Pole des Rotors anzuziehen, werden die Statorwicklungen zyklisch nacheinander erregt.

Schrittmotoren mit einem Dauermagnet-Rotor stellen aufgrund der Art und Weise, wie sie funktionieren, eine besondere Herausforderung dar. Ein wesentliches Problem bei diesen Motoren ist es, die beiden Elektromagneten im Stator zyklisch umzupolen, um den Rotor mitzuziehen. Der Stator setzt sich aus zwei Elektromagneten zusammen, die jeweils im 90-Grad-Winkel zueinander angeordnet sind (vgl. Abb. 2.2). Durch diese Anordnung entsteht ein Magnetfeld, das den Rotor in seiner Position hält und ihn bewegt, wenn die Polung des Magnetfeldes verändert wird. Zur Veranschaulichung der Funktionsweise ist es hinreichend, eine Annahme von nur 4 Schritten pro Umdrehung bzw. einem Schrittwinkel von 90 Grad (Abb. 2.3 und 2.4) zu treffen.

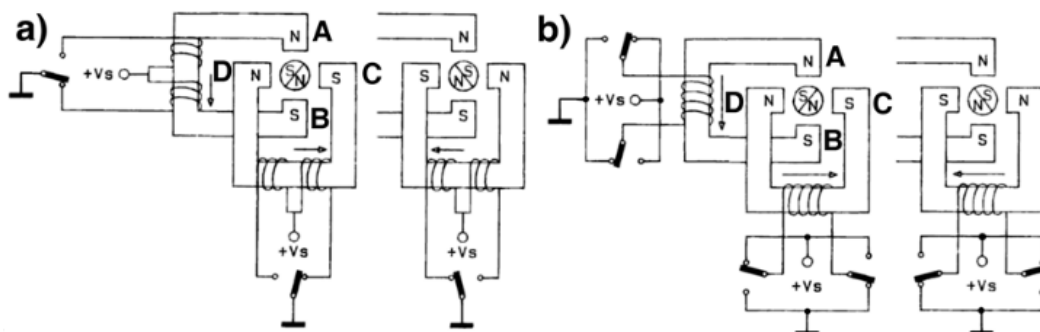


Abbildung 2.2: Schrittmotoren mit Dauermagnet-Rotor. a) unipolar; b) bipolar. Die Elektromagnete A – B und C – D sind gegeneinander in einem Winkel von 90° angeordnet. [16]

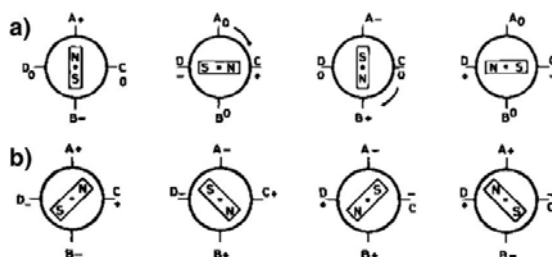


Abbildung 2.3: Vollschrittbetrieb [16]

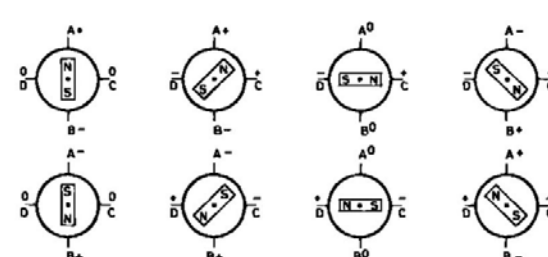


Abbildung 2.4: Halbschrittbetrieb [16]

Im Vollschrittbetrieb der Einphasensteuerung sind nur jene Statorpole aktiv, denen ein Pol des Rotors direkt gegenüberliegt. Die Anziehungskräfte halten den Rotor in seiner Lage. Um den Rotor um einen Schritt weiterzubewegen, ist die aktuelle Erregung auszuschalten, und die jeweils im gewünschten Drehsinne nachfolgenden Statorpole sind so zu erregen, dass die betreffenden Pole des Rotors angezogen werden. Durch zyklisches Aus- und Einschalten von Statorwicklungen kann so der Rotor Schritt für Schritt weiterbewegt werden. Das Erregungsschema für den Vollschrittbetrieb ist wie folgt: AB – CD – BA – DC, wobei AB und BA die Erregung des Elektromagneten mit den Polen A und B bezeichnen, während CD und DC die Erregung des Elektromagneten mit den Polen C und D darstellen. Dabei bedeuten AB/BA und CD/DC jeweils die entgegengesetzten Polungen (vgl. Abb. 2.3 a).

Im Vollschrittbetrieb kann der Dauermagnet-Rotor von Schrittmotoren alternativ durch Zweiphasensteuerung betrieben werden, wobei jeweils zwei benachbarte Pole des Stators so erregt werden, dass der Rotor durch die resultierenden Magnetkräfte in der Mitte zwischen beiden gehalten wird. Diese Betriebsweise erzeugt das höchste Drehmoment. Das Erregungsschema im Vollschrittbetrieb mit Zweiphasensteuerung ist wie folgt: AB/CD – BA/CD – BA/DC – AB/DC (vgl. Abb. 2.3 b). Die Umkehrung der Drehrichtung bei Schrittmotoren erfordert eine entsprechende Änderung der Erregungsreihenfolge der Statorwicklungen. Bei der Anwendung des Vollschrittbetriebs mit Einphasensteuerung kann die Erregung der jeweils im gewünschten Drehsinn nachfolgenden Statorpole umgepolt werden, um den Rotor in die entgegengesetzte Richtung zu bewegen (Erregungsschema: AB – DC – BA – CD). Alternativ dazu kann bei Zweiphasensteuerung jeder der benachbarten Statorpole erregt werden, um ein Drehmoment zu erzeugen, das den Rotor in der Mitte zwischen beiden hält. In diesem Fall muss die Erregungsreihenfolge entsprechend umgekehrt werden (Erregungsschema: AB/CD – AB/CD – BA/DC – BA/CD).

Im Halbschrittbetrieb, der in Abbildung 2.4 dargestellt ist, wird eine Kombination aus den beiden zuvor beschriebenen Betriebsweisen verwendet, wobei sie sich zyklisch abwechseln. Dadurch kann der Schrittwinkel halbiert werden, was zu einer höheren Genauigkeit führt. Das Erregungsschema

sieht folgendermaßen aus: AB – AB/CD – CD – BA/CD – BA – BA/DC – DC – AB/DC. Dabei wird zuerst nur ein Elektromagnet mit den Polen A und B erregt, danach wird die Erregung auf die benachbarten Pole C und D erweitert (AB/CD). Der Rotor wird nun um einen halben Schritt weiterbewegt. Anschließend wird nur der Elektromagnet mit den Polen C und D erregt (CD), gefolgt von der Erweiterung auf die benachbarten Pole A und B (BA/CD), um den Rotor um einen weiteren halben Schritt zu bewegen. Dieser Zyklus wird fortgesetzt, bis die volle Umdrehung erreicht ist.

Unipolare Schrittmotoren sind mit vier Spulen ausgestattet, die in nur eine Richtung vom Strom durchflossen werden. Jeder der beiden Elektromagnete besteht aus zwei Phasen, deren Wicklungen jeweils entgegengesetzten Wickelsinn haben (vgl. Abb. 2.2 a). Um den Elektromagneten zu erregen, wird entweder die eine oder die andere Phase aktiviert, was zur Polung N – S oder S – N führt. Die Ansteuerung gestaltet sich vergleichsweise einfach, da der Stromfluss lediglich ein- und auszuschalten ist und nicht umgepolt werden muss. Aus diesem Grund können einfache Leistungsstufen, wie zum Beispiel vier einzelne Darlington-Transistoren oder ein Vierfach-Darlington-Array, verwendet werden. Der Nachteil dieser Betriebsart liegt in einem geringeren Drehmoment, da jeweils nur der halbe Wickelraum – also die Hälfte der insgesamt untergebrachten Windungen – zur Erregung genutzt werden kann. Das wiederum führt dazu, dass die Wicklungen mit dünnerem Draht ausgeführt werden müssen, was eine entsprechende Beschränkung der maximalen Stromstärke zur Folge hat.

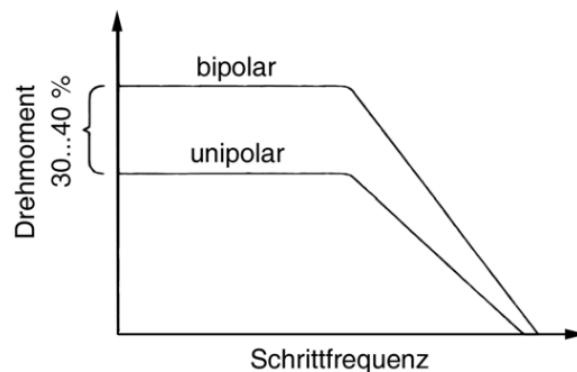


Abbildung 2.5: Das Drehmoment in Abhängigkeit von Bauart und Schrittfrequenz [10]

Bipolare Schrittmotoren sind aufgrund ihrer technischen Eigenschaften eine attraktive Option für viele Anwendungen. Im Gegensatz zu unipolaren Schrittmotoren haben bipolare Schrittmotoren nur zwei Phasen, was bedeutet, dass die Umkehrung des Magnetfelds durch das Umpolen der Stromrichtung erreicht wird. Diese Umpolung erfordert jedoch den Einsatz von zwei Brückenschaltungen mit Leistungsbau-elementen. Der größte Vorteil von bipolaren Schrittmotoren liegt darin, dass der verfügbare Wickelraum voll ausgenutzt werden kann, was den Einsatz von hinreichend dickem Draht ermöglicht und durch entsprechende Ströme ein hohes Drehmoment erreicht werden kann. In der Tat hat ein gleich großer bipolarer Motor im Vergleich zu einem unipolaren Motor ein um 30 bis 40 Prozent höheres Drehmoment (vgl. Abb. 2.5). Der Einsatz von integrierten Leistungsschaltungen macht den erforderlichen Aufwand jedoch erträglich. Aus diesen Gründen bevorzugen die meisten Anwendungen bipolare Schrittmotoren.

In der Regel umfasst eine Schrittmotorsteuerung drei Funktionseinheiten, nämlich den Mikrocontroller, die Steuerschaltung und die Treiberstufen. In einfachen Konfigurationen ist der Mikrocontroller direkt mit den Treiberstufen (siehe auch Abb. 2.6) verbunden, wobei alle Steuerfunktionen programm-

seitig realisiert werden. Es gibt eine Vielzahl von integrierten Schrittmotorsteuerschaltkreisen mit unterschiedlichen Typenreihen, die sich hinsichtlich der Interfaces, der Steuerfunktionen und der Schaltungsintegration unterscheiden.

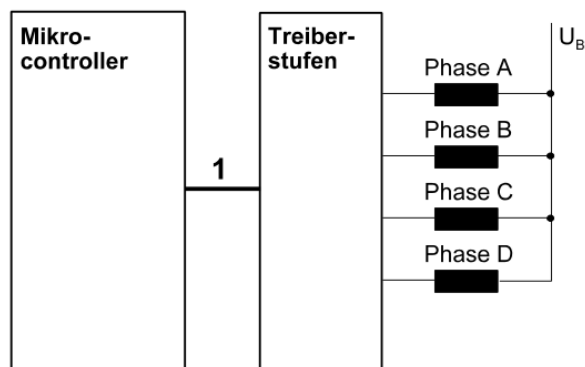


Abbildung 2.6: Schrittmotorsteuerung mit der Direktsteuerung durch Mikrocontroller [10]

Im Hinblick auf die Interfaces unterscheiden sich die Typenreihen der integrierten Schrittmotorsteuerschaltkreise in mehreren Aspekten. Es gibt Varianten, die Einzelanschlüsse zur Steuerung der Betriebsart, zur Schrittauslösung und anderen Funktionen bereitstellen. Auch einfache Busschnittstellen wie ein 8-Bit-Bus sind verfügbar. Darüber hinaus sind typische Mikrocontroller-Schnittstellen, wie zum Beispiel Serial Peripheral Interface (SPI) oder die Serial-Interface-Standard RS-232, weit verbreitet.

Von Steuerfunktionen sind insbesondere zwei Varianten von dem Interesse. Zum einen ist es möglich, einzelne Schritte manuell von außen auszulösen und die Betriebsart von außen einzustellen. Zum anderen können auch mehr oder weniger komplexe Bewegungsabläufe realisiert werden, die durch sogenannte Motion Control Functions realisiert werden. Diese erlauben es, vordefinierte oder gar selbst konfigurierte Bewegungsmuster abzurufen, die in Abhängigkeit von verschiedenen Parametern, wie beispielsweise Geschwindigkeit oder Beschleunigung, ablaufen können.

In Bezug auf die Schaltungsintegration bieten sich bei der Schrittmotorsteuerung zwei grundsätzliche Möglichkeiten: getrennte Schaltkreise für Steuerung und Treiberstufen oder eine gemeinsame Integration beider Schaltungen in einem Schaltkreis. Während die erstgenannte Variante eine höhere Flexibilität hinsichtlich der Anpassung an individuelle Anforderungen erlaubt, punktet die zweite Variante durch eine reduzierte Anzahl von Komponenten, was wiederum Vorteile in Bezug auf Kosten, Platzbedarf und Systemkomplexität mit sich bringt.

Die Erregung von Schrittmotoren mit Dauermagnetrotoren ist mittels acht aufeinanderfolgender Zustände zyklisch darstellbar (vgl. Tabelle 2.2). Die drei rechten Spalten geben Aufschluss darüber, welche der Zustände in der entsprechenden Betriebsart verwendet („ja“) oder ausgelassen („nein“) werden. Die Ansteuerung von unipolaren und bipolaren Schrittmotoren erfolgt grundsätzlich auf ähnliche Weise. Es ist entscheidend, abwechselnd die Elektromagnete AB und CD umzupolen bzw. (Einphasensteuerung, Halbschrittbetrieb) zeitweise auszuschalten.

Die in Tabelle 2.2 dargestellte Codierung weist den folgenden Schlüssel auf:

- „0“ bedeutet ausgeschaltet
- „1“ bedeutet eingeschaltet

Tabelle 2.2: Die Erregungszustände der Schrittmotoren mit Dauermagnetrotor [10]

Zustand	unipolar				bipolar		Halbschritt	Einphasensteuerung	Zweiphasensteuerung
	A	B	C	D	AB	CD			
1	0	1	0	1	-	-	ja	nein	ja
2	0	0	0	1	0	-	ja	ja	nein
3	1	0	0	1	+	-	ja	nein	ja
4	1	0	0	0	+	0	ja	ja	nein
5	1	0	1	0	+	+	ja	nein	ja
6	0	0	1	0	0	+	ja	ja	nein
7	0	1	1	0	-	+	ja	nein	ja
8	0	1	0	0	-	0	ja	ja	nein

- „+“ bedeutet Polung AB oder CD
- „-“ bedeutet Polung BA oder DC

Die korrekte Ausführung der Zustandsübergänge ist für die Steuerung von wesentlicher Bedeutung. Die Art und Weise, wie die Elektromagnete im nächsten Schritt erregt werden sollen, hängt von der aktuellen Erregung, der gewählten Drehrichtung und der Betriebsart ab. Dabei muss darauf geachtet werden, dass die Schritte nicht zu schnell aufeinander folgen. Folglich müssen zwei Herausforderungen bewältigt werden: die Durchführung der Zustandsübergänge und die Einhaltung der Zeitvorgabe.

Es existieren verschiedene Ansätze für die Implementierung von Schrittmotorsteuerungen, wobei sowohl reine Software- als auch reine Hardwarelösungen zum Einsatz kommen. Bei reinen Softwarelösungen werden beispielsweise die Zustandsübergänge durch zyklisches Adressieren einer Wertetabelle oder durch das zyklische Rotieren von Registerinhalten erzeugt, während die Zeit zwischen den Schritten mittels einer Programmschleife abgezählt oder über eine Zähler-Zeitgeber-Einheit dargestellt werden kann. Letzteres ist dabei die Vorzugslösung, da es dem Prozessor Zeit gibt, weitere Aufgaben auszuführen, und die Schrittmotorsteuerung als Unterbrechungsbehandlung implementiert werden kann (wobei die Zeitgeberunterbrechung das Ausgeben des jeweils nächsten Schrittes veranlasst). Reine Hardwarelösungen hingegen führen das Weiterschalten um eine vorgegebene Anzahl von Schritten autonom aus, wozu ein Zustandsautomat, ein Zeitzähler und ein Schrittzähler erforderlich sind.

Verbundlösungen integrieren schaltungstechnische Realisierungen des Zustandsautomaten mit programmseitigen Lösungen für die Schrittauslösung, Zeitvorgabe, Schrittzählung und anderen Aspekten. Hierbei wird der Zustandsautomat auf Hardwareebene realisiert, während die softwarebasierten Funktionen programmtechnisch implementiert werden. Dies ermöglicht eine effiziente Steuerung des Schrittmotors durch die Kombination von hardware- und softwarebasierten Ansätzen.

2.4 Funktionsweise von Kugelschaltern und Mikrovibrationssensoren

Kugelsensoren stellen ein wichtiges Hilfsmittel zur Bewegungserkennung dar und sind durch ihre Struktur gekennzeichnet, welche aus mindestens drei Kontaktflächen besteht, wobei zwei dieser Flächen fixiert und die dritte beweglich ist. Innerhalb des Sensors befindet sich eine Kugel, welche bei Bewegung die beiden fixierten Kontaktflächen elektrisch miteinander verbindet und somit den Widerstand zwischen den beiden äußeren Anschlussflächen verändert. Anhand der Änderung des

Widerstands lässt sich der Bewegungszustand des Sensors ermitteln. Demzufolge kann eine Veränderung des Widerstands als ein Indikator für eine Bewegung des Sensors interpretiert werden. Die Abbildung 2.7 veranschaulicht die Funktionsweise eines Kugelsensors mittels zwei Skizzen. Die

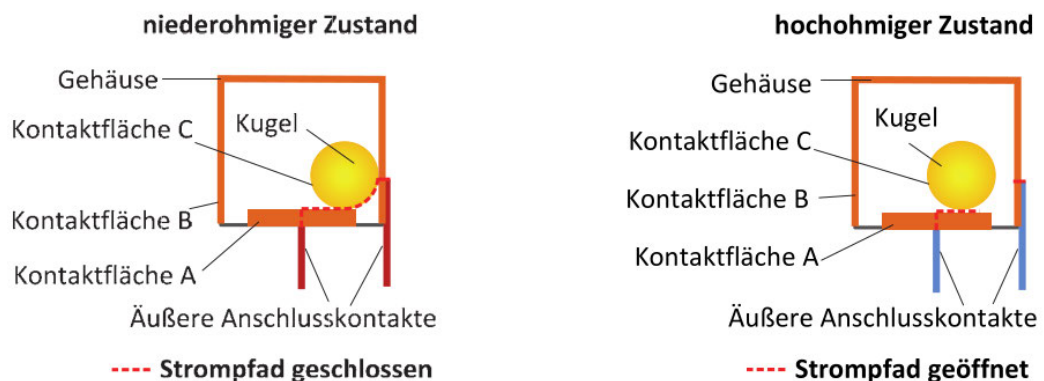


Abbildung 2.7: Funktionsweise eines Kugelsensors [3]

linke Skizze verdeutlicht den Sensor im geschlossenen, niederohmigen Zustand, wobei die elektrisch leitfähige Kugel die beiden Kontaktflächen A und B überbrückt und somit den Widerstand zwischen ihnen auf unter 100Ω reduziert. Die rechtsseitige Skizze der Abbildung illustriert den Kugelsensor im offenen, hochohmigen Zustand. Hierbei findet keine Überbrückung der Kontaktflächen A und B statt, wodurch der Widerstand zwischen den Kontaktflächen hoch bleibt und weit über $50M\Omega$ beträgt. Die Bestimmung der Sensitivität eines Kugelsensors ist durch verschiedene Parameter wie die Masse der Kugel und die geometrischen Eigenschaften der Kontaktflächen bedingt. Eine geringe Kugelmasse führt dazu, dass die erforderliche externe Kraft zum Bewegen der Kugel verringert wird. Die Gewichtskraft einer Stahlkugel hängt kubisch von deren Durchmesser ab, was eine weitere Herausforderung bei der Optimierung der Sensitivität darstellt[3].

Im Bereich der Sensorik sind viele Bauteile von beträchtlicher Größe und lediglich als bedrahtete Komponenten erhältlich, was eine automatische Verarbeitung als SMD-Komponente unmöglich macht. Systemintegratoren sind auf Spezialmaschinen angewiesen, um diese Bauteile wirtschaftlich verarbeiten zu können. Infolgedessen wurde ein sehr kleiner Kugelsensor entwickelt, der vollautomatisch mit den gängigen Pick-and-Place-Automaten der Elektronikbranche bestückt werden kann. Der neue Mikrovibrationssensor zeichnet sich durch seine Bauart aus, die im Vergleich zu ähnlichen Produkten wesentlich kleiner ist. Zudem ermöglicht seine elektronische Testung eine parallele Durchführung. Der Mikrovibrationssensor wird zur Erkennung geringfügiger Bewegungen und Vibrationen mithilfe einer mobilen Mikrokugel eingesetzt. Die Kugel überbrückt zwei Kontakte und reduziert den Widerstand zwischen den beiden externen Anschlusspads von mehreren Megaohm ($>30 M\Omega$) auf unter 100Ω . Der Sensor ist vollständig passiv, erfordert keine Signalkonditionierung und arbeitet mit Strömen von nur $0,2\mu A$. Die Mikrovibrationssensoren können durch die Nutzung von werkzeugspezifischen Auswerteelektroniken die Steuerung von bewegungsempfindlichen Geräten übernehmen. Diese Sensoren können dabei helfen, viele Systeme umweltfreundlicher zu gestalten, indem sie die Wake-up- und Power-Down-Logik implementieren, um den Batterieverbrauch zu minimieren und den Energieverbrauch auf ein Minimum zu bringen. So wird die Verfügbarkeit von grüner Technologie und grüner Elektronik in neuen Bereichen von Design und Anwendung vorangetrieben. Typischerweise werden diese Sensoren in Anwendungen wie Fahrradcomputern, Fernbedienungen, elektronischen Schlosssystemen, RFID-Transpondern, GPS-Tracking-Systemen, drahtlosen Sensornetzwerken, beleuchteten Hunde-Halsbändern, Zugangskontrollsystemen und Datenloggern eingesetzt, die nur bei

Bewegung eingeschaltet werden.

Eigenschaften des Mikrovibrationssensors MVS0409.02:[11]

- Der vorgestellte Mikro-Vibrationssensor wird für die Detektion von leichten Bewegungen und Vibrationen verwendet. Es ist ein omnidirektionaler Sensor, halogenfrei und arbeitet mit einer breiten Versorgungsspannung von 1,80 V bis 15 V.
- Der Sensor ist vollständig passiv und erfordert keine Signalverarbeitung. Er arbeitet mit Strömen von nur 0,2 μA .
- Der Sensor hat eine automatisierte SMD-Montage, ist RoHS-konform und bleifrei.
- Der Sensor ist spezifiziert für eine Betriebstemperatur von -40 °C bis +85 °C und hat eine Größe von 2,85 mm x 2,45 mm x 1,7 mm.
- Der Reaktionspunkt des Sensors liegt bei ca. 50 mg.
- Reflow-Lötprozess 260 °C, 10 s.

Innerhalb des Sensors befindet sich eine mobile, vergoldete Mikrokugel im Hohlraum. Diese Mikrokugel überbrückt beim Bewegen zwei vergoldete Kontakte, indem sie von einem hohen auf einen niedrigen Widerstandszustand umschaltet. Wenn der Sensor in Ruhe ist, ist er nicht unbedingt geschlossen. Nur in 70% - 99% der Zeit ist der Sensor geschlossen, wenn er sich im Ruhezustand befindet.[11] Die Abbildung 2.8 zeigt die typischen Eigenschaften des Sensors während der Erregung und im Ruhezustand.

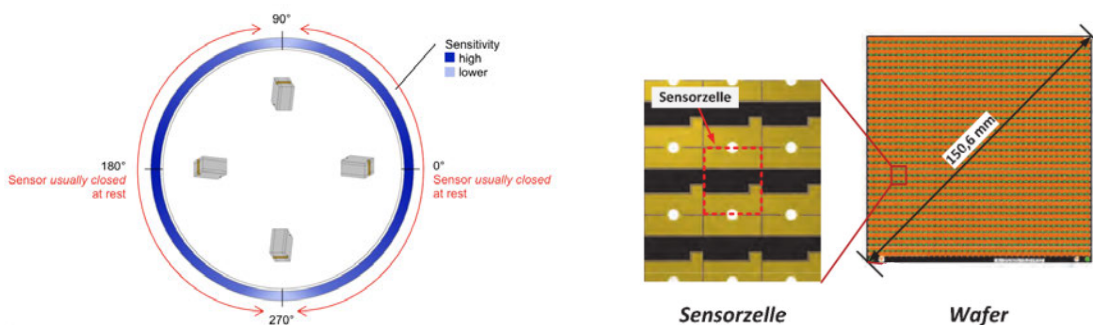


Abbildung 2.8: Merkmale des Sensors MVS0409 [11] **Abbildung 2.9:** Anordnung der Sensoren im Wafer [3]

Durch die Verwendung von Mikrovibrationssensoren in Plattenbauweise kann eine regelmäßige Matrix aus Einzelsensoren, auch Wafer genannt, vervielfältigt werden. Zusätzlich ist es durch diese Bauweise möglich, die Messdaten jedes einzelnen Sensors den entsprechenden Spalten und Reihen des Wafers zuzuordnen, um eine optimale Datenstruktur zu schaffen und die Datenanalyse zu vereinfachen (vgl. Abb. 2.9). In der Produktion enthält ein Wafer 1360 MVS.

3 Untersuchung des Verhaltens der Mikrovibrationssensoren

3.1 Ziel der Untersuchung

Die spezielle Kundenapplikation verzeichnet einen Fehler in Bezug auf die MVS, welche im Abschnitt 2.4 beschrieben wurden. Die besondere Natur der Applikation erfordert eine Analyse der bisher wenig untersuchten spezifischen Eigenschaften der MVS. Die Abbildung 3.1 stellt diese Applikation dar.

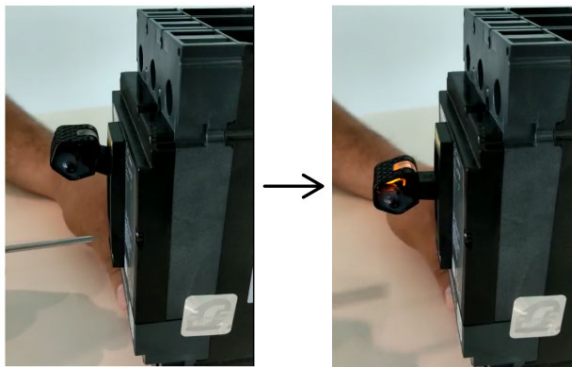


Abbildung 3.1: Kundenapplikation

Kurze Beschreibung der Applikation:

Die Platzierung der MVS auf dem Motorschutzschalter hatte das Ziel, den Zustand des Schalters (eingeschaltet oder ausgeschaltet) zu erfassen. Bei der Bewegung des Schalters sollte eine entsprechende Signalisierung durch die MVS erfolgen. Es kam jedoch vor, dass die Signalisierung entweder zu spät oder gar nicht stattfand. Eine Evaluierung ergab, dass der Fehler nur bei den MVS auftritt, welche im Ruhezustand hochohmig sind.

Das Ziel der Untersuchung ist, durch die Korrelation des Ruhezustands der MVS mit den bekannten Messgrößen **Anzahl der Impulse pro Messung** (n_p) und **Hightime** (t_h), zu beurteilen, ob eine Klassifizierung der Sensoren anhand der bekannten Beurteilungsmatrix[3] für den spezifischen Kundenanwendungsfall möglich ist.

3.2 Bekannte Messdaten

Das Testsystem EIFuTe wurde entwickelt, um die mechanische Anregung und elektrische Prüfung von Sensoren in beliebigen Raumlagen zu ermöglichen.

Das System besteht aus einem Steuer-PC (2), der die Prüfsequenz steuert, die Testdaten verarbeitet und in einer Datenbank archiviert. Nachdem der Wafer auf die Wafer-Auflageplatte (1) platziert wurde und die Testsequenz gestartet wurde, bewegt sich der Pneumatikzylinder (4) in seine Endposition. Anschließend kommt der Nadeladapter der Testelektronik (6) mit den Sensoren auf dem Wafer in Kontakt. Der Schwenkmotor (5) neigt die Waferoberfläche in einen vordefinierten Winkel zur Erdoberfläche, während der Rotationsmotor (7) die Sensoren auf dem Wafer in eine Drehbewegung versetzt. Sobald die Soll-Rotationsgeschwindigkeit erreicht ist, erfolgt die Charakterisierung der elektrischen Eigenschaften der Sensoren.[3] Die Charakterisierung der omnidirektionalen und lageempfindlichen Sensoren erfolgt in einem zweistufigen Testverfahren.

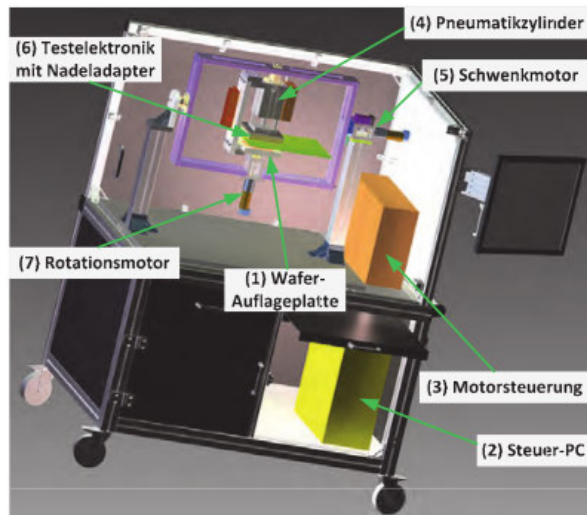


Abbildung 3.2: Messanlage EIFuTe [3]

In der ersten Phase sind die äußeren Anschlusspads des Sensors mit der Erde verbunden, um die elektrischen Merkmale zwischen dem Boden, der Kammer und der Kugel zu ermitteln. Nachdem der Wafer eingelegt wurde (siehe Abb. 3.4), erfolgt eine Drehung um 165° in die Prüfposition mittels des Schwenkmotors. Sobald die Prüfung der Bodenseite abgeschlossen ist, erfolgt eine weitere Schwenkbewegung des Wafers um 180° . Im zweiten Testschritt werden die Eigenschaften zwischen dem Deckel, der Kammer und der Kugel ermittelt. Durch dieses zweistufige Verfahren wird eine umfassende Charakterisierung der Sensoren ermöglicht, indem die elektrischen Merkmale in verschiedenen Konfigurationen zwischen Boden, Kammer, Deckel und Kugel untersucht werden.

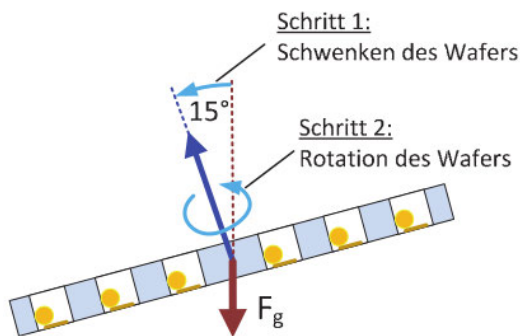


Abbildung 3.3: Erläuterung des Testkonzepts - Ausrichtung des Wafers im Raum

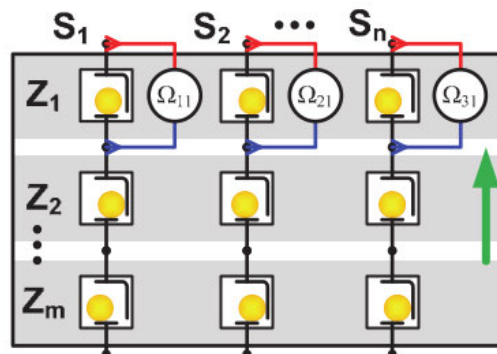


Abbildung 3.4: ESB der Sensoren des Wafers

Die Messung der Sensoren erfolgt parallel unter Verwendung mehrerer Messspitzen über einen definierten Zeitraum pro Sektor. Ein Sektor bezeichnet dabei die Position der Kugel in Bezug auf die Kammer. Insgesamt werden 8 Sektoren betrachtet, wobei 4 Sektoren zur Prüfung der Kontaktierung zwischen Deckel und Kammer dienen und 4 Sektoren zur Prüfung der Kontaktierung zwischen Boden und Kammer. Das Ergebnis wird in Form einer Tabelle im .csv-Format festgehalten, die die berechneten Größen „ n_p “ und „ t_h “ je nach Sektor enthält. Die „ n_p “ bezeichnet die Anzahl der Impulse eines MVS pro Messdauer in jeweiligem Sektor, die „ t_h “ die Zeit, die das MVS im leitenden Zustand in jeweiligem Sektor bleibt.

Für die Klassifizierung von Sensoren kann eine Matrix (siehe Abb. 3.5) verwendet werden[3], welche die Anzahl der Impulse pro Messung und das prozentuales Verhältnis von Hightime zur Gesamtzeit der Messung beinhaltet.[3]

<i>OnRatio</i> Pulse	niedrig	mittel	hoch
niedrig	✗	✗	✗
mittel	✗	✓	✓
hoch	✓	✓	✓

✓ Sensor gut ✗ Sensor schlecht

Abbildung 3.5: Beurteilungsmatrix [3]

Das erste Gütekriterium – *OnRatio* wird folgendermaßen bestimmt:

Das Sensorsignal $u(t)$ wird mit der Auswerteschaltung auf der Abb. 3.6 gemessen und digitalisiert.

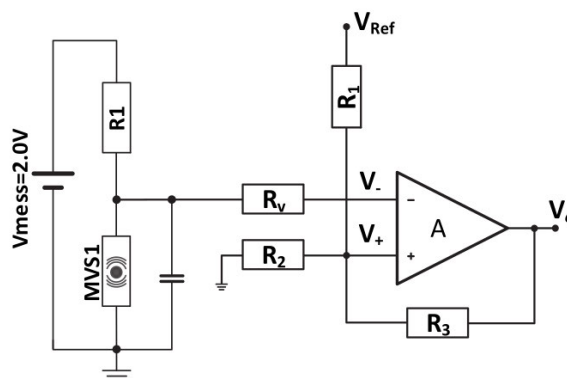


Abbildung 3.6: Optimiertes elektronisches Auswertekonzept für Testsysteme [3]

Die Dauer, in der der Schalter geschlossen ist, kann durch das Produkt der Abtastgeschwindigkeit τ_{Tast} und der Anzahl der Abtastwerte unterhalb von R_L berechnet werden. Dieses Produkt ergibt den Zeitraum t_{RL} .

OnRatio für einen Sensor im Anregungszeitraum – $t_{Anregung}$ kann aus diesem Zusammenhang abgeleitet werden (siehe Gl. 3.1).[3]

$$OnRatio = \frac{t_{RL}}{t_{Anregung}} \tag{3.1}$$

Das zweite Gütekriterium ergibt sich aus der Anzahl der erfassten Widerstandsänderungen des Sensors innerhalb des Messintervalls. Wenn sich der Widerstandswert von R_L über die Schaltschwelle R_H oder umgekehrt ändert, wird ein Impulzzähler für den Sensor im FPGA erhöht. Wenn man den digitalen Zustand am Ausgang des Schmitt-Triggers zum Abtastzeitpunkt i mit $V_i \in (0,1)$ bezeichnet, ergibt sich die Anzahl der Pulse wie folgt:[3]

$$n_{Pulse} = \sum_{i=1}^M (V_i - V_{i-1})^2 \tag{3.2}$$

Im Bild 3.7 sind Histogramme dargestellt, die die mittlere Impulszahl und den On_{Ratio} von 20.000 Entwicklungssensoren darstellen. Der Mittelwert der Impulsanzahl beträgt 74, mit einer Standardabweichung von 14 Pulsen. Die Sensoren sind im Mittel zu 80 % der Zeit geschlossen.[3]

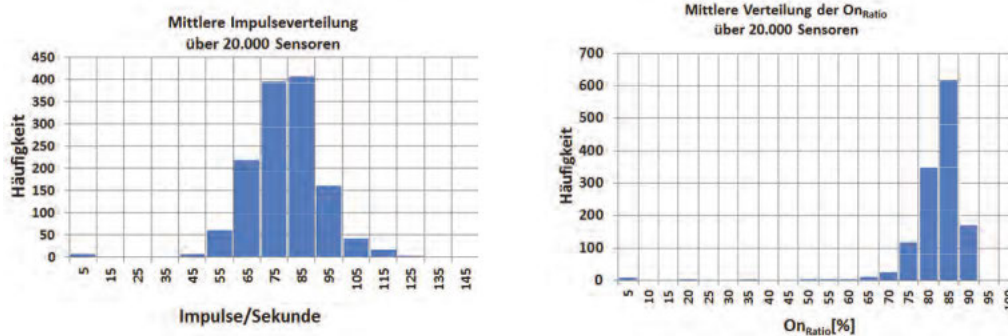


Abbildung 3.7: Gemittelte Pulszahl und On_{Ratio} über 20.000 Entwicklungssensoren

3.3 Versuchsaufbau

Die Prüflinge (MVS) wurde auf dem FI-Schutzschalter platziert, um die Kundenapplikation zu simulieren (siehe Abb. 3.8). Durch die Erzeugung des Fehlerstromes wird die Versorgungsspannung unterbrochen, was durch den schnellen Wechsel des Schalters zur Generierung eines mechanischen Impulses führt. Dieser Übergang von „High zu Low“ wird durch die grafische Darstellung auf einem Oszilloskop veranschaulicht. Der Vibrationssensor ist in der Messschaltung serialisiert mit einem $2k\Omega$ Widerstand verbunden. Die über diesen Widerstand fließende Spannung wird am Oszilloskop unter Verwendung des Triggerungsvorgangs auf der Versorgungsspannung visualisiert.

Komponente:

- FI-Schutzschalter
- Oszilloskop Tektronix
- Messschaltung auf der Steckplatine
- MVS

Die Abb. 3.9 zeigt ein Ersatzschaltbild von der verwendeten Messschaltung. Der MVS und der FI-Schutzschalter werden als elektrische Schalter dargestellt.

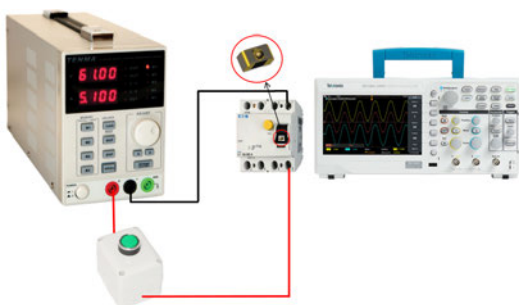


Abbildung 3.8: Versuchsaufbau

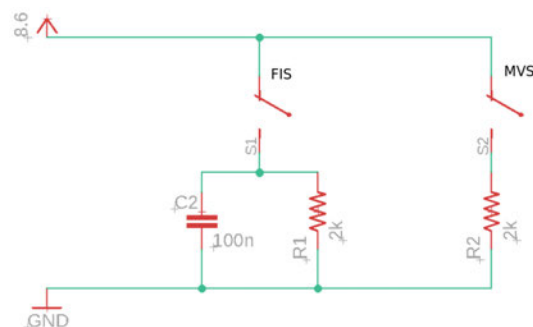


Abbildung 3.9: ESB Messschaltung

Die elektrischen Potenziale zwischen Spannungsquelle, FI-Schutzschalter und MVS sind parallel gekoppelt, was zu einer Reduktion des Bedarfs an einer zweiten Spannungsquelle führt. Direkte Messungen der Spannungen an den Schaltern sind wenig informativ, da sie ein invertiertes Oszillogramm zeigen würden. Daher werden die Spannungen an seriell geschalteten Widerständen gemessen, was auch den Bedarf an einem zusätzlichen Vorwiderstand reduziert. Ein parallel zu dem Widerstand des Zweigs des FI-Schutzschalters gekoppelter Kondensator verlängert die Abfallzeit des vom FI-Schutzschalter erzeugten Sprunges und glättet den Übergang von „High“ zu „Low“, wodurch das Triggerungsfehler am Oszilloskop vermieden werden kann. Diese Maßnahme stellt sicher, dass der Übergang kontinuierlich und langsam genug ist, um bei der ausgewählten Auflösung des Oszilloskops eine Triggerung zu ermöglichen.

3.4 Datenaufnahme

Zunächst ist es erforderlich, die Impulsantwort eines minderwertigen Sensors oder eines qualitativ hochwertigen Sensors zu bestimmen. Folgende Oszillogramme (siehe Abb.3.10) wurden nach der Messung an der Versuchsanlage (siehe Abschnitt 3.3) erstellt.

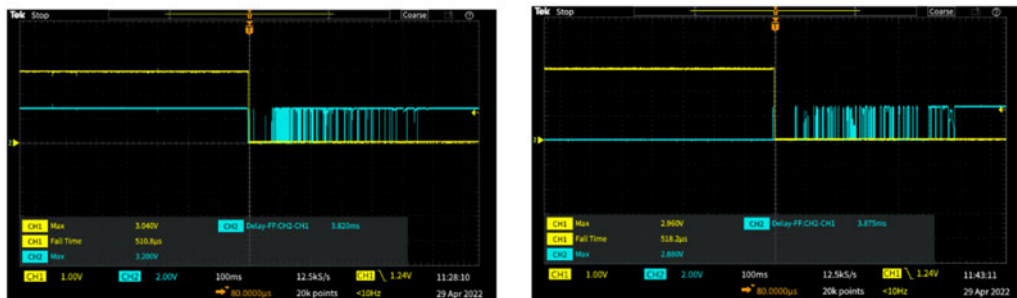


Abbildung 3.10: Ansprechverhalten eines hochwertigen Sensors (links) und eines minderwertigen Sensors (rechts)

Das Ansprechverhalten des Sensors links wird als hochwertig von der Kunden bestimmt und ist bei mindestens 90% von Sensoren gewünscht. Die genauere Beschreibung eines hochwertigen Sensors wäre:

- der MVS in seinem Ruhezustand ist niederohmig
- der MVS weist eine gute Sensitivität auf (angemessene Anzahl von Pulsen pro Anregungszeit)

Basierend auf den vorherigen Erläuterungen ergibt sich die folgende mathematische Beschreibung für ein qualitativ hochwertiges Signal des Mikrovibrationssensors (Gl. 3.3 und 3.4: Sei $U(t)$ das Signal des qualitativ hochwertigen Mikrovibrationssensors, wobei t_a die Zeit des Anfangs der Anregung und t_e das Ende der Anregung darstellt.

$$U(t) \geq U_{HIGH} \quad \text{bei} \quad 0 \leq t < t_a \quad \text{und} \quad t_e < t \leq \infty \quad (3.3)$$

$$U_{min} \leq U(t) \leq U_{max} \quad \text{bei} \quad t_a \leq t \leq t_e \quad (3.4)$$

Die Bezeichnung U_{HIGH} wird verwendet, um den leitenden Zustand des Signals zu repräsentieren. Dieser Zustand zeigt an, dass das Signal einen hohen Spannungswert aufweist, der auf eine bestimmte Aktivität oder einen bestimmten Zustand des Systems hinweist. Des Weiteren werden U_{min}

und U_{max} als Bezeichnungen für die Extremwerte des Signals verwendet. U_{min} stellt den minimalen Spannungswert dar, den das Signal annehmen kann, während U_{max} den maximalen Spannungswert repräsentiert, den das Signal erreichen kann.

Die MVS des ausgewählten Wafers wurden während der Messung an der Messanlage EIfuTe durch Vibration angeregt und gemessen. Jeder Prüfling (MVS) auf dem Wafer ist eindeutig nummeriert und verfügt über die entsprechenden Messdaten. Um den Zusammenhang zwischen der Qualität der Prüflinge und den bereits gesammelten Messdaten (n_p und t_h) zu untersuchen, wurde eine Wiederholmessung mithilfe des in Abbildung 3.8 dargestellten Versuchsaufbaus durchgeführt. Dabei wurden die MVS aus dem Wafer jeweils 50 Mal gemessen, und die resultierenden Oszillogramme wurden manuell ausgewertet. Die erzielten Ergebnisse wurden zusammen mit den bekannten Messdaten (n_p und t_h) in einer Tabelle festgehalten. Idealerweise sollten alle 1360 Sensoren pro Wafer gemessen und ausgewertet werden, um aussagekräftige statistische Ergebnisse zu erzielen. Aufgrund des damit verbundenen hohen Zeitaufwands ist dies jedoch nicht praktikabel. Daher wurden die Prüflinge in kleinere Gruppen unterteilt, indem eine spezifische Filterung angewendet wurde. Pro Gruppe wurden jeweils 5 Sensoren ausgewählt und gemessen.

Tabelle 3.1: Gruppen und Filtern

Gruppe	Filter
C	$5500ms \leq \bar{t}_h \leq 5520ms$
D	$5600ms \leq \bar{t}_h \leq 5620ms$
E	$5800ms \leq \bar{t}_h \leq 5820ms$
F	$250 \frac{n}{s} \leq \bar{n}_p \leq 400 \frac{n}{s}$
G	$400 \frac{n}{s} \leq \bar{n}_p \leq \frac{n}{s}$

Die Tabelle 3.1 zeigt die Gruppen mit ihren entsprechenden Auswahlkriterien aus dem ersten Wafer, dabei wurden die Gruppen „C, D, E“ nach den Mittelwerten t_h von allen Sektoren und die Gruppen „F, G“ nach den Mittelwerten n_p von allen Sektoren sortiert. Im Folgenden wird eine Übersicht über die verwendeten Daten gegeben. Die Gruppe „C“ ist in der Tabelle 3.2 dargestellt und repräsentiert damit das Vorgehen der Strukturierung aller weiteren Gruppen.

Tabelle 3.2: Datenrepräsentation Gruppen C, D, E

group	bod_hightime_s1	bod_hightime_s2	bod_hightime_s3	bod_hightime_s4	...	n_o
C1	5872	4667	5872	5163	...	12
C2	4631	5768	5112	4846	...	4
C3	4281	5563	5490	5863	...	15
C4	5320	6155	6561	4680	...	5
C5	6164	4667	6181	4964	...	6
...

3.5 Ergebnisse

Folgende Nebeneigenschaften von MVS wurden während Untersuchung ermittelt:

- Die Einschwingungsdauer des MVS bis es in seinen Ruhezustand zurückkehrt, beträgt maximal 400ms.

- Der Bereich der kürzesten Impulse beträgt $30\mu s$ bis $40\mu s$.
- Anzahl der Impulse und die Impulsbreite hängen von der Stärke der Anregung ab.

Zur effektiven Beurteilung der Prüflinge (MVS) wird eine externe Größe eingeführt, die als ξ bezeichnet wird und die Güte eines Prüflings, anhand der Randbedingungen 3.3 und 3.4 beschreibt. Die Berechnung dieser Größe erfolgt gemäß der folgenden Gleichung 3.5:

$$\xi = \left(1 - \frac{n_o}{n_M}\right) \cdot 100\% \tag{3.5}$$

Hierbei bezeichnet n_o die Anzahl der qualitativ minderwertigen Mikrovibrationssensoren, während n_m die Gesamtzahl der durchgeführten Wiederholmessungen repräsentiert.

Zunächst wurden die Messdaten (n_p und t_h) für jeden einzelnen Sektor separat in Bezug auf ξ analysiert, um die elektrische Kontaktierung zwischen den Kontaktoberflächen in verschiedenen Sektoren zu untersuchen. Die Abb. 3.11, 3.12, 3.13 und 3.14 zeigen die Visualisierung der Parameter n_p und t_h in Bezug auf Güte eines MVS in Prozent (ξ). Basierend auf den gesammelten

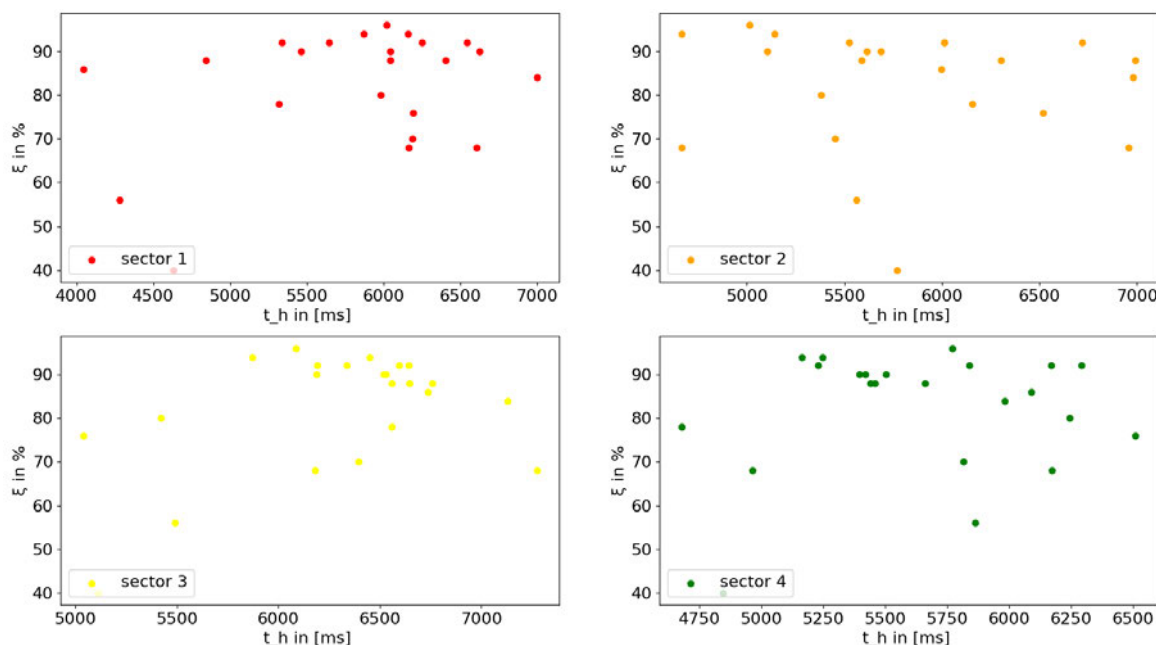


Abbildung 3.11: Sektor 1 - 4, ξ in Abhängigkeit von t_h

Daten (ξ) sowie den bereits bekannten Informationen von EIFuTe (t_h und n_p) lässt sich feststellen, dass es nur geringfügige Unterschiede zwischen den einzelnen Sektoren gibt. Darüber hinaus lässt sich ableiten, dass die Kontaktflächen eine gleichmäßig gute Leitfähigkeit aufweisen und es keinen eindeutigen Bereich gibt, in dem die Kontaktierung geschwächt ist. Zudem lässt sich erkennen, dass kein spezifischer Sektor identifiziert werden kann, der einen expliziten Einfluss auf die Größe der ξ aufweist. Die Korrelationskoeffizienten variieren der Sektoren sich zwischen -0,38 und 0,44. In anderen Worten, es gibt keine deutliche Korrelation zwischen einem bestimmten Sektor und der Größe ξ . Da es keinen spezifischen Sektor gibt, der einen signifikanten Einfluss auf die Größe ξ aufweist, sind die Mittelwerte von t_h und n_p über alle Sektoren eine repräsentative Maß, die die weiterführende Analyse in diesem Kontext erleichtern können. In Abbildung 3.15 wird die Beziehung zwischen der Größe ξ und den Parametern \bar{t}_h und \bar{n}_p dargestellt. Die Größe ξ wird dabei durch eine farbige Skala veranschaulicht, während jeder Sensor als Punkt repräsentiert wird.

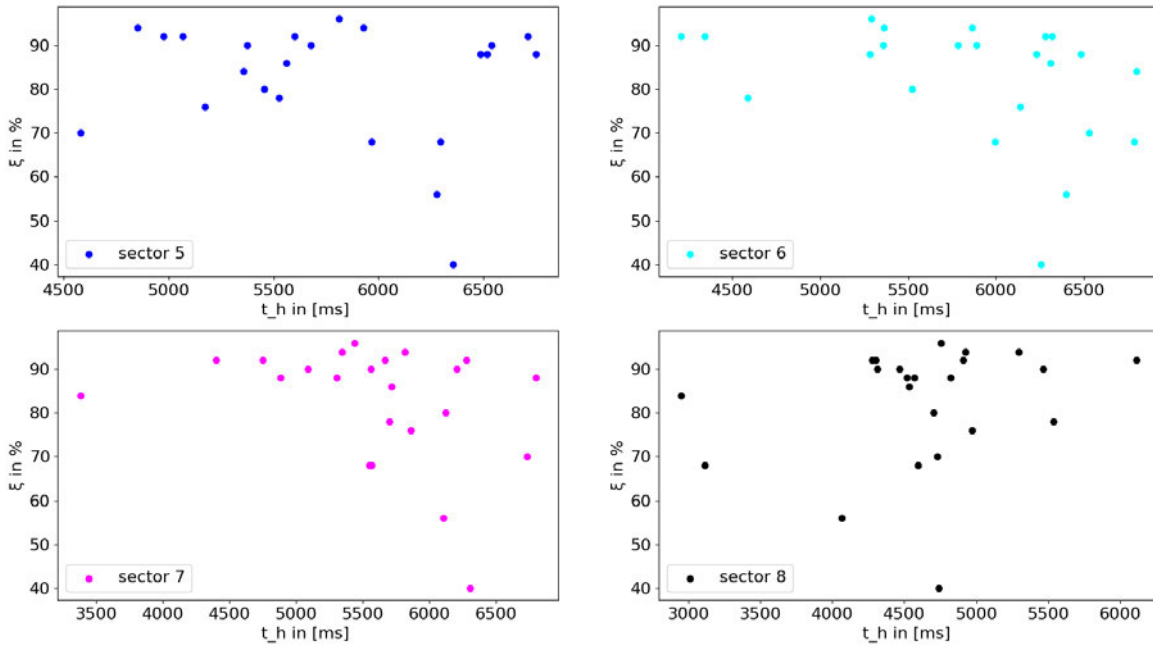


Abbildung 3.12: Sektor 5 - 8, ξ in Abhängigkeit von t_h

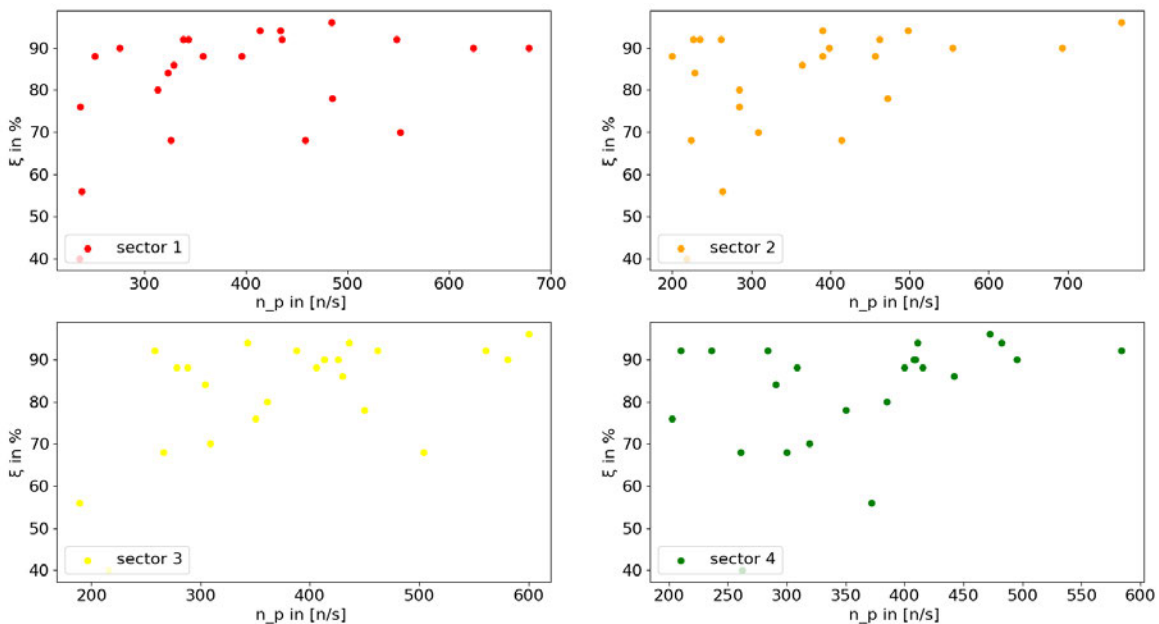


Abbildung 3.13: Sektor 1 - 4, ξ in Abhängigkeit von n_p

Die vorliegende Auswertung erlaubt weder eine Bestätigung noch eine Widerlegung der Hypothese, dass die Güte des Prüflings in Bezug auf eine spezifische Applikation 3.1 mit den Parametern t_h und n_p korreliert. In Abbildung 3.15 ist zu beobachten, dass die qualitativ hochwertigen und minderwertigen Prüflinge stark miteinander vermischt sind. Diese Beobachtung lässt keine eindeutige Schlussfolgerung auf eine signifikante Korrelation zwischen t_h , n_p und der Prüflingsgüte zu.

Die vorläufigen Ergebnisse haben gezeigt, dass die eindeutige Klassifizierung der MVS anhand der verfügbaren Messdaten derzeit nicht möglich ist.

Angesichts des Zeit- und Arbeitsaufwands für eine manuelle Messung ist es unerlässlich, den Prozess zu automatisieren. Durch die Automatisierung des Prozesses können nicht nur präzisere Signal-

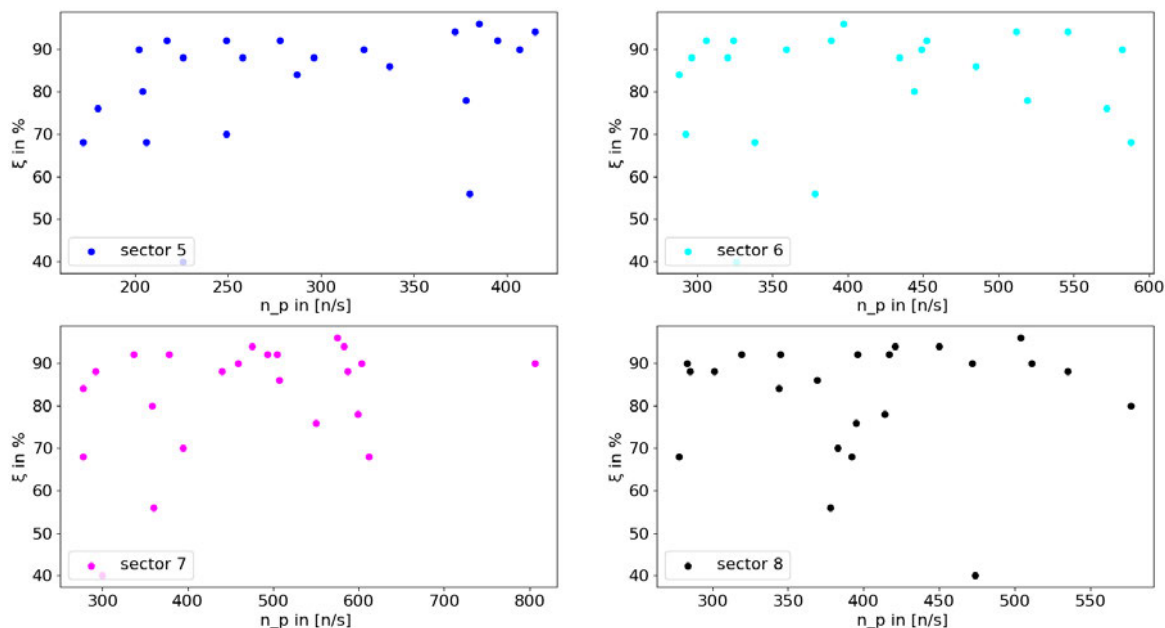


Abbildung 3.14: Sektor 5 - 8, ξ in Abhängigkeit von n_p

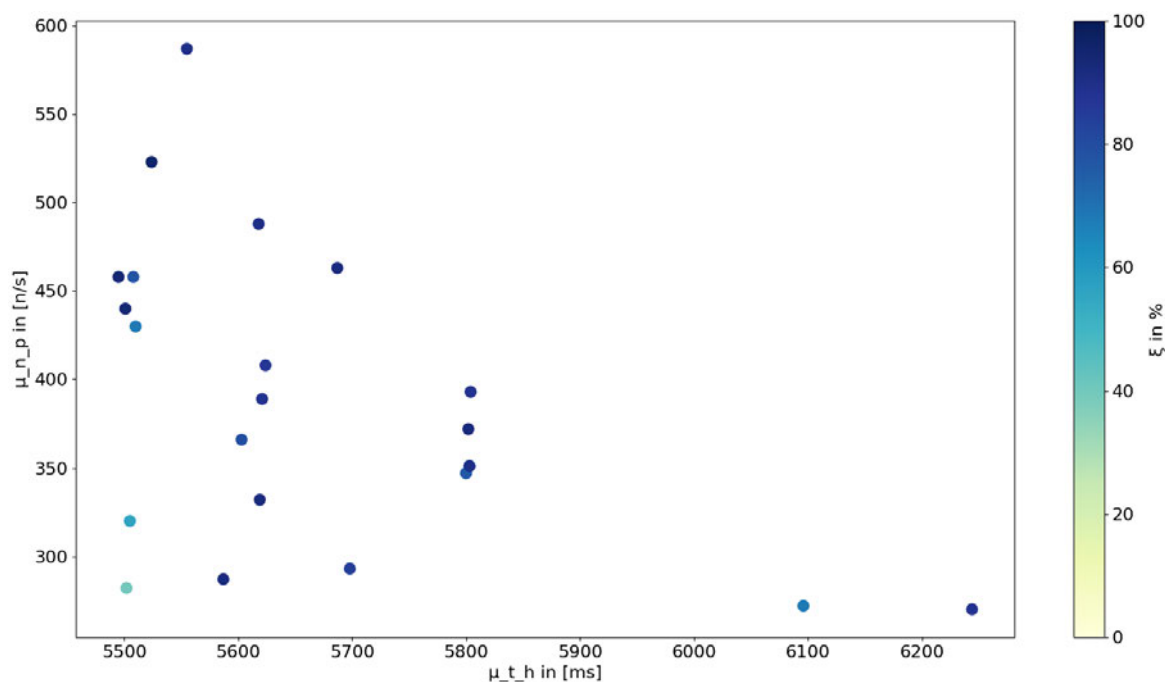


Abbildung 3.15: Beziehung zwischen ξ und den Parameter \bar{t}_h und \bar{n}_p

verarbeitungsergebnisse erzielt, sondern auch skalierbare Datenerfassungsmethoden verwendet werden, um bessere Statistiken für die MVS zu erhalten.

Nach jedem Versuch wurden Sensoren, die ungewöhnliche Ergebnisse zeigten, entnommen und untersucht. Mithilfe eines digitalen Mikroskops wurde festgestellt, dass diese Sensoren eine deformierte Kugel und beschädigte innere Leitungsrouten hatten, die während der Herstellung entstanden sein könnten.

4 Entwurf eines multifunktionalen Moduls

Die durchgeführte Untersuchung hat nicht nur zu einer detaillierten Erfassung aller relevanten Messdaten der MVS geführt, sondern ermöglichte auch, ein tieferes Verständnis des Messprozesses zu erwerben. Auf dieser Grundlage könnte ein automatisiertes System zur Messung und Auswertung der Prüflinge konzipiert werden.

Bevor das Konzept im Detail erläutert werden kann, ist es notwendig, auf einige wichtige Details der Herstellung einzugehen, um das Gesamtbild besser zu verstehen. Wie bereits in den vorherigen Abschnitten erläutert wurde, werden die MVS während der Produktion in einen Wafer integriert, was eine schnellere Qualitätskontrolle durch die Auswertungsmaschine EIFuTe 3.2 ermöglicht. Um die MVS jedoch an Kunden zu liefern, müssen sie zuerst einzeln aus dem Wafer herausgesägt und dann mithilfe einer Gurtungsmaschine[4] in spezielle Bauteilgurte verpackt werden. Diese Gurtungsmaschine besteht aus mehreren Standardmaschinen, von denen insbesondere der Wendelförderer von Bedeutung ist, da dieser direkt mit der konzipierten „Blackbox“ verbunden wird.

4.1 Konzept

Das Konzept zielt darauf ab, ein multifunktionales Modul zu entwickeln, die als einfache Zwischenstufe in die Gurtungsmaschine integriert werden kann. Dieses Modul soll dabei in der Lage sein, die vereinzelt MVS kontinuierlich zu messen und die Messergebnisse für weitere statistische Untersuchungen zu speichern. Darüber hinaus soll es in der Lage sein, die MVS dynamisch auszuwerten und zu sortieren. Dieses Konzept baut auf der Idee auf, dass eine automatisierte Sortierung von MVS während des Verpackungsprozesses dazu beitragen kann, die Effizienz und Genauigkeit der Prozesse zu verbessern. Zur besseren Veranschaulichung des Konzeptes sind in der Abbildung 4.1 exemplarisch die CAD-Modelle des konzipierten Moduls dargestellt.



Abbildung 4.1: Visualisierung des Konzeptes mittel CAD-Modellierung

Diese Darstellung soll das Verständnis des Konzeptes unterstützen und eine anschauliche Basis für weitere Erläuterungen bieten. Es sei jedoch darauf hingewiesen, dass die tatsächliche Implementierung des Moduls in der Praxis von den hier dargestellten Modellen abweichen kann und aufgrund von technischen Anforderungen oder Anpassungen angepasst werden muss. Im Folgenden werden weitere Details zum konzipierten Modul und damit verbundenen Wendelförderer erläutert.

4.2 Wendelförderer

Wendelförderer (eng. Bowl Feeder) ist eine Maschine, die zur automatischen Handhabung von Kleinteilen verwendet wird [5]. Es besteht aus einer Schüssel, die eine bestimmte Menge an Kleinteilen aufnimmt und sie dann auf einen Fördermechanismus überträgt, der die Teile zu ihrer Weiterverarbeitung transportiert. Die Schüssel selbst besteht aus einer Schüsselwand und einem Schlüsselboden, die aus Materialien wie Edelstahl oder Kunststoff hergestellt werden. Die Schüssel wird durch eine vibrierende Platte oder einen elektromagnetischen Antrieb in Schwingungen versetzt, um die Kleinteile in Bewegung zu setzen. Der Fördermechanismus besteht aus verschiedenen mechanischen Komponenten wie Förderbändern, Schwerkraftrutschen, Schwingarmen und Schneckenförderern, die je nach Art der Kleinteile und dem gewünschten Verarbeitungsprozess ausgewählt werden. Eine wichtige Komponente des Wendelförderers ist der Sensor, der den Zustand und die Position der Kleinteile erfasst und die Steuerung der Maschine anpasst. Wendelförderer wird in vielen industriellen Bereichen eingesetzt, wie zum Beispiel in der Automobilindustrie, der Elektronikindustrie und der Verpackungsindustrie. Es bietet eine effiziente Möglichkeit, Kleinteile in großen Mengen zu handhaben und zu verarbeiten, was die Produktivität erhöht und die menschliche Arbeitsbelastung verringert. Gemäß einer Studie von Boothroyd [5] werden die Wendelförderer auch als Teilezuführsysteme bezeichnet und haben in der Automobilindustrie eine wichtige Rolle bei der Montage von Motoren gespielt. Eine weitere Studie von Fathi [7] beschäftigt sich mit der Optimierung des Fördermechanismus und der Steuerung des Bowl Feeders, um die Effizienz und Genauigkeit der Handhabung von Kleinteilen zu verbessern. Insgesamt bietet der Wendelförderer eine zuverlässige und effektive Lösung für die Handhabung von Kleinteilen in der industriellen Produktion. Durch die Integration von fortschrittlichen Technologien wie Sensoren und Steuerungssystemen kann die Effizienz und Genauigkeit weiter verbessert werden. Im Betrieb des Wendelförderers der vorliegenden Gurtung-

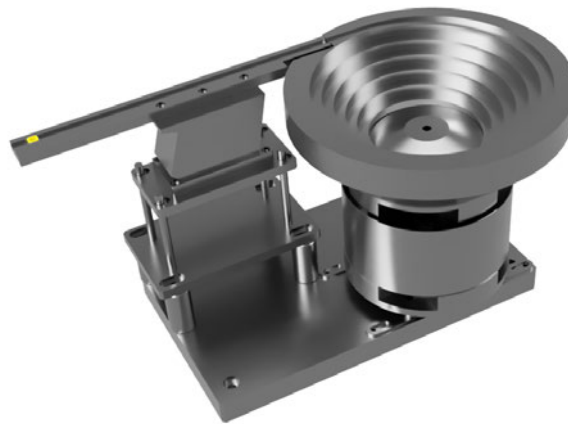


Abbildung 4.2: CAD-Modell eines Wendelförderers

maschine (siehe die Abb. 4.2) sind folgende Einstellungen zu beachten. Die Schüsselspirale ist mit einer speziellen Fräse und Stoßstangen ausgestattet, um sicherzustellen, dass die MVS ausschließlich vertikal (hochkant) entlang der Spirale bewegt werden können. Die Einstellung der Vibration ermöglicht es den MVS, erfolgreich durch die Spirale befördert zu werden. Um sicherzustellen, dass die MVS seitenrichtig in die Vibrationsrinne gelangen, ist am Ende der Spirale ein optischer Sensor installiert. Der Sensor erkennt, ob die MVS korrekt ausgerichtet sind, indem er einen Laserstrahl auf sie richtet und ihre Reflexion erfasst. Falls eine Reflexion auftritt, wird automatisch erkannt, dass die Kupferkontakte der MVS dem optischen Sensor entgegengesetzt sind. In diesem Fall wird das

System durch Druckluft angehalten, und die MVS werden aus der Spirale entfernt. Die Vibrationsrinne selbst ist mit einem weiteren optischen Sensor am Anfang der Förderstrecke ausgestattet. Ein Sensor am Anfang der Rinne stellt sicher, dass die Rinne nicht überlastet wird. Wenn das Lichtsignal durch einen MVS für einen bestimmten Zeitraum blockiert wird, wird der Vibrationstopf umgehend ausgeschaltet und die weitere Förderung gestoppt, bis die Rinne wieder leer ist und das Lichtsignal störungsfrei empfangen wird.

4.3 Multifunktionales Modul

Im Anschluss an die Vibrationsrinne werden die Prüflinge auf einem mechanisch entkoppelten, metallischen Würfel abgelegt. Sobald der MVS auf diesem Würfel platziert wird und somit das Lasersignal blockiert, wird der Prüfling mithilfe eines pneumatischen Manipulators auf den Gurt weiterbewegt.

An diesem Punkt ist es sinnvoll, das konzipierte multifunktionale Modul einzusetzen (siehe Abb. 4.3). Geplant ist, den metallischen Würfel durch einen speziellen Messmanipulator zu ersetzen. Der

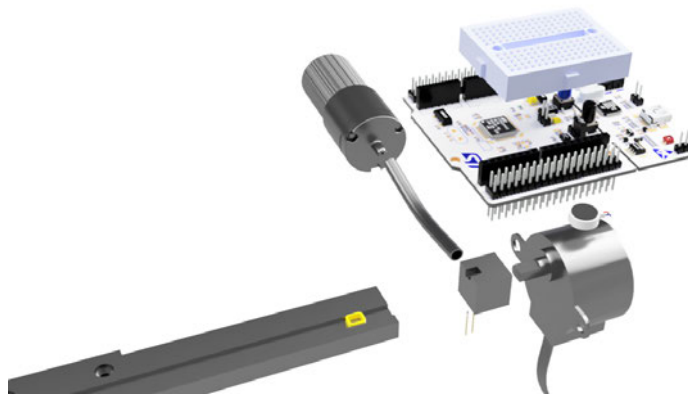


Abbildung 4.3: Konzept eines multifunktionalen Moduls als CAD-Modell

Messmanipulator soll nicht nur eine sichere elektrische Kontaktierung gewährleisten, sondern auch in der Lage sein, den Prüfling in zwei verschiedenen Ausrichtungen, nämlich bei 0 Grad und 180 Grad, anzuregen und zu messen. Das Signal des MVS soll durch eine elektrische Schaltung erfasst und durch einen MCU ausgewertet werden. Anhand dieser Auswertung sollen die Prüflinge entsprechend sortiert werden. Darüber hinaus soll der gesamte Messprozess vollständig durch das MCU gesteuert werden. In den folgenden Abschnitten (4.3.1 - 5.2) erfolgt eine umfassende Erläuterung der einzelnen Submodule des konzipierten Gerätes. Dabei wird zunächst auf den Messmanipulator eingegangen und danach auf MCU konzentriert.

4.3.1 Messmanipulator

Die Realisierung des Submoduls zur Anregung und Messung der Prüflinge erfordert eine Vielzahl kreativer Lösungen, um den technischen Anforderungen gerecht zu werden.

Insbesondere müssen folgende Herausforderungen bewältigt werden:

1. eine sichere und zuverlässige elektrische Kontaktierung zwischen den Messspitzen und dem Prüfling

2. eine präzise Rotation der Prüfstation und geeignete Anregungsquelle
3. eine kompakte Konstruktion

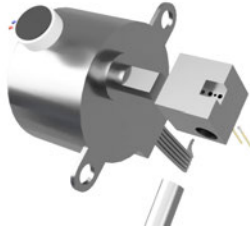


Abbildung 4.4: CAD-Modelle eines Messmanipulators

Um die elektrische Kontaktierung zwischen dem Prüfling und den Messspitzen zu gewährleisten, wurde eine spezielle würfelförmige Station entwickelt, welche den Prüfling durch Saugluftdruck auf die Messkontakte anpresst. Dadurch bleibt die Kontaktierung auch bei einer Rotation um 180 Grad stabil. In der Abbildung 4.5 ist ein Teil der technischen Zeichnung der Station zu sehen. Die Draufsicht

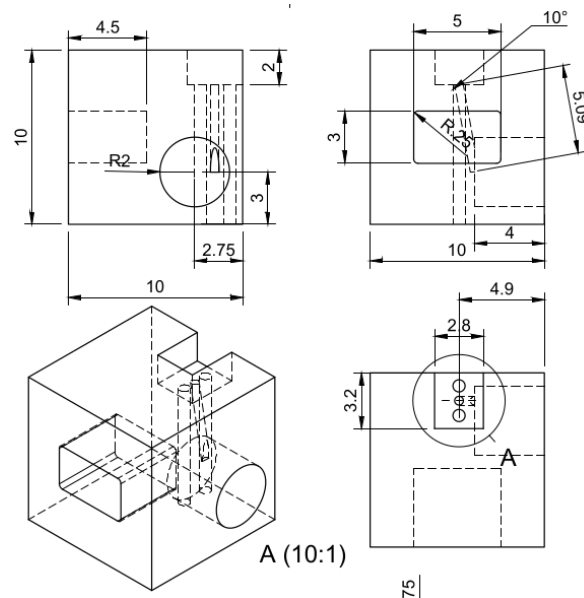


Abbildung 4.5: Technische Zeichnung einer würfelförmigen Messstation

zeigt eine quadratische Vertiefung, die an die Größe des Prüflings angepasst ist. In dieser Vertiefung befinden sich drei Bohrungen, wobei die beiden äußeren für die Halterung der Messspitzen vorgesehen sind. Als Messspitzen kommen spezielle Pogopins² zum Einsatz, da sie aufgrund ihrer Bauart eine höhere Haltbarkeit und Widerstandsfähigkeit gegenüber mechanischen Stößen und Vibrationen aufweisen. Die mittlere Bohrung führt schräg zu einem Saugdruckanschluss, welcher auch in der Vorderansicht dargestellt ist. Als Saugdruckquelle wird eine kleine Luftpumpe verwendet, die aus einem Gleichstrommotor besteht. Die Steuerung der Pumpe ist einfach gehalten, da lediglich das Ein- und Ausschalten des Motors erforderlich ist. Der Schaltungsentwurf zur elektrischen Steuerung der Pumpe wird im nachfolgenden Abschnitt 5.2 detailliert beschrieben. Die Seitenansicht zeigt eine rechteckige Bohrung für die Welle eines Schrittmotors. Der unipolare Schrittmotor ermöglicht eine präzise Steuerung der Drehbewegung der Prüfstation. Für die Anregung des Prüflings wurde ein Vibrationsmotor auf dem Körper eines Schrittmotors platziert. Da beide mechanisch gekoppelt sind,

²Ein Pogo-Pin oder federbelasteter Stift ist eine Art elektrischer Verbindungsmechanismus, der in vielen modernen elektronischen Anwendungen und in der Elektronikprüfindustrie verwendet wird.

werden die Vibrationen von dem Vibrationsmotor zum Prüfling über die Welle des Schrittmotors übertragen. Der Vibrationsmotor soll so ausgewählt werden, dass er eine ausreichende Schwingung für die Anregung des Prüflings erzeugen kann. Durch diese Konstruktion wird eine effektive Anregung des Prüflings und eine zuverlässige Übertragung des Messsignals durch die Messspitzen gewährleistet.

Die Messschaltung wird im Abschnitt 5.2 beschrieben.

4.3.2 Mikrocontroller

4.3.2.1 Auswahl eines Mikrocontrollers

Zur erfolgreichen Durchführung des Projektes spielt die Programmierung eines geeigneten Mikrocontrollers eine zentrale Rolle, da dieser die Steuerung und Datenverarbeitung ermöglicht. Um einen passenden Mikrocontroller auszuwählen, müssen zunächst die erforderlichen Peripheriegeräte sowie die technischen Eigenschaften wie Referenzspannung und Abtastfrequenz berücksichtigt werden. In der Anwendung sollen mehrere Aktoren über den Mikrocontroller angesteuert werden, daher muss der Mikrocontroller über ausreichend viele GPIOs verfügen. Da es beim Messen vor allem um analoge Signale geht, müssen diese zuerst digitalisiert werden, damit eine weitere Signalverarbeitung möglich ist. Hierfür sollte der Mikrocontroller über einen ADC (siehe auch 2.1) verfügen. Bei der Auswahl eines ADC sind insbesondere die Länge des kürzesten abzutastenden Impulses sowie die gewünschte Auflösung des ADCs zu berücksichtigen. Im Rahmen der vorliegenden Untersuchung wurde die erforderliche Länge eines Impulses ermittelt, welche sich auf $30\mu s$ beläuft. Ausgehend von dem Shannon-Nyquist-Abtasttheorem lässt sich ableiten, dass die Abtastfrequenz f_A größer als $66,6\text{ kHz}$ sein sollte, um das Signal angemessen abtasten und digitalisieren zu können. Zur Sicherstellung einer präzisen Signalabbildung ist ein ADC mit einer Auflösung von 16 Bit empfehlenswert. Dies würde bedeuten, dass für die Konvertierung $2^{16} = 65536$ mögliche Digits zur Verfügung stehen. Eine höhere Auflösung würde zwar eine genauere Signalabbildung ermöglichen, aber auch den Preis für den ADC erhöhen und die Prozessorkapazität des Mikrocontrollers beanspruchen. Daher muss ein Kompromiss zwischen der notwendigen Genauigkeit und der Leistungsfähigkeit des Systems gefunden werden. Nach Festlegung der erforderlichen Messspezifikationen muss die für die Datenverarbeitung erforderliche Hardware bestimmt werden. Die Messung muss innerhalb einer Sekunde durchgeführt werden, weshalb der Einsatz von Counter/Timer unerlässlich ist. Die meisten MCU verfügen in der Regel über Counter/Timer, es ist jedoch wichtig, einen geeigneten Prescaler einzustellen, um sicherzustellen, dass die erforderliche Genauigkeit und Frequenz erreicht wird. Um eine reibungslose Datenverarbeitung während der Messung zu gewährleisten, ist eine angemessene Speicherkapazität des Arbeitsspeichers (SRAM³) von entscheidender Bedeutung, insbesondere bei hohen Abtastfrequenzen, da hierbei große Datenmengen generiert werden. Daher sollte bei der Auswahl des Mikrocontrollers die SRAM-Größe berücksichtigt werden, um sicherzustellen, dass genügend Speicherplatz für die zeitnahe Verarbeitung und Analyse der Daten zur Verfügung steht. Für unseren Fall hieße es: mit 16 Bit ADC und minimaler Abtastfrequenz von $66,6\text{ kHz}$ bräuchte man einen Array aus 16 Bit Integer der Mindestgröße 66666 , damit soll die SRAM Größe größer als $133,3\text{ kB}$ sein.

Zusammenfassend sind folgende Kriterien beim Auswahl eines Mikrocontrollers für die vorliegende Applikation zu berücksichtigen:

³Static random-access memory (deutsch: statisches RAM, Abkürzung: SRAM) bezeichnet einen elektronischen Speicherbaustein. Zusammen mit dem dynamischen RAM (DRAM) bildet es die Gruppe der flüchtigen (volatil; engl. volatile) Speicher, das heißt, die gespeicherte Information geht bei Abschaltung der Betriebsspannung verloren.

- angemessene Anzahl von GPIOs
 - Eingang für das Lesen eines analogen Signals
 - Ausgänge für die Aktoren (Schrittmotor, Pumpe, Vibrationsmotor)
- ein passender Analog-Digital-Wandler (ADC)
 - Auflösung 16-Bit
 - Abtastfrequenz $f_A > 66,6\text{kHz}$
- breit einstellbare Prescaler
- ausreichende SRAM
 - SRAM > 133,3 kB

4.3.2.2 Algorithmus

Nach der Festlegung der Spezifikationen ist es notwendig, einen geeigneten Algorithmus für die Prozesssteuerung zu entwickeln. Ein Flussdiagramm des vorgeschlagenen Algorithmus wird in Abbildung 4.6 dargestellt.

Es wird angenommen, dass der zu prüfende Gegenstand (MVS) bereits zu Beginn des Verfahrens erkannt wurde, beispielsweise durch einen optischen Sensor. In diesem Fall ist es sinnvoll, den Gegenstand innerhalb einer bestimmten Periode zu messen, indem er zu festgelegten Zeitpunkten angeregt und die daraus resultierenden Messdaten in einem Speicher abgelegt werden. Eine solche Vorgehensweise ist besonders wichtig, um die Genauigkeit der Messdaten zu gewährleisten und mögliche Fehler rechtzeitig zu erkennen.

Ein Beispiel für diesen Messvorgang ist wie folgt: Der eine sekundenlange Messvorgang beginnt zum Zeitpunkt t_0 , wobei der Gegenstand durch ein Vibrationssignal $x(t)$ zum Zeitpunkt t_1 für eine Dauer von Δt angeregt wird. Das Ausgangssignal des Gegenstands $a(t)$ wird dabei durch die Messschaltung am Eingang des Mikrocontrollers gelesen (Digitalisierung durch ADC $a(t) \rightarrow a[t]$) und in einem Feld gespeichert. Da alle Zeitpunkte bekannt sind und das Messsignal ebenfalls verfügbar ist, kann die Analyse des Signals vergleichsweise einfach durchgeführt werden. Zusätzlich ist es möglich, das Messsignal in Echtzeit auf einem Oszilloskop zu beobachten, um sicherzustellen, dass das gespeicherte Signal dem tatsächlichen Signal entspricht.

Im nächsten Schritt des Algorithmus, wie im Flussdiagramm dargestellt, werden die Messdaten analysiert. Hierbei werden zunächst zwei Gütekriterien aus dem Messsignal bestimmt, nämlich der On_{Ratio} und n_{Pulse} (vgl. Abschnitt 3.2). Durch den Einsatz eines leistungsstarken Mikrocontrollers

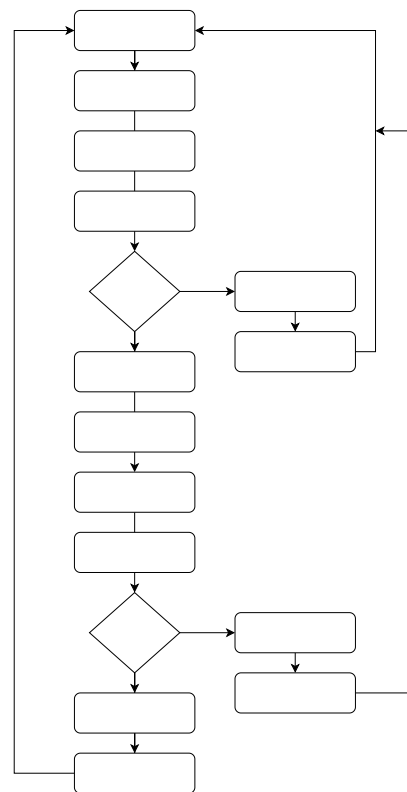


Abbildung 4.6: Flussdiagramm des entwickelten Algorithmus

ist es möglich, das Messsignal präziser auszuwerten und damit eine genauere Klassifizierung der Sensoren abhängig von den Kundenapplikationen zu erzielen.

Insbesondere für die Kundenapplikation, die in Abschnitt 3.1 beschrieben ist, ist die Verwendung des Gütekriteriums On_{Ratio} nicht adäquat. Dieses Kriterium gibt lediglich die aufsummierte Dauer an, in der der Schalter geschlossen ist, und liefert keine Aussage darüber, ob der Sensor im Ruhezustand tatsächlich niederohmig ist. An dieser Stelle ist eine prozedurale Auswertung des Messsignals sinnvoll. Hierbei wird das Signal in drei Phasen unterteilt:

1. Die erste Phase dauert bis zum Zeitpunkt t_1 , welcher dem Anfang der Vibration entspricht. In diesem Abschnitt des Signals kann der Bereich zwischen t_0 und t_1 auf die Änderung des Spannungswertes geprüft werden.
2. Die zweite Phase kann als summierte Dauer der Vibration und der während der Untersuchung ermittelten Einschwingungsdauer bestimmt werden: $t_1 + \Delta t + 400ms$. Da in dieser Phase die Anregung erfolgt, ist es sinnvoll, die Anzahl der Pulse n_{Pulse} zu berechnen und mit einem bestimmen Sollwert der Pulse zu vergleichen.
3. Die folgende dritte Phase dauert bis zum Ende der Messung und erfordert wie die erste Phase die Überprüfung des Spannungswertes.

Die Erfüllung von Bedingungen aller drei Phasen bestimmt einen qualitativ hochwertigen Prüfling. Es ist von Bedeutung, dass die Ergebnisse einschließlich der Rohdaten nach der Analyse gespeichert werden, um eine spätere Fehleranalyse zu ermöglichen. Insbesondere bei großen Datensätzen, die getrennt von dem Mikrocontroller verwaltet werden sollten, empfiehlt sich die Übertragung der Daten über den UART-Port nach jeder Messung und die Speicherung auf einem externen Rechner. Im Anschluss an die Analyse der Messdaten wird, sofern der Prüfling als qualitativ hochwertig eingestuft wurde, eine weitere Testreihe mit dem Ziel durchgeführt, die Ergebnisse der ersten Testreihe zu bestätigen. Diese zweite Testreihe besteht in einer erneuten Durchführung des Testverfahrens nach einer Rotation des Prüflings um 180 Grad. Sollte das Ergebnis auch in diesem Fall positiv ausfallen, wird der Prüfling in die Kategorie der hochwertigen Sensoren einsortiert. Andernfalls erfolgt eine Klassifikation als minderwertiger Prüfling.

5 Implementation eines multifunktionalen Moduls

5.1 Aufbau eines Prototyps

Der Prototyp des Moduls wurde mit geringfügigen Abweichungen gemäß dem im Kapitel 4 beschriebenen Entwurf aufgebaut. Dieser Prototyp ist aus einem Experimentierboard mit einem Mikrocontroller, drei unterschiedlichen Motoren und einer Schnittstelle mit integrierten elektrischen Schaltungen aus Abschnitt 5.2 aufgebaut. Dabei wurde das STM32 Nucleo-144 [15] mit einem STM32H7A3ZI Mikrocontroller als Experimentierboard gewählt, da es alle im Abschnitt 4.3.2 beschriebenen Anforderungen erfüllt.

Der eingesetzte μC verfügt über:

- eine hohe Anzahl von GPIOs (144) mit unterschiedlichen Arten von Konnektoren
- einen 16-Bit ADC
- die Abtastfrequenz bis zu $96 MHz$
- eine breit einstellbare Vorteilerkaskade
- 2 MB RAM und 1.4 MB SRAM
- einen UART Port
- eine Vielzahl von Timer-Optionen

Ferner ist der ST-LINK-Debugger/Programmierer bereits in das verwendete Experimentierboard integriert. Die Programmierung des μC erfolgt durch die Anwendung der Programmiersprache C.

Als Schrittmotor wurde Modell 28BYJ-48 [1] ausgewählt. Dieses Modell ist ein unipolarer Schrittmotor und wurde aufgrund seiner günstigen Kosten und technischen Eigenschaften für die Applikation als passend erachtet. Der Schrittmotor arbeitet mit einer Spannung von 5V und benötigt daher keine externe Spannungsquelle, sondern kann direkt von dem μC gespeist werden. Der Schrittwinkel des Motors beträgt 5.625° , was für Testzwecke ausreichend ist. Als Treiberstufe wurde der ULN2003-IC ausgewählt, die im Abschnitt 5.2 beschrieben wird.

Der Saugluftdruck wird mithilfe einer Vakuumpumpe – „Magt Air Pump“ [6] erzeugt. Die verwendete Saugpumpe besitzt folgende spezifischen Eigenschaften: eine Nennspannung von DC 12 V, eine Nennstromaufnahme von unter $400 mA$, eine Luftströmung ohne Last zwischen $2 LPM$ und $3.2 LPM$ und eine maximale Vakuumleistung von $-420 mmHg$. Nach einer Vielzahl von Experimenten konnte festgestellt werden, dass die Pumpe eine ausreichende Vakuumleistung aufweist, um den Prüfling auch bei einer Drehung um 180 Grad in Position zu halten und somit die elektrische Verbindung zu den Messspitzen aufrechtzuerhalten. Des Weiteren ist der Geräuschpegel der Pumpe mit weniger als $65 dB$ angegeben. Die Schnittstelle des Luftauslasses ist für Schläuche mit einem inneren Durchmesser von $4 mm$ ausgelegt. Die Pumpe hat ein ungefähres Gewicht von $64g$. Für die Steuerung der Pumpe ist eine Transistorschaltung mit Schutzdiode erforderlich. Diese wird detaillierter im Abschnitt 5.2 beschrieben.

Die Anregung des Prüflings passiert durch Einschalten eines Knopfvibrationsmotors von AZDelivery [9]. Der Motor weist eine Nennspannung von 5V auf und kann bei einer Betriebsspannung zwischen 3.0 und 5.3 V betrieben werden. Die Nenndrehzahl des Motors beträgt mindestens 9000 U/min, während der Nennstrom bis zu 75 mA betragen kann. Der Anlaufstrom kann jedoch bis zu 90 mA erreichen und die Anlaufspannung beträgt 3,7 VDC. Der Motor hat einen Durchmesser von 10mm und eine Dicke von 2,7mm und sein Isolationswiderstand beträgt $10\text{ M}\Omega$. Der digitale Eingang wird durch ein hohes Signal aktiviert und durch ein niedriges Signal deaktiviert. Der Motor verfügt zudem über ein Befestigungsloch mit einem Durchmesser von 3mm. Zum Vorteil kommt der Knopfvibrationsmotor in einem Modul mit integrierter Transistorschaltung, was die direkte Verbindung mit dem μC ermöglicht. Die Abb. 5.1 und 5.2 stellen den aufgebauten Prototyp mit oben beschriebenen Komponenten dar.

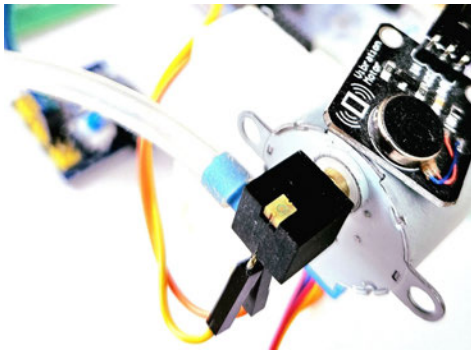


Abbildung 5.1: Messmanipulator

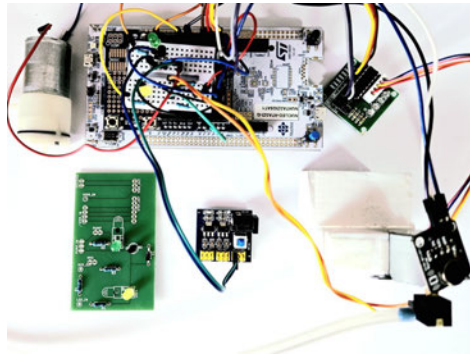


Abbildung 5.2: multifunktionales Modul

5.2 Elektrische Schaltungen

Um die optimale Verbindung zwischen dem Mikrocontroller und äußeren Peripherie zu ermöglichen, ist es unerlässlich, auf das Grundwissen der Elektrotechnik zuzugreifen, nämlich die Entwicklung der elektrischen Schaltkreise. Wie im Abschnitt 4.3.1 schon beschrieben wurde, besteht der Messmanipulator aus vier Komponenten: Messstation, Saugpumpe, Schrittmotor und Vibrationsmotor, die jeweils mit dem Mikrocontroller elektrisch zu verbinden sind. Im folgenden Abschnitt werden die elektrischen Schaltungen, die speziell für die Anwendung entwickelt wurden, sowie die erforderlichen integrierten Schaltungen beschrieben.

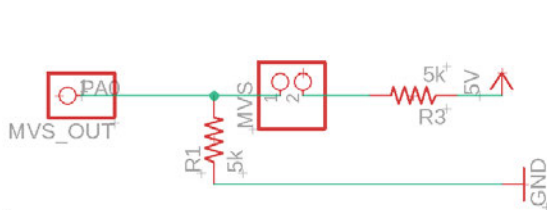


Abbildung 5.3: Messschaltung

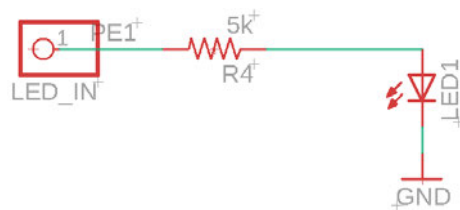


Abbildung 5.4: Hilfeschtaltung

Die in der Abbildung 5.3 dargestellte Schaltung dient der Erfassung des Messsignals und unterscheidet sich wesentlich von der ersten (Abbildung 3.9), insbesondere aufgrund der Möglichkeit der Einbindung des Mikrocontrollers. Das grundlegende Prinzip, die Messung der Spannungsänderung an einem seriell geschalteten Widerstand, bleibt jedoch unverändert. Die Messschaltung besteht aus einem einfachen Spannungsteiler und enthält keinen externen Bauteil zur Aufzeichnung des Beginns der Vibration, dabei wird der Prüfling zwischen den Widerständen R_1 und R_5 platziert. An

dem Widerstand R_1 erfolgt die Messung des Signals, welches sowohl dem Messeingang des Mikrocontrollers als auch Kanal 1 des Oszilloskops zugeführt wird. Für die visuelle Darstellung der Dauer des Messvorganges wurde eine einfache Schaltung entworfen, welche in Abbildung 5.4 dargestellt ist. Die Schaltung besteht aus einer Serienschaltung des Widerstands R_4 und einer Leuchtdiode LED_1 . Das Signal wird über den Widerstand R_4 abgegriffen und am zweiten Kanal des Oszilloskops angezeigt, um die Dauer der Messung in Echtzeit darstellen zu können. Damit kann die eingestellte Abtastdauer relativ einfach kontrolliert werden. Die Leuchtdiode LED_1 dient der optischen Anzeige des Signals.

Induktive elektrische Lasten können nicht direkt an einen Mikrocontroller angeschlossen werden, da dies zu einer Überlastung des μC führen würde. Aus diesem Grund müssen die Schaltungen mit Transistoren verwendet werden.

In der Elektrotechnik eignen sich Transistoren zur kontaktlosen Steuerung von kleinen bis mittleren Leistungen. Dabei fungiert die Kollektor-Emitter-Strecke (CE-Strecke) des Transistors als Schaltstrecke, während der Basisanschluss als Steuerelektrode dient. Um den Transistor als Schalter für induktive Lasten zu verwenden, ist die Anwesenheit einer Freilaufdiode in der Schaltung unerlässlich. Ohne sie kann selbst der Transistor durch die Schaltvorgänge einer induktiven Last überlastet werden. Wenn eine Induktivität im Kollektor-Stromkreis vorhanden ist und abgeschaltet wird, kollabiert das Magnetfeld, das durch den Stromfluss in der Induktivität aufgebaut wurde, schlagartig. Die Spule versucht, die abgeschaltete Spannung aufrechtzuerhalten und erzeugt so eine Induktionsspannung.

Um dies zu verhindern, wird eine Diode parallel zur Induktivität geschaltet, um die Spannungsbegrenzung sicherzustellen. Die in der Abbildung 5.5 dargestellte Transistorschaltung mit Freilaufdiode wird zur Steuerung eines Gleichstrommotors eingesetzt, der auch als Saugpumpe in der Applikation verwendet wird. Die Freilaufdiode wird parallel zum Motor oder, elektrisch gesehen, der Induktivität geschaltet, um eine hohe Induktionsspannung zu vermeiden, die bei einem sperrenden Transistor auftreten kann. Die Diode fungiert hier als Schutzdiode und schließt die Induktionsspannung kurz, wobei sie auf den Wert der Diodendurchflussspannung begrenzt wird.

Für die Steuerung eines unipolaren Schrittmotors wird die Verwendung einer Treiberstufe benötigt (siehe auch Abschnitt 2.3). Eine integrierte Lösung für diesen Zweck ist der ULN2003-IC [8] (siehe Abb. 5.6), welcher im Wesentlichen aus Darlington-Paaren von acht Transistoren besteht. Dieser Chip dient als Schnittstelle zwischen den vier Ausgangspins des Mikrocontrollers und den vier Wicklungen des Schrittmotors und muss mit der für den Antrieb des Motors erforderlichen Stromversorgung verbunden werden.

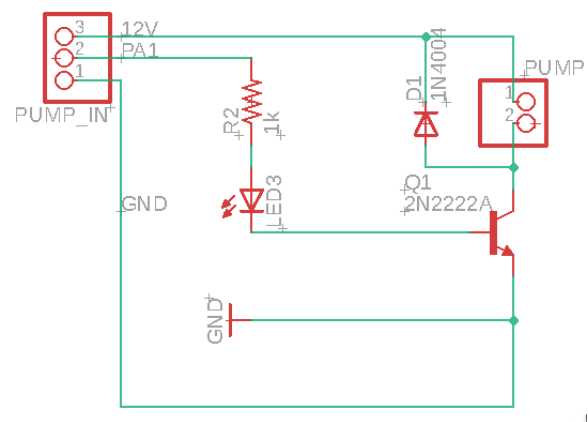


Abbildung 5.5: Elektrische Schaltung zur Steuerung der Pumpe

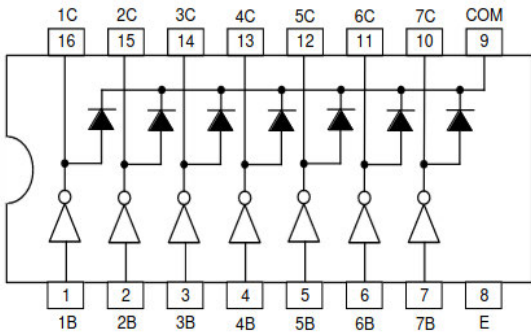


Abbildung 5.6: Pinbelegung von ULN2003 [8]

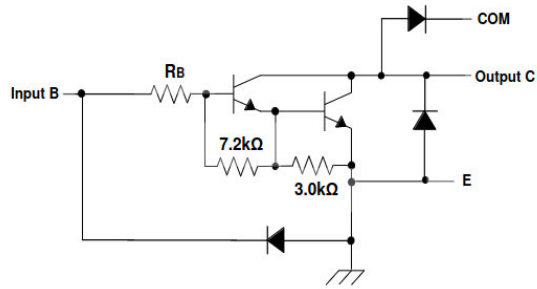


Abbildung 5.7: funktionales Blockschaltbild [8]

In der Schaltung (siehe Abb. 5.7) ist das Eingangssignal B direkt über den Widerstand R_B mit der Basis der in Parallelschaltung mit Freilaufdiode stehenden Darlington-Transistor-Paare verbunden. Dabei wirken die Transistorpaare wie ein einzelner Transistor und führen zu einer Erhöhung des Verstärkungsfaktors B . Die gesamte Verstärkung B entspricht dabei ungefähr dem Produkt der Verstärkungen der beiden Einzeltransistoren (B_1 bzw. B_2).

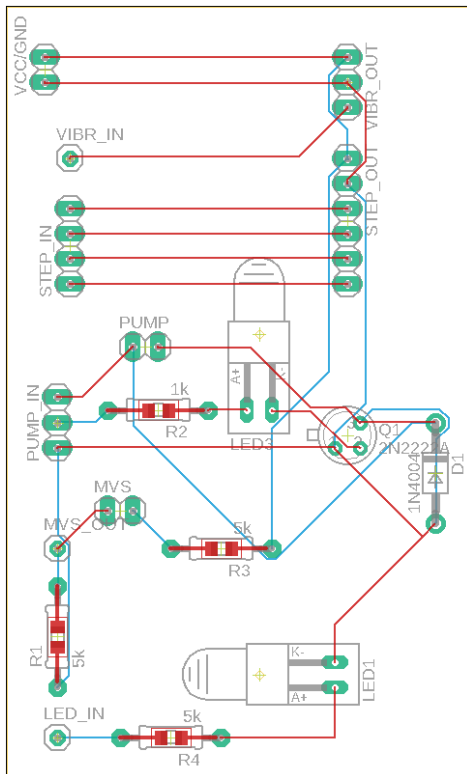


Abbildung 5.8: Schaltungsentwurf mittels EAGLE

Die Implementierung der oben beschriebenen Schaltungen wurde auf einer eigens dafür entwickelten Leiterplatte durchgeführt. Der Entwurf erfolgte mithilfe des CAD-Programms Eagle und wurde von der B&B-Gruppe hergestellt. Die Abbildung 5.8 zeigt diesen Entwurf. Die Leiterplatte besteht aus zwei Schichten – oberer und unterer Schicht, wobei die roten Leitungen die Verbindungen im oberen Schicht und die blauen im unteren Schicht darstellen. Aufgrund der fehlenden Bestückungsmöglichkeiten ist die Leiterplatte für THT-Bauteile⁴ ausgelegt. Die Leiterplatte dient als Schnittstelle zwischen dem μC und den Aktoren. Entsprechend sind die Steuereingänge der Aktoren auf der Leiterplatte als Ausgänge (OUT) bezeichnet, während die Eingänge (IN) der Leiterplatte direkt mit dem μC zu verbinden sind.

Da die meisten elektrischen Komponenten zunächst auf ihre Kompatibilität getestet werden sollten, werden die Schaltungen auf einem Steckbrett aufgebaut, um die Möglichkeit zu haben, die Schaltungselemente beliebig oft zu ändern.

Nachdem alle elektrischen Bauteile erfolgreich getestet werden, wird die entwickelte Leiterplatte bestückt.

⁴Als Durchsteckmontage oder auch Einsteckmontage (englisch through-hole technology, THT; pin-in-hole technology, PIH) bezeichnet man in der Aufbau- und Verbindungstechnik eine Montageweise von bedrahteten elektronischen Bauelementen.

5.3 Softwaretechnische Lösungen

Die Implementierung des im Abschnitt 4.3.2 beschriebenen Algorithmus erfolgte in der Programmiersprache C. Diese Programmiersprache zählt im Bereich der Mikrocontrollertechnik aufgrund ihrer Geschwindigkeit, Speicherhandhabung und Struktur zu den gängigsten. Als integrierte Entwicklungsumgebung wurde die STM32CubeIDE von STM32-Familie verwendet, die mit zahlreichen Tools wie beispielsweise einem eingebetteten Debugger, grafischen Konfigurationspaneelen und Texteditor ausgestattet ist.

Während der Softwareentwicklung wurden folgende Tools genutzt:

- Linux-Betriebssystem
- GCC-Compiler
- GDB-Debugger
- STM32CubeIDE
- HAL Bibliothek
- Stepper Driver ECUAL

In Bezug auf die Signalverarbeitung ist es von grundlegender Bedeutung, zuerst eine geeignete Datenstruktur zu definieren, die das gemessene Signal und die dazugehörigen markanten Daten enthält. In Quelltext A.1 wird die Implementierung dieser Datenstruktur in der Programmiersprache C dargestellt. Die Datenstruktur *MeasureData* besteht aus einem Feld *signal*, das das Messsignal speichert, sowie 4 Variablen vom Typ *int* und einer Variable vom Typ *bool*. In den Variablen *t_vibr_start* und *t_end* werden die Zeitpunkte des Beginns der Anregung und des Endes der Messung gespeichert. Die Variablen *pulses* und *htime* enthalten die berechneten Werte für die [Anzahl der Impulse pro Messung](#) und die [Hightime](#), die für die weitere Signalanalyse relevant sind.

Die Funktion zur Realisierung der Messung (siehe auch Abschnitt 4.3.2.2), trägt den Namen *measure()* und ist in Quelltext A.3 definiert. Das Ziel dieser Funktion besteht darin, ein möglichst genaues Zeitfenster von einer Sekunde zu erzeugen und innerhalb dieses Zeitraums das Signal zu messen. Während der Messung wird eine Anregung von 100 Millisekunden Dauer durchgeführt, deren Zeitpunkt entsprechend in der zuvor beschriebenen Datenstruktur gespeichert wird. Die Implementierung dieser Funktion erfordert zunächst die konkrete Anpassung der Hardware. Es ist notwendig, die korrekte Funktion von [Analog-Digital-Wandler](#), Timern und GPIO-Pins sicherzustellen. Diese Hardwarekomponenten sind wesentlich für die korrekte Erfassung des Signals und die zeitliche Steuerung der Messung. Ein reibungsloser Betrieb dieser Komponenten stellt somit eine notwendige Voraussetzung für die korrekte Funktion der Messfunktion dar. Der Quelltext A.2 zeigt die Konfigurationen des Analog-Digital-Wandlers (ADW) und des 16-Bit Timers. Der ADW ist auf eine Abtastfrequenz von 32 MHz und eine Auflösung von 16 Bit eingestellt. Damit werden die wesentlichen Voraussetzungen für eine verlustarme Abtastung erfüllt. Dank der relativ hohen Auflösung können die Messwerte, in unserem Fall die Spannung, präzise erfasst werden, da sie den Wertebereich zwischen 0 und 65536 abdecken können. Die Abtastfrequenz ist weit höher als die kleinstmögliche Impulsrate des Sensors und führt daher zu einem äußerst geringen Informationsverlust. Zur Erzeugung eines Zeitfensters wird Timer 16 verwendet, ein 16-Bit Timer, der bis zu 65536 zählen kann. Da der Prozessortakt 96 MHz beträgt, würde der Timer sehr schnell überlaufen, wenn er mit dieser Frequenz zählen würde. Daher wird ein Prescaler mit einem geeigneten Skalierungsfaktor zwischen dem Prozessor und dem Timer eingesetzt. Mit einem Skalierungsfaktor von 9600 kann die Taktfrequenz auf 10 kHz reduziert werden, sodass der Timer alle 0,1 ms inkrementiert wird. Um eine Sekunde zu erreichen, muss der

Timer bis 10000 gezählt werden. Die Schrittdauer von 0,1 ms ermöglicht eine ausreichend präzise Zeitmarkierung (z. B. das Schreiben eines Signals auf GPIO nach 200 ms, wobei der gewünschte Zeitpunkt mit einer Abweichung von +/-0,1 ms erreicht werden kann).

Die Messfunktion wird folgendermaßen technisch realisiert: Zunächst wird der Timer 16 gestartet und der Ausgang PG9 sofort auf High gesetzt, um ein Hilfssignal zu erzeugen, welches parallel zur Messung läuft und den Messzeitfenster visualisiert. Dies ist besonders nützlich für weitere Debugging-Aufgaben. Die eigentliche Messung wird in einer „for-Schleife“ durchgeführt, in der der [ADW](#) ständig ausgelöst wird, um den Messwert zu konvertieren und in eine Variable *raw* zu schreiben, welche als *i*-te Element in das oben beschriebene Feld gespeichert wird. Die Messfunktion beinhaltet auch „if-Bedingungen“, die das zeitgerechte Ein- und Ausschalten des Vibrationsmotors durch die Steuerung des Ausganges PG12 realisieren. Das Einschalten erfolgt nach 200 ms und das Ausschalten nach 300 ms. Die erforderlichen Zeitpunkte werden ebenfalls in der oben beschriebenen Datenstruktur gespeichert. Sobald der Timer eine Sekunde überschritten hat, wird er gestoppt und wieder auf 0 gesetzt.

Nach Abschluss der Messung können die Messwerte analysiert werden. Hierfür wurde die Funktion *analyse()* (siehe auch Quelltext [A.4](#)) implementiert. Technisch gesehen führt die Funktion folgende Schritte aus: Zunächst wird das gemessene Signal diskretisiert, indem jeder Messwert mit einer Triggergrenze verglichen wird. Wenn der Messwert größer als die Triggergrenze ist, wird an dieser Stelle eine „1“ geschrieben, ansonsten eine „0“. Durch die Diskretisierung wird die Verarbeitung des Signals erheblich erleichtert. Anschließend wird in einer „for-Schleife“ die Anzahl der negativen Impulse gezählt. Hierzu werden zuerst die aufsteigenden Flanken des Signals gefunden und alle Einsen gezählt, um die High-Time zu berechnen. In den letzten beiden „for-Schleifen“ wird geprüft, ob das Signal vor Beginn der Vibration und 200 ms vor dem Ende des Signals auf High war. Wenn dies nicht der Fall ist, wird die Zustandsvariable *bad* auf 1 gesetzt, andernfalls auf 0.

Basierend auf den Variablenwerten von *bad* und *pulses* wird entschieden, ob eine weitere Prüfung mit einer 180-Grad-Drehung durchgeführt werden soll. Hierbei wird ein Schrittmotor durch die Ansteuerung der Driverbibliothek *ECUAL* gesteuert. Die Implementierung des Algorithmus, der im Abschnitt [4.3.2.2](#) beschrieben wurde, erfolgt vollständig im C-Code der Quelltextdatei [A.5](#).

5.4 Testmethodik

Nach der Implementierung des Algorithmus ist es unerlässlich, den geschriebenen Code zu testen. Eine Möglichkeit dazu besteht darin, das Testobjekt mit einer wahren Referenz zu vergleichen. Im vorliegenden Fall handelt es sich um die Signalverarbeitung, bei der die Frage im Raum steht, ob das gemessene Signal mit dem realen übereinstimmt. Als Referenz dient das parallel am Oszilloskop gemessene Signal. Im Anschluss muss geprüft werden, ob die gemessenen oder gespeicherten Daten visuell mit den Referenzdaten übereinstimmen können. Sobald keine oder nur minimale Unterschiede in den Plots zu erkennen sind, kann der Test als erfolgreich beurteilt werden.

Für die Durchführung des Testverfahrens gibt es verschiedene Möglichkeiten. Eine simple Variante besteht darin, die Messdaten vom Mikrocontroller zum Computer zu transportieren und dort weiter zu verarbeiten. Als zukunftsfähige Option bietet sich hier das Modell Raspberry Pi 4b als Rechner und externer Datenspeicher an. Die gespeicherten, gemessenen Daten können dabei über den UART-Port vom μC transportiert und von einem Operationsfile (tty) in ein anderes File (CSV bzw.

TXT) eingelesen werden. Nachdem die Daten in einem gängigen Format vorliegen, können sie mithilfe von verschiedenen Programmiersprachen und -tools bearbeitet werden. Hierfür bietet sich insbesondere die Programmiersprache Python an, da sie über zahlreiche Bibliotheken verfügt, die die Datenverarbeitung und -visualisierung erleichtern, wie beispielsweise Matplotlib, Numpy und Pandas. Ein weiterer Vorteil von Python besteht darin, dass die Skripte plattformübergreifend einsetzbar sind.

Darüber hinaus besteht das Testschema aus vier einfache Schritten:

1. Übertragung der gemessenen Daten über UART
2. Einlesen und Aufbereiten der Daten mittels Bash-Kommandos
3. Bearbeitung und Visualisierung der Daten mittels Python
4. Vergleich der entstandenen Plots mit dem Signal auf dem Oszilloskop

Die Übertragung der Messdaten erfolgt mittels UART-Kommunikation, die Teil der HAL-Bibliothek ist. Es ist jedoch wichtig, die Baudrate⁵ zu berücksichtigen, da in unserem Fall die Datenübertragung mit einer Baudrate von 115200 erfolgt. Zur Durchführung der UART-Kommunikation werden folgende Funktionen verwendet (siehe Quellcode A.6): Nach Aufruf dieser Funktionen wird ein Datenstrom mit Schlüsselwörtern erzeugt, die den neuen Datensatz kennzeichnen. Aufgrund der Übertragung großer Datenmengen kann die Datenübertragung eine längere Zeit in Anspruch nehmen (ca. 2 Minuten). Daher sollten diese Funktionen nur während der Debugging-Phase verwendet werden. Um diesen Datenstrom in eine Datei einlesen zu können, werden die Bash-Befehle *stty* und *tee* verwendet. Der *stty*-Befehl wird verwendet, um die Baudrate des Linux-Terminals einzustellen, um das Einlesen vom *tty*-Port zu ermöglichen. Der *tee*-Befehl ermöglicht die Kopie des Datenstroms vom *tty*-Port in ein übliches .txt-Dateiformat. Die Verwendung der genannten Befehle ist im Quelltext B.1 exemplarisch aufgeführt.

Ein Beispiel des erzeugten Datenstroms, siehe unten:

```
Analog:
0, 45701
...
72275, 1243
Digital:
0, 1
...
72275, 0
Measurements:
htime = 482 ms; pulses = 12
bad state = 1; vibr = 208 ms
```

Nachdem die Messdaten übertragen und gespeichert wurden, erfolgt die Auswertung mittels Python mithilfe der Bibliotheken *numpy*, *pandas* und *matplotlib*. Die Bibliothek *numpy* stellt mathematische Funktionen zur Verfügung, die für die Verarbeitung von Messdaten notwendig sind. Mit

⁵1 Baud ist die Geschwindigkeit, wenn 1 Symbol pro Sekunde übertragen wird. Jedes Symbol entspricht einer definierten, messbaren Signaländerung im physischen Übertragungsmedium. Die Baudrate einer Datenübertragung muss auf Sende- und Empfangsseite gleich sein.

pandas können die Daten in Form von Dataframes organisiert und manipuliert werden. Die Bibliothek *matplotlib* wird für die Visualisierung der Daten eingesetzt und ermöglicht die Erstellung von Diagrammen und Graphen zur weiteren Analyse der Daten.

Während der Datenverarbeitung traten viele Herausforderungen auf, die auf die unterschiedlichen Datenquellen zurückzuführen sind. Einerseits wurden die Messdaten des Oszilloskops als Referenz verwendet, andererseits wurden die Messsignale des Mikrocontrollers aufgezeichnet. Beide Quellen haben sehr unterschiedliche technische Eigenschaften, wodurch unterschiedlich große Datenmengen entstanden sind, die angepasst werden mussten, um eine sinnvolle Visualisierung zu ermöglichen.

Das Oszilloskop hat eine Auflösung von 10 Bit und eine Abtastfrequenz von 200 MHz. Dabei wird das Hilfssignal am Kanal 1 gemessen und das Messsignal des Prüflings am Kanal 2. Während der Messung erzeugt das Oszilloskop ein Datenpaket mit 20.000 Zeilen und 3 Spalten. Die erste Spalte stellt die Zeit in Sekunden dar, während die beiden anderen Spalten für jeden Kanal reserviert sind. Anhand dieser Daten kann das Messsignal visualisiert werden. Die Triggerung des Oszilloskops ist auf eine steigende Flanke des ersten Kanals eingestellt. Abbildung 5.9 zeigt die Visualisierung der vom Oszilloskop generierten Messdaten. Das gelbe Signal auf dem Plot 5.9 ist ein Hilfssignal, das

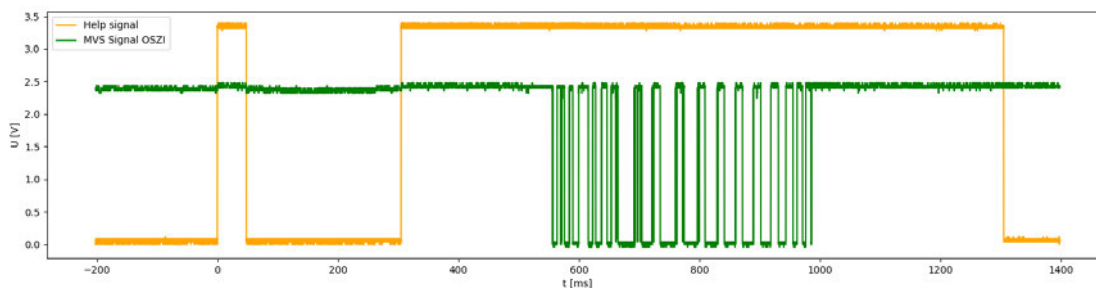


Abbildung 5.9: Visualisierung der rohen Messdaten des Oszilloskops

nur während der Messung „High“ sein muss. Allerdings ist zu beobachten, dass vor der eigentlichen Messung ein kurzer Impuls auftritt. Da die Triggerung auf die steigende Flanke dieses Kanals eingestellt ist, hat der Oszilloskop den Koordinatenursprung ebenfalls auf diese Flanke gesetzt. Der Impuls tritt immer unvermeidbar zum Start des Mikrocontrollers auf und zeichnet damit die Dauer des Initialisierungsprozesses auf. Um die Messdaten des Oszilloskops mit den Messdaten des Mikrocontrollers vergleichen zu können, muss das Messsignal, während der eigentlichen Messung (ein sekundenlanger Vorgang), aus diesem Signal extrahiert werden. Eine mögliche Lösung besteht darin, alle Messwerte innerhalb des größeren Rechtecks zu filtern. Die Abbildung 5.10 illustriert die erfolgreiche Filterung der Messdaten des Oszilloskops und die Visualisierung der Messdaten des Mikrocontrollers. Die Messeigenschaften des Mikrocontrollers umfassen eine Abtastrate von 32 MHz und eine Auflösung des Analog-Digital-Wandlers von 16 Bit. Bei einer Taktfrequenz von 96 MHz werden damit etwa 72.000 Messwerte pro Sekunde erzeugt. Obwohl die beiden Signale auf den ersten Blick fast identisch erscheinen, ist eine leichte zeitliche Verschiebung des Messsignals des Mikrocontrollers erkennbar. Um diese Verschiebung zu berechnen und auszugleichen, wurden die Zeitpunkte der ersten Impulse beider Signale voneinander subtrahiert. Anschließend wurde das Signal des Mikrocontrollers um den berechneten Betrag auf der Abszisse verschoben. Dadurch wurden die beiden Signale synchronisiert und können nun präzise verglichen werden. Es wird angenommen, dass die Verzögerung des roten Signals durch die parasitäre Kapazität innerhalb der Messschaltung hervorgerufen wurde. Die Abbildung 5.11 visualisiert die Überlagerung des gefilterten Messsignals

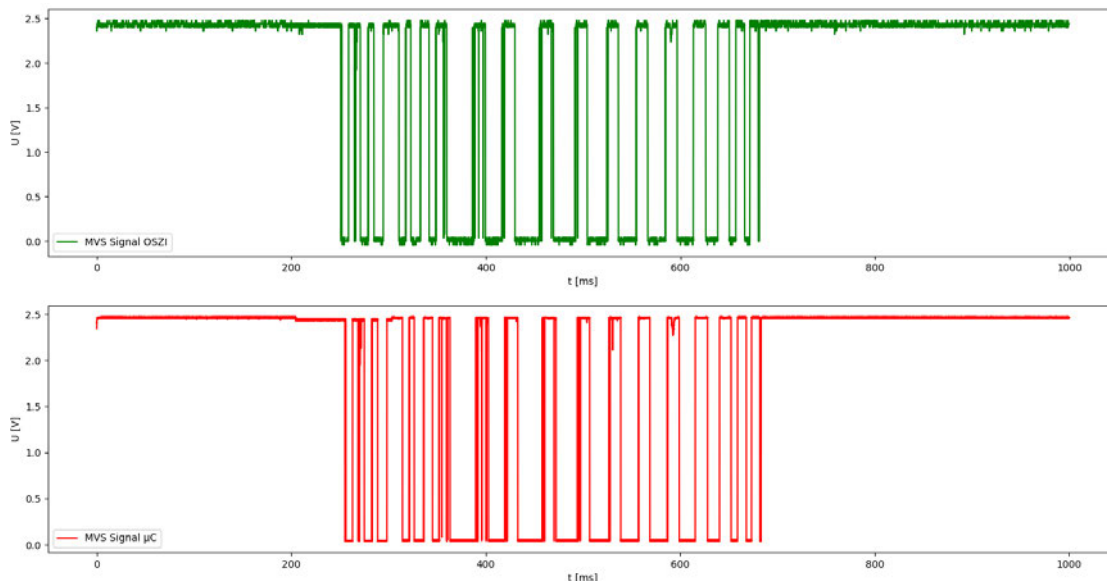


Abbildung 5.10: Filterung der Messdaten des Oszilloskops und Visualisierung der Messdaten des Mikrocontrollers

des Oszilloskops und des zeitlich angepassten Messsignals des Mikrocontrollers. Die Abbildung

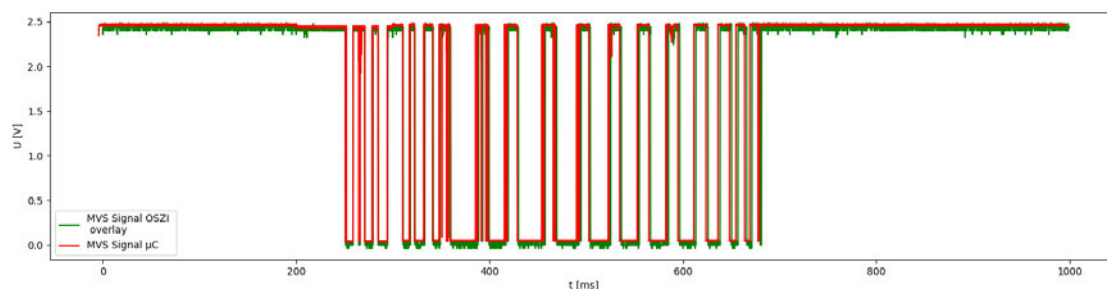


Abbildung 5.11: Filterung der Messdaten des Oszilloskops und Visualisierung der Messdaten des Mikrocontrollers

zeigt, dass die Messergebnisse des gefilterten Messsignals des Oszilloskops und des zeitlich angepassten Messsignals des Mikrocontrollers trotz unterschiedlicher Quellen übereinstimmen. Dies bestätigt die korrekte Funktionsweise der Messfunktion *measure()*.

Um die Funktion *analyse()* zu prüfen, wurde das diskretisierte Messsignal des μC visualisiert, da anhand dieses Signals die Zustandsvariable *bad* und die Anzahl der Impulse n_p bestimmt werden. In **Abbildung 5.12** wird das diskretisierte Messsignal mit zusätzlichen Informationen dargestellt, nämlich den ausgerechneten Werten für $n_p = 52$ und $t_h = 756$ ms, den Auswertungszustand $bad = 0$ (MVS ist als gut markiert) und den Zeitpunkt des Beginns der Anregung $vibr = 204$ ms. Die Ergebnisse der

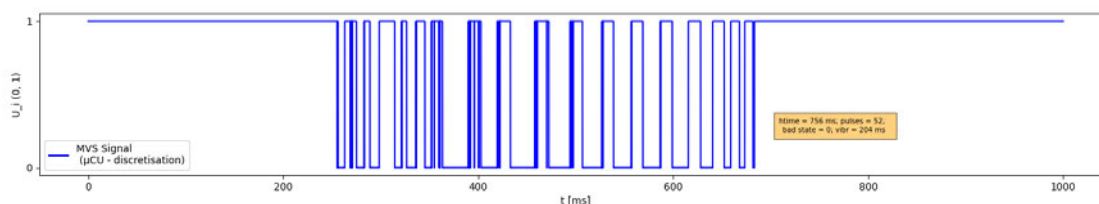


Abbildung 5.12: Visualisierung des diskretisierten Messsignals

Funktion *analyse()* werden durch die **Abbildung 5.12** veranschaulicht. Das diskretisierte Messsignal

des μC zeigt deutlich, dass der Prüfling niederohmig in seinem Ruhezustand ist, was zu einem Zustand „0“ bzw. „false“ der Zustandsvariable *bad* führt. Weiterhin ist zu erkennen, dass der Prüfling eine ausreichend hohe Empfindlichkeit besitzt, da er mehr als 10 Impulse pro Messung erzeugt hat. Diese Ergebnisse bestätigen die Funktionalität der *analyse()* Funktion und ihre Fähigkeit, die Zustände des Prüflings korrekt zu bestimmen.

6 Fazit und Ausblick

Die vorliegende Forschungsarbeit hat erfolgreich ihre Ziele erreicht und die gestellten Fragen beantwortet, die für die Automatisierung des Auswertungsprozesses von entscheidender Bedeutung waren. Im Rahmen dieser Arbeit wurden qualitativ hoch- und minderwertige MVS definiert und die Möglichkeit einer Qualitätsklassifizierung auf Basis bekannter Messdaten untersucht. Ein Sensor von hoher Qualität zeichnet sich aus, durch eine niederohmige Ruhezustandscharakteristik des MVS, eine gute Sensitivität des MVS, die durch eine angemessene Anzahl von Pulsen pro Anregungszeit erreicht wird und eine Fehlerwahrscheinlichkeit bei der einzelnen MVS, die kleiner als 10% ist. Die vorläufigen Ergebnisse haben gezeigt, dass eine eindeutige Klassifizierung der MVS anhand der verfügbaren Messdaten derzeit nicht möglich ist.

Darüber hinaus trug die Untersuchung dazu bei, den Auswertungsprozess tief zu verstehen, was die Konzipierung eines Einbaumoduls für Messung und Auswertung von MVS ermöglichte. Während des Aufbaus des Prototyps traten mehrere Herausforderungen auf, die erfolgreich bewältigt wurden. Dazu zählen unter anderem eine sichere und zuverlässige elektrische Kontaktierung zwischen den Messspitzen und dem Prüfling, eine präzise Rotation der Prüfstation und geeignete Anregungsquelle, eine kompakte Konstruktion, die Auswahl der passenden Hardware, die Entwicklung und das Design der elektrischen Schaltungen mittels Eagle, die Implementation des entworfenen Algorithmus in der Programmiersprache C und die Erstellung des Testskripts für das Debugging in der Programmiersprache Python.

Der Prototyp hat gezeigt, dass die erzeugte Saugdruckluft genug stark war, um die sichere Kontaktierung zwischen Messspitzen und dem Prüfling zu gewährleisten, die entworfene elektrische Schaltungen fehlerfrei arbeiteten und der entwickelte Sortieralgorithmus erfolgreich implementiert wurde. Zusammenfassend basierend auf Ergebnissen dieser Bachelorarbeit kann die Automatisierung des Auswerteprozesses von MVS weitergeführt werden, mit einer positiven Erwartung über der reibungslosen Integration in den Produktionsprozess des Unternehmens B&B-Gruppe.

Im Folgenden werden die nächsten Schritte und mögliche Verbesserungen aufgeführt, um das Auswertesystem für die Gurtmaschine weiter zu verbessern. Zunächst ist es notwendig, den verfügbaren Platz für die Installation des Systems zu vermessen. Anschließend kann das Gehäuse für das Auswertesystem entworfen werden, um es optimal in die Gurtmaschine integrieren zu können. Nach der Integration des multifunktionalen Moduls in die Gurtmaschine ist ein weiterer Test notwendig, um sicherzustellen, dass das System ordnungsgemäß funktioniert. Es ist geplant, einen zweiten Steuerungsmodus zu programmieren, um eine manuelle Steuerung des Systems zu ermöglichen. Als externer Datenspeicher kann der SOC – Raspberry Pi 4 Model B genutzt werden, für den ein kleines Bash-Skript erstellt werden soll, um die Filestruktur zu organisieren. Weitere mögliche Verbesserungen umfassen die Programmierung einer kleinen grafischen Benutzeroberfläche (GUI) auf Python, die auf dem Raspberry Pi ausgeführt werden kann, um Konfigurationen auf die Messungen anzuwenden.

Anhang A: Implementation der Funktionen in der Programmiersprache C

Quelltext A.1: Erzeugung einer Datenstruktur

```
struct MeasureData {
    uint16_t signal[100000];
    int t_vibr_start;
    int t_end;
    int pulses;
    int htime;
    bool bad;
} Meas;
```

Quelltext A.2: Konfiguration des Analog-Digital-Wandlers und Timers

```
//ADC conf.
ADC_HandleTypeDef hadc1;
hadc1.Instance = ADC1;
hadc1.Init.ClockPrescaler = ADC_CLOCK_ASYNC_DIV1;
hadc1.Init.Resolution = ADC_RESOLUTION_16B;
//Timer 16 conf.
TIM_HandleTypeDef htim16;
htim16.Instance = TIM16;
htim16.Init.Prescaler = 9600 - 1;
htim16.Init.CounterMode = TIM_COUNTERMODE_UP;
```

Quelltext A.3: Definition der Messfunktion

```
void measure(struct MeasureData *s) {
    const int ONESEC = 10000 - 1;
    // store our counter value
    int tim_val_ms = 0; // tick is 0.1 ms

    bool is_vibrating = false;
    // store our 16bit ADC reading
    uint16_t raw;
    // starting timer 16
    HAL_TIM_Base_Start(&htim16);
    // set help signal (yellow led)
    HAL_GPIO_WritePin(GPIOG, GPIO_PIN_9, GPIO_PIN_SET);

    for (int i = 0; tim_val_ms < ONESEC; i++) {

        if (!is_vibrating && tim_val_ms >= ONESEC * 0.2 &&
            tim_val_ms < ONESEC * 0.3) {
            HAL_GPIO_WritePin(GPIOG, GPIO_PIN_12, GPIO_PIN_SET); // vibrator ON
            // saving time of begin vibration
            s->t_vibr_start = i;
            is_vibrating = true;
        }
    }
}
```

```

} else if (is_vibrating && tim_val_ms >= ONESEC * 0.3) {
    HAL_GPIO_WritePin(GPIOD, GPIO_PIN_12, GPIO_PIN_RESET); // vibrator OFF
    is_vibrating = false;
}

// ADC routine
HAL_ADC_Start(&hadc1);
HAL_ADC_PollForConversion(&hadc1, HAL_MAX_DELAY);
raw = HAL_ADC_GetValue(&hadc1);
// needed to save the last iterator i
s->t_end = i;
s->signal[i] = raw;
tim_val_ms = __HAL_TIM_GET_COUNTER(&htim16);
}
// reset help signal (yellow led)
HAL_GPIO_WritePin(GPIOD, GPIO_PIN_9, GPIO_PIN_RESET);
// stoping and resetting timer 16
HAL_TIM_Base_Stop_IT(&htim16);
__HAL_TIM_SET_COUNTER(&htim16, 0);
}

```

Quelltext A.4: Definition der Analysefunktion

```

void analyse(struct MeasureData *s) {
    // digitalize signal v:[0,1]
    const int TRIGGER = 26589; // approx. 1V
    const int SCALE = 1000;

    for (int i = 0; i < s->t_end; i++) {
        if (s->signal[i] < TRIGGER) {
            s->signal[i] = 0;
        } else {
            s->signal[i] = 1;
        }
    }
    // count pulses
    for (int i = 0; i < s->t_end; i++) {
        int edge = s->signal[i + 1] - s->signal[i];
        if (edge == 1) { // count only negative pulses
            s->pulses++;
        }
        if (s->signal[i] == 1) {
            s->htime++;
        }
    }
    // check if sensor has been open before vibration
    s->bad = false;
    for (int i = 0; i <= s->t_vibr_start; i++) {
        if (s->signal[i] == 0) {
            s->bad = true;
            break;
        }
    }
}

```

```

}
// check if sensor has been open after vibration (after 750 ms)
for (int i = (int)(s->t_end * 0.75); i < s->t_end; i++) {
    if (s->signal[i] == 0) {
        s->bad = true;
        break;
    }
}
}
// resolve the variables to human readable values
s->htime = s->htime * SCALE / s->t_end;
s->t_vibr_start = s->t_vibr_start * SCALE / s->t_end;
}

```

Quelltext A.5: Implementation des kompletten Algorithmus

```

// stepper init
STEPPERS_Init_TMR(&htim17);
STEPPER_SetSpeed(STEPPER_MOTOR1, 14); // set RPM
// pump on
HAL_GPIO_WritePin(GPIOA, GPIO_PIN_5, GPIO_PIN_SET);
HAL_Delay(1000);
measure(&Meas);
analyse(&Meas);
uart_transmit_info();

if (Meas.bad == false && Meas.pulses >= SENSITIVITY) {
    // rotate 180
    STEPPER_Step_NonBlocking(STEPPER_MOTOR1, PI, DIR_CW);
    HAL_Delay(1500);
    measure(&Meas);
    analyse(&Meas);
    uart_transmit_info();

    if (Meas.bad == false && Meas.pulses >= SENSITIVITY) {
        // rotate -45 and reset pump
        STEPPER_Step_NonBlocking(STEPPER_MOTOR1, PI / 4, DIR_CCW);
        HAL_Delay(1500);
        HAL_GPIO_WritePin(GPIOA, GPIO_PIN_5, GPIO_PIN_RESET);
        HAL_Delay(1000);
        // rotate to 0
        STEPPER_Step_NonBlocking(STEPPER_MOTOR1, PI * 3 / 4, DIR_CCW);
    } else {
        // rotate +45 and reset pump
        STEPPER_Step_NonBlocking(STEPPER_MOTOR1, PI / 4, DIR_CW);
        HAL_Delay(1500);
        HAL_GPIO_WritePin(GPIOA, GPIO_PIN_5, GPIO_PIN_RESET);
        HAL_Delay(1000);
        // rotate to 0
        STEPPER_Step_NonBlocking(STEPPER_MOTOR1, PI * 3 / 4, DIR_CW);
    }
} else {
    // rotate 180

```

```

STEPPER_Step_NonBlocking(STEPPER_MOTOR1, PI, DIR_CW);
HAL_Delay(1500);
// rotate +45 and reset pump
STEPPER_Step_NonBlocking(STEPPER_MOTOR1, PI / 4, DIR_CW);
HAL_Delay(1500);
HAL_GPIO_WritePin(GPIOA, GPIO_PIN_5, GPIO_PIN_RESET);
HAL_Delay(1000);
// rotate to 0
STEPPER_Step_NonBlocking(STEPPER_MOTOR1, PI * 3 / 4, DIR_CW);
}

```

Quelltext A.6: UART-Übertragung

```

void uart_transmit_analog(void) {
    /* Transmit an array with ANALOG data via uart */
    for (int j = 0; j < Meas.t_end; j++) {
        uart_buf_len = sprintf(uart_buf, "%d, %u \r\n", j, Meas.signal[j]);
        HAL_UART_Transmit(&huart3, (uint8_t *)uart_buf, uart_buf_len, 100);
    }
}

void uart_transmit_digital(void) {
    /* Transmit an array with DIGITAL data via uart */
    uart_buf_len = sprintf(uart_buf, "Digital \n");
    HAL_UART_Transmit(&huart3, (uint8_t *)uart_buf, uart_buf_len, 100);
    for (int j = 0; j < Meas.t_end; j++) {
        uart_buf_len = sprintf(uart_buf, "%d, %u \r\n", j, Meas.signal[j]);
        HAL_UART_Transmit(&huart3, (uint8_t *)uart_buf, uart_buf_len, 100);
    }
}

void uart_transmit_info(void) {
    /* Transmit extra measurement data via uart */
    uart_buf_len = sprintf(uart_buf, "Measurements \n");
    HAL_UART_Transmit(&huart3, (uint8_t *)uart_buf, uart_buf_len, 100);
    uart_buf_len = sprintf(uart_buf, "htime = %d ms; pulses = %d; \r\n",
        Meas.htime, Meas.pulses);
    HAL_UART_Transmit(&huart3, (uint8_t *)uart_buf, uart_buf_len, 100);
    uart_buf_len = sprintf(uart_buf, "bad state = %d; vibr = %d ms \r\n",
        Meas.bad, Meas.t_vibr_start);
    HAL_UART_Transmit(&huart3, (uint8_t *)uart_buf, uart_buf_len, 100);
}

```

Anhang B: Bash-Befehle

Quelltext B.1: Bash Befehle

```
stty 115200 -F /dev/ttyACM0 raw -echo  
echo "data saved in tmp_data.txt" && cat /dev/ttyACM0 | tee tmp_data.txt
```

Literaturverzeichnis

- [1] 28BYJ-48 – 5V Stepper Motor. Kiatronics.
- [2] Jens Benecke. „Modellierung der Hochfrequenz-Eigenschaften kleiner Gleichstrommotoren“. In: (2009).
- [3] Thomas Blank. *Funktionale Schichten in Leiterplatten Entwicklung von intelligenten Mikrovi-
brationssensoren in mehrdimensionaler Leiterplattentechnologie*. ger. Institut für Prozessda-
tenverarbeitung und Elektronik (IPE), 2013.
- [4] *Blistergurtmaschinen*. <http://weresch-automat.de/sgm101auto.htm>. Accessed: 2023-
01-30.
- [5] G Boothroyd. „Automatic handling of small parts“. In: *Annals of CIRP* 24.1 (1975), S. 393–401.
- [6] *DC 12V Micro Vacuum Pump*. MAGT.
- [7] M. Fathi, P. Keshavarz und M. R. Rahimpour. „An intelligent control approach for a vibratory
bowl feeder: A model-based optimization“. In: *Journal of Intelligent Manufacturing* 30.4 (2019),
S. 1431–1444.
- [8] *HIGH VOLTAGE, HIGH CURRENT DARLINGTON TRANSISTOR ARRAYS*. DS35313. Rev.
7-2. Diodes Incorporated. Jan. 2017.
- [9] *high-quality vibration motor module*. AZDelivery.
- [10] *Introduction to Motor Control with STM32*. URL: [https://wiki.st.com/stm32mcu/wiki/
STM32MotorControl:Introduction_to_Motor_Control_with_STM32](https://wiki.st.com/stm32mcu/wiki/STM32MotorControl:Introduction_to_Motor_Control_with_STM32).
- [11] *Micro Vibration Sensor MVS0409.02*. MVS0409.02. Rev. 2.6. B&B-Gruppe. März 2020.
- [12] Rainer Parthier. *Messtechnik*. ger. 10. Aufl. 2022. ISBN: 978-3-658-37970-4.
- [13] Sheldon M. Ross. *Statistik für Ingenieure und Naturwissenschaftler*. eng. 3. Aufl. 2006. ISBN:
9783827416216.
- [14] R Scheithauer. „Signale und Systeme“. In: (2005).
- [15] *STM32 Nucleo-144 boards*. DB3171. Rev. 16. ST life.augmented. Feb. 2023.
- [16] Hans-Dieter Stölting und Achim Beisse. „Schrittmotoren“. In: *Elektrische Kleinmaschinen: Eine
Einführung*. Wiesbaden: Vieweg+Teubner Verlag, 1987, S. 189–213. ISBN: 978-3-663-01104-0.
DOI: [10.1007/978-3-663-01104-0_7](https://doi.org/10.1007/978-3-663-01104-0_7). URL: [https://doi.org/10.1007/978-3-663-
01104-0_7](https://doi.org/10.1007/978-3-663-01104-0_7).

Eidesstattliche Erklärung

Hiermit versichere ich – Shyngys Muratbek – an Eides statt, dass ich die vorliegende Arbeit selbstständig und nur unter Verwendung der angegebenen Quellen und Hilfsmittel angefertigt habe.

Sämtliche Stellen der Arbeit, die im Wortlaut oder dem Sinn nach Publikationen oder Vorträgen anderer Autoren entnommen sind, habe ich als solche kenntlich gemacht.

Diese Arbeit wurde in gleicher oder ähnlicher Form noch keiner anderen Prüfungsbehörde vorgelegt oder anderweitig veröffentlicht.

Mittweida, 26. Mai 2023



Ort, Datum

Shyngys Muratbek, Ing.