

Bachelorarbeit

Experimentelle Verifizierung optischer Velocimeter mit LabVIEW

Zur Erlangung des akademischen Grades

Bachelor of Engineering (B. Eng.)

Sommersemester 2016

Autor:	Tobias Kanigowski
Matrikelnummer:	012202109
Studiengang:	Elektrotechnik
Hochschule:	Hochschule Bochum
Erstprüfer:	Prof. Dr.-Ing. Arno Bergmann
Zweitprüfer:	Dipl. Ing. Thorsten Bartsch

Eidesstattliche Erklärung

„Ich versichere, dass ich die Arbeit selbständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt, sowie Zitate kenntlich gemacht habe und die Regelungen der geltenden Prüfungsordnung zu Versäumnis, Rücktritt, Täuschung und Ordnungsverstoß zur Kenntnis genommen habe.“

Bochum, den _____

(Tobias Kanigowski)

Inhaltsverzeichnis

1	EINLEITUNG	4
2	ZIELE DES PROJEKTS	6
3	GRUNDLAGEN	8
3.1	SENSOREN ZUR LÄNGENMESSUNG VON BEWEGTEN OBERFLÄCHEN	8
3.2	LABVIEW	12
3.3	MESSABWEICHUNG	12
3.4	TESTSTAND.....	15
3.4.1	<i>Aufbau</i>	16
3.4.2	<i>Referenz</i>	16
3.4.3	<i>Steuerung</i>	17
3.4.4	<i>Bedienung des Teststandes</i>	18
4	UMSETZUNG DER DATENAUSWERTUNG	20
4.1	AUFSCHLÜSSELN DER MESSDATEN	20
4.2	AUFBAU DER BENUTZEROBERFLÄCHE.....	29
4.3	VERGLEICH DER GESAMTEN FAHRSTRECKE	31
4.4	VERGLEICH DER BESCHLEUNIGUNGS- UND BREMSVORGÄNGE.....	35
4.5	VERGLEICH DER FAHRT MIT KONSTANTER GESCHWINDIGKEIT	38
4.6	CSV & EXCEL PROTOKOLL.....	41
4.6.1	<i>Umsetzung CSV</i>	41
4.6.2	<i>Umsetzung Excel Protokoll</i>	43
5	ECHTZEITFÄHIGES SYSTEM	45
5.1	HARDWARE & INBETRIEBNAHME	45
5.2	STRUKTUR.....	45
5.3	KOMMUNIKATION	46
5.4	ANPASSUNG DES SYSTEMS.....	48
6	MÖGLICHE WEITERENTWICKLUNGEN	50
6.1	EINBINDUNG EINES ABSOLUTEN WEGGEBERS	50
6.2	EINBINDUNG EINES FRAMEGRABBERS	50

6.3	EINBINDUNG EINER HÖHENVERSTELLUNG.....	51
6.4	EINSATZ EINER LEISTUNGSSTÄRKEREN CPU.....	51
6.5	AUTOMATISCHE ERKENNUNG DES BEWEGUNGSZUSTANDES	51
I	ABKÜRZUNGSVERZEICHNIS	53
II	ABBILDUNGS- UND TABELLENVERZEICHNIS	54
III	QUELLENVERZEICHNIS	56
IV	ANHANG	60

Vorwort

Dieser Arbeit geht eine Praxisphase voraus, die im Labor für elektrische Antriebe der Hochschule Bochum absolviert wurde. Diese Praxisphase umfasste ein Projekt zur Weiterentwicklung eines Teststandes. Ziel dieses Teststandes ist die Verifikation von optischen Geschwindigkeitssensoren.

1 Einleitung

Die Ermittlung von Geschwindigkeiten spielt in der Industrie eine zentrale Rolle. Sensoren mit dieser Aufgabe kommen zum Beispiel bei der Prozesssteuerung und -überwachung zum Einsatz. Auch Längenfortschritte lassen sich aus den gemessenen Geschwindigkeiten ermitteln. [1]

Die präzise Ermittlung der Länge ist beispielweise bei der Ablängung von Materialien ein entscheidender Faktor, hier hat sich eine typische Messabweichung von weniger als 0,05% etabliert [2]. Je nach Einsatzgebiet unterscheiden sich die Anforderungen, die an Geschwindigkeitssensoren gestellt werden. Diese Anforderungen können unter anderem die Messgenauigkeit oder Zuverlässigkeit des Sensors betreffen, aber auch Umgebungseinflüsse oder Wirtschaftlichkeit berücksichtigen [3]. Der Nachweis der Erfüllung dieser Anforderungen an einen Sensor wird auch Verifikation genannt. Die ISO 9000:2005, Abschnitt 3.8.4 definiert die Verifikation als „Bestätigung durch einen objektiven Nachweis, dass Anforderungen erfüllt werden“. Die Verifikation der Sensoren ist somit von besonderer Bedeutung für die Gewährleistung der Qualität von Prozessen und Produkten.

Mittels einer Testumgebung werden die Anforderungen an die Messgenauigkeit der Sensoren verifiziert. Der objektive Nachweis wird hierbei durch Testresultate erbracht. Gegenstand dieser Arbeit ist eine Testumgebung für optische Geschwindigkeitssensoren (Velocimeter, siehe Kapitel 3.1). Dabei handelt es sich um einen Linearteststand, an dem reversierende Oberflächen dargestellt werden (Abbildung 1-1).



Abbildung 1-1: Teststand für optische Geschwindigkeitssensoren

Um die Verifikation der Messgeräte an diesem Teststand zu optimieren, wurde im Rahmen dieser Arbeit eine Funktion zur Analyse der Messergebnisse entwickelt und implementiert. In den folgenden Kapiteln wird ein Überblick über die Ziele des Projektes gegeben. Nach einer Zusammenfassung der technischen Grundlagen, wird die Umsetzung der Weiterentwicklung ausführlicher beschrieben. Abschließend folgt eine Sammlung von Ideen zur Weiterentwicklung des Teststandes.

2 Ziele des Projekts

Im Fokus dieser Arbeit steht die Weiterentwicklung eines industriellen linearen Prüfstandes für verschiedene kontaktlose Messsysteme zur Geschwindigkeits- und Längenerfassung. Ziel des Projektes ist unter anderem die Implementierung einer Messdatenauswertung in der Software des Teststandes. Weitere Ziele wurden zuvor in einem Projektauftrag sowie in einem Lastenheft in Form von Anforderungen (Anf.) festgehalten [4]. In den folgenden Kapiteln werden die Anforderungen an die Weiterentwicklung und deren Umsetzung beschrieben.

Um die Weiterentwicklung erkennbar zu machen, folgt zunächst eine Beschreibung des Ausgangszustandes. Der Zustand des Systems zu Beginn des Projekts resultierte aus der Umstellung der Steuerung auf ein System von National Instruments (NI). Diese Umstellung ist in *Gerhmann 2016* „Industrielles Testsystem für optische Velocimeter“ näher beschrieben [5].

Die im Zuge der Umstellung geschriebene Software wurde in LabVIEW implementiert und beinhaltete eine Benutzeroberfläche (Graphical User Interface; GUI). Sie umfasste das Steuern des Teststands und die Messdatenerfassung. Über die GUI wurden die verschiedenen Fahrprogramme (siehe Kapitel 3.4.4) gesteuert. Das Programm lief auf einem PC und kommunizierte über eine Peripheral-Component-Interconnect (PCI) Express und einer PXIe-8360 Karte im NI Chassis mit den weiteren NI Modulen.

Da die Messdaten bisher lediglich in eine Datei geschrieben wurden, wird nun innerhalb der GUI bereits eine Auswertung der Daten erfolgen. Durch diese Auswertung hat der Nutzer die Möglichkeit, die Funktionsfähigkeit des Sensors zu beurteilen. Die Auswertung erfolgt in einem zusätzlichen Reiter in der GUI (Anf_01). Um die Speicherung der Auswertungen zu ermöglichen, ist diese innerhalb eines Excel-Protokolls (Anf_02) oder einer CSV Datei (Anf_03) darstellbar. Um die Eigenschaften des Sensors in unterschiedlichen Fahrsituationen zu untersuchen, werden die Messdaten in Bewegungsabläufe aufgeteilt und verglichen. Daraus folgt zum einen ein Vergleich der gesamten Fahrstrecke (Anf_04), in dem die Messwerte der absoluten Gesamtstrecke vom untersuchten Sensor und des Referenzsystems verglichen werden. Weiterhin werden die

Beschleunigungs- und Bremsphasen (Anf_5) und die Fahrt mit konstanter Geschwindigkeit (Anf_6) untersucht. Hierzu werden ebenfalls die zurückgelegten Strecken miteinander verglichen.

Zudem wird das komplette System auf einen echtzeitfähigen Controller von NI umgestellt (Anf_08) und ein zweiter Sensor zur Unterstützung des Referenzsystems eingebracht. [4]

3 Grundlagen

3.1 Sensoren zur Längenmessung von bewegten Oberflächen

Sensoren zur optischen Längenmessung werden in Bereichen eingesetzt, in denen eine berührungsfreie Längenmessung notwendig bzw. sinnvoll ist. Sie dienen zur Ermittlung von Längen, Geschwindigkeiten oder Beschleunigungen bewegter Objekte [1].

Ein Anwendungsfall wäre eine Walzstraße in der Stahlverarbeitenden Industrie, in der empfindliche und heiße Materialoberflächen eine kontaktfreie Messung erfordern. Weitere Einsatzmöglichkeiten finden sich in Längen- und Vorschubmessungen in der Stahl-, Rohr- und Papierproduktion oder in anderen heißen und kalten Fertigungsprozessen der Kunststoff-, Textil- oder Kabelindustrie [6]. Ein Vorteil gegenüber taktilen Systemen liegt in der schlupffreien Messung [1]. Zu den Nachteilen zählt die hohe Empfindlichkeit gegenüber Verschmutzungen [7].

Für die Umsetzung dieser Messsysteme haben sich verschiedene Methoden etabliert. Zwei dieser Methoden sind das Korrelations- und das Ortsfrequenzfilterverfahren. Im Verlauf werden diese Verfahren in den Grundlagen skizziert.

Die Geschwindigkeitsmessung mittels **Korrelationsverfahren** basiert auf der Laufzeitmessung oder Längenmessung einer bewegten Bildszene. Hier ist es notwendig, markante Merkmale dieser Bildszene zu extrahieren. [8]

Ein solcher Aufbau könnte wie folgt aussehen. Eine Kamera erfasst in einem festen zeitlichen Abstand Bilder einer bewegten Oberfläche. Zwei aufeinander folgende Bilder werden durch den Vergleich der Pixelzeilen auf gemeinsame Merkmale untersucht. Bei den Zeilen mit der maximalen Übereinstimmung wird die Wegdifferenz Δx bestimmt. Mit dem bekannten Zeitabstand Δt lässt sich die Geschwindigkeit v mit Gleichung (1) berechnen.

$$v = \frac{\Delta x}{\Delta t} \quad (1)$$

Durch den festen Zeitabstand ist die maximal messbare Geschwindigkeit, sowie durch die Auflösung der Kamera die Genauigkeit, des Systems begrenzt.

Bei der Messmethode nach dem **Ortsfrequenzfilterverfahren** wird eine bewegte Oberfläche mit ortsfesten, gitterförmigen und periodischen Strukturen überlagert. Die Geschwindigkeit wird hier aus der Frequenz des reflektierten Lichts bestimmt. Verfahren nach dem Ortsfrequenzfilterverfahren sind die Laser-Doppler-Velocimetrie und das Verfahren mit einer Gitterabtastung. [8]

Die Laser-Doppler-Velocimetrie (LDV), in der Strömungsmesstechnik auch Laser-Doppler-Anemometrie (LDA) genannt, ist ein optisches Messverfahren [8], bei dem die Geschwindigkeit einer Oberfläche durch die Messung von dopplerverschobenem Licht ermittelt wird [9]. In Bezug auf die Geschwindigkeitsmessung von Oberflächen ist auch der Begriff Laser-Surface-Velocimeter (LSV) verbreitet [10].

Das Messprinzip eines LDV lässt sich unter anderem mit dem Interferenz-Streifenmodell an einem Zweistrahl-Laser-Doppler System erklären. [11]

Das Zweistrahl-Laser-Doppler System basiert auf zwei kohärenten Laserstrahlen. Die Kohärenz wird erreicht, indem ein Laserstrahl durch einen Strahlenteiler aufgeteilt wird. Diese Laserstrahlen werden durch eine Sendeoptik auf dem Brennpunkt zur Überlagerung gebracht. Das Schnittvolumen dieser Laserstrahlen bildet das Messvolumen, in dem sich die zu messende Oberfläche befinden muss. Dies erfordert eine genaue Justierung des Sensors über der Oberfläche. Im Messvolumen entstehen durch die Überlagerung der beiden Laserstrahlen Zonen mit minimaler und maximaler Helligkeit, die Interferenzstreifen genannt werden (siehe Abbildung 3-1). [11] [8]

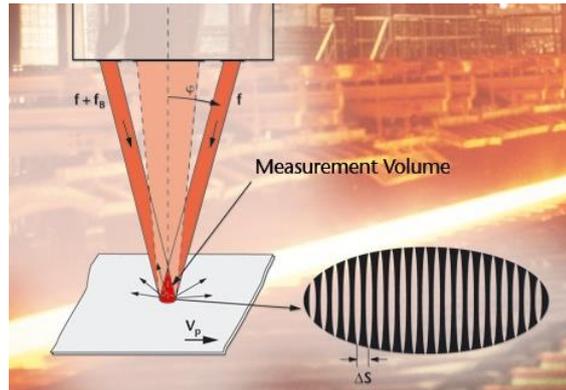


Abbildung 3-1: Funktionsprinzip eines LDV [12]

Der Abstand dieser Interferenzstreifen Δs ist abhängig von der Wellenlänge des Lasers λ und des Schnittwinkels der Laserstrahlen 2φ [4.10] (Gleichung 2) [12]:

$$\Delta s = \frac{\lambda}{2 \cdot \sin \varphi} \quad (2)$$

Bewegt sich die Oberfläche durch das Streifenmuster, wird das reflektierte Licht durch die Interferenzstreifen in seiner Intensität moduliert. Dieses Licht wird auf einen Photodetektor geführt und in ein elektrisches Signal umgewandelt. Die Frequenz f dieses Signals entspricht der Dopplerfrequenz und ist proportional zur Geschwindigkeit der Oberfläche v . Zusammen mit dem Abstand der Interferenzstreifen Δs ergibt sich die Gleichung (3) [11] [12]:

$$v = f \cdot \Delta s = \frac{f \cdot \lambda}{2 \cdot \sin \varphi} \quad (3)$$

Die Geschwindigkeit wird von einem Controller berechnet. Bei einer Längenmessung wird die Länge durch die Integration der Geschwindigkeit gebildet. Für eine genaue Messung ist die exakte Bestimmung der Wellenlänge des Laserstrahls notwendig.

Eine weitere Umsetzung des Ortsfrequenzfilterverfahrens erfolgt mit einer sogenannten Gitterabtastung. [8]

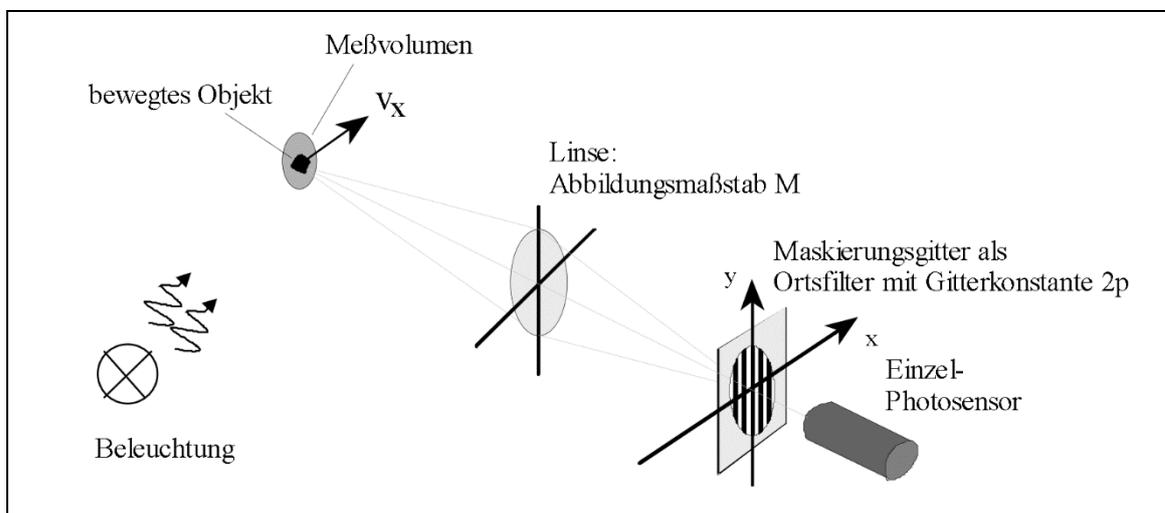


Abbildung 3-2: Prinzip zur Geschwindigkeitsmessung mit einem Maskierungsgitter [8]

Abbildung 3-2 zeigt einen möglichen Aufbau nach diesem Prinzip. Hierbei wird ein bewegtes Objekt oder eine Oberfläche von einer Lichtquelle bestrahlt. Das reflektierte Licht wird von einem Objektiv auf ein optisches Gitter (Maskierungsgitter) übertragen. Dieses Gitter besteht aus lichtdurchlässigen und -undurchlässigen Zonen. Die Zonen modulieren die Intensität des Lichts. Das optische Gitter lässt sich somit mit den Interferenzstreifen des Laser-Doppler Verfahrens vergleichen. Anschließend wandelt ein Photosensor das erfasste Licht in ein periodisches Signal um. Die Frequenz f_0 des empfangenen Signals enthält auch hier die Geschwindigkeit der Oberfläche. Sie lässt sich für Abbildung 3-2 mit Gleichung (4) beschreiben.

$$v_x = \frac{2p f_0}{M} \quad (4)$$

In dieser Gleichung lässt sich der Vorteil gegenüber einem LDV erkennen, da hier die Wellenlänge der Beleuchtungsquelle nicht relevant ist.

Das Maskierungsgitter muss nicht, wie im einfachsten Fall in Abbildung 3-2 zu sehen, als zusätzliches Bauteil im System vorhanden sein. Es kann auch als aktives, integriertes Gitter durch ein Photosensorenarray nachgebildet werden. Hierbei stellt die alternierende Gewichtung der einzelnen Photosensoren die Gitterzonen dar. [8]

3.2 LabVIEW

LabVIEW ist eine Entwicklungsumgebung der Firma National Instruments (Akronym für „**L**aboratory **V**irtual **I**nstrumentation **E**ngineering **W**orkbench“, [13]). Die graphisch orientierte Programmiersprache ähnelt einer Funktionsbausteinsprache (FBS) für Speicherprogrammierbare Steuerungen (SPS). Funktionsbausteine werden als Virtuelle Instrumente (VI) bezeichnet. Die Programmierung erfolgt in diesen VIs, die wiederum beliebig verschachtelt werden können. Daher wird bei untergeordneten VIs auch von Sub-VIs gesprochen. Ein wichtiges Element von LabVIEW ist das Frontpanel. Es ermöglicht dem Nutzer die Entwicklung einer individualisierten, anwendungsorientierten Benutzeroberfläche. [13]

3.3 Messabweichung

Zum besseren Verständnis werden in diesem Kapitel einige Definitionen der Messtechnik erläutert. Die Grundlage der späteren Auswertungen basiert auf den Messabweichungen der gemessenen Werte. Die Messabweichung ist der Unterschied zwischen dem erhaltenen und dem wahren Wert [14]. Da der wahre Wert einer Messgröße nicht bekannt sein kann, wird die Messabweichung in dem „Internationalen Wörterbuch der Metrologie“ auch als „Messwert minus eines Referenzwertes“ definiert [15]. Diese Referenzwerte dienen somit als Ersatz für den prinzipiell unbekanntem „wahren Wert“ und werden in der Praxis mit Hilfe von Messgeräten mit höherer Auflösung ermittelt [16]. Die Auflösung ist definiert, als die kleinste Änderung einer Messgröße, die in der entsprechenden Anzeige eine merkliche Änderung verursacht [15]. Innerhalb dieser Arbeit werden folgende definierte Abweichungen berechnet.

Die **Absolute Messabweichung** F ist die Differenz aus dem Messwert x_M und dem Referenzwert x_R (Gleichung (5)).

$$F = x_M - x_R \quad (5)$$

Die **relative Messabweichung** f ist das Verhältnis der Abweichung F zum Referenzwert x_R in Prozent (Gleichung (6)).

$$f = \frac{F}{x_R} \cdot 100\% = \frac{x_M - x_R}{x_R} \cdot 100\% \quad (6)$$

Der **arithmetische Mittelwert** \bar{F} der Messabweichungen ist die Summe aller Messabweichungen F_i dividiert durch die Anzahl der Messungen n (Gleichung (7)).

$$\bar{F} = \frac{1}{n} \cdot \sum_i^n F_i \quad (7)$$

Die **mittlere absolute Messabweichung** D ist der arithmetische Mittelwert aller Abweichungen der Messabweichung F_i vom Mittelwert der Messabweichungen \bar{F} (Gleichung (8)).

$$D = \frac{1}{n} \cdot \sum_i^n |F_i - \bar{F}| \quad (8)$$

Die **Standardabweichung** S ist die Wurzel aus dem arithmetischen Mittelwert der Quadrate aller Abweichungen der Messabweichung F_i vom Mittelwert der Messabweichungen \bar{F} (Gleichung (9)). Sie beschreibt die Abweichung der Einzelwerte zum Mittelwert und bezieht sich auf die Grundgesamtheit. [17]

$$s = \sqrt{\frac{1}{n} \cdot \sum_i^n (F_i - \bar{F})^2} \quad (9)$$

Die **empirische Standardabweichung** $s_{emp.}$ ist eine korrigierte Standardabweichung. Sie bezieht sich auf eine Strichprobe. (Gleichung (10)). [17]

$$s_{emp.} = \sqrt{\frac{1}{n-1} \cdot \sum_i^n (F_i - \bar{F})^2} \quad (10)$$

Die **Varianz** s^2 ist die quadratische Standardabweichung s . [17]

Die Berechnung dieser Abweichungen erfolgt mit den entsprechenden VIs aus den LabVIEW Paletten „Numerische Funktionen“ und „Wahrscheinlichkeit & Statistik“.

3.4 Teststand

Der Teststand dient der Entwicklung und Überprüfung von Velocimeter und befindet sich im Labor für elektrische Antriebe der Hochschule Bochum. Ursprünglich wurde er von der Firma GIA mbH entwickelt. Mit Hilfe des Teststandes werden optische Messsysteme für Längen- und Geschwindigkeitsmessungen untersucht. Seine Grundfunktion besteht darin, gemessene Längen einer bewegten Oberfläche mittels eines Referenzsystems zu vergleichen. Der Aufbau ist in Abbildung 3-3 skizziert.

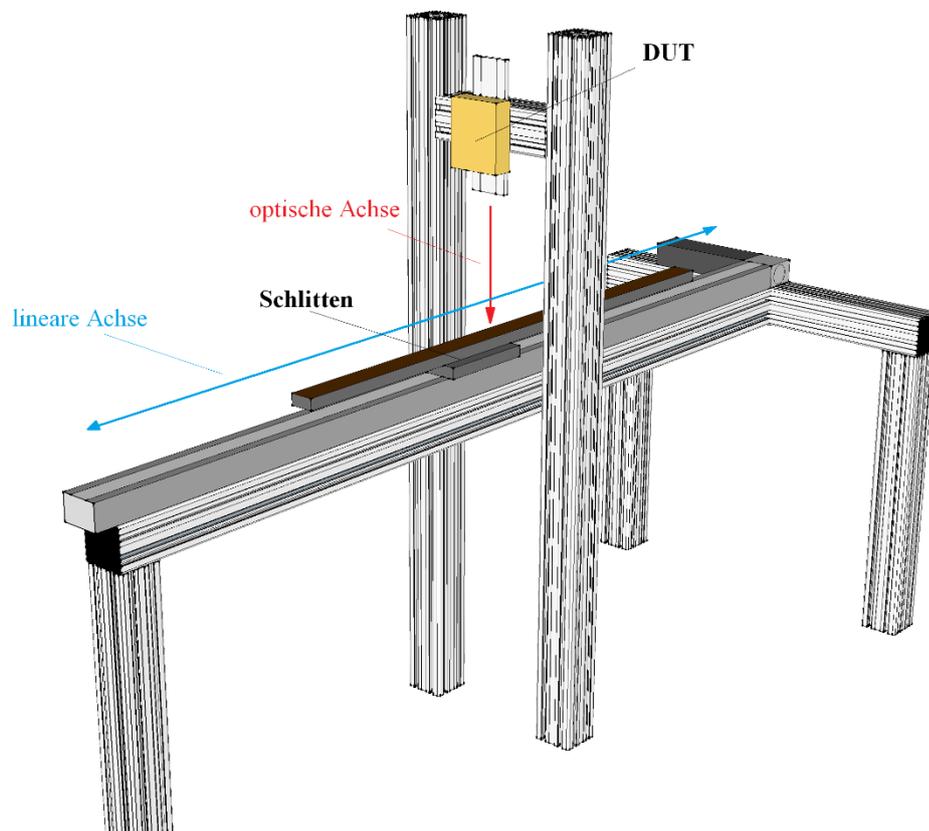


Abbildung 3-3: Skizze des Teststands mit linearer Verfahrensachse und optische Achse des DUT

3.4.1 Aufbau

Das Hauptelement des Teststandes ist ein Lineartisch. Der Lineartisch verfügt über einen Schlitten, der auf einer linearen Achse verfahren wird (Abbildung 3-3). Der Schlitten ist auf einem Führungswagen montiert, der mittels einer Kugelschienenführung auf einer Zahnriemenachse geführt wird.

Die Zahnriemenachse wird durch einen Servomotor angetrieben, dessen Ansteuerung durch einen Servoverstärker (Ecovario) erfolgt. Für die Positionierung des Führungswagens nutzt der Servoverstärker einen internen Motorencoder, sowie zwei induktive Sensoren als Endschalter und einen weiteren induktiven Sensor als Referenzpunkt in der Mitte des Lineartisches.

Auf den Schlitten können verschiedene Materialien, wie zum Beispiel Schleifpapiere, appliziert werden. Diese stellen die Oberfläche für die Messung dar. Mittig über dem Lineartisch befindet sich der „Galgen“, eine Vorrichtung zur Montage des zu vermessenden Sensors. Der Sensor, in diesem Fall auch „Device Under Test“ (DUT) genannt, ist auf einer Montageplatte installiert. Diese ist über einen manuellen Spindeltrieb in der Höhe verstellbar, um die Arbeitshöhe des DUTs einzustellen.

Der Teststand beinhaltet ungeschützte, bewegte Maschinenteile und ist dadurch eine Gefährdungsquelle, die durch eine Notabschaltung gesichert wird. Diese wird durch eine Lichtschrankenanlage realisiert. Eine Ampelanzeige gibt zudem Auskunft über den Betriebszustand der Anlage.

3.4.2 Referenz

Das Referenzsystem ist ein Magnetsensor der Firma SIKO (MSK 1000). Dabei handelt es sich um einen Wegsensor mit magnetisch kodierter Maßverkörperung. Der Sensorkopf ist seitlich am Schlitten montiert. Die Maßverkörperung besteht aus einem magnetisch kodierten Kunststoffband, das mit einem Stahlband verbunden ist. [18]

Die magnetische Kodierung wird durch eine alternierende magnetische Polarisierung erreicht [18]. Dieses Magnetband ist unterhalb des Schlittens starr mit der Führungsschiene verbunden.

Bei dem Verfahren bewegt sich der Sensorkopf über das Magnetband und misst über zwei Sensoren das Magnetfeld des Maßkörpers. Aus diesen gemessenen Signalen ermittelt das System die Position innerhalb der Magnetfelder. Bei einer Positionsveränderung wird, je nach Verfahrensrichtung, ein positives bzw. negatives Inkrement ausgegeben. [19]

3.4.3 Steuerung

Die primären Elemente der Steuerung sind ein PXI System der Firma National Instruments und ein Servoverstärker der Firma Jenaer Antriebstechnik (Ecovario).

Das PXI System ist eine PC-basierte Plattform zur Steuerung von verschiedenen Mess-, Steuer- und Regelanwendungen. Die Hardwarearchitektur des PXI Systems besteht aus drei grundlegenden Komponenten: Chassis, Systemcontroller und Peripheriemodule. [20]

Das verwendete Chassis (PXIe-1078) ist aktuell wie folgt bestückt:

Slot	Bezeichnung	Bedeutung
1	PXIe-8100 RT	Systemcontroller
2	PXIe-6356	Datenerfassungskarte
3	PXIe-1435	Framegrabber
4	PXIe-8512	CAN-Bus Schnittstelle

Tabelle 3-1: Bestückung des PXI Chassis PXIe-1078

Die zur Steuerung verwendete Software läuft auf dem Systemcontroller und dem am Teststand befindlichen Host PC (Siehe Kapitel 5.2). Das PXI System ist über ein Controller Area Network (CAN) Bussystem mit dem Servoverstärker verbunden. Die Datenerfassung des DUT und des Referenzsystems erfolgt über die Datenerfassungskarte (PXIe-6356). Durch die Einbindung des Systemcontrollers ergibt sich die in Abbildung 3-4 zu sehende, vereinfachte Hardwarearchitektur.

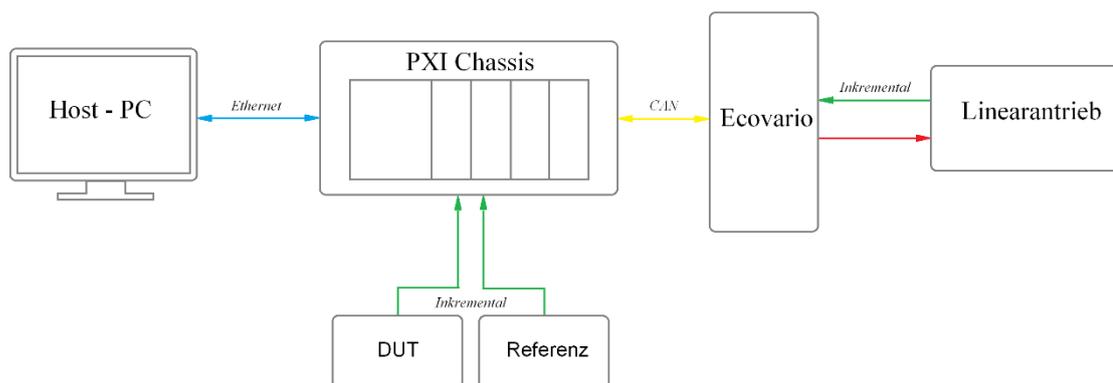


Abbildung 3-4: Vereinfachte Hardware Architektur des Teststandes

3.4.4 Bedienung des Teststandes

Die Bedienung des Teststandes läuft über die Benutzeroberfläche (Graphical User Interface, GUI), welche auf dem Host PC dargestellt wird. Über die GUI ist es möglich, folgende Fahrprogramme zu starten:

Referenzfahrt

Die Referenzfahrt dient zur Positionsfindung des Schlittens. Wird eine Referenzfahrt über den Button „Start Homing“ angefordert, verfährt der Schlitten bis die Referenz, in diesem Fall der mittige induktive Sensor, gefunden wird. Die Referenzfahrt muss nach jedem Einschalten des Teststandes durchgeführt werden.

Handbetrieb

Der Schlitten wird im absoluten oder relativen Modus auf eine gewünschte Position verfahren. Die absolute Positionierung ist nur verfügbar, wenn zuvor eine Referenzfahrt durchgeführt und die Referenz gefunden wurde.

Reversierfahrt

Die Reversierfahrt ist die eigentliche Messfahrt. Sie ist über mehrere Einstellungsparameter definiert, welche über die Benutzeroberfläche vorgegeben werden (Abbildung 3-5).

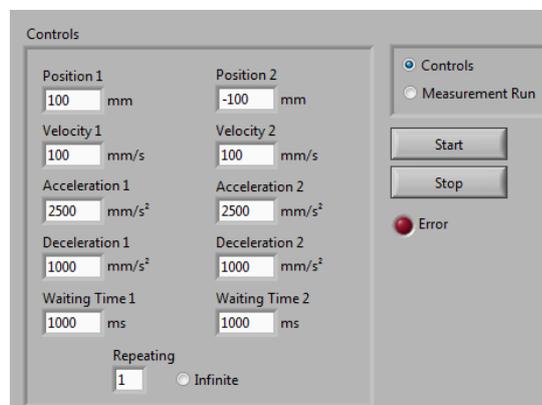


Abbildung 3-5: Bedienfeld zur Konfiguration der Reversierfahrt in der GUI

Die Positionen 1 und 2 deklarieren die Positionen zwischen denen der Schlitten reversieren soll. Über die Eingabe „Acceleration 1“ und „Acceleration 2“ werden die Beschleunigungen eingestellt, mit denen der Schlitten die jeweilige Position anfährt. „Target Velocity 1“ und „Target Velocity 2“ bestimmen die Zielgeschwindigkeiten. Mittels „Deceleration 1“ und „Deceleration 2“ werden die negativen Beschleunigungen zur Bremsung des Schlittens konfiguriert. Über „Waiting Time 1“ und „Waiting Time 2“ wird die Wartezeit nach Erreichen der Position eingestellt. Diese Fahrparameter bestimmen somit das Fahrverhalten eines Messdurchlaufes. Über das Eingabefeld „Repeating“ wird die Anzahl der Messdurchläufe vorgegeben. Durch die Auswahl „Infinity“ reversiert der Schlitten bis zur Beendigung der Fahrt durch den „Stop“ Button.

4 Umsetzung der Datenauswertung

In diesem Kapitel ist die Implementierung der Datenauswertung in LabVIEW beschrieben.

Im aktuellen Aufbau des Teststands werden die gemessenen Wegstrecken des DUT und der Referenz, entsprechend der Wegrichtung in Form von gezählten Inkrementen erfasst. Diese Primärdaten bilden die Messdaten, die durch den Vergleich der Messwerte vom DUT und der Referenz als Grundlage der Auswertungen dienen.

Die Umsetzung erfordert zunächst eine Aufschlüsselung dieser Messdaten, um sie für die jeweiligen Auswertungen zu nutzen. Aus den Anforderungen aus Kapitel 2 ergibt sich die Aufteilung in die Bewegungsabläufe: Beschleunigung, Bremsung und Fahrt mit konstanter Geschwindigkeit. Die Unterscheidung dieser Bewegungsabläufe ist notwendig, um das Verhalten und die Empfindlichkeit des DUT in diesen Abschnitten der Messung zu untersuchen. Damit werden zum Beispiel signifikante Messabweichungen, wie der Schleppfehler in einer beschleunigten Bewegung erkannt [2]. Auch das Abdriften der Messwerte oder andere systematische Fehler können somit identifiziert und gegebenenfalls einem spezifischen Bewegungsablauf zugeordnet werden. Zudem geben die Messdaten von einer Beschleunigung aus dem Stillstand, Aufschluss über eine zeitliche Verzögerung der Messwertausgabe.

Nachfolgend werden die Funktionen des in LabVIEW erstellten Programms zur Messdatenauswertung beschrieben.

4.1 Aufschlüsseln der Messdaten

Zur Auswertung der einzelnen Bewegungsabläufe ist es notwendig, die Messfahrt in die jeweiligen Abläufe aufzuschlüsseln (Abbildung 4-1). Daraus ergeben sich einzelne Zeitfenster. Um die Messdaten den einzelnen Zeitfenstern zuzuordnen, kann auf die theoretischen Zeiten zurückgegriffen werden.

Eine Alternative wäre hier die automatische Erkennung des Bewegungsablaufes anhand der Messwerte. Für diese Umsetzung wurde jedoch die theoretische Zuordnung gewählt, da hier die statistische Auswertung innerhalb gleicher Zeitfenster erfolgen kann. Die

zuvor genannte Alternative wird in Kapitel 6.5 als mögliche Weiterentwicklung ausführlicher beschrieben.

Die theoretischen Zeitfenster werden für jeden Bewegungsablauf berechnet. So werden zum Beispiel die zurückgelegten Strecken während der Beschleunigungszeit von Position 1 zu Position 2 im Zeitfenster der theoretischen Beschleunigungszeit miteinander verglichen. Um die Zeitfenster in LabVIEW zu ermitteln, wurde das Sub-VI *Messzeiten.vi* erstellt.

Dieses VI ermittelt aus den in der GUI angegebenen Fahrparametern die Beschleunigungszeiten $t_{a\ 12}$, $t_{a\ 21}$, die Bremszeiten $t_{d\ 12}$, $t_{d\ 21}$, die Fahrzeiten mit konstanter Geschwindigkeiten $t_{v\ 12}$, $t_{v\ 21}$, sowie die gesamte Fahrzeit $t_{ges.}$.

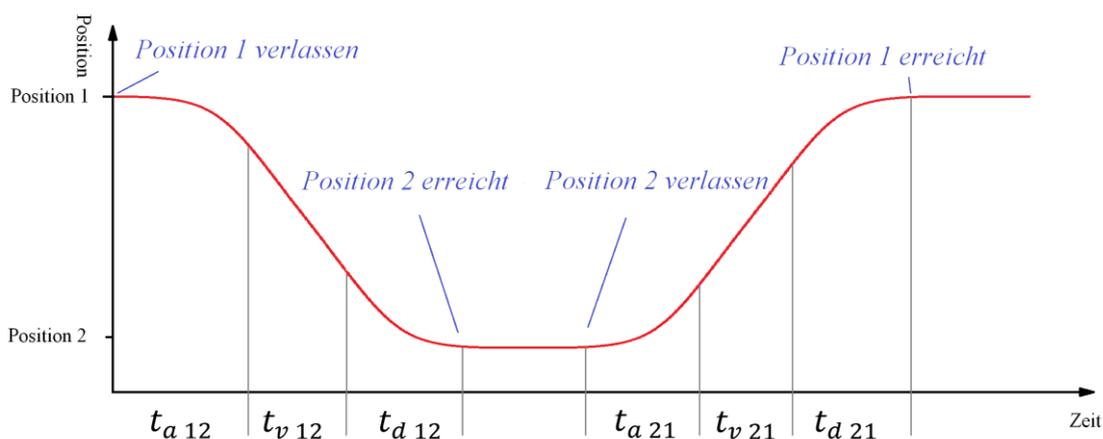


Abbildung 4-1: Skizze eines Messdurchlaufes mit Einteilung der Zeitfenster und Orientierungspunkten

Um die Messdaten für die weitere Auswertung nutzen zu können, werden sie zunächst in einer lokalen Variable gespeichert. Erst durch diese lokale Variable stehen die Messdaten in LabVIEW zur Verfügung. Sie wird bei jedem Start einer Messung zurückgesetzt und besteht aus einem zweidimensionalen Array aus dem numerischen Datentyp Double (DBL). Dieses Array enthält die gemessenen Strecken der Referenz $x_{i,Ref}$ und des DUT

$x_{i,DUT}$. Bei einer laufenden Messung wird jeder erfasste Messwert unter einem fortlaufenden Index in dem Array gespeichert. Da die Datenerfassung nur ausgegebene Inkremente zählt, werden die Messdaten entsprechend der Auflösung der Sensoren berechnet. Die Auflösung wird vom Nutzer in der GUI angegeben.

In Tabelle 4-1 ist der Aufbau des Messdaten-Arrays dargestellt. Der Index i gibt nicht nur Auskunft über die Anordnung der Daten, er enthält zudem Zeitinformationen. Diese ergeben sich aus der *Sample Rate*, welche die Anzahl der Messungen pro Sekunde angibt. Ist die *Sample Rate* beispielsweise auf 100 Samples pro Sekunde festgelegt, ergibt sich ein zeitlicher Abstand Δt von 10 Millisekunden zwischen jedem Messwert. Somit lässt der Index i Rückschlüsse auf die Zeitabstände der einzelnen Messwerte zu. Die Zeitdifferenz zwischen zwei Indizes lässt sich mit der Gleichung (11) berechnen.

Index i	Dimension 1	Dimension 2
0	$x_{1,DUT}$	$x_{1,Ref}$
1	$x_{2,DUT}$	$x_{2,Ref}$
2	$x_{3,DUT}$	$x_{3,Ref}$
3	$x_{4,DUT}$	$x_{4,Ref}$
...

Tabelle 4-1: Aufbau des Messdaten-Arrays

$$\Delta t = \frac{1}{\text{Sample Rate}} \quad [\Delta t] = \frac{1}{\frac{1}{s}} \quad (11)$$

Um die Bewegungsabläufe in die jeweiligen Zeitfenster aufzuteilen, werden Informationen über die aktuelle Lage des Schlittens benötigt. Dies geschieht über die Rückmeldung des Servoverstärkers, welche über die CAN-Schnittstelle in LabVIEW eingelesen wird. Die Rückmeldung bestätigt, dass der Schlitten die anzufahrende Position erreicht hat. Diese Rückmeldung wird genutzt, um das Erreichen und Verlassen der Position zu ermitteln. Wird eine Position erreicht oder verlassen, wird der zu diesem Zeitpunkt beschriebene Index des Messdaten-Arrays erfasst. Das Verlassen einer Position wird in diesem Fall durch eine fallende Flanke der Rückmeldung detektiert. Diese Indizes werden als Orientierungspunkte in einem weiteren Array gespeichert.

Tabelle 4-2 zeigt den Aufbau dieses Arrays für das Erreichen der jeweiligen Position, das Array zum Verlassen der Position ist nach dem gleichen Schema aufgebaut. Die Anordnung dieser Orientierungspunkte lässt sich durch Abbildung 4-1 nachvollziehen.

Index i	Dimension 1	Bedeutung	Messdurchlauf
1	$i_{\text{Pos.2}}$ erreicht	Der Schlitten hat Position 2 erreicht.	1
2	$i_{\text{Pos.1}}$ erreicht	Der Schlitten hat Position 1 erreicht.	1
3	$i_{\text{Pos.2}}$ erreicht	Der Schlitten hat Position 2 erreicht.	2
4	$i_{\text{Pos.1}}$ erreicht	Der Schlitten hat Position 1 erreicht.	2
...

Tabelle 4-2: Aufbau des Arrays mit Orientierungspunkten des Messdaten-Arrays für das Erreichen einer Position.

Die Ermittlung des Index i für den Index eines Orientierungspunktes einer beliebigen Messung n lässt sich durch die Gleichungen (12) bis (15) darstellen:

für den Index i mit $i_{\text{Pos.1}}$ erreicht

$$i = (2 \cdot n) \quad (12)$$

für den Index i mit $i_{\text{Pos.2}}$ erreicht

$$i = (2 \cdot n) - 1 \quad (13)$$

für den Index i mit $i_{\text{Pos.1}}$ verlassen

$$i = (2 \cdot n) - 1 \quad (14)$$

für den Index i mit $i_{\text{Pos.2}}$ verlassen :

$$i = (2 \cdot n) \quad (15)$$

Um nun die Messdaten für die einzelnen Phasen mit den jeweiligen Zeitfenstern zu erhalten, müssen die passenden Indizes des Messdaten-Arrays ermittelt werden. Die theoretischen Zeiten werden mit Hilfe der *Sample Rate* in die Anzahl der entsprechenden Indizes Δi umgewandelt. Mit den zuvor ermittelten Orientierungspunkten werden für die einzelnen Phasen die Indizes der Start- und Endpunkte, i_{Start} und i_{Ende} , des Messdaten-Arrays berechnet. Anschließend lässt sich aus dem Messdaten-Array ein Teil-Array mit den gewünschten Messdaten extrahieren.

Um die für die jeweilige Auswertung relevanten Messdaten zu bestimmen, sind folgende Indizes für die einzelnen Phasen des Messdurchlaufes definiert:

Index-Ermittlung für die Beschleunigungsphasen:

$$\Delta i_{\text{Beschleunigung } 12} = t_{a\ 12} \cdot \text{Sample Rate} \quad (16)$$

$$i_{\text{Start } a\ 12} = i_{\text{Pos.1 verlassen}} \quad (17)$$

$$i_{\text{Ende } a\ 12} = i_{\text{Pos.1 verlassen}} + \Delta i_{\text{Beschleunigung } 12} \quad (18)$$

$$\Delta i_{\text{Beschleunigung } 21} = t_{a\ 21} \cdot \text{Sample Rate} \quad (19)$$

$$i_{\text{Start } a\ 21} = i_{\text{Pos.2 verlassen}} \quad (20)$$

$$i_{\text{Ende } a\ 21} = i_{\text{Pos.2 verlassen}} + \Delta i_{\text{Beschleunigung } 21} \quad (21)$$

Index-Ermittlung für die Bremsphasen:

$$\Delta i_{\text{Bremsung } 12} = t_{d\ 12} \cdot \text{Sample Rate} \quad (22)$$

$$i_{\text{Start } d\ 12} = i_{\text{Pos.2 erreicht}} - \Delta i_{\text{Bremsung } 12} \quad (23)$$

$$i_{\text{Ende } d\ 12} = i_{\text{Pos.2 erreicht}} \quad (24)$$

$$\Delta i_{\text{Bremsung } 21} = t_{d\ 21} \cdot \text{Sample Rate} \quad (25)$$

$$i_{\text{Start } d\ 21} = i_{\text{Pos.1 erreicht}} - \Delta i_{\text{Bremsung}} \quad (26)$$

$$i_{\text{Ende } d\ 21} = i_{\text{Pos.1 erreicht}} \quad (27)$$

Index-Ermittlung für die Fahrt mit konstanter Geschwindigkeit:

$$i_{Start\ v\ 12} = i_{Pos.1\ verlassen} + \Delta i_{Beschleunigung\ 12} \quad (28)$$

$$i_{Ende\ v\ 12} = i_{Pos.2\ erreicht} - \Delta i_{Bremsung\ 12} \quad (29)$$

$$i_{Start\ v\ 21} = i_{Pos.2\ verlassen} + \Delta i_{Beschleunigung\ 21} \quad (30)$$

$$i_{Ende\ v\ 21} = i_{Pos.1\ erreicht} - \Delta i_{Bremsung\ 21} \quad (31)$$

Im nachfolgenden Beispiel wird dieses Verfahren verdeutlicht. Als Grundlage wird eine Messung aus vier Messdurchläufen betrachtet.

Das Ziel ist die Bestimmung der Messdaten innerhalb der Beschleunigungszeit von Position 2 nach Position 1 im dritten Messdurchlauf. Hierzu müssen die Indizes $i_{Start\ a\ 21}$ und $i_{Ende\ a\ 21}$ für $n = 3$, bestimmt werden.

Die eingestellten Fahrparameter werden wie folgt angenommen:

$$Sample\ Rate = 50 \frac{1}{s}, \quad v_{21} = 400 \frac{mm}{s}, \quad a_{21} = 400 \frac{mm}{s^2}, \quad Position\ 1 = 300\ mm, \\ Position\ 2 = -300\ mm.$$

Für die Bewegungsrichtung von Position 2 nach Position 1 ist der Orientierungspunkt beim Verlassen der Position 2 relevant. Mit Gleichung (15) lässt sich dieser aus dem Array mit den Orientierungspunkten (Tabelle 4-3) ermitteln.

Für $n = 3$ ergibt sich $i = (2 \cdot n) = 6$, unter diesem Index wird nun der Index des Messdaten-Arrays $i_{Start\ a\ 21} = 1250$ ausgelesen. (Siehe Makierung Tabelle 4-3)

i	Index	Bedeutung	Messdurchlauf
1	0	Position 1 verlassen	1
2	250	Position 2 verlassen	1
3	500	Position 1 verlassen	2
4	750	Position 2 verlassen	2
5	1000	Position 1 verlassen	3
6	1250	Position 2 verlassen	3
7	1500	Position 1 verlassen	4
8	1750	Position 2 verlassen	4

Tabelle 4-3: Beispiel eines Arrays mit Orientierungspunkten

Mit der theoretischen Beschleunigungszeit $t_{a\ 21}$ aus Gleichung (36) und der *Sample Rate* wird die Anzahl der Messwerte $\Delta i_{\text{Beschleunigung 21}}$ innerhalb der Beschleunigungszeit mit Gleichung (19) berechnet.

$$\text{mit } t_{a\ 21} = \frac{v_{21}}{a_{21}} = \frac{400 \frac{\text{mm}}{\text{s}}}{400 \frac{\text{mm}}{\text{s}^2}} = 1 \text{ s}$$

$$\Delta i_{\text{Beschleunigung 21}} = 1 \text{ s} \cdot 50 \frac{1}{\text{s}} = 50$$

Durch Gleichung (21) wird der Index $i_{\text{Ende a 21}}$ berechnet.

$$i_{\text{Ende a 21}} = 1250 + 50 = 1300$$

Mit den Indizes $i_{\text{Start a 21}} = 1250$ und $i_{\text{Ende a 21}} = 1300$ lassen sich nun die relevanten Messdaten aus dem Messdaten-Array extrahieren (Abbildung 4-2).

i	DUT	REF	
...	Stillstand
1247	-300,8	-300,1	
1248	-300,8	-300,1	
1249	-300,8	-300,1	Beschleunigung
1250	-300,6	-299,8	
1251	-299,1	-299,2	
1252	-289,1	-289,2	
1253	-275,7	-275,4	
1254	-258,8	-256,9	
...	
1296	-125,6	-126,3	
1297	-115,7	-114,9	
1298	-107,1	-107,6	
1299	-101,8	-102,1	
1300	-100	-99,9	konst. Geschwindigkeit
1301	-98,2	-97,7	
1302	-96,4	-95,5	
1303	-94,6	-93,3	
...	

$\Delta i_{\text{Beschleunigung 21}}$ {

Abbildung 4-2: Beispiel eines Messdaten-Arrays mit gekennzeichneter Beschleunigungsphase

4.2 Aufbau der Benutzeroberfläche

Um eine Visualisierung der Auswertungen zu ermöglichen, wurde die vorhandene Benutzeroberfläche des Teststands durch die Registerkarte „Measurement Analysis“ ergänzt. Die Registerkarte wurde so entwickelt, dass sie aktiviert wird sobald eine erste Messung startet. Sie enthält fünf weitere Registerkarten. Die erste Registerkarte „General Info“ (Abbildung 4-3) enthält allgemeine Informationen zur aktuellen Messung, wie die eingestellten Fahrparameter die Messzeiten und eine Fortschrittsanzeige. Zudem werden die aktuellen Messdaten in einem Graphen dargestellt. Diese Darstellung umfasst die gemessenen Positionen des DUT und der Referenz über die Zeit und deren Abweichungen. Über ein Auswahlfenster können die Graphen eingestellt werden. Hier lässt sich die Darstellung der gesamten oder einer einzelnen Messung, die Skalierung oder eine Cursor-Funktion einstellen.

Die Registerkarten „Total Length“ „Deviation & Velocity“, „Acceleration & Deceleration“ und „Constant Velocity“ werden in Kapitel 4.3 bis 4.5 ausführlicher beschrieben. Diese Aufteilung wurde gewählt, um die Visualisierung der jeweiligen Auswertung übersichtlich und detailliert darzustellen.

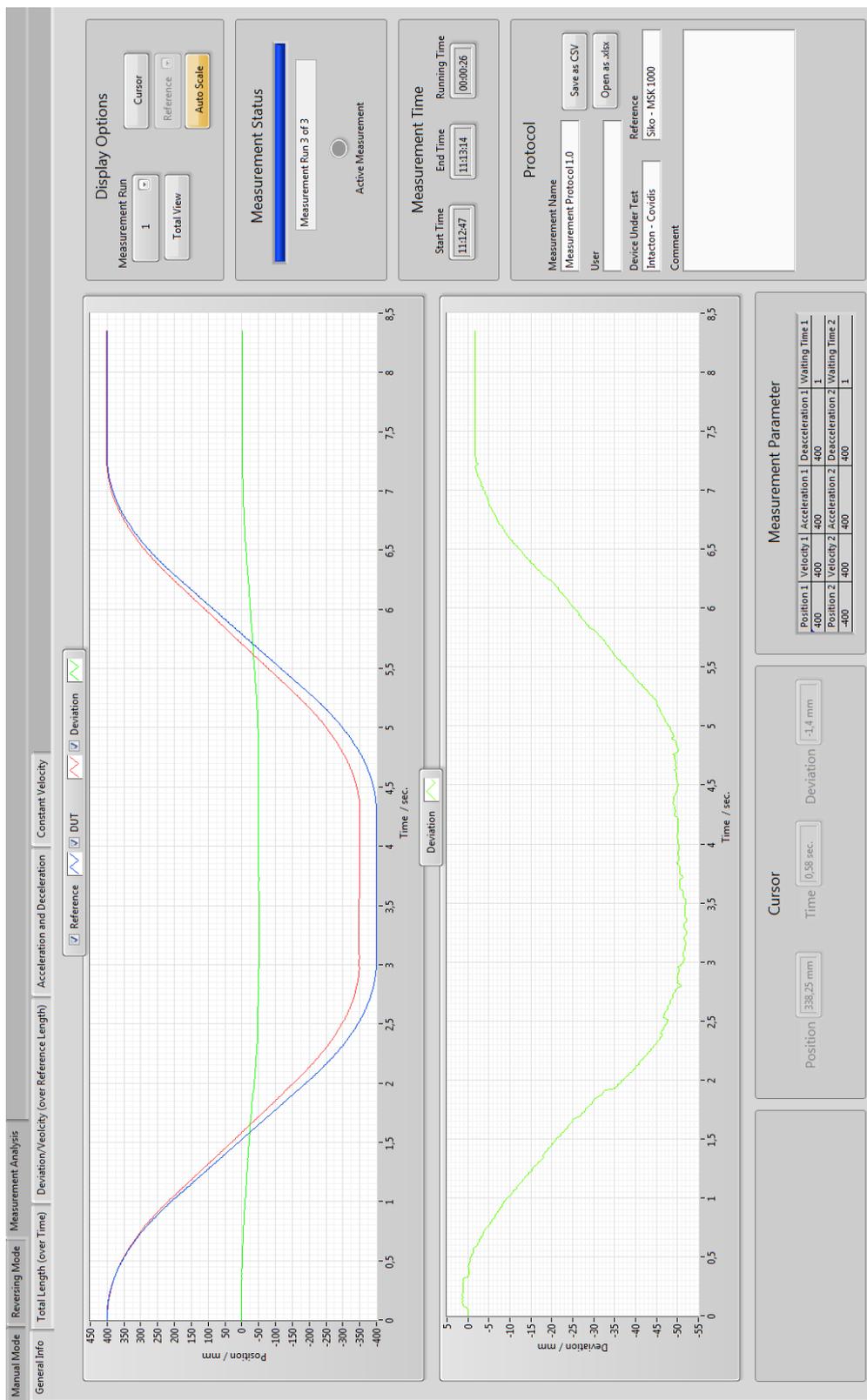


Abbildung 4-3: Screenshot der Benutzeroberfläche mit offener Registerkarte „General Info“

4.3 Vergleich der gesamten Fahrstrecke

Das Ziel beim Vergleich der gesamten Fahrstrecke ist es, die absolut zurückgelegten Strecken der Referenz und des DUT miteinander zu vergleichen. Die Bildung der absoluten Strecken lässt sich mit den Gleichungen (32) und (33) beschreiben.

$s_{\text{Ref. abs.}}$ = absolute zurückgelegte Strecke der Referenz

$s_{\text{DUT abs.}}$ = absolute zurückgelegte Strecke des DUT

$s_{1>2}$ = Strecke von Position 1 zu Position 2 (aus Messdaten)

$s_{2>1}$ = Strecke von Position 2 zu Position 1 (aus Messdaten)

$$s_{\text{Ref. abs.}} = |s_{\text{Ref. 1>2}}| + |s_{\text{Ref. 2>1}}| \quad (32)$$

$$s_{\text{DUT abs.}} = |s_{\text{DUT 1>2}}| + |s_{\text{DUT 2>1}}| \quad (33)$$

Die Strecken $s_{\text{Ref. abs.}}$ und $s_{\text{DUT abs.}}$ dienen somit als Grundlage für diese Auswertung. In Abbildung 4-4 wird ein absoluter Streckenverlauf in einem Weg-Zeit Diagramm dargestellt.

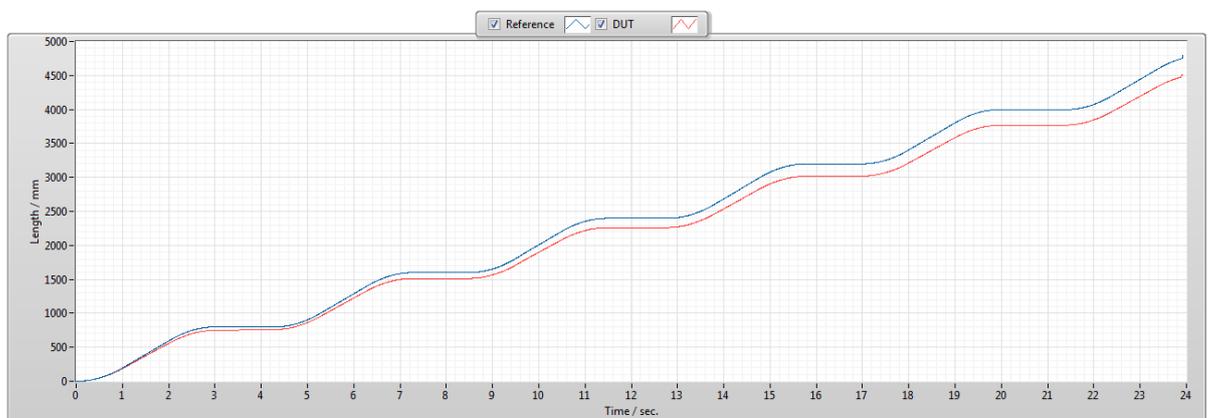


Abbildung 4-4: Beispiel eines absoluten Streckenverlaufs

Die Messdaten werden für diesen Vergleich in die absolute gefahrene Strecke umgerechnet und weiterhin in einem Array gespeichert. Die Umwandlung erfolgt durch das Aufaddieren jedes einzelnen Messwertes und wird durch eine For-Schleife realisiert, die jeden Index des Messdaten-Arrays neu berechnet. Sie wird durch Gleichung (34) mit den originalen Messdaten x_i und der aufaddierten Strecke s_i beschrieben.

$$s_i = |x_i - x_{i-1}| + s_{i-1} \quad (34)$$

Der Vergleich der gesamten Fahrstrecke wird in zwei Registerkarten dargestellt. Der Unterschied zwischen den beiden Registerkarten ist die Auflegung der Graphen. In der einen werden die Gesamtstrecken des DUT und der Referenz über die Zeit angezeigt. In der anderen wird die Abweichung der beiden Strecken über die Referenzlänge gelegt.

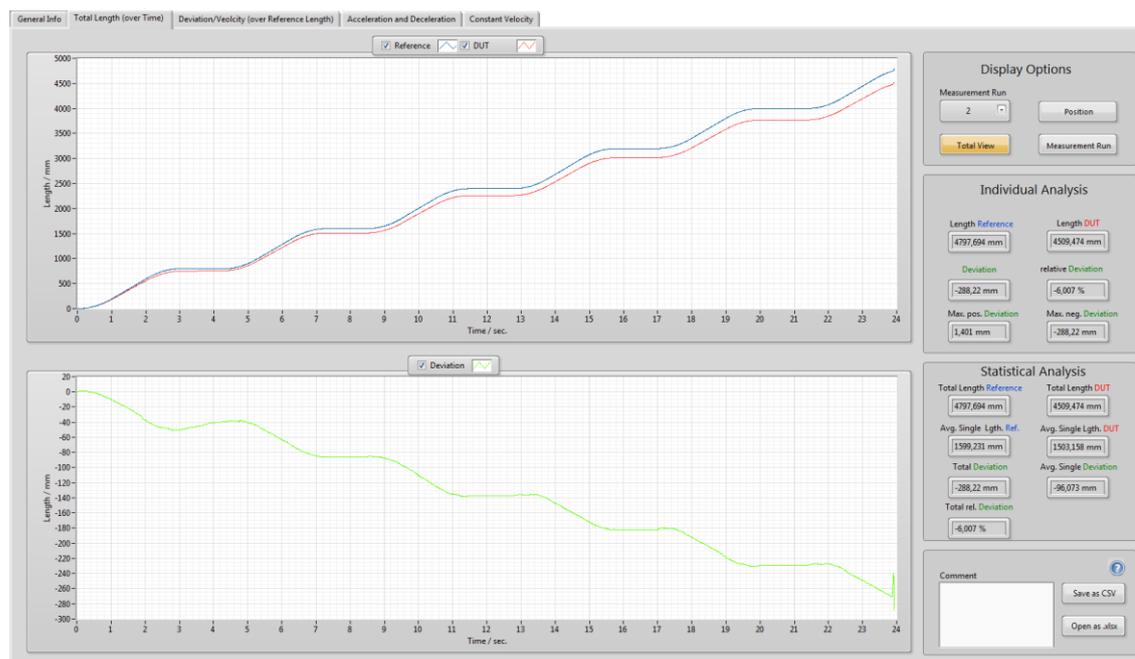


Abbildung 4-5: Visualisierung der Auswertung der gesamten Fahrstrecke in der GUI als Registerkarte

Die Visualisierung erfolgt durch zwei Diagramme (Abbildung 4-5). Das obere Diagramm zeigt die Werte der Referenz und des DUT an. Die Abweichung der DUT Werte gegenüber der Referenz wird im unteren Diagramm dargestellt. Die Berechnung dieser Datenpunkte der Abweichung beruht auf Gleichung (5) und wurde durch die Subtraktion der Messdaten Arrays realisiert. Über das Fenster „Display Options“ kann ein einzelner Messdurchgang oder die gesamte Messung angezeigt werden. Zudem lassen sich bei der Anzeige der gesamten Messung die jeweiligen Messdurchgänge und erreichten Positionen einblenden.

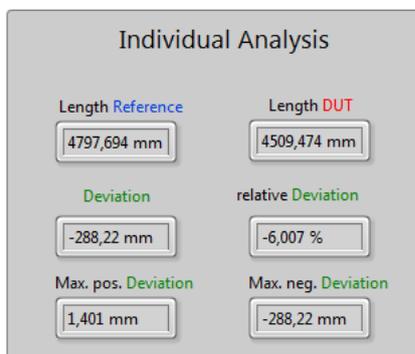


Abbildung 4-6: Anzeige der Auswertung „Individual Analysis“

Im Fenster „Individual Analysis“ (Abbildung 4-6) wird die Auswertung der im Diagramm dargestellten Werte gezeigt. Sie enthält die Längen der Referenz und des DUT. Die folgenden absoluten und relativen Messabweichungen beziehen sich auf diese beiden Längen. Weiterhin wird die maximale positive und maximale negative Abweichung der Messung dargestellt. Die in diesem Fenster gezeigten Auswertungen sind für den unter „Display Options“ ausgewählten Messdurchlauf gültig. Wird die gesamte Messung im Diagramm dargestellt, beziehen sich diese Werte ebenfalls auf die gesamten Messdaten.

Unter „Statistical Analysis“ finden sich die Gesamtstrecken beider Sensoren, die durchschnittlichen gefahrenen Strecken pro Messdurchlauf und die absolute und relative Messabweichung.

In der Registerkarte „Deviation & Velocity“ kommt eine Darstellung der Geschwindigkeit hinzu. Die Geschwindigkeit v wird über Gleichung (35) ermittelt.

$$v = \frac{\Delta s}{\Delta t} \quad (35)$$

Die Zeitspanne Δt wird vom Nutzer unter „Display Options“ eingestellt. Die jeweilige Wegstrecke Δs wird der Zeitspanne entsprechend aus dem Array gebildet. Die ermittelte Geschwindigkeit wird ebenfalls in einem Diagramm über die Referenzlänge dargestellt (Abbildung 4-7). Das Diagramm zeigt die Geschwindigkeiten des DUT und der Referenz und die Abweichung dieser beiden Geschwindigkeiten.

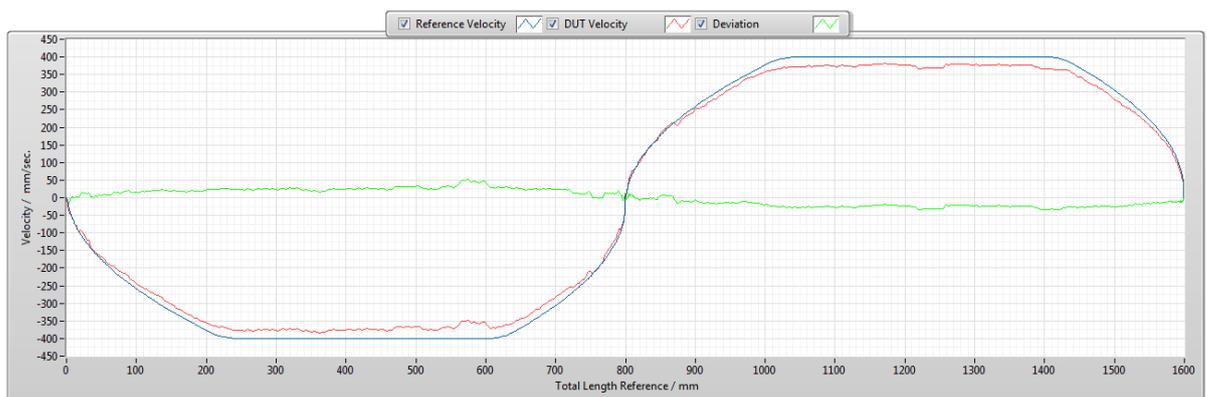


Abbildung 4-7: Geschwindigkeitsanzeige eines einzelnen Messdurchlaufes

4.4 Vergleich der Beschleunigungs- und Bremsvorgänge

In jedem Messdurchlauf gibt es zwei Beschleunigungsphasen, die Beschleunigung von Position 1 zur Position 2 und von Position 2 zu Position 1 (Abbildung 4-8) Die zur Auswertung relevanten Messdaten werden mit dem in Kapitel 4.1 beschriebenen Verfahren ermittelt.

Es werden die in den Beschleunigungszeiten zurückgelegten Strecken $s_{a\ 12}$, $s_{a\ 21}$ der Referenz und des DUT verglichen.

Die Beschleunigungszeiten $t_{a\ 12}$ und $t_{a\ 21}$ werden durch die vorgegebenen Geschwindigkeiten v_{12} , v_{21} und Beschleunigungen a_{12} und a_{21} berechnet (Gleichung 36).

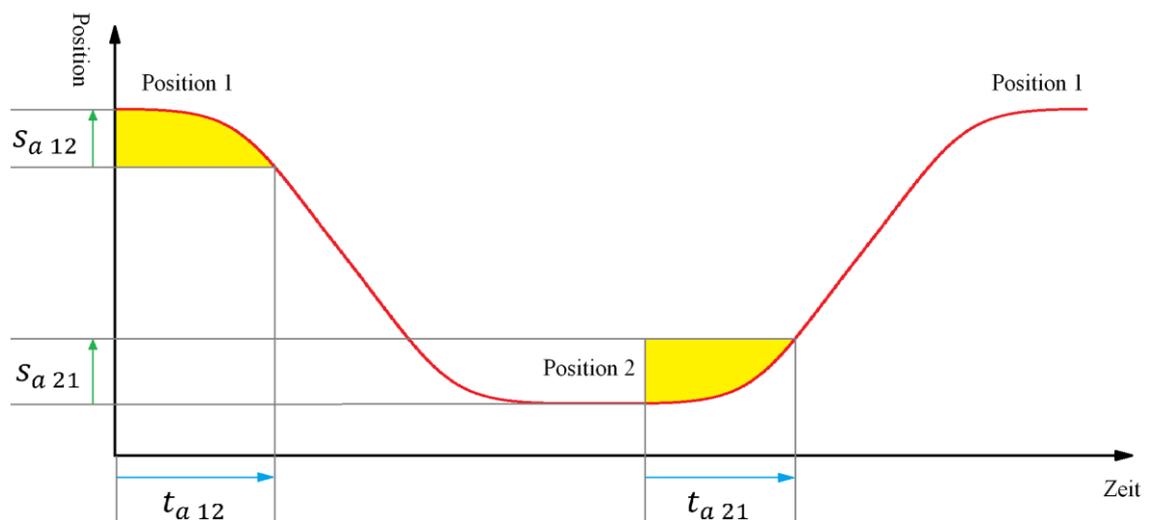


Abbildung 4-8: Skizze der Beschleunigungsphasen eines Messdurchlaufes

$$t_{a\ 12} = \frac{v_{12}}{a_{12}} \quad ; \quad t_{a\ 21} = \frac{v_{21}}{a_{21}} \quad (36)$$

Für die beiden Bremsvorgänge (Abbildung 4-9) werden ebenfalls die zurückgelegten Strecken $s_{d\ 12}$, $s_{d\ 21}$ des DUT mit denen der Referenz verglichen.

Die Bremszeiten $t_{d\ 12}$ und $t_{d\ 21}$ werden mit den angegebenen negativen Beschleunigungen $a_{d\ 12}$, $a_{d\ 21}$ und den Geschwindigkeiten v_{12} , v_{21} ermittelt (Gleichung ((37))).

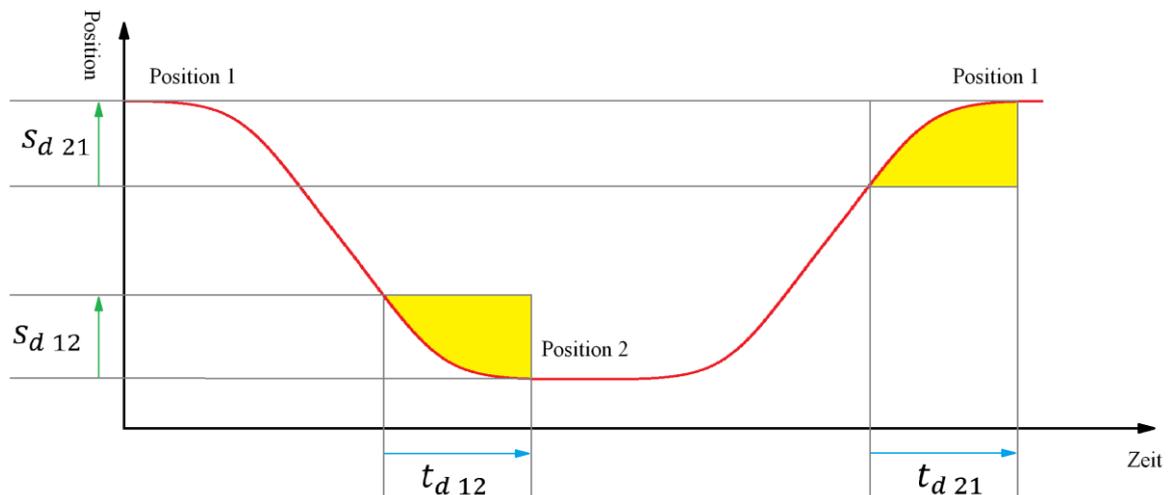


Abbildung 4-9: Skizze der Bremsphasen eines Messdurchlaufes

$$t_{d\ 12} = \frac{v_{12}}{a_{d\ 12}} ; t_{d\ 21} = \frac{v_{21}}{a_{d\ 21}} \quad (37)$$

Die Visualisierung der Auswertungen erfolgt unter einer gemeinsamen Registerkarte „Acceleration & Deceleration“ (Abbildung 4-10).

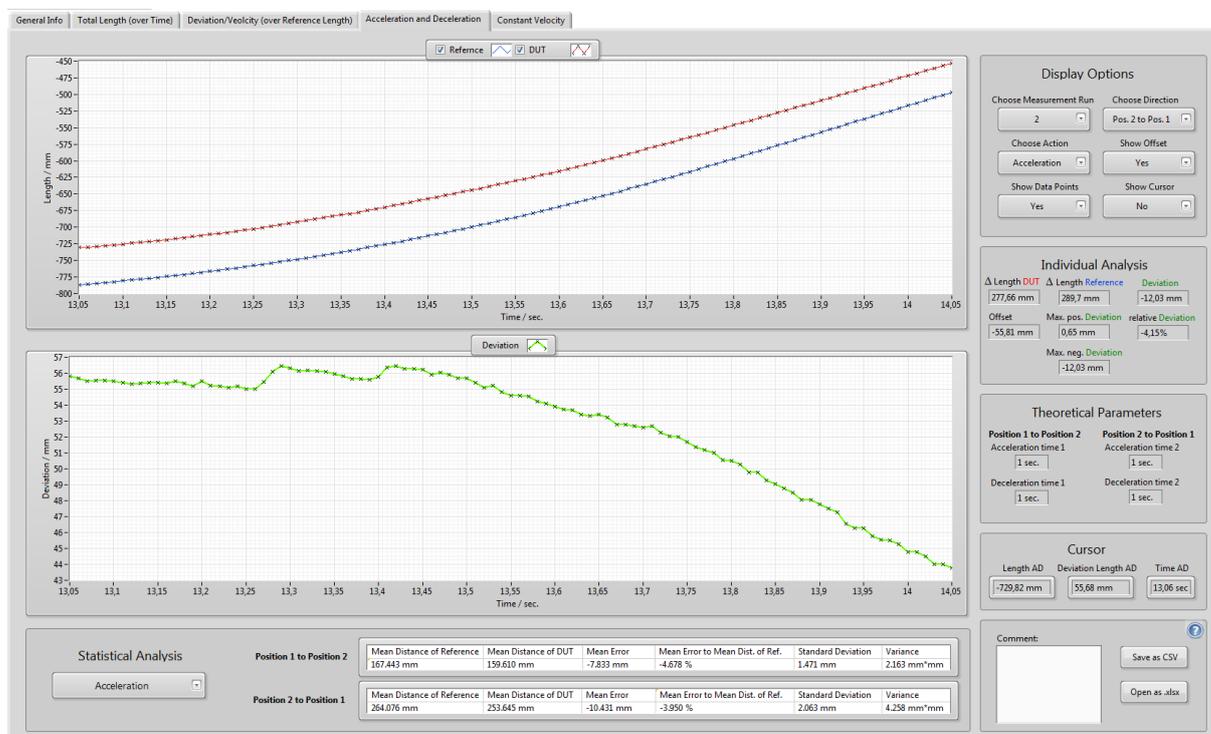


Abbildung 4-10: Visualisierung der Auswertung von Beschleunigung und Bremsung in der GUI

Durch die Auswahl des Messdurchlaufes unter „Display Options“, werden die Messdaten (oberes Diagramm) und die Abweichung (unteres Diagramm) dargestellt. Die Anzeige kann in der Auswahlliste „Choose Action“ zwischen Beschleunigung und Bremsung gewechselt werden. Weiterhin stehen noch Möglichkeiten zur Konfiguration des Diagramms zur Verfügung. Es kann die Anzeige der Datenpunkte, das Ausblenden des Offsets in der Abweichung, oder eine Cursor Funktion ausgewählt werden. Zudem sind unter „Theoretical Parameters“ die theoretischen Zeitspannen der Beschleunigungen und Bremsungen vermerkt.

Die Auswertung erfolgt im Fenster „Individual Analysis“. Sie umfasst die Strecken der Referenz und des DUT, die absolute Abweichung und relative Messabweichung, den Offset und die maximalen positiven und negativen Abweichungen. Diese Auswertung bezieht sich nur auf die in den Diagrammen dargestellten Messdaten.

Die statistische Auswertung ist unterhalb der Graphen in tabellarischer Form dargestellt. Die Anzeige kann zwischen Beschleunigung und Bremsung gewechselt werden und enthält die Auswertungen für beide Verfahrensrichtungen. Zu denen gehören die Mittelwerte der zurückgelegten Strecken, die mittleren Abweichungen, die relativen Abweichungen, die Standardabweichungen und die Varianz.

4.5 Vergleich der Fahrt mit konstanter Geschwindigkeit

In dieser Auswertung werden die in der theoretischen Fahrzeit mit konstanter Geschwindigkeit v_{12} , v_{21} zurückgelegten Strecken $s_{cv\ 12}$, $s_{cv\ 21}$ der Referenz und des DUT verglichen (Abbildung 4-11). Die Zeitfenster ergeben sich durch die Gleichungen (38) und (39).

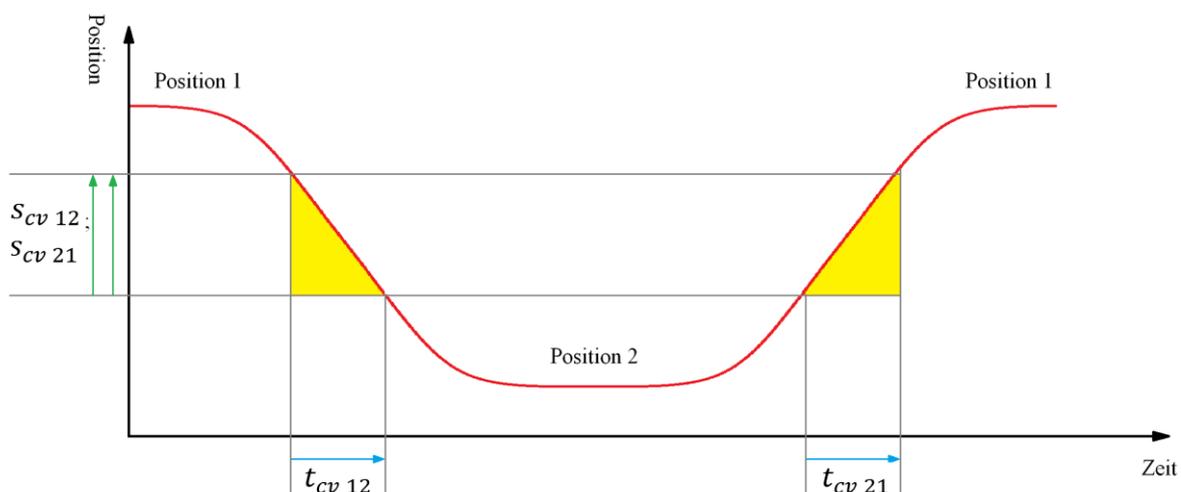


Abbildung 4-11: Skizze der Fahrzeiten mit konstanter Geschwindigkeit eines Messdurchlaufes

$$t_{cv 12} = \frac{|Position 1 - Position 2| - \frac{a_{12} \cdot \left(\frac{v_{12}}{a_{12}}\right)^2}{2} - \frac{a_{d 12} \cdot \left(\frac{v_{12}}{a_{d 12}}\right)^2}{2}}{v_{12}} \quad (38)$$

$$t_{cv 21} = \frac{|Position 1 - Position 2| - \frac{a_{21} \cdot \left(\frac{v_{21}}{a_{21}}\right)^2}{2} - \frac{a_{d 12} \cdot \left(\frac{v_{21}}{a_{d 21}}\right)^2}{2}}{v_{21}} \quad (39)$$

Für diesen Vergleich ist keine zusätzliche Berechnung nach den oben genannten Gleichungen nötig, da diese bereits durch die Endpunkte der Beschleunigungsphasen und den Startpunkten der Bremsphasen bekannt sind.

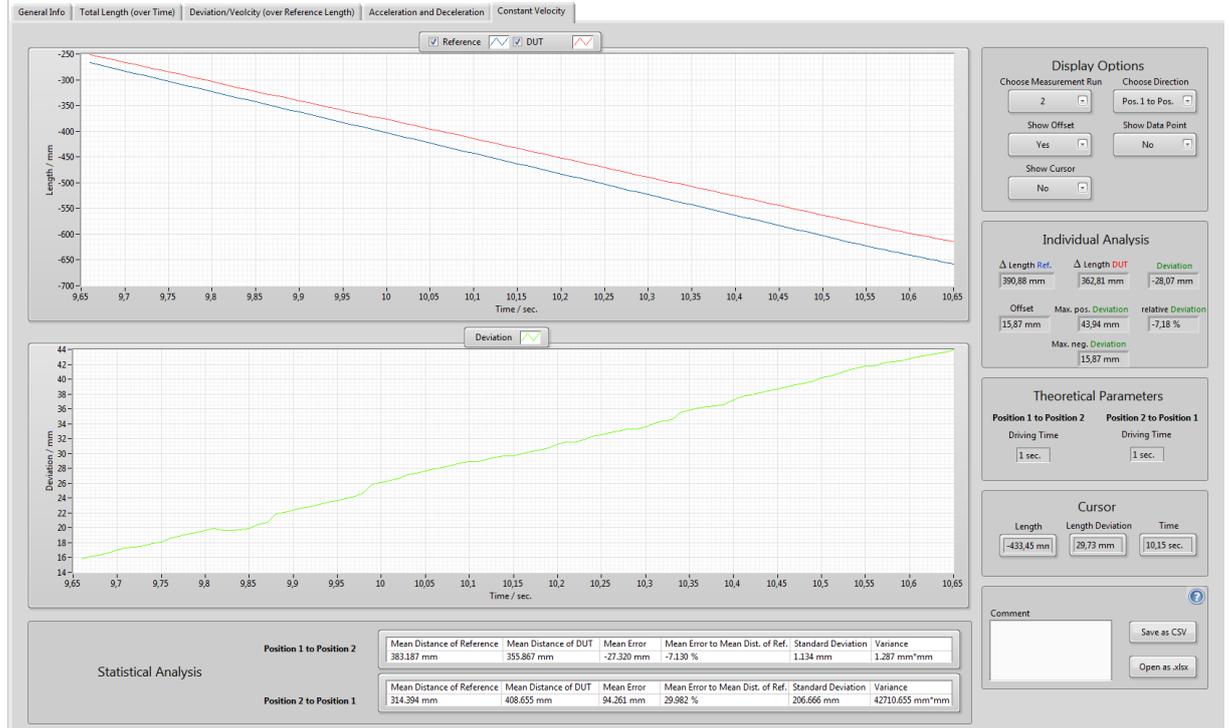


Abbildung 4-12: Visualisierung der Auswertung der Fahrt mit konstanter Geschwindigkeit in der GUI

Die Auswertung erfolgt nach dem gleichen Prinzip wie bei dem Vergleich der Beschleunigungs- und Bremsvorgänge. Die Visualisierung ist in der GUI unter der Registerkarte „Constant Velocity“ zu finden (Abbildung 4-12).

4.6 CSV & Excel Protokoll

Zur Speicherung der Auswertungen besteht die Möglichkeit, die jeweiligen Daten in Form eines Excel Protokolls im .xlsm Format oder als CSV-Datei im .dat Format umzuwandeln. Die Bedienung befindet sich in jeder Registerkarte der Auswertung unten rechts (Abbildung 4-13).

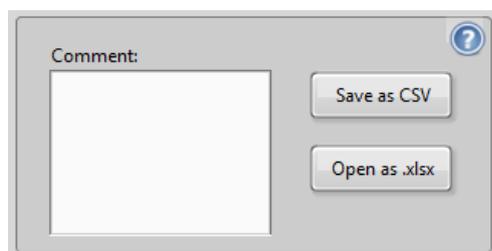


Abbildung 4-13: Bedienfeld für das Excel Protokoll und die CSV Datei

In dem Fenster „General Info“ wird über die Buttons „Open as .xlsm“ und „Save as CSV“ die statistische Auswertung in das jeweilige Format umgewandelt. Bei Auswertungen der gesamten Fahrstrecke, der Beschleunigung, der Bremsung oder der Fahrt mit konstanter Geschwindigkeit, wird die im Graphen der GUI eingestellte Anzeige dargestellt. Im Excel Format erfolgt hier die graphische, sowie die tabellarische Darstellung der Messwerte. Im CSV Format lediglich die tabellarische.

4.6.1 Umsetzung CSV

Die Darstellung im Comma-Separated-Value (CSV) Format ist eine einfache Darstellung von Tabellen. Das Format nutzt definierte Trennzeichen zur Unterscheidung von Zeilen und Spalten. In diesem Fall werden die Trennzeichen „Zeilenumbruch“ und „Tabulator“ genutzt. CSV-Dateien eignen sich für den Export von einfachen Datensätzen. Diese können von Programmen wie Excel oder Matlab importiert werden. Somit wird dem Nutzer das Exportieren von einzelnen Messdaten und Auswertungen zur weiteren Analyse ermöglicht und vereinfacht.

Die erstellte CSV-Datei besitzt einen Header mit Informationen zu den Messdaten. Dieser Header dient dazu, die Messdaten einer bestimmten Messung zuzuordnen und den Bewegungsablauf zu erkennen. Der Aufbau für die individuellen Auswertungen ist in Tabelle 4-4 dargestellt.

Bezeichnung:	Bedeutung:
File path to original measuring data:	Dateipfad zu den originalen Messdaten.
Measurement Run:	Angabe zum aktuellen Messdurchlauf der Messung
Movement Process:	Betrachteter Bewegungsablauf: „Acceleration“, „Deceleration“ oder „Constant Velocity“
Movement Direction:	Bewegungsrichtung: „Position 1 to Position 2“ oder „Position 2 to Position 1“
Delta Length Reference:	Gibt die im Betrachtungszeitraum zurückgelegte Strecke der Referenz in mm an.
Delta Length DUT:	Gibt die im Betrachtungszeitraum zurückgelegte Strecke des DUT in mm an.
Deviation:	Gibt die Abweichung der DUT-Strecke zur Referenz-Strecke in mm an.
Deviation [%]:	Gibt die Abweichung der DUT-Strecke zur Referenz-Strecke im Bezug zur Referenz-Strecke in % an.
Offset:	Gibt die Abweichung von dem DUT zur Referenz zum Startzeitpunkt der Betrachtung in mm an.
Max. Positive Deviation	Gibt die maximale positive Abweichung während der Betrachtungszeit in mm an.
Max. negative Deviation	Gibt die maximale negative Abweichung während der Betrachtungszeit in mm an.

Tabelle 4-4: Aufbau des Headers in der CSV-Datei

Dem Header folgen die Messdaten von der Referenz und dem DUT in Millimeter, die Werte der Abweichungen in Millimeter und der Zeit in Sekunden. Die Aufteilung ist beispielhaft in Tabelle 4-5 dargestellt.

Ref.	DUT	Dev.	Time
-0,628	-0,936	-0,308	0,15
-0,757	-0,702	0,055	0,16
-0,899	-0,468	0,431	0,17
...

Tabelle 4-5: Anordnung der Messdaten unter dem Header

Die CSV-Datei wird mit Hilfe einer LabVIEW Funktion erstellt, die es erlaubt Arrays als Tabelle in eine Datei zu schreiben. Da dieses Arrays aus Strings besteht, werden alle Zahlenwerte zuvor zu Strings konvertiert. Die CSV-Dateien werden unter den Namen der betrachteten Bewegungsabläufe und einer fortlaufenden Nummerierung gespeichert. Der Speicherort ist fest eingestellt und befindet sich auf dem Desktop des Host-PCs. Wird die CSV-Datei über die Registerkarte „General Info“ erstellt, enthält diese die statistischen Werte aus den Einzelbetrachtungen.

4.6.2 Umsetzung Excel Protokoll

Das Excel Protokoll beinhaltet die gleichen Informationen wie die CSV-Datei. Sie enthält zusätzlich noch eine graphische Darstellung der Daten. Dies dient dazu, die Auswertungen inklusive der Visualisierung zu speichern. Bei den Protokollen der individuellen Auswertung wird die aktuelle Auswahl der Messdaten in einem Diagramm, wie in der GUI abgebildet, dargestellt. Um auch über das Excel Protokoll Zugriff auf die Messdaten zu bekommen, werden diese auf der zweiten Tabellenseite des Dokuments dargestellt.

Für die Umsetzung wurde eine Protokollvorlage im Excel Format erstellt und die von LabVIEW bereitgestellten VIs zur Bearbeitung von Excel Dokumenten genutzt. Wird der Button „Open as .xlsx“ betätigt, wird die Vorlage geöffnet und in einer Sequenzstruktur mit den entsprechenden Daten gefüllt. Die Vorlage enthält alle Bezeichnungen und Formen für das spätere Protokoll, sodass nur die Daten eingefügt werden müssen. Da die automatische Achsenanpassung in Excel unzureichend ist, wird zudem über LabVIEW ein Visual Basic Code als Makro zur Achsenanpassung aktiviert. In Abbildung 4-14 ist ein Protokoll für eine Beschleunigungsphase von Position 2 zur Position 1 dargestellt. Weitere Beispiele befinden sich im Anhang dieser Arbeit.

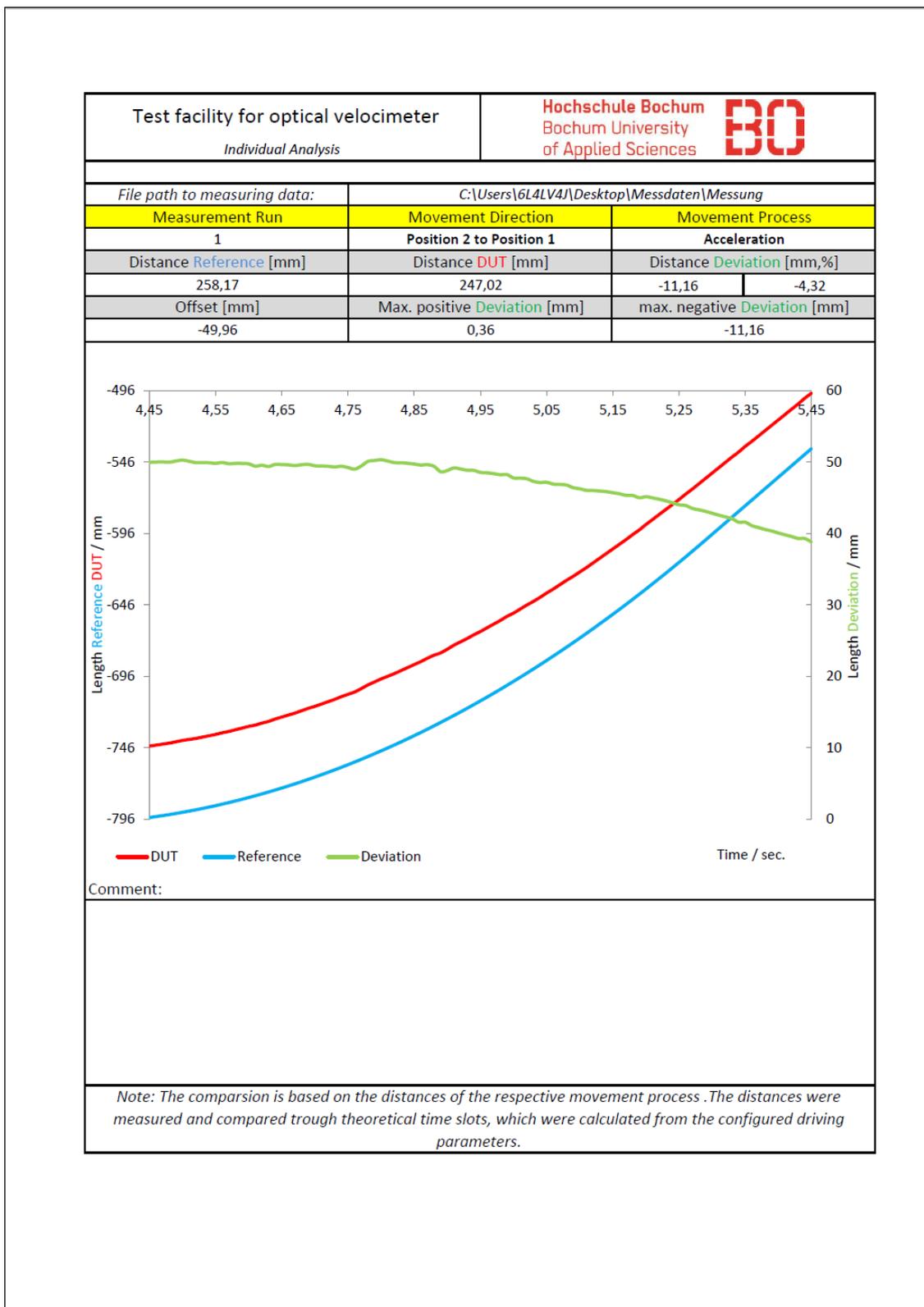


Abbildung 4-14: Beispiel eines automatisch erstellten Excel Protokolls für eine Beschleunigungsphase

5 Echtzeitfähiges System

Eine weitere Anforderung an die Weiterentwicklung des Teststandes war die Umstellung auf ein echtzeitfähiges System. Hierzu stand bereits ein Embedded Controller der Firma National Instruments zur Verfügung. Dieser Controller ist echtzeitfähig und unterstützt die Software Bibliothek *RealTime Module* von NI [21]. Er löst die bisherige Steuerung über den Teststand-PC, mittels einer PCI Express Schnittstelle zum PXI Chassis, ab.

5.1 Hardware & Inbetriebnahme

Der im Rahmen der Weiterentwicklung eingebaute, echtzeitfähige Controller (PXIe-8100 RT) wird von NI als RealTime Target bezeichnet. Er ist für den Einbau in ein PXI System konstruiert und verfügt über ein *Intel® Atom™* Prozessor. [22]

Es lassen sich verschiedene Betriebssysteme auf diesem Controller installieren. Im Einsatz befindet sich das von der Firma Phar Lap Inc., speziell für LabVIEW, entwickelte Echtzeitbetriebssystem Phar Lap ETS [23]. Der Einbau erfolgt im System Slot des PXI Chassis.

Zur Inbetriebnahme war eine Formatierung und Neuinstallation des Controllers erforderlich. Die Vorgehensweise dieser Neuinstallation wurde zur Dokumentation in einer Anleitung zusammengefasst (siehe Anhang).

5.2 Struktur

Die Umstellung hatte zur Folge, dass die Struktur der Software geändert wurde. Das Programm läuft nun auf dem Controller. Da dieser nicht zur Visualisierung fähig ist, erfolgt die Visualisierung und Bedienung weiterhin auf dem Teststand-PC. Dadurch wurde es erforderlich, das Programm aufzuteilen. Aus dieser Aufteilung entsteht ein Netzwerk aus dem RealTime Target und dem Teststand-PC, welcher aufgrund der Beherbergung der Software und Steuerung des Netzwerkes weiterhin als Host-PC bezeichnet wird.

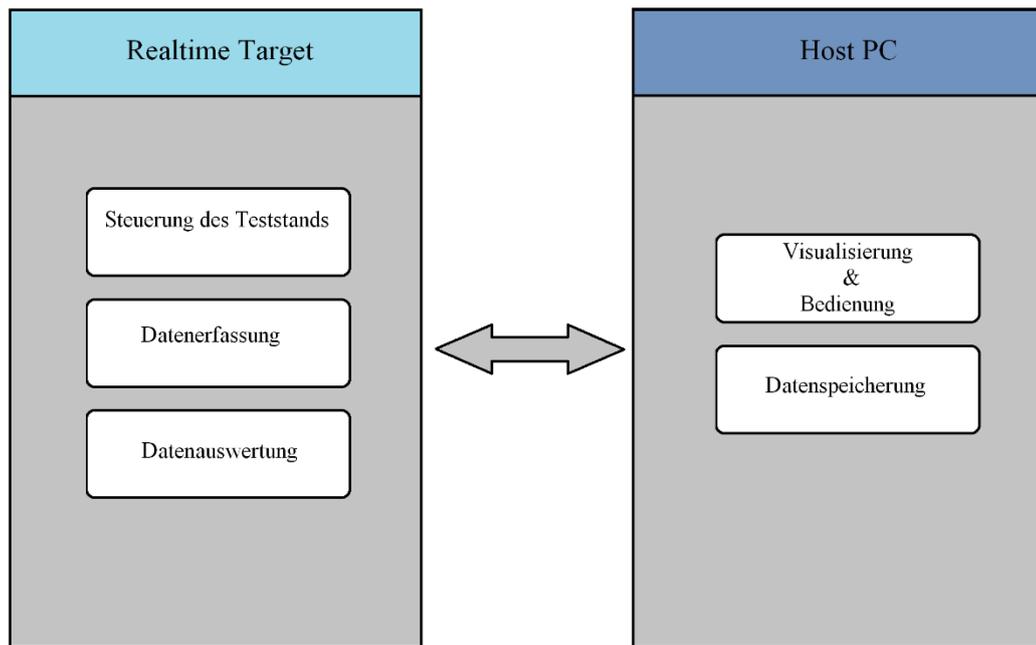


Abbildung 5-1: Skizzierte Struktur des Systems aus Host-PC und Realtime Target

Durch dieses Netzwerk ist es möglich, Programmabschnitte auf zwei verschiedenen Systemen gleichzeitig auszuführen. Die Aufteilung des Programms ist teils variabel, was eine bessere Optimierung der Auslastung zulässt. Die entstandene Struktur ist in Abbildung 5-1 dargestellt. Der Host-PC visualisiert das Programm auf dem Target, das Programm wird weiterhin über ihn gestartet und bedient. Auch die Speicherung der Messdaten und Protokolle wird vom Host-PC in einem separaten Programm übernommen. Die Strukturen der Steuerung des Teststandes, der Datenerfassung und der Datenauswertung wurden weitestgehend beibehalten und laufen nun über das RealTime Target.

5.3 Kommunikation

Die Kommunikation zwischen dem RealTime Target und dem Host-PC erfolgt über eine Ethernet Verbindung. Der Datenaustausch wird hierbei automatisch über LabVIEW realisiert.

LabVIEW bietet mehrere Möglichkeiten zum Austausch von Daten zwischen zwei LabVIEW Anwendungen an. Eine Möglichkeit sind Network-Streams. Ein Network-Stream ist ein unidirektionaler Kommunikationskanal. Der Datenaustausch erfolgt mit

der Network-Stream-Engine (NSE) und Sende- und Empfangsendpunkten. Diese Endpunkte müssen in der LabVIEW Anwendung definiert werden. Erst wenn ein Sendeendpunkt mit einem Empfangsendpunkt verbunden ist, entsteht ein Stream. Jeder Endpunkt verwendet zur Datenübertragung einen „First In - First Out“ (FIFO) Puffer. Der Sendeendpunkt schreibt die Daten in seinen FIFO Puffer, dieser wird über die NSE an den FIFO Puffer des Empfangsendpunkts gesendet und dort ausgelesen. Mit Hilfe eines Network-Streams können alle gängigen Datentypen in LabVIEW verschickt werden. [24]

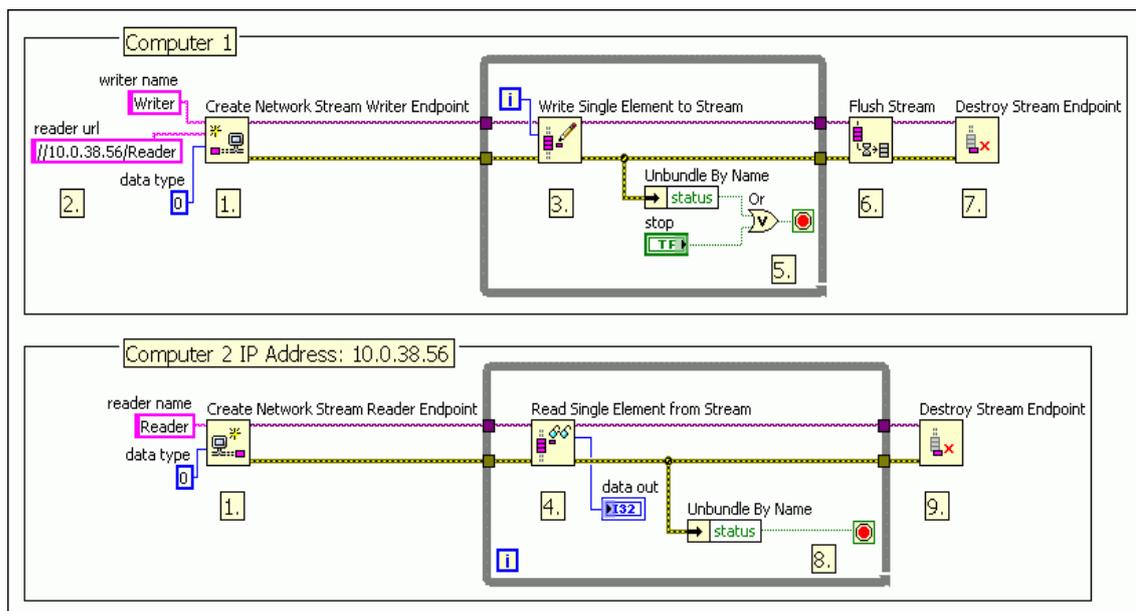


Abbildung 5-2: Beispiel einer Network-Stream Anwendung [24]

In Abbildung 5-2 ist eine Beispielanwendung von Network-Stream dargestellt. Die gezeigte Anwendung dient zur Übertragung eines Integer-Wertes von einem Computer zu einem anderen Computer innerhalb eines Netzwerkes.

Eine weitere Möglichkeit zum Datenaustausch ist der Zugriff auf Umgebungsvariablen innerhalb eines Netzwerkes. Der Austausch erfolgt über die Engine für Umgebungsvariablen Shared Variable Engine (SVE). Die Umgebungsvariablen werden zunächst in einer Bibliothek angelegt. Um diese Variablen im Netzwerk öffentlich zu machen, werden sie an die SVE übertragen. Bei Umgebungsvariablen kann zudem die Echtzeit FIFO Funktion aus dem RealTime Modul aktiviert werden. [25]

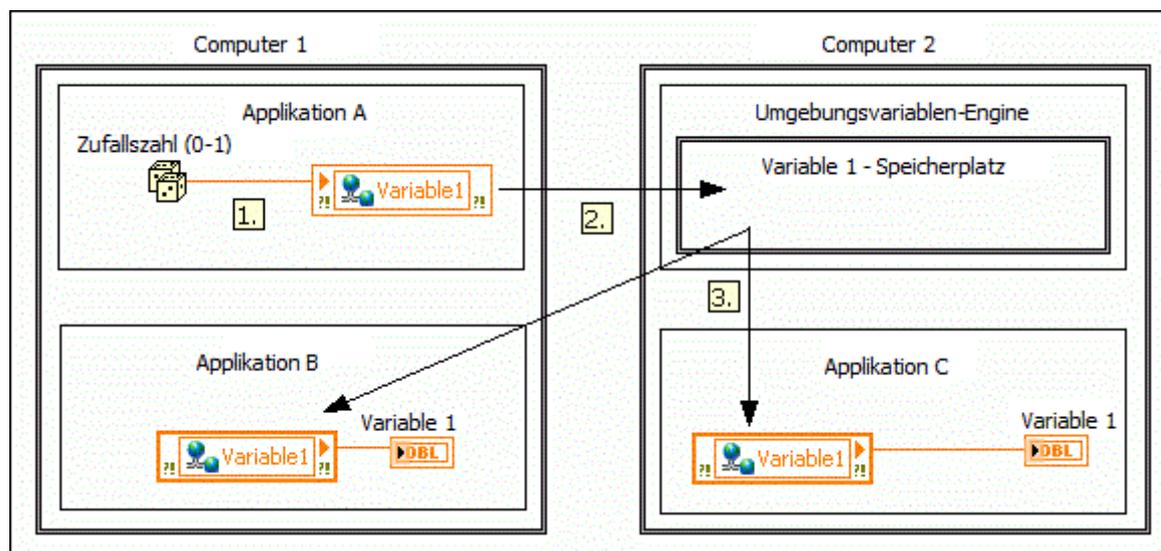


Abbildung 5-3: Beispiel einer Umgebungsvariablen-Anwendung [25]

Abbildung 5-3 zeigt ein Beispiel für Umgebungsvariablen. In diesem Fall wird eine Zufallszahl in einem Netzwerk zugänglich gemacht.

5.4 Anpassung des Systems

Vor der Realisierung des Projektes wurden die Hauptschleifen des vorhandenen Programmes ohne Steuerung der Zykluszeiten realisiert. Hierdurch wurden die Schleifen ungebremst ausgeführt. Dies hatte zur Folge, dass die CPU-Auslastung des neuen Systems zu hoch wurde. Um das Programm auf dem neuen System lauffähig zu machen, war es nötig die Hauptschleifen „auszubremsen“. Hierzu bietet LabVIEW die Funktion einer zeitgesteuerten Schleife an.

Zeitgesteuerte Schleifen ermöglichen neben der Anpassung der Zykluszeit noch weitere Funktionen zur Optimierung der Anwendung. So lassen sich Prioritäten einzelner Schleifen festlegen. Bei mehrkernigen Prozessoren besteht die Möglichkeit, einzelne

Kerne für eine Schleife zu reservieren. In der aktuellen Version des Programmes wurde der Datenerfassung die höchste Priorität zugeordnet. Die Datenauswertung musste aufgrund knapper Ressourcen und ihrer niedrigen Priorität in Bezug zur Lauffähigkeit verlangsamt werden. In diesem Fall zeigt sich, dass der Einsatz einer leistungsstärkeren CPU vom Vorteil wäre (für weitere Informationen siehe Kapitel 6.4). Um die CPU-Auslastung darzustellen, wurde auf dem Controller die Software *Real-Time System Manger* installiert. Die CPU-Auslastung wird in einer Umgebungsvariable gespeichert und kann zudem beim Anschluss eines Monitors an den VGA Ausgang des Controllers, graphisch dargestellt werden.

Die Datenspeicherung wurde mit Hilfe von Umgebungsvariablen in einen VI auf dem Host-PC ausgelagert. Um die Datenspeicherung zu nutzen, ist es daher notwendig dieses VI auf dem Host-PC auszuführen.

6 Mögliche Weiterentwicklungen

6.1 Einbindung eines absoluten Weggebers

Die Anforderung eines zweiten Wegsensors zur Unterstützung des Referenzsystems konnte aufgrund von Lieferschwierigkeiten nicht umgesetzt werden. Die Umsetzung wird daher als mögliche Weiterentwicklung beschrieben.

Die Wahl fiel auf einen Seilzuggeber der Firma Posital. Er ist mit einer SSI Schnittstelle (Synchron Serielle Schnittstelle) und einer Inkrementellen Schnittstelle ausgestattet [26]. Dieser eignet sich für das Projekt aufgrund seiner hohen Auflösung. Zudem könnten beide Schnittstellen durch die vorhandenen Ein- und Ausgänge des PXI Systems (PXIe-6356) realisiert werden.

Ein zweiter Wegsensor kann zum einem das Referenzsystem unterstützen oder dieses überwachen. Darüber hinaus könnte dem Nutzer eine Auswahlmöglichkeit zur Verfügung stehen, welcher Sensor als Referenz genutzt werden soll. Weiterhin besitzt der Servoverstärker die Möglichkeit einen externen Sensor als Weggeber zu nutzen. Hierdurch könnte auf eine Referenzfahrt zur Kalibrierung verzichtet werden.

Eine mögliche Umsetzung einer SSI Schnittstelle in LabVIEW findet sich im Anhang.

6.2 Einbindung eines Framegrabbers

Im Chassis des Teststandes ist eine Framegrabber-Karte installiert (PXIe-1435) mit Hilfe dieser Karte lassen sich Bilder einer Industriekamera in LabVIEW einlesen. Eine mögliche Weiterentwicklung wäre es, die Bildaufnahme in LabVIEW zu implementieren. Dadurch könnten parallel zur Messwertaufnahme eines optischen Sensors und des Referenzsystems die zugehörigen Bilder des Maßsystems abgespeichert werden.

Der vorhandene Framegrabber besitzt eine Camera Link Schnittstelle, sowie die Möglichkeit der Spannungsversorgung über „Power over Camera Link“ [27].

Zur Erkennung einer angeschlossenen Kamera ist es notwendig, den passenden Treiber in LabVIEW zu installieren. Hierzu stellt National Instruments eine Online- Datenbank zur Verfügung [28]. Dieser Treiber wird Camera File genannt. Lässt sich keine Camera

File für die benutzte Kamera finden, kann über die Software „NI Camera File Generator“ eine eigene Camera File erstellt werden [29].

6.3 Einbindung einer Höhenverstellung

Der bei der Vorstellung des Teststandes erwähnte „Galgen“ wird über eine manuelle Spindel in der Höhe verstellt. Eine denkbare Erweiterung wäre es, diese Höhenverstellung über einen Schrittmotor und ein Abstandmessgerät zu steuern. Hierdurch könnte eine unregelmäßige Arbeitshöhe, die in der Realität durchaus vorkommen kann, simuliert werden. Die Arbeitshöhe könnte zudem ebenfalls in den Messdaten erfasst werden, um den Einfluss der Arbeitshöhe auf den Sensor zu untersuchen.

6.4 Einsatz einer leistungstärkeren CPU

Bereits bei der Umstellung des Systems auf einen echtzeitfähigen Controller hat sich gezeigt, dass die CPU des Controllers unterdimensioniert ist. Auch unter Berücksichtigung der eventuellen Einbindung eines Framegrabbers wäre der Einsatz einer leistungstärkeren CPU notwendig, da hier mit wesentlich höheren Datenraten zu rechnen ist.

Mit der Unterstützung des technischen Supports von National Instruments fiel die Wahl auf einen Controller mit einem i5 4400e Prozessor (PXIe 8840). Neben der höheren Taktfrequenz hat dieser Prozessor den Vorteil, dass er aus zwei Kernen besteht [21]. Dies ermöglicht die Verteilung verschiedener Schleifen auf die Kerne, zum Beispiel könnte ein Kern nur für die Datenerfassung reserviert sein. In diesem Fall wäre eine komplette Überarbeitung und Aufteilung der Software zwischen Target und Host-PC vorteilhaft.

6.5 Automatische Erkennung des Bewegungszustandes

In Kapitel 4.1 wurde das Verfahren zur Aufschlüsselung der Messdaten anhand der theoretischen Zeitfenster beschrieben. Hier besteht der Nachteil, dass die theoretischen Zeitfenster nicht den exakten Zeitfenstern in der Praxis entsprechen. Diese Diskrepanz hat mehrere Ursachen. So arbeitet der Servoverstärker mit Toleranzen in Bezug auf die Sollgeschwindigkeit, der Positionierung und der Zeit für die Rückmeldung. Aber auch

der Schleppfehler der linearen Achse sowie die Zykluszeiten spielen eine Rolle. Durch diese Nichtübereinstimmung der Zeitfenster können relevante Messdaten eines Bewegungsablaufes „abgeschnitten“ sein.

Eine alternative Lösung ist die Implementierung einer automatischen Erkennung. Hierzu wird zum Beispiel für jeden Messwert im Messdaten-Array die Beschleunigung a_i [30] berechnet (Gleichung (40)). Zur Berechnung muss die zurückgelegte Strecke Δs innerhalb einer gewählten Abtastzeit t_s bestimmt werden.

$$a_i = \frac{2\Delta s}{t_s^2} \quad (40)$$

Diese Strecke kann mit Gleichung (41) und die in Indizes umgerechnete Abtastzeit Δi (Gleichung (42)) aus den Referenzdaten des Messdaten-Array bestimmt werden.

$$\Delta s = x_{i,Ref} - x_{i-\Delta i, Ref} \quad (41)$$

$$\Delta i = \text{Sample Rate} \cdot t_s \quad (42)$$

Anschließend lässt sich mit booleschen Operatoren überprüfen, ob eine Beschleunigung vorliegt. Für die Bedingung nach Gleichung 43 könnten somit positive Beschleunigungen erkannt werden, die eine noch zu bestimmende Toleranz-Beschleunigung a_{min} überschreiten.

$$a_i > a_{min} \quad (43)$$

I Abkürzungsverzeichnis

NI	National Instruments
GUI	Graphical User Interface
PCI	Peripheral Component Interconnect
CSV	Comma Separated Values
DUT	Device Under Test
LDV	Laser-Doppler-Velocimetrie
LDA	Laser-Doppler-Anemometrie
LSV	Laser-Surface-Velocimeter
FBS	Funktionsbausteinsprache
SPS	Speicherprogrammierbare Steuerung
VI	Virutal Instrument
DUT	Device Under Test
CAN	Controller Area Network
DBL	Double
NSE	Network Stream Engine
FIFO	First In First Out
SVE	Shared Variable Engine
SSI	Synchron Serielle Schnittstelle

II Abbildungs- und Tabellenverzeichnis

Abbildung 1-1: Teststand für optische Geschwindigkeitssensoren	5
Abbildung 3-1: Funktionsprinzip eines LDV [12]	10
Abbildung 3-2: Prinzip zur Geschwindigkeitsmessung mit einem Maskierungsgitter [8]	11
Abbildung 3-3: Skizze des Teststands mit linearer Verfahrensachse und optische Achse des DUT	15
Abbildung 4-1: Skizze eines Messdurchlaufes mit Einteilung der Zeitfenster und Orientierungspunkten.....	21
Abbildung 4-2: Beispiel eines Messdaten-Arrays mit gekennzeichneteter Beschleunigungsphase	28
Abbildung 4-3: Screenshot der Benutzeroberfläche mit offener Registerkarte „General Info“	30
Abbildung 4-4: Beispiel eines absoluten Streckenverlaufs	31
Abbildung 4-5: Visualisierung der Auswertung der gesamten Fahrstrecke in der GUI als Registerkarte	32
Abbildung 4-6: Anzeige der Auswertung „Individual Analysis“	33
Abbildung 4-7: Geschwindigkeitsanzeige eines einzelnen Messdurchlaufes	34
Abbildung 4-8: Skizze der Beschleunigungsphasen eines Messdurchlaufes	35
Abbildung 4-9: Skizze der Bremsphasen eines Messdurchlaufes	36
Abbildung 4-10: Visualisierung der Auswertung von Beschleunigung und Bremsung in der GUI	37
Abbildung 4-11: Skizze der Fahrzeiten mit konstanter Geschwindigkeit eines Messdurchlaufes	38

Abbildung 4-12: Visualisierung der Auswertung der Fahrt mit konstanter Geschwindigkeit in der GUI.....	39
Abbildung 4-13: Bedienfeld für das Excel Protokoll und die CSV Datei	41
Abbildung 4-14: Beispiel eines automatisch erstellten Excel Protokolls für eine Beschleunigungsphase	44
Abbildung 5-1: Skizzierte Struktur des Systems aus Host-PC und Realtime Target	46
Abbildung 5-2: Beispiel einer Network-Stream Anwendung [24].....	47
Abbildung 5-3: Beispiel einer Umgebungsvariablen-Anwendung [25]	48
Tabelle 3-1: Bestückung des PXI Chassis PXIe-1078.....	17
Tabelle 4-1: Aufbau des Messdaten-Arrays	22
Tabelle 4-2: Aufbau des Arrays mit Orientierungspunkten des Messdaten-Arrays für das Erreichen einer Position.....	23
Tabelle 4-3: Beispiel eines Arrays mit Orientierungspunkten.....	27
Tabelle 4-4: Aufbau des Headers in der CSV-Datei.....	42
Tabelle 4-5: Anordnung der Messdaten unter dem Header	42

III Quellenverzeichnis

- [1] M. Schaeper, Mehrdimensionale Ortsfiltertechnik, Springer Vieweg, 2014.

- [2] A. Bergmann, Verbesserung des Ortsfrequenzfilterverfahrens zur kamerabasierten Messung der translatorischen Geschwindigkeit bewegter Objekte, Shaker Verlag, 2010.

- [3] G. F. H.-R. Tränkler, Das Ingenieurwissen Messtechnik, Springer Vieweg, 2014.

- [4] T. Kanigowski, Lastenheft: Weiterentwicklung Linearteststand, 2016.

- [5] L. Gehrman, Industrielles Testsystem für optische Velocimeter, 2016.

- [6] Polytech, Funktionsprinzip der Laser Surfave Velocimetrie, 2011.

- [7] H.-R.-. Tränkler und M. Leonhard, Sensortechnik, Springer Verlag, 2014.

- [8] M. Schanz, Eindimensionale CMOS Bildsensorik mit integrierter Singalverarbeitung, 1999.

- [9] M. Linden, Entwicklung und Evaluierung eines serientauglichen LDA-Sensors für die hochdynamische und hochgenaue Messung der Garngeschwindigkeit in der Spulerei, Instiut für Textil- und Verfahrenstrechnik der Deuteschen Institute für Textil- und Faserforschung Denkendorf, 2010.

- [10] Polytech, Datenblatt: LSV-1000 Laser Surface Velocimeter.
- [11] D. Wulff, Fachlabor für Strömungsmaschinen, Pfeiderer-Institut für Strömungsmaschinen, 2005.
- [12] Polytech GmbH, „www.polytec.com,“ [Online]. Available: <http://www.polytec.com/us/solutions/industrial-length-and-speed-control/basic-principles-of-velocimetry/>. [Zugriff am 1 August 2016].
- [13] National Instruments, „www.ni.com/labview,“ [Online]. Available: <http://www.ni.com/labview/why/d/>. [Zugriff am 1 August 2016].
- [14] T. Mühl, Einführung in die elektrische Messtechnik, Springer Vieweg, 2014.
- [15] B. Brinkmann, Internationales Wörterbuch der Metrologie, Beuth Verlag, 2012.
- [16] W. Hässelbarth, BAM-Leitfaden zur Ermittlung von Messunsicherheiten bei quantitativen Prüfergebnissen, Bundesanstalt für Materialforschung und -prüfung, 2004.
- [17] L. Papula, Mathematische Formelsammlung für Ingenieure und Naturwissenschaftler, Vieweg & Sohn Verlag, 2006.
- [18] Siko GmbH, Magnetic Length and Angle Measurement Systems.
- [19] E. Schönfelder und G. . Hering, Sensoren in Wissenschaft und Technik, Vieweg+Teubner Verlag, 2012.

- [20] National Instruments, „www.ni.com/pxi/“, [Online]. Available:
<http://www.ni.com/pxi/whatis/d/>. [Zugriff am 1 August 2016].
- [21] National Instruments, Datasheet : NI PXIe-8840 RT, 2014.
- [22] National Instruments , NI PXIe-8100 RT User Manual, 2011.
- [23] National Instruments, „www.ni.com“, [Online]. Available:
<http://www.ni.com/tutorial/14669/en/#toc1>. [Zugriff am 3 August 2016].
- [24] National Instruments, „zone.ni.com“, [Online]. Available:
<https://zone.ni.com/reference/en-XX/help/371361G-01/lvconcepts/usingstreams/>.
[Zugriff am 3 August 2016].
- [25] National Instruments, „zone.ni.com“, [Online]. Available:
http://zone.ni.com/reference/de-XX/help/371361H-0113/lvconcepts/ni_psp/.
[Zugriff am 3 August 2016].
- [26] Posital Fraba, Datasheet LD0-S5E1B-1213-2P00-AAW.
- [27] National Instruments, NI PXIe-1435 User Guide and Specifications.
- [28] National Instruments, „digital.ni.com“, [Online]. Available:
<http://digital.ni.com/public.nsf/allkb/05DCE3868362783586256FC8004F123C>.
[Zugriff am 3 August 2016].

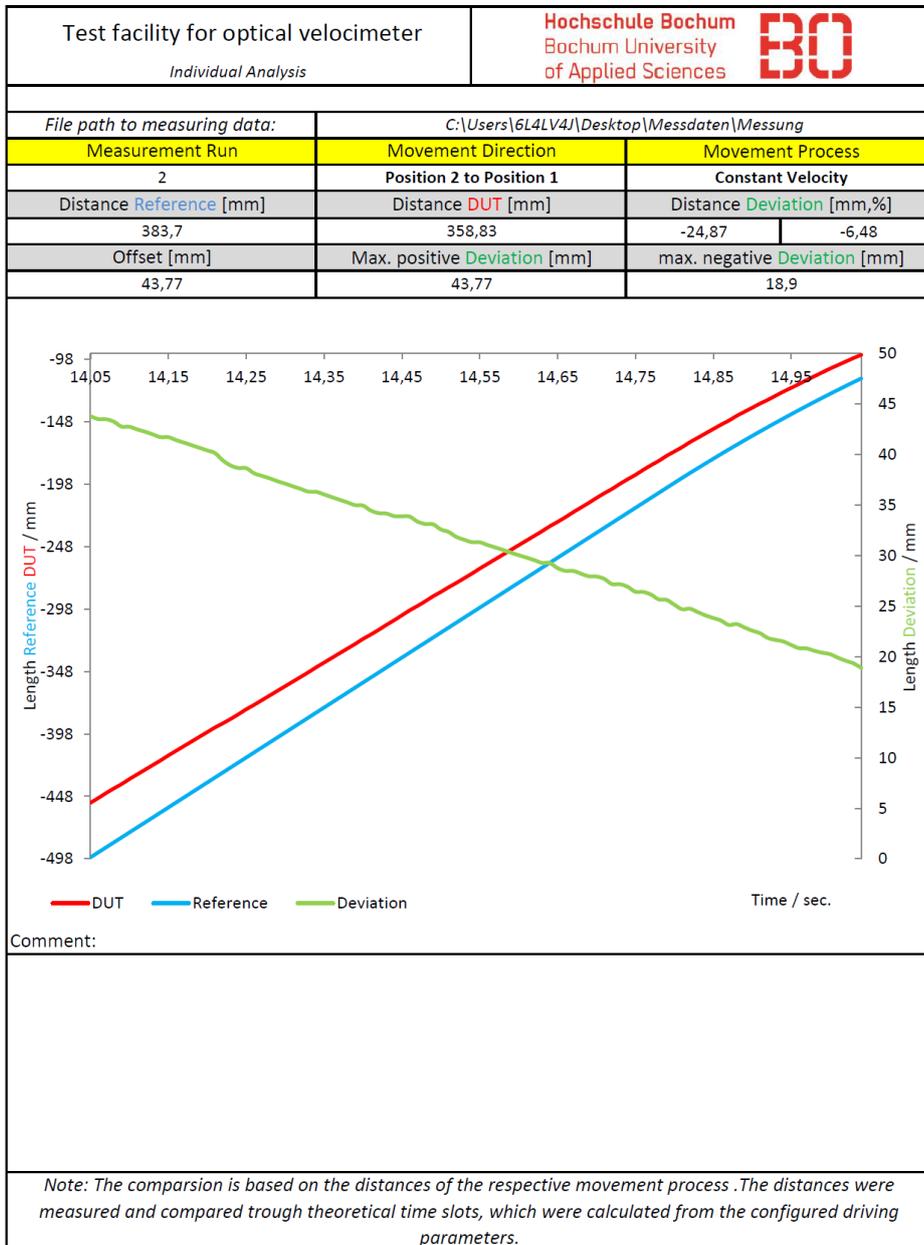
- [29] National Instruments, „sine.ni.com,“ [Online]. Available: <http://sine.ni.com/nips/cds/view/p/lang/de/nid/14207>. [Zugriff am 3 August 2016].
- [30] S. Hesse und G. Schnell, Sensoren für die Prozess- und Fabrikautomation, Vieweg + Teubner Verlag, 2009.
- [31] Hering und E. , Sensoren in Wissenschaft und Technik, 2012: Vieweg + Teubner.
- [32] W. Pläßmann, Handbuch Elektrotechnik, Springer Vieweg, 2013.
- [33] National Instruments, Getting Started with the LabVIEW Real-Time Module.

IV Anhang

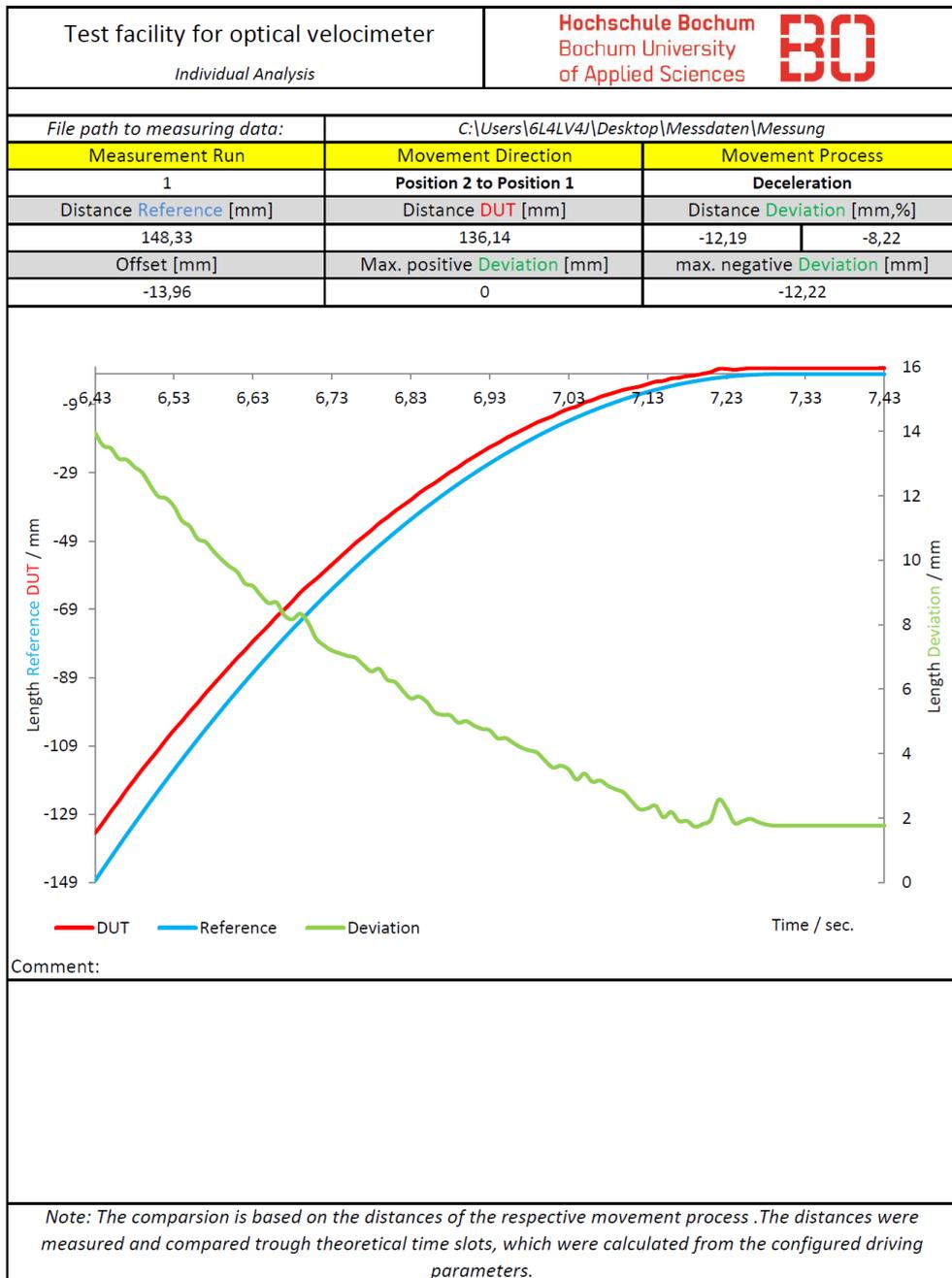
Anhang 1: Excel Protokoll - Auswertung einer Fahrt mit konstanter Geschwindigkeit	61
Anhang 2: Excel Protokoll – Auswertung einer Bremsphase.....	62
Anhang 3: Excel Protokoll – Auswertung des Vergleichs der gesamten Fahrt.....	63
Anhang 4: Excel Protokoll – Auswertung der Abweichung mit Diagramm über die Referenzlänge	64
Anhang 5: Excel Protokoll – Statistische Auswertung mit Allgemeinen Angaben und Vergleich der Gesamtlänge.....	65
Anhang 6: Excel Protokoll – Statistische Auswertung der Beschleunigungsphasen	66
Anhang 7: Excel Protokoll – Statistische Auswertung der Bremsphasen	67
Anhang 8: Excel Protokoll – Statistische Auswertung der Fahrten mit konstanter Geschwindigkeit	68
Anhang 9: Anleitung zur Neuinstallation von LabVIEW RealTime Targets 1 von 2....	69
Anhang 10: Anleitung zur Neuinstallation von LabVIEW RealTime Targets 2 von 2..	70

Anhang CD:

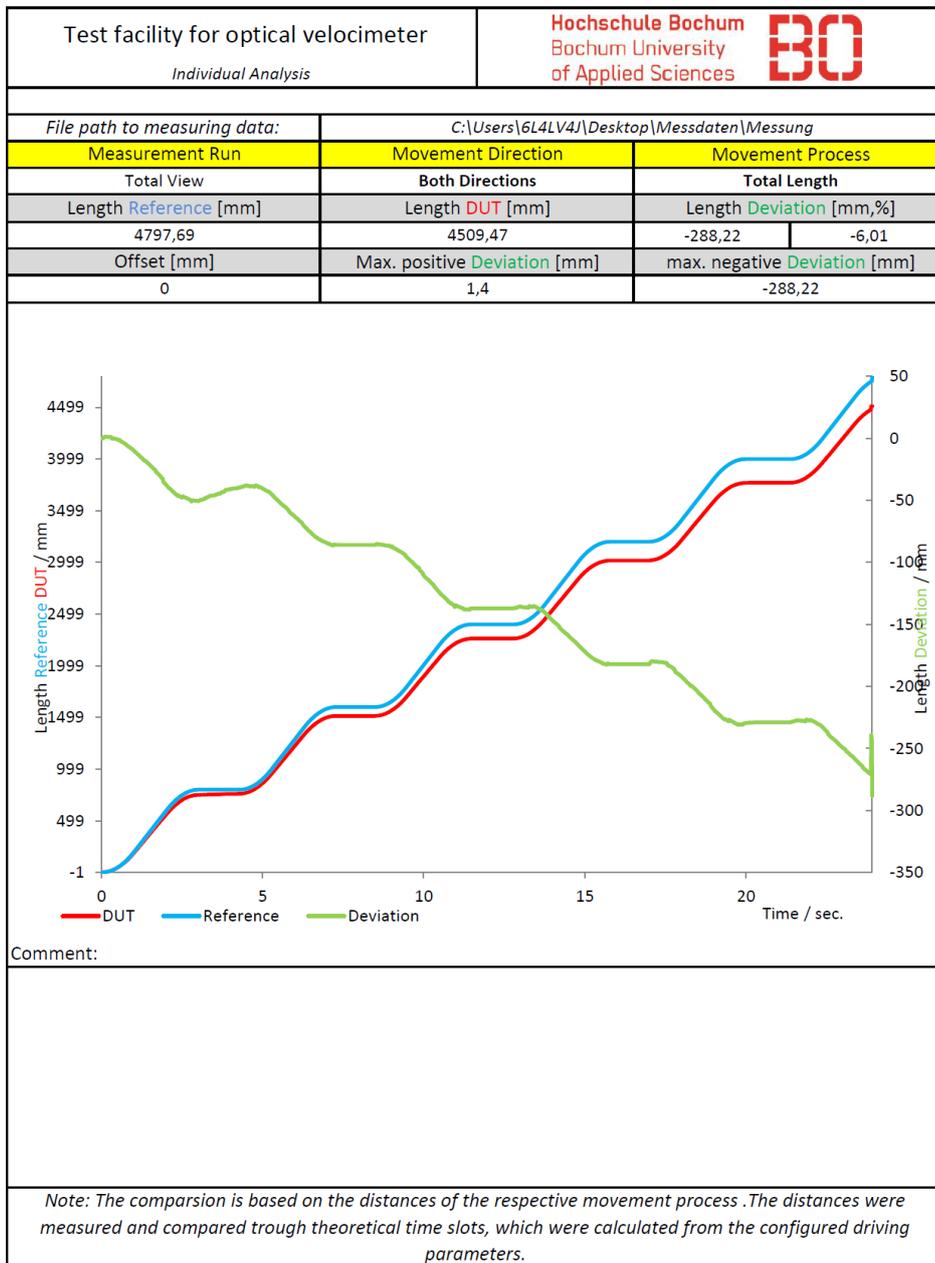
- Datenblätter
- Lastenheft
- Literatur
- Bachelorarbeit
- Beispiel zur SSI Implementierung
- LabVIEW Programm



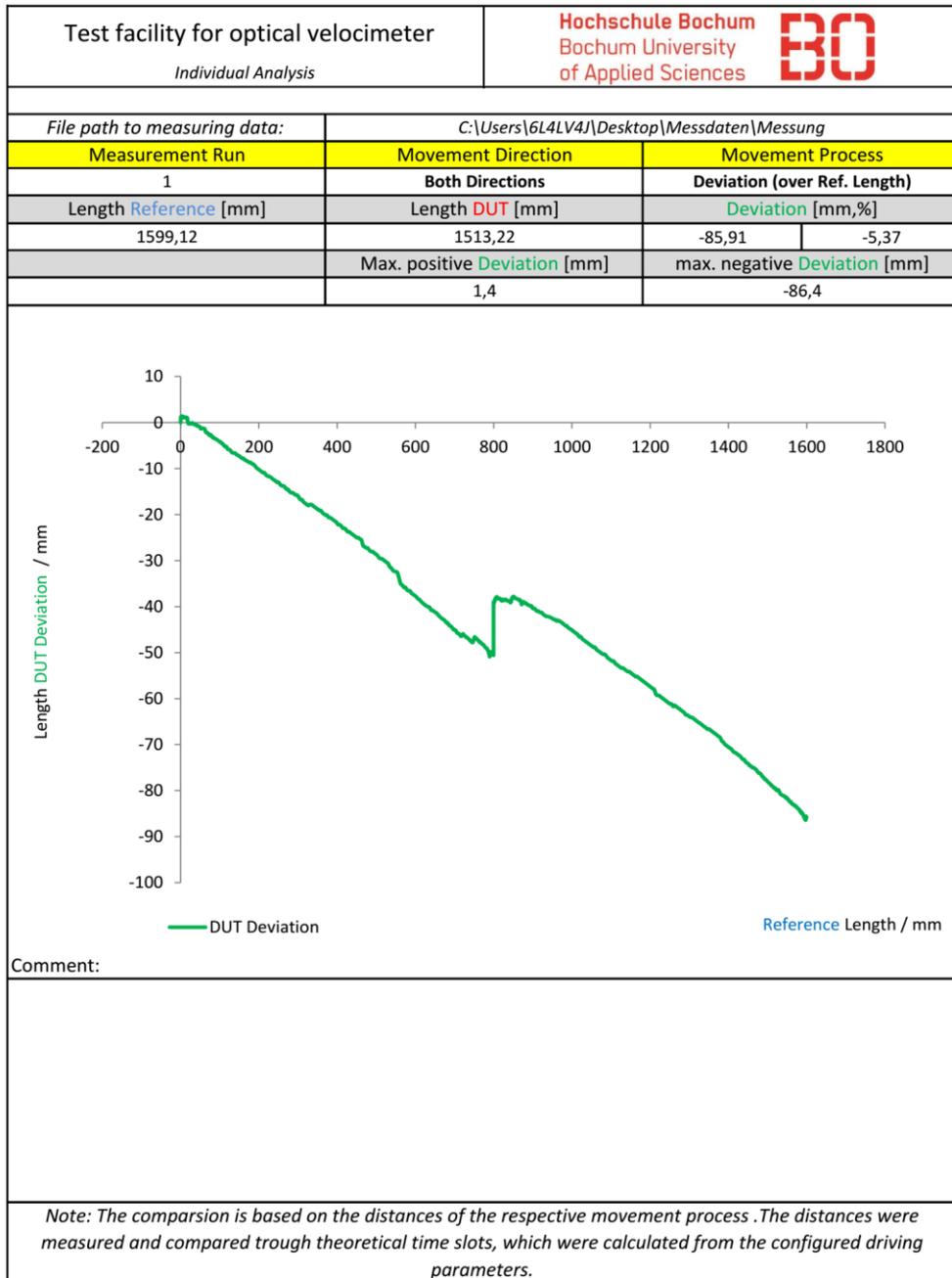
Anhang 1: Excel Protokoll - Auswertung einer Fahrt mit konstanter Geschwindigkeit



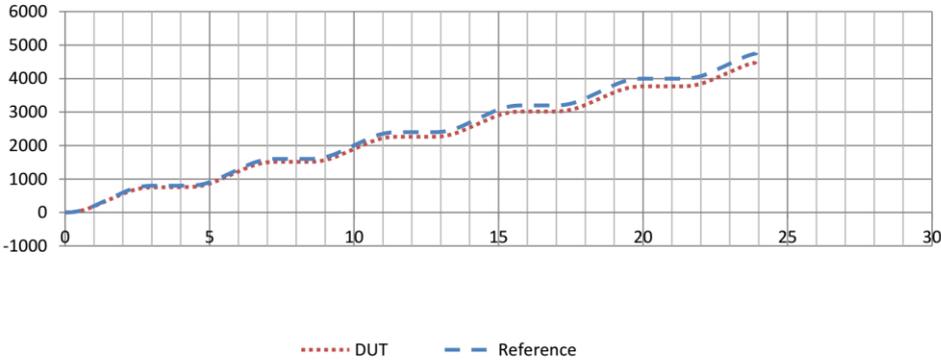
Anhang 2: Excel Protokoll – Auswertung einer Bremsphase



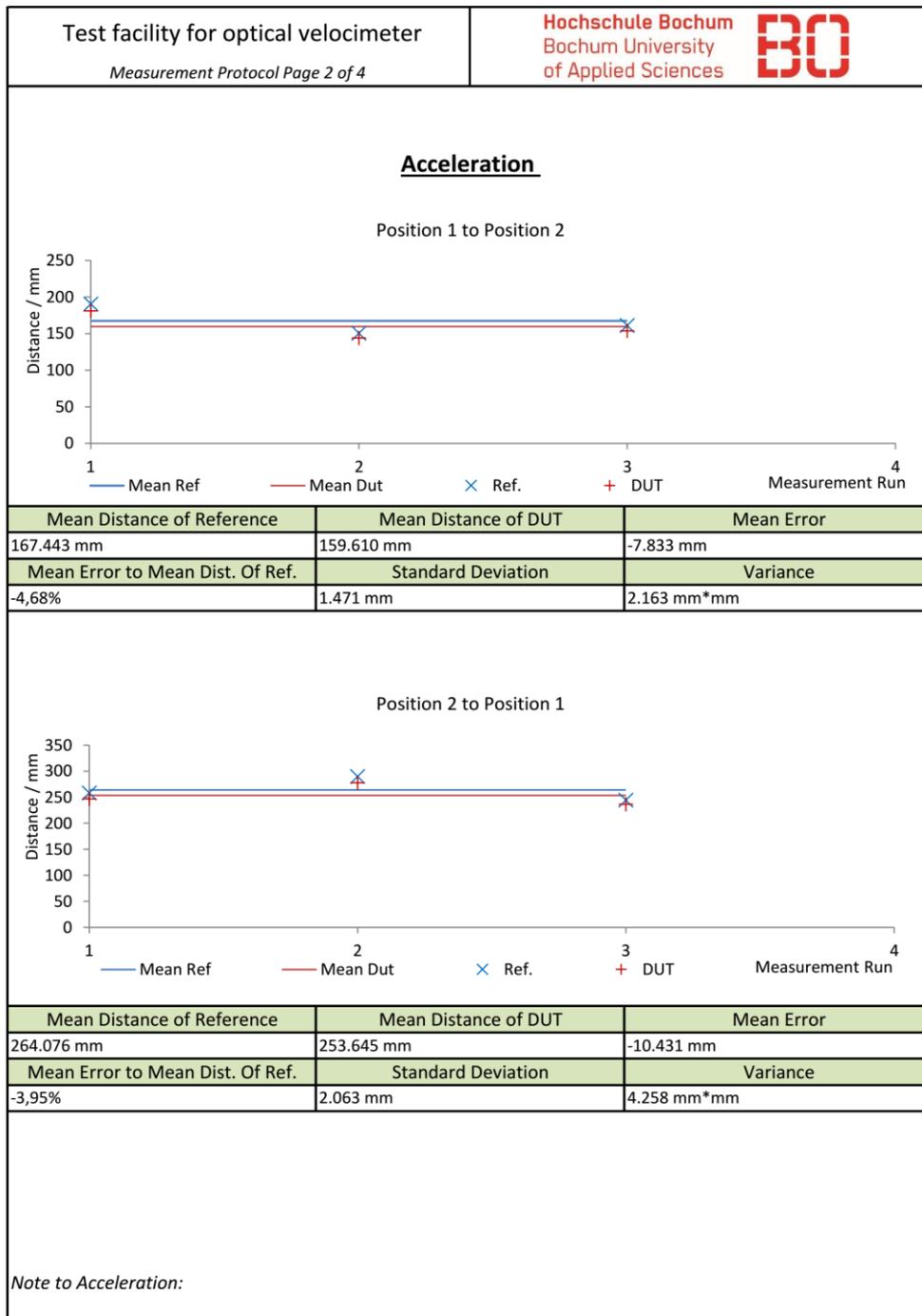
Anhang 3: Excel Protokoll – Auswertung des Vergleichs der gesamten Fahrt



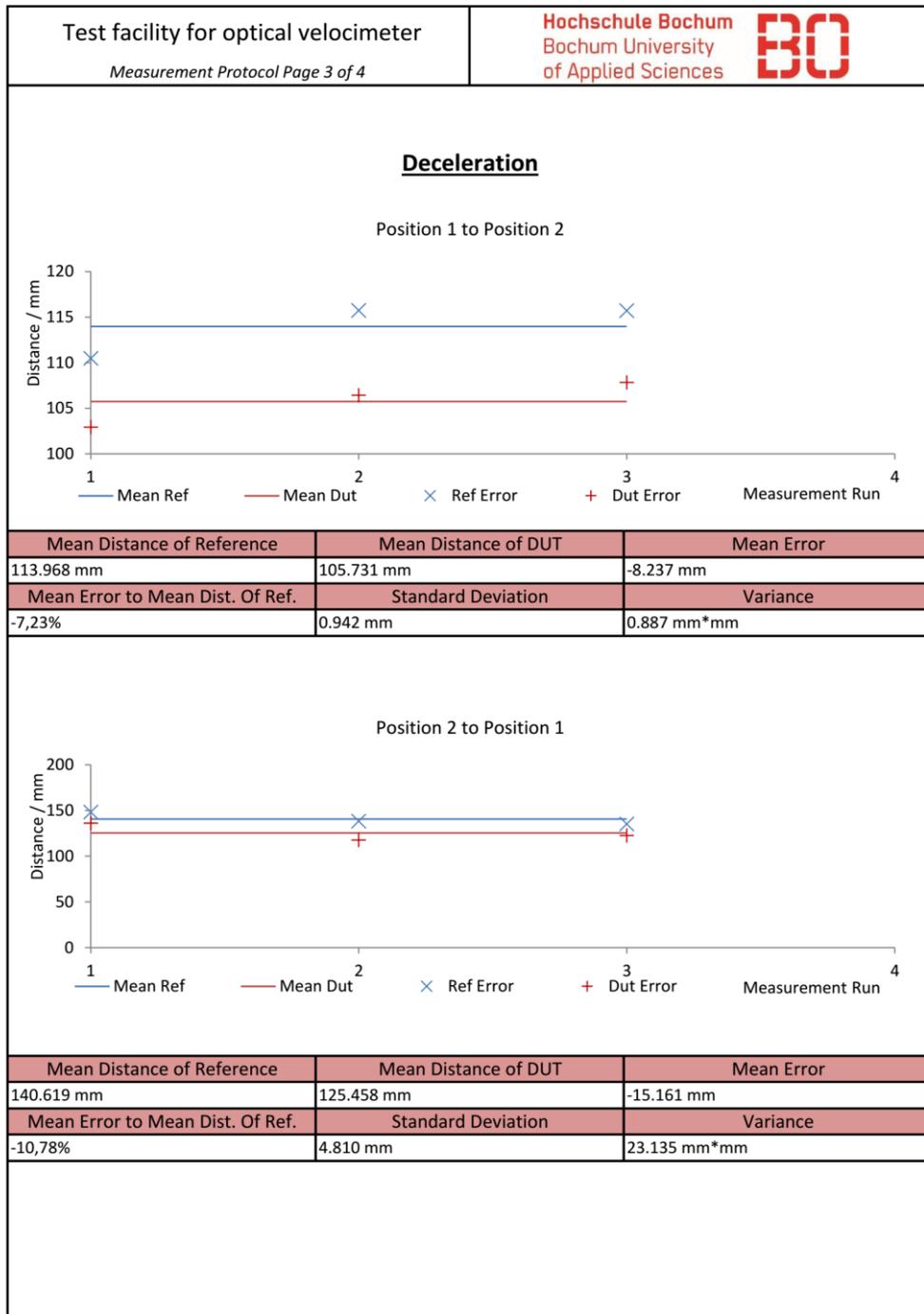
Anhang 4: Excel Protokoll – Auswertung der Abweichung mit Diagramm über die Referenzlänge

Test facility for optical velocimeter <i>Measurement Protocol Page 1 of 4</i>			Hochschule Bochum Bochum University of Applied Sciences 		
Measurement:		Measurement Protocol 1.0			
File Path To Measuring Data:		C:\Users\6L4LV4\\Desktop\Messdaten\Messung			
Date	Start Time	End Time	Running Time	Test Facility	Location
02.08.16	11:12:47	11:13:14	00:26	Linear Test Facility	D3-34
Device Under Test	Reference	Resolution DUT	Resolution Ref.	User	
Intacton - Covidis	Siko - MSK 1000	4,275	1000		
Measurement Parameters					
Position 1	Verlocity 1	Acceleration 1	Deceleration 1	Waiting Time 1	Repeatings
400	400	400	400	1	3
Position 2	Verlocity 2	Acceleration 2	Deceleration 2	Waiting Time 2	
-400	400	400	400	1	
Comment					
Total Length					
					
Total Length Ref.	Total Length DUT	Total Deviation	Total Length Error %		
4797,69	4509,47	-288,22	-6,01		

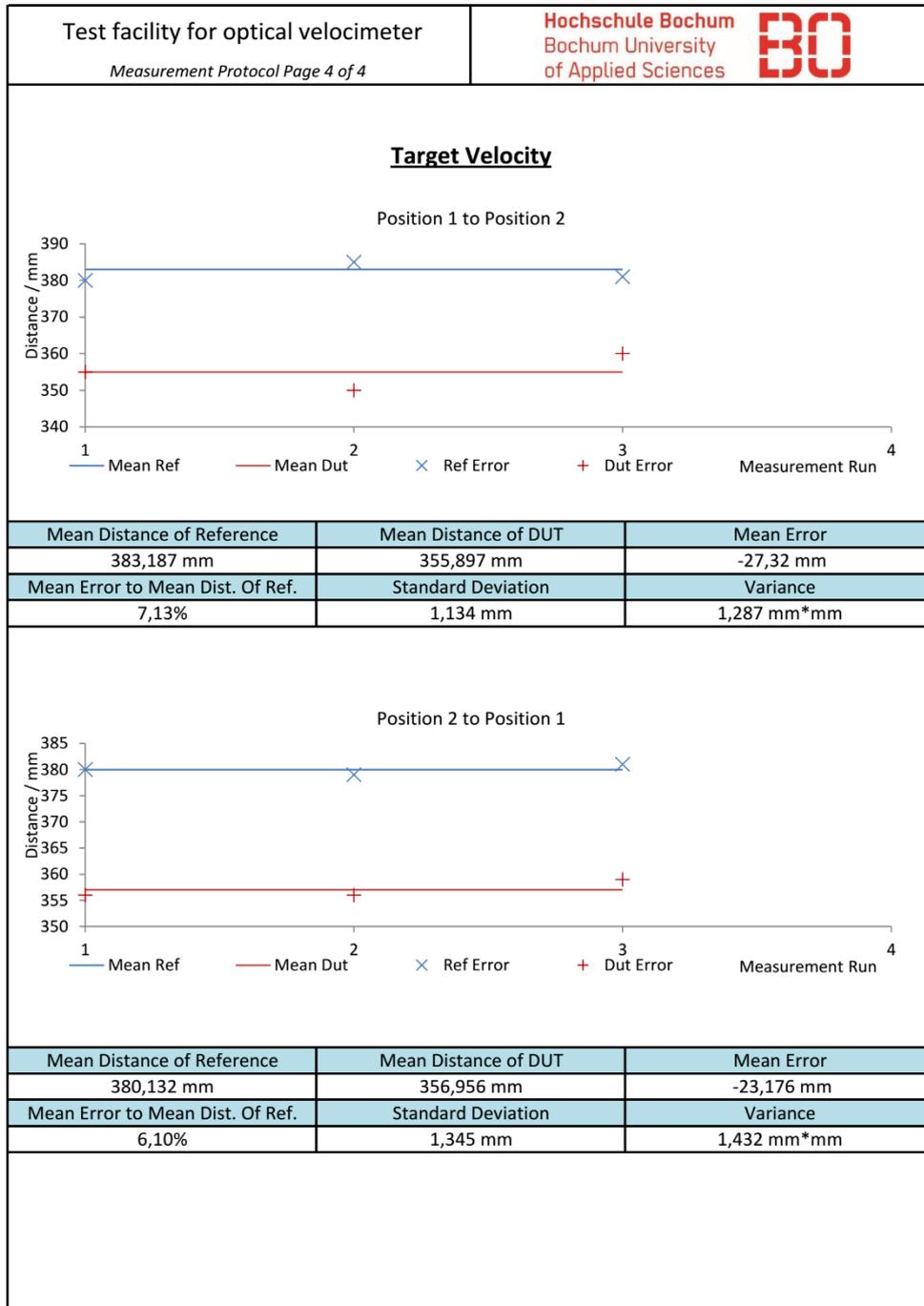
Anhang 5: Excel Protokoll – Statistische Auswertung mit Allgemeinen Angaben und Vergleich der Gesamtlänge



Anhang 6: Excel Protokoll – Statistische Auswertung der Beschleunigungsphasen



Anhang 7: Excel Protokoll – Statistische Auswertung der Bremsphasen



Anhang 8: Excel Protokoll – Statistische Auswertung der Fahrten mit konstanter Geschwindigkeit

<p>Teststand für optische Velocimeter</p>	<p>Anleitung zur Neuinstallation von LabVIEW RealTime Targets Autor : Tobias Kanigowski</p> <p>Seite 1/2</p>	<p>Hochschule Bochum Bochum University of Applied Sciences</p> 
---	--	---

Diese Anleitung dient zur Neuinstallation von PXI Controllern. Sie wurde für den Controller PXIe 8100 RT im Zusammenhang mit dem Linearen Teststand für optische Velocimeter im Labor für elektrische Antriebe D3-32 erstellt. Zur Neuinstallation werden benötigt : PXI Chassis, Monitor, Software NI MAX und die gewünschte Software für den Controller.

Schritt 1 : IP Adresse herausfinden

Zunächst muss der Controller in einem Chassis auf dem System Slot installiert werden. Der System Slot eines Chassis ist mit dem Symbol ▲ gekennzeichnet. Über den VGA Ausgang des Controller wird nun ein Monitor angeschlossen und das Chassis eingeschaltet. Auf dem Monitor wird die IP Adresse des Controllers angezeigt, diese muss notiert werden. Das Chassis kann ausgeschaltet werden.

Schritt 2 : Safe Boot Modus aktivieren

Der Controller muss nun wieder aus dem Chassis ausgebaut werden. Auf dem Controller befinden sich 4 Dip-Schalter (siehe Abbildung 1). Switch 2 mus auf ON geschaltet werden um den Safe Boot Modus zu aktivieren. Anschließend muss der Controller wieder eingebaut werden.

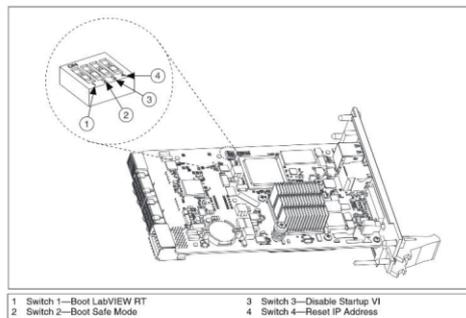


Abbildung 1 : Switch Schalter auf dem Controller Quelle: NI PXIe-8100 User Manual

<p>Teststand für optische Velocimeter</p>	<p>Anleitung zur Neuinstallation von LabVIEW RealTime Targets <i>Autor : Tobias Kanigowski</i></p> <p>Seite 2/2</p>	<p>Hochschule Bochum Bochum University of Applied Sciences </p>
<p>Schritt 3 : System finden</p> <p>Für die weitere Konfiguration muss die Software NI MAX (Measurement & Automation Explorer) gestartet werden. Nachdem das Chassis eingeschaltet wurde kann man hier unter „Netzwerkumgebung“ eines neues System hinzufügen. Hierzu wird die zuvor notierte IP Adresse benötigt. Das System sollte nun angezeigt werden.</p> <p>Schritt 4 : Formatierung</p> <p>Wurde das System gefunden lässt sich mit einem Rechtsklick auf das gefunde System die Option „Festplatte formatieren“ wählen. Hier gibt es noch die Option die Netzwerkeinstellungen zu behalten. Als Dateisystem wird das „Reliance“ Format gewählt, welches von NI für PXI Systeme empfohlen wird.</p> <p>Schritt 5 : Installation</p> <p>Nach der Formatierung kann auf dem Controller neue Software installier werden. Hierzu kann über einen Rechtsklick auf das System die Option „Software Installieren“ gewählt werden. Als Software muss mindestens <i>LabVIEW RealTime Modul</i>, <i>Platform Service</i> und <i>DAQ Software</i> ausgewählt werden, damit das Chassis erkannt und die PXI Karten gefunden werden können.</p> <p>Schritt 6 : Realtime Boot aktivieren</p> <p>Nach der Installation muss der Realtime Boot Modus aktiviert werden. Hierzu muss der Controller erneut ausgebaut werden und Switch 2 auf OFF und Schwitch 1 auf ON geschaltet werden (Abbildung 1). Die Neuinstallation ist nun abgeschlossen.</p>		

Anhang 10: Anleitung zur Neuinstallation von LabVIEW RealTime Targets 2 von 2