



MASTERTHESIS

**Entwicklung eines Konzeptes zur
Digitalisierung des Labors der mechanischen
Verfahrenstechnik der HAW, inklusive einer
ersten Teilumsetzung**

B. Sc. Daniel Ludwig Matr.: 2241272

Erstprüfer: Prof. Dr. Enno Stöver

Zweitprüfer: Prof. Dr. Martin Geweke

28. Februar 2021

Inhaltsverzeichnis

Abkürzungsverzeichnis	I
Symbolverzeichnis	III
Indexverzeichnis	V
Glossar englischer Begriffe	VII
Tools	VIII
1 Einleitung	1
1.1 Ist-Analyse des Verfahrenstechnik Labors	2
1.2 Digitale Transformation	6
1.3 Motivation und Zielsetzung	9
2 Stand der Technik	11
2.1 Messtechnik	11
2.1.1 Digitale Durchflussmesstechnik für Gase	11
Thermischer Massendurchflusssensor	15
2.1.2 Druckmessung	16
Halbleiter Drucksensoren	19
2.2 Verfahrenstechnischer Prozess: Durchströmung von Schüttungen	20
2.2.1 Darcygleichung	22
2.2.2 Ableitung der Filtergleichung aus der Darcygleichung	24
2.2.3 Ableitung der Filtergleichung für die Betriebsweise mit einer konstanten Druckdifferenz	26
2.3 Verfahrenstechnischer Prozess: Gas-Feststoff Wirbelschicht	29
2.3.1 Festbett-/Wirbelschichtparameter	29
2.3.2 Klassifizierung von Schüttgütern nach Geldart	30
Gruppe C	30
Gruppe A	30
Gruppe B	30
Gruppe D	30
2.3.3 Wirbelschichtzustände eines für die Fluidisierung optimalen Schüttguts	31
Anmerkung	32
2.3.4 Druckverlust beim Betrieb einer Wirbelschicht	33
Anmerkung	34

2.4	Signalverarbeitung	35
2.4.1	Vom Bit bis zur „vom Menschen lesbaren“ Alphanumeric	35
2.4.2	Analoge und digitale Signale	36
2.4.3	Schnittstellen	37
2.4.4	Verbinder	38
2.4.5	Bitserielle Datenübertragung	39
2.4.6	RS-232	40
2.4.7	RS-485	41
2.4.8	USB	42
2.4.9	Bluetooth	43
2.4.10	Verarbeitung digitaler Signale in Labview	44
2.5	Datenverwaltung	45
2.5.1	Informationsmanagementsysteme	45
	SQL-Datenbanken	48
	Big Data	49
2.5.2	PERA-ICS Modell	51
2.5.3	ISA-S95	53
2.5.4	B2MML	54
3	Konzeptentwicklung für die Labordigitalisierung	57
3.1	Konzeptentwurf 3.0 (ohne Datenbank)	58
3.2	Konzeptentwurf 4.0 (mit Datenbank und Softwarevernetzungsmöglichkeiten)	60
3.3	Fazit	64
3.4	Strukturkonzept einer relationalen Datenbank	67
3.4.1	Fazit	73
4	Signal- und Datenverarbeitung von Messeinrichtungen	75
4.1	Serielle Schnittstellenkonfiguration und Signalverarbeitung von Messeinrichtungen	76
4.2	Serielle Schnittstelle im Betriebssystem finden und Debuggen	78
	Tool zur Identifikation und Debuggen serieller Schnittstellen	78
4.3	Kontinuierliche Daten via RS-232 und LabVIEW verarbeiten	78
4.3.1	Anpassung der LabVIEW-Einstellungen	79
4.3.2	Serielle Sub VI programmierung	81
	Erster Schritt: Serielle Objekte	81
	Zweiter Schritt: Port Initialisierungs- und Abbruchskonfiguration	81
	Dritter Schritt: N = 1 For Loop, XON flow control	82
	Vierter Schritt: Optimierung	84
4.3.3	Kontinuierliche Messdatenerfassung RS-232 im Haupt VI	85
	Erster Schritt: Port Konfiguration und String verketteten/anzeigen lassen	86
	Zweiter Schritt: „String Slicing“	86

Dritter Schritt: „Dateninterpretation“	89
Vierter Schritt: Zeilenkonstruktion	92
Fünfter Schritt: Zeitstempelprogrammierung	93
Sechster Schritt: Versuchsbeobachtungen	95
Siebter Schritt: Dataloggerprogrammierung	95
Siebter Schritt ergänzung gemäß neuster Applikationen: Graph mit zwei Plots	96
Achter Schritt: Diskrete Daten am Beispiel des Datalog Headers	98
Neunter Schritt: Auslastungsoptimierung	98
4.3.4 NI DAQmx Programmierung am Beispiel der Wirbelschichtanlage	99
Haupt Programm Dataloggerinitialisierung am Beispiel der Filterkuchenversuchsanlage	104
Haupt Programm Dataloggerheader am Beispiel der Filterkuchenversuchsanlage	105
Haupt Programm Endsequenz	106
4.4 Verwendung beider Programme	107
4.4.1 Fazit	112
5 Praktische Teilumsetzung des Konzepts 3.0 an ausgewählten Versuchsständen	113
5.1 Filterkuchenversuchsanlage	113
5.1.1 Sensorauswahl	114
5.1.2 Elektrotechnik der Filterkuchenversuchsanlage	117
5.1.3 Filterkuchenversuchsanlage nach den Modifikationen	118
5.1.4 Workflow der Filterkuchenversuchsanlagen	118
5.2 Wirbelschichtversuchsanlage	122
5.2.1 Sensorauswahl	124
5.2.2 Elektrotechnik der Wirbelschichtversuchsanlage	125
5.2.3 Wirbelschichtversuchsanlage nach den Modifikationen	126
5.2.4 Workflow der Wirbelschichtanlage	127
5.2.5 Fazit	128
6 Fazit und Vision	130
6.1 Bilanz in Bezug auf die Ist-Analyse zu Beginn des Projekts	130
6.2 Fazit und Vision des Kapitels Konzeptentwicklung	132
6.3 Fazit und Vision des Kapitels Signalverarbeitung	135
6.4 Fazit und Vision der Teilumsetzung an den ausgewählten Versuchsanlagen	137
Abbildungsverzeichnis	viii
Anhang	i
Einleitung	i
Ergänzung des Abschnitts Dehnungsresistive Sensoren	iii

Inhaltsverzeichnis

Anhang des Abschnitts Datenverwaltung im Stand der Technik	v
Anhang der Konzeptentwicklung	viii
Machine Learning	xiv
Anhang der Teilumsetzung	xvi
Anhang der Schnittstellenprogrammierung	xxi

Veröffentlichungserklärung

Die vorgelegte Masterthesis mit dem Titel „Entwicklung eines Konzeptes zur Digitalisierung des Labors der mechanischen Verfahrenstechnik der HAW, inklusive einer ersten Teilumsetzung“ kann der Öffentlichkeit zugänglich gemacht werden.

Die Vervielfältigung und die Veröffentlichung der Masterthesis ist grundsätzlich erlaubt.

Datum, Ort

Unterschrift

Datum, Ort

Unterschrift

Datum, Ort

Unterschrift

Eidesstattliche Erklärung

Ich versichere, dass ich vorliegende Arbeit ohne fremde Hilfe selbstständig verfasst und nur die angegebenen Hilfsmittel benutzt habe. Wörtlich oder dem Sinn nach, aus anderen Werken entnommene Stellen sind unter Angabe der Quelle kenntlich gemacht.

Datum, Ort

Unterschrift

Danksagung

Ich möchte mich herzlich bei allen bedanken, die mich bei der Erreichung meines akademischen Ziels unterstützt haben. Allen voran selbstverständlich meine geliebte Mutter, die mir in Momenten, in denen ich aufgrund nicht offensichtlicher technischer Probleme fast verzweifelt bin oder in den Zeiten in denen ich nicht wusste wie ich neues, jedoch notwendiges neues Wissen erschließen kann, mich motiviert oder aber gezwungen hat an Tagen an denen ich 12-14 Stunden am Projekt gearbeitet habe Feierabend zu machen, naja.. bis auf die letzte Woche, hahaha. Auch mein bester Freund Muhammad, aka. Mo, war ein unheimlich wichtiger Beistand, ich Danke dir besonders!

Ich bedanke mich aus ganzem Herzen bei meinen Teamkollegen M. Hannappel und S. Wittkowski und den Professoren C. Frank, M. Geweke, M. Hölling, A. Sievers, E. Stöver, die mich und meine unkonventionelle, energetische Art und Weise sowie Humor... „ertragen“... haben. Die Welt kann ein grauer Ort sein und unsere Zeit auf Erden ist definitiv begrenzt, wenn nicht ich das weiß, wer dann. Aus diesem Grunde muss man jede Minute nutzen und zugleich genießen. Die beste Art dies zu tun ist meines Erachtens NUR durch Humor und durch die Wertschätzung jeden Moments und den Menschen in deinem Orbit, die auch nur das Beste für einen wollen.

Auf technischer Ebene gilt mein außerordentlicher Dank Prof. C. Frank. Ich weiß nicht wo ich ohne Sie in Bezug auf die elektrotechnische sowie programmiertechnische Hilfe bei meinen Herausforderungen wäre. Sie haben es mir ermöglicht, nach äußerst unangenehmen Startschwierigkeiten, ein Fuß in die mir noch unbekannte Welt der seriellen Schnittstellen Programmierung mit Python sowie LabVIEW zu setzen.

Sie alle haben mir geholfen, als mir mein Privatleben meine Arbeit erschwert hat.

VIELEN DANK!!!

Abkürzungsverzeichnis

AI	artificial intelligence
ANSI	American National Standards Institute
ASCII	American Standard Code for Information Interchange
Bd	Baudrate
BD	Blockdiagramm
BD	Blockdiagramm
BuB	Bedinung und Beobachtung
Com	Serieller Anschluss unter Windows
CTS	Clear to Send
DAQ	data acquisition, Datenakquisition
DBMS	Datenbank Management Systemen
DEE	Datenendeinrichtung
DÜE	Datenübertragungseinrichtung
EIA	Electronic Industrial Alliance
ELN	Electronic Laboratory Notebook
ERP	Enterprise Resource Planning
FP	Front Panel
FP	Front Panel
GND	Signal Ground
HMI	Human-Machine-Interface
ICS	Industrial Control System (deutsch Industrielle Kontrollsysteme)
IMS	Informationsmanagementsysteme
IoT	Internet of Things
ISA	International Society of Automation

ISO	International Organization for Standardization
KI	künstliche Intelligenz
LIMS	Labor Management Informations Systeme
MES	Manufacturing Execution System
MESA	Manufacturing Enterprise Solutions Association
MOM	Manufacturing Operations Management
MVT	mechanische Verfahrenstechnik
PERA	Purdue Enterprise Reference Architecture
rM	rechte Maustaste
RS-232	Recommended Standard oder Radio Section 232
RTS	Request to Send
RxD	Receive Data
SETSVI	Simple Elapsed Time Sub VI
SKDM	Schwebekörperdurchflussmesser
SQL	Structured Query Language
SSVI	Serial Sub VI
TxD	Transmit Data
USB	Universal Serial Bus
VI	Virtuelles Instrument
VT	Verfahrenstechnik
WBF	World Batch Forum
WSN	Wireless Sensor Network
XML	Extensible Markup Language
XSD	[XML] Schema Definition

Symbolverzeichnis

Symbol	Bezeichnung	Einheit
A	Anströmfläche	m^2
\vec{a}	Beschleunigung	m/s^2
Bd	Baudrate	Byte/s
c_p	Wärmekapazität	$\text{kJ}/\text{kg} \cdot \text{K}$
c_w	Widerstandsbeiwert	-
d	Durchmesser	mm^2
ε	Porosität	-
η	dynamische Viskosität	Pas
Eu	Eulergleichung	
F	Kraft	N
\vec{F}	Kraftvector	$\text{kg} \cdot \text{m}/\text{s}^2$
F	Kraft	N
g	Gravitationskonstante	m/s^2
I	Stromstärke	A
k	Dehnungsempfindlichkeit	-
m	Masse	kg
ν	Querkontraktionszahl (Poisson-Konstante)	-
P, p	Druck	N/m^2
ψ	Dehnung	-
$\pi_{l,t}$	longitudinale/transversale Piezokonstante	Pa^{-1}
R	elektrischer Widerstand	Ω
Re	Reynoldszahl	-
ρ	Dichte	kg/m^3

ρ_{el}	spezifischer elektrischer Widerstand	$\Omega \cdot \text{m}$
S	Surface	mm^2
s	Filterkuchendicke	
$\sigma_{l,t}$	longi./transv. mechanische Spannungskomponente	Pa
T	Temperatur	K
U	elektrische Spannung	V
\dot{V}	Volumenstrom	m^2/s
v	Geschwindigkeit, Leerohrgeschwindigkeit	m/s
w	charakteristische Geschwindigkeit	m/s
x	Feinheitsmerkmal	mm

Indexverzeichnis

Index	Bezeichnung
32	Sauter(-durchmesser)
A	Auftrieb
a	Anström(-fläche)
Br	Wheatstone Brücke
c	Coriolis
el	elektrisch
f	Fluid
fb	Festbett
FM	Filtermittel
fs	Feststoff
G	Gravitation
h	hydraulisch
irr	irreversibel
K	Filterkuchen
konst	konstant
L	Länge
l	longitudinal
m	Massen(-konzentration)
mf	minimal Fluidisation
Q	Querkontraktion
s	Sink(-geschwindigkeit)
t	transversal
V	Volumen

vs	Volumen(-konzentration)
W, w	Widerstand
ws	Wirbelschicht
x	Feinheitsmerkmal

Glossar englischer Begriffe

Englisch	Deutsch
AI	künstliche Intelligenz
classification	Klassifizieren
DAQ	Datenakquisition
digital Twin	digitaler Zwilling
dryfreezer	Gefriertrockner
dustfiltration	Staubabscheidung
fixed-bed	Festbett
fluidization	Fludisierung
high pressure homogenizer	Hochdruckhomogenisator
laser diffraction device	Laserbeugerpunktelanalysator
machien learning	maschinelles Lernen
particle bed, fixed-bed filtration	Festbett- / Kuchenfiltration
unit operation	verfahrenstechnische Grundoperation
principal	Auftraggeber ⇒ Dozenten
rollcrusher	Walzenbrecher
sieveing	Sieben
standard operating procedures (SOP)	Handlungsanweisung, Versuchsanweisung
supervisor	Schichtleiter ⇒ wissenschaftlicher Mitarbeiter
tensil tester	Zugversuchapparatur

Tools

Alle Tools die im Verlauf des Projekts eingesetzt wurden sind in der folgenden Liste aufgelistet:

- Hardware
 - Smartphone P30+ (privat) für die Fotodokumentation
 - MacBook Pro 2011 (privat)
 - Windows 10 Desktop PC (privat)
 - Windows 10 Desktop PC
- Software
 - Powerpoint für Präsentation, Grafikerstellung (z.B. elektrisches Schaltschema, Datenbank) und Bildmanipulationen.
 - PDF24
 - * zum beschneiden der PDF Ränder
 - * Zum "Drucken" von Dateien in ein PDF, anstelle eines Papiers
 - Tabellenkalkulationsprogramm: Excel
 - Bildmanipulation: GIMP
 - Textsatz: \LaTeX zur Anfertigung des Dokuments
 - Fonts (XeLaTeX notwendig, um die Fonts nutzen zu können)
 - 1 Arno Pro
 - 2 Hypatia Sans
 - 3 Menlo (Python font)
 - Bibliotheksmanagement: JabRef
 - * Für Formeln im Hauptdokument wurden die folgenden Normen beachtet:
 - DIN 1338 Formelschreibweise und Formelsatz
 - DIN 1304 Formelzeichen
 - DIN 1301 Einheitennamen, Einheitenzeichen

- DIN 1313 Größen
- LabVIEW 2019 auf den HMI's an der HAW Life Sciences
- LabVIEW 2020 auf dem MacBook pro und dem privatem PC
- LabVIEW 2021 auf den HMI's an der HAW Life Sciences
- Anaconda als Python Distribution
 - * Spyder als Python Editor

Zusammenfassung

Im Verlauf des Projekts wurden Konzepte zur Digitalisierung des verfahrenstechnischen Labors erstellt. Es sind zwei Konzepte erstellt worden, die aufeinander aufbauen. Das erste Konzept mit dem Titel „Konzept 3.0“ beschreibt die Generierung von Messwertdaten mittels Sensoren und die softwaretechnische Verarbeitung und Archivierung. Das zweite Konzept, mit dem Titel „Konzept 4.0“ beschreibt die zukünftige, potentielle Implementierung weitere Applikationen im Rahmen von Big Data. Da das zweite Konzept Datenbankmanagementsysteme im Zentrum hat, ist ein Datenbankkonzept, angelehnt an den ISA-S95 and B2MML, erstellt und dieses anschließend auf ein relationales Datenbankkonzept transferiert worden. Abschließend wurden zwei Versuchsstände, die Wirbelschichtversuchsanlage und die Filterkuchenversuchsanlage, mit digitaler Sensorik ausgestattet und Programme in LabVIEW geschrieben, die die Datenströme beider Versuchsstände verarbeiten können und Tabstopgetrennte *.txt-Dateien generieren.

Abstract

In the course of the project, concepts for digitizing the process engineering laboratory were created. Two concepts have been created. The second one build on top of the first one. The first concept, entitled Concept 3.0, describes the generation of measured value data by means of sensors and the software processing and storage. The second concept, titled Concept 4.0, describes the future, potential implementation of further applications in the context of Big Data. Since the second concept has database management systems at its core, a database concept, based on ISA-S95 and B2MML, has been created and then transferred to a relational database concept. Finally two test rigs, the fluidization bed test rig and the filtration (fixed-bed) test rig, have been equipped with digital sensors. Programs have been written in LabVIEW that can process the data streams of both test rigs and generate tab-stop separated *.txt files.

1 Einleitung

Die Digitalisierung in der Wirtschaft hat im Jahre 2020 Dimensionen erreicht, dass Forscher mittels „Supercomputern“ und *künstlicher Intelligenz* (KI) beispielsweise chemische Verbindungen für die Reifenherstellung oder Enzymformulierung errechnen, mit der die Ethanolproduktion optimiert werden kann [2]. Mittels *Digitaler Zwilling*, im englischen *digital Twin*, können Produkte, Produktionsabläufe oder ein gesamter Prozess, durch die Kopplung von Produktions- und Produkt Twin, abgebildet und somit optimiert werden. Mittels Digital Twins lassen sich sogenannte Soft-Sensoren integrieren (angewandte numerische Simulation). In Produktionsprozessabschnitten, in denen die Bedingungen es nicht erlauben einen realen Langzeitsensor zu implementieren, können, mittels Digital Twins, Sensoren in Simulationssoftware generiert werden, die mit einem Prozess synchronisiert sind, um Totzeiten zu verringern oder zu eliminieren, bis schlechte Produktionsbedingungen detektiert werden können. Dadurch kann die Effizienz eines Prozesses signifikant gesteigert werden [36].

Abhängig vom Kontinent, Staat, Region und Branche wird der digitale Wandel anders wahrgenommen. Eine Einteilung nach Kondratjew ist nicht Eindeutig, dennoch erfüllt der digitale Wandel gemäß [5] alle Kriterien, um den Platz des 6. Kondratjew-Zyklus einzunehmen. Im Privatsektor nimmt die Zahl an „smartem“ Geräten stetig zu. Das ein Smartphone mit einem Smart TV autonom kommuniziert ist bereits geübter Alltag. Auch in der Industrie kommunizieren Sensoren, Geräte, Maschinen, Aggregate bereits autonom untereinander, aber auch mit Enterprise Resource Planning (ERP)-, Manufacturing Execution Systemen (MES) oder *Manufacturing Operations Management* (MOM) und ähnlichem. Smarte Sensoren sind im Stande, bei autonomer Erkennung einer Erhöhung der Standardabweichung sich selbst zu kalibrieren [22].

Es lässt sich erahnen, dass sich die Datenmengen pro Zeiteinheit, Prozess o.ä. die nächsten Jahre stetig erhöhen wird. Nun sind Konzepte gefragt, wie sich die „relevanten“ Daten intelligent archivieren lassen, um sie in Echtzeit oder zu einem späteren Zeitpunkt verhältnismäßig einfach verarbeiten und auswerten zu können, um beispielsweise Produktionsprozesse zu optimieren.

Damit zukünftige Absolventen der HAW diesen Anforderungen schneller gerecht werden können, ist das erste Ziel dieser Arbeit, den Status quo des verfahrenstechnischen Labors zu erfassen, um die nächsten Schritte einzuleiten, damit der Stand der Technik in der Industrie noch besser approximiert werden kann.

1.1 Ist-Analyse des Verfahrenstechnik Labors

In der Verfahrenstechnik gibt es für die unterschiedlichsten verfahrenstechnischen Operationen (Unit Operations), wie z.B. das Trennen von Stoffgemischen oder das Zerkleinern von Partikeln, verschiedene Versuchsstände. Um sich einen Überblick über das verfahrenstechnische (VT) Labor zu machen, kann das Labor auf mindestens drei unterschiedliche Weisen betrachtet werden.

1. Das VT-Labor kann in die **Unit Operations**, mit den Versuchsständen und den jeweiligen Komponenten wie Apparate, Messtechnik und Datenverarbeitung, unterteilt werden. Diese Sichtweise eignet sich besonders, um sich einen Überblick zu verschaffen.
2. Betrachtet man das VT-Labor unter dem Aspekt der **anfallenden Daten** (siehe Tabelle 16 im Anhang), die bei den Versuchsständen generiert werden, dann kann das VT-Labor in diskrete und kontinuierlich anfallende Daten unterteilt werden. Folglich sind im Rahmen einer Datenverarbeitung mindestens zwei Lösungsansätze notwendig.
3. Eine differenzierte Betrachtung der anfallenden Daten führt zur nächsten Klassifizierung, die Einteilung nach **Messaufgabe**. Verfahrenstechnische Messaufgaben könnten die folgenden sein:
 - Volumenstrom-/Massenstrommessung (kontinuierlich)
 - Temperaturmessung (kontinuierlich/diskret)
 - Wiegen (kontinuierlich/diskret)
 - Druckmessung (kontinuierlich/diskret)
 - Konzentrationsmessung (kontinuierlich/diskret)
 - Luftfeuchtebestimmung (diskret/kontinuierlich)
 - Feuchtebestimmung von Feststoffen (diskret)
 - pH-Wert Messung (kontinuierlich/diskret)
 - Gaskonzentrationsmessung (kontinuierlich)

Die Ausgangssituation des mechanischen Teils, des verfahrenstechnischen Labors (MVT-Labor), wurde in einer Excel-Datei dokumentiert (siehe Anhang: Ist-Analyse-VT-Labor.xlsx). In der Tabelle 1 sind die Versuchsstände der MVT; drei Versuchsstände der Unit Operation Trennen, der thermischen Verfahrenstechnik (**hellrot eingefärbt**), die häufig Ihre Anwendung in der MVT finden sowie einige Analyseeinrichtungen (**hellblau eingefärbt**); aufgelistet. Das Ziel der Ist-Analyse war es, den „Grad der Digitalisierung“ zu Beginn des Projekts zu eruieren. Mit „Grad der Digitalisierung“ ist gemeint, wie viele

Tabelle 1: Mechanische Unit Operations, deren Versuchsstände und messtechnische Einrichtungen

Unit Operations		Versuchsstände	digitale Schnittstelle
Trennen	Klassieren	Sichten	keine
	Filtration	Kuchenfiltration	keine
		Blaine	keine
		Staubabscheidung	USB
	thermisch	Gefriertrocknen	USB
		Trocknungsofen	USB, LAN
		Vakuumtrockner	keine
Fluidisieren		Wirbelschicht	keine
Partikelzerkleinerung	Partikelzerkleinerung	Backenbrecher	keine
		Planetenkugelmühle	RS-232, RS-485
		Prallmühle	keine
		Walzenmühle	keine
		Schneidmühle	keine
		Hochdruckhomogenisator	keine
		Intensivmixer	USB
Mischen	Mischen	Conchiermaschine	keine
		Rührversuch	RS-232
		Zugversuch	RS-232
	MVT	Siebturmanalyse	keine
		Luftstrahlsiebung	RS-232, USB
		Scherzelle	RS-232
		Stampfdichte	keine
		Laserbeugungs-Partikelgrößenanalyse	USB
Analyseeinrichtungen	Analyseeinrichtungen	Waagen	USB
		IR-Waagen	RS-232
		Refractometer	RS-232
		Fluiddichthemesser	RS-232
		Gaspyknometer	RS-232
		Mikroskope	USB
		Viskosimeter	USB

Tabelle 2: „Grad der Digitalisierung“ der MVT Geräte und Anlagen

MVT Versuchsstände	Sensorik zweckmäßig	Sensorik vorhanden	„effektive“ Schnittstelle	priorisierte Einrichtungen
Sichten				
Kuchenfiltration	○			✗
Blaine				
Staubabscheidung	○	✓	✓	✓
Wirbelschicht	○			✗
Backenbrecher				
Planetenkugelmühle	○	✓	✗	
Prallmühle				
Walzenmühle				
Schneidmühle				
Hochdruckhomogenisator				
Intensivmischer		✓	✗	
Conchiermaschine				
Rührversuch	○	✓	✓	✓
Summe aller ✓ oder ○	5	4	2	2
Summe aller ✗ oder ✗			2	2
prozentualer Anteil, relativ zu allen Einrichtungen Spalteneinträgen	36 %	29 %	50 %	50 %
MVT Analyseeinrichtungen	Sensorik zweckmäßig	Sensorik vorhanden	„effektive“ Schnittstelle	priorisierte Einrichtungen
Zugversuch	○	✓	✓	
Siebturmanalyse zzgl. digital Waage	○	✓	✗	✗
Luftstrahlsiebung	○	✓	✓	✓
Scherzelle	○	✓	✗	✗
Stampfdichte				
Laserbeugungs- Partikelgrößenanalyse	○	✓	✓	✓
Summe aller ✓ oder ○	5	4	2	2
Summe aller ✗ oder ✗			2	2
prozentualer Anteil, relativ zu allen Einrichtungen Spalteneinträgen	83 %	83 %	60 %	50 %

Versuchsstände digitale Messtechnik verwenden, mit denen Daten, zwecks Weiterverarbeitung, generiert werden können. **Messtechnikkomponenten mit einer digitalen Schnittstelle werden im Verlauf dieser Arbeit nur Sensoren genannt.** Die Zellen in der Tabelle 1, bei denen ein Upgrade auf Sensoren sinnhaft wäre, werden grau schraffiert eingefärbt.

Der Tabelle 1 kann entnommen werden, dass 11 der 15 MVT Versuchsstände keine Sensoren besitzen. Einige Analyseeinrichtungen werden i. d. R. nur im Rahmen der MVT genutzt und sind in der Tabelle 1 mit MVT markiert. Alle weiteren messtechnischen Einrichtungen werden, neben der MVT, auch in der chemischen und thermischen Verfahrenstechnik genutzt. Ein Upgrade mit Sensoren ist bei dem Wirbelschicht- und dem Filterkuchenversuchsstand sinnhaft, daher sind diese zwei Versuchsstände in der Tabelle grau schraffiert.

In der Tabelle 3 ist die Schnittstellenverteilung der Versuchsstände im MVT Labor dargestellt. Der Tabelle ist zu entnehmen, dass 20 (15 DB-9 und 25 DB-25 Stecker, siehe Abbildung 15 in Abschnitt 2.4.4) Geräte die RS-232 Schnittstelle besitzen, sieben Geräte haben eine USB Schnittstelle, drei Geräte haben eine LAN Schnittstelle und ein Gerät hat eine RS-485 Schnittstelle. In der Tabelle 2 sind die Versuchsstände und Analyseeinrichtungen der MVT aufgelistet. Neben der Spalte mit den Versuchsständen- und Analyseeinrichtungen gibt es vier weitere Spalten:

Tabelle 3: Schnittstellenverteilung im MVT-Labor

Schnittstelle	Anzahl
RS-232 (DB-9)	15
RS-232 (DB-25)	5
USB	7
LAN	3
RS-485 (DB-9)	1

- Sensorik zweckmäßig
 - Versuchsstände und Analyseeinrichtungen, bei denen Sensoren Zweckmäßig sind, sind in dieser Spalte mit einem Kreis (○) markiert. Der prozentuale Anteil, der jeweils letzten Zeile, ist relativ zu allen Versuchsständen bzw. Analyseeinrichtungen.
- Sensorik vorhanden
 - In dieser Spalte haben alle Versuchsstände und Analyseeinrichtungen einen Haken, die bereits über Sensorik verfügen. Der prozentuale Anteil, der jeweils letzten Zeile, ist relativ zu allen Versuchsständen bzw. Analyseeinrichtungen.
- „effektive“ Schnittstelle

- In dieser Spalte haben die Versuchsstände und Analyseeinrichtungen, die eine digitale Schnittstelle besitzen und auch bei den Versuchen verwendet werden, einen Haken. Der prozentuale **Anteil**, der jeweils letzten Zeile, ist relativ zu allen Spalteneinträgen.
- priorisierte Einrichtungen
 - Versuchsstände und Analyseeinrichtungen die in dieser Spalte aufgeführt sind, haben für die Präsenzveranstaltung eine hohe Bedeutung **und** sollten, wenn nicht bereits vorhanden, unverzüglich eine Signalverarbeitungsapplikation erhalten oder mit Sensoren ausgestattet werden. Die roten **Kreuze** betonen die *Wichtigkeit* der Versuchsstände, die noch nicht auf dem Stand der *anerkannten Regeln der Technik* sind. Der prozentuale **Anteil**, der jeweils letzten Zeile, ist relativ zu allen Spalteneinträgen.

1.2 Digitale Transformation

An dieser Stelle bietet es sich an die Bedeutung der Schlagwortkombination „digitale Transformation“ kurz zu erläutern. Soll ein Unternehmen oder dergleichen digital transformiert werden, dann ist nicht von einem Projekt die Rede. Die digitale Transformation ist ein Prozess, welche lediglich mit einem Projekt initiiert wird. Solch ein Projekt kann, wie im Rahmen dieses Projekts mit dem erstellen eines Digitalisierungskonzepts, dem Aufrüsten von Versuchsanlagen im Labormaßstab mit Sensoren beginnen, doch darf mit der Beendigung des Projekts nicht als abgeschlossen betrachtet werden! Im akademischen Rahmen sind die Haupttreiber (*engl. enabler*) digitale Technologien, wie die Ermöglichung von Infrastrukturen/Vernetzungen oder Anwendungen. Doch auch Akteure wie der Staat, Unternehmen, die Wissenschaft (Forschung und Lehre) und Gemeinschaften können eine digitale Transformation triggern. In Anlehnung an [45], tragen die Stufen der digitalen Transformation im akademischen Rahmen folgende Label:

- Digitale Transformation
 - Weiterentwicklung der Lehre, des Mindsets der Dozenten sowie Studierenden
 - des **Kundenerlebnisses** -> die Kompetenzen, welche **Absolventen** in die **Unternehmen** einbringen werden,

- der Geschäftsmodelle, in Analogie der akademischen Modelle, (die Vision des „Lernraums digitale Umformtechnik“ von Prof. Dr. E. Stöver, siehe Abschnitt 3.2 Absatz 7, nach dem Paragraph *künstliche Intelligenz*)
- Digitale Nutzung
 - Anwendung der digitalen Tools von den Dozenten sowie Studierenden und somit zugleich Generierung von praktischen Erfahrungen der Studierenden, im Rahmen der digitaler Dokumentation und Datenverarbeitung,
- Digitale Kompetenz
 - Einfacher Einstieg im Umgang mit neuen Technologien

Um die digitale Transformation erfolgreich einzuleiten, ist zielgerichtetes Handeln erforderlich. Wie *sollte* die digitale Infrastruktur aussehen? Wie sieht das operative Geschäft im Unternehmen heute, in naher Zukunft sowie visionär betrachtet aus und wie kann das als Hochschule approximiert werden? Was für Kompetenzen *sollte* der Hochschulabsolvent von morgen in ein Unternehmen einbringen können? Wer sind die Stakeholder in der Hochschule und wie können diese abgeholt werden? Das Stichwort an der Stelle ist **Change Management**. Wie ist die Organisation anzupassen, um den Ansprüchen gerecht zu werden?

Diese Fragen sollten im akademischen Rahmen beantwortet werden, um einen erfolgreichen Start gewährleisten zu können. In der folgenden Auflistung, nach [45], sind einige Hilfestellungen, um die oben genannten Fragen präziser stellen zu können (grau markierte Punkte spielen im akademischen Rahmen eine eher untergeordnete Rolle):

- Digitale Infrastruktur
 - Datenintegration,
 - erweiterte Analysen,
 - Datenschutz,
 - Datensicherheit, ...
- Kunde
 - Unternehmen,
 - Kundenverständnis,
 - Customer Experience Management, ...
- Transformationsmanagement
 - Digitale Strategie,
 - Change Management, ...
- Operatives Geschäft
 - Integrierte IT,
 - Digitale Wertschöpfungskette,
 - Digitale Produktion, ...
- Organisation
 - Agilität,
 - Arbeitsplatz der Zukunft,
 - Digitales Denken & Handeln, ...
- Wertversprechen
 - Smarte Produkte,
 - Individualisierung,
 - Digitale Ökosysteme, ...

Neben dem Change Management, sind oftmals ein fehlendes Bewusstsein, eine nicht vorhandene Dringlichkeit und die fehlende Vision der Entscheider, in Bezug auf die Notwendigkeit einer digitalen Transformation, wesentliche Herausforderungen. Des Weiteren haben wir derzeit das Zeitalter der digitalen Vernetzung. Im Privatsektor sind es Phänomene wie *Social Media* oder „smarte Geräte“, die autonom kommunizieren. In der Industrie sind es Sensoren, Prozessketten, Softwarelösungen (ERP, MES, uvm...), aber auch entlang der **gesamten** Wertschöpfungskette, d.h. vom Forstwirt bis zum Toilettenpapierverkauf im Einzelhandel findet oder besser sollte eine Vernetzung bzw. Kommunikation stattfinden (Stichwort *Bullwhip Effekt*). Aus der Sicht der Hochschule könnte es **mehr** Kommunikation zwischen den Fakultäten sowie **mehr** Kommunikation und interdisziplinäre Projekte zwischen den Departements einer Hochschule, aber auch zwischen verschiedenen Hochschulen und Universitäten, als auch mit Unternehmen (Stichwort ist z.B. die Vision des stellvertretendem Departementleiters Prof. Dr. E. Stöver, dem digitalen Lernraum am Berliner Tor) sein.

1.3 Motivation und Zielsetzung

In der Industrie ist der Einsatz (digitaler) Sensoren, der Anlagensteuerung mittels *Process Logic Controllern* (PLC, unter Siemens SPS), als auch der Einsatz von Datenbanken *anerkannter Stand der Technik*. Die Dokumentation mittels digitaler Medien ist im operativem Geschehen von Unternehmen, aber auch in der Forschung und Entwicklung bereits Stand der Technik. Durch die strukturierte Archivierung in Datenbanken ist eine präzise Abfrage möglich, wodurch die Daten von verschiedenen Softwarelösungen wie Simulationssoftware, *Machine Learning* Applikationen, *Virtual* oder *Augmented Reality*, CAD, uvm. abgegriffen werden können [2, 27].

Mit dem Schlagwort Industrie 4.0 assoziiert man die monetäre Nutzung der genannten Softwarelösungen, aber auch die Vernetzung von Applikationen, Sensoren und Prozessketten sowie die multimedialen Dokumentationsmöglichkeiten.

Im Gegensatz zu den Datenbanken, werden die Daten in sogenannten *Data Lake's* unstrukturiert abgelegt. In Data Lakes können die unterschiedlichsten Datentypen (Textdateien, Audios, Grafiken, Bilder etc.) archiviert werden. Es ist demnach mit der Verwendung einer Cloud (OneDrive, Dropbox etc.) im Privatbereich vergleichbar [27]. Um die folgenden Generation von Studenten optimal auf das operative Geschäft vorbereiten zu können, ist es notwendig, neben den Einsatz von Sensoren, digitale Medien für die Protokollierung einzuführen sowie Applikationen im Rahmen von Big Data inkrementell zu implementieren.

Im Rahmen dieses Projekts soll ein Konzept entworfen werden, dass die digitale Transformation des mechanischen Verfahrenstechnik Labors im Fokus hat. Das Konzept soll neben dem mechanischen Verfahrenstechnik Labor unter anderem auch auf das chemische und thermische Verfahrenstechnik Labor anwendbar sein. Bei der Digitalisierung soll der Grad der Automatisierung der Laborversuche minimal sein, um den Lerneffekt für die Studierenden zu erhalten, den man durch die praktische Durchführung der Versuche erhält. Zu Beginn ist zu eruieren, wie viele Laborversuche digital Messwerte bereitstellen und welche Schnittstellen zur Verfügung stehen. Bei Bedarf soll alternative Messtechnik oder ganz Versuchsanlagen recherchiert und diskutiert werden. Es ist eine Lösung zu eruieren, die digitalen Rohdaten zu archivieren, um sie zu einem beliebigen Zeitpunkt verwenden zu können. Abschließend soll das ganzheitliche Digitalisierungskonzept zum Teil umgesetzt und validiert werden. Folgend sind die Ziele des Projekts noch einmal aufgelistet.

- Erstellung eines Digitalisierungskonzept für das Verfahrenstechnik (VT) Labor
- Erstellung einer Ist-Analyse des mechanischen Verfahrenstechnik Labors
- Die Akquise und Speicherung digitaler Daten (Signalverarbeitung)
- im Bedarfsfall digitale Messtechnik eruiieren
- Validierung des Konzepts, durch Teilumsetzung anhand einer ausgewählten Versuchsanlage

2 Stand der Technik

In diesem Abschnitt wird der Stand der Technik in den Bereichen Messtechnik, ausgewählter verfahrenstechnischer Prozesse, digitale Schnittstellen und Signalverarbeitung sowie Datenverwaltung, in Bezug auf die *digitale Transformation* des VT-Labors, erläutert.

2.1 Messtechnik

In diesem Abschnitt werden ausgewählte digitale Druck- und Volumenstrommesstechniken aufgezeigt, die in Laboren Anwendung finden.

2.1.1 Digitale Durchflussmesstechnik für Gase

Zum messen von Durchflüssen oder Mengen gibt es eine Vielzahl von Verfahren. In der Abbildung 1 ist eine Übersicht der verschiedenen Messverfahren abgebildet. Nicht alle Messverfahren der Abbildung 1 sind dafür geeignet, um Volumen- oder Massenströme von Gasen zu messen. Auf die Messprinzipien aller Sensoren dieses Abschnitts wird im Rahmen dieser Arbeit nicht eingegangen. Für tiefgreifende Informationen wird auf Fachliteratur, wie z.B. [43, 47, 50, 52] verwiesen. Messprinzipien, die für die Messung von Luft geeignet sind und in Sensoren ihre Anwendung finden, sind die folgenden:

- „Kraft auf Körper“ Durchflussmessung (Volumenstrom, non-digital)
- Differenzdruckverfahren (Volumenstrom)
 - Lochblende
 - Venturi-Rohr
- Vortexdurchflußmessung (Volumenstrom, für Gase nur in Kombination mit Dichtemessung Massenstrom)
- thermische Massendurchflussmessung (Massenstrom)
- Coriolis Durchflussmessung (Massenstrom)

Kraft auf Körper Durchflussmessung Da ein Versuchsstand im Rahmen dieser Arbeit, zu Beginn des Projekts, einen (*analog*) Schwebekörperdurchflussmesser (SKDM; siehe Abbildung 2a) besitzt und eine diversitäre Redundanz, mittels *digitalem* Sensor, erfolgen soll, wird das Wirkprinzip dieses rein analogen Messverfahrens erläutert. Gemäß [47, S. 827 ff.] wird dieses Messverfahren „Kraft auf

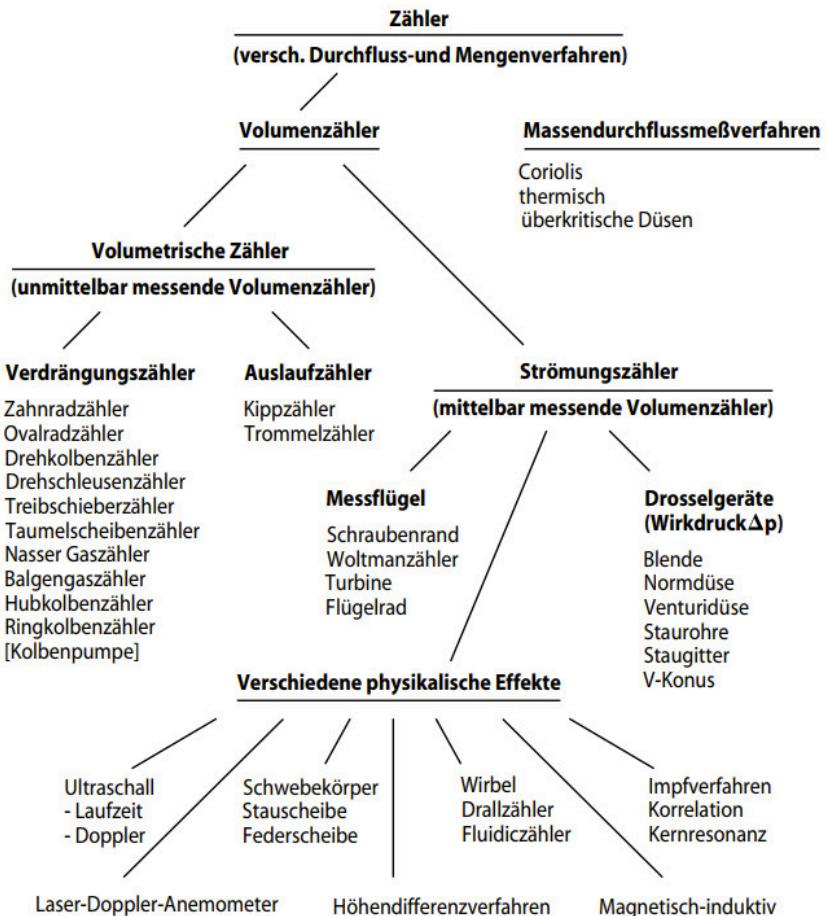


Abbildung 1: Übersicht verschiedener Durchfluss- und Mengenmessverfahren [52, S. 794]

Körper genannt". Ein *Festkörper* wird in einer konischen Rohrleitung angeströmt und wird angehoben oder ausgelenkt, im Falle eines **Federscheibendurchflussmessers**, bei hinreichend großer Anströmgeschwindigkeit. Eine äquivalente Lösung, ist dass installieren eines konischen Schwebekörper in eine Blendenöffnung. Sobald sich ein Kräftegleichgewicht einstellt, verharrt der Festkörper in der Position. Die geometrischen Formen der Festkörper können z.B. ein Konus (SKDM) oder eine Scheibe (Feder-scheibendurchflussmessers) sein [47, S. 827]. Der abzulesende Volumenstrom ist somit unter anderem eine Funktion des durchströmten Querschnitts A . Das physikalische Wirkprinzip ist ein Kräftegleichgewicht. Wenn sich die Gravitationskraft F_G , abzüglich der Auftriebskraft F_A eines Partikels oder Festkörpers, mit der Widerstandskraft F_W im Gleichgewicht befindet (siehe Gleichung 1), dann befindet sich die Strömung in einem stationären Zustand. Für ein frei bewegliches Festkörperpartikel, welches

sich im Gravitationsfeld befindet und vertikal von unten angeströmt wird (z.B. SKDM), gilt Gleichung 1 [43, S. 105 ff.].

$$F_G - F_A = F_W \quad (1)$$

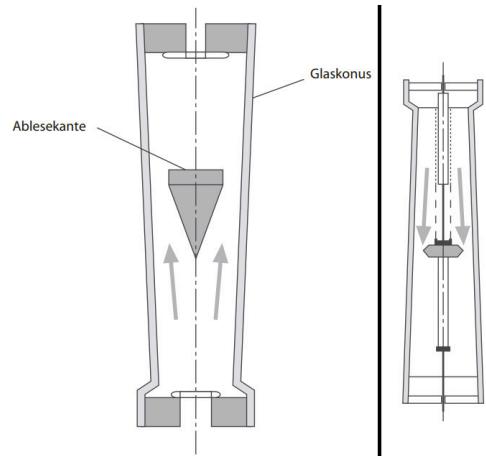
Die Widerstandskraft F_W eines Festkörpers in einer Strömung lautet:

$$\vec{F}_W = c_w(\text{Re}_x) \cdot A \cdot \rho_f \cdot \frac{\vec{v} \cdot |v|}{2} \quad (2)$$

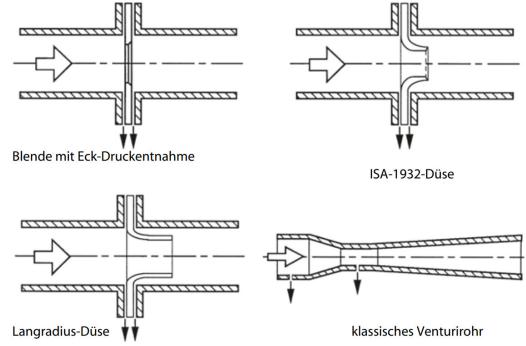
Der c_w Wert gibt den Widerstand eines Partikels in einer Strömung an. Er ist abhängig von der Form sowie Größe eines Partikels (Index x) und ist eine Funktion der Reynoldszahl.

Differenzdruckbasierte Sensoren Differenzdruckbasierte Sensoren sind robust, bieten eine hohe Prozesssicherheit und sind kostengünstig. Sie eignen sich besonders für große Durchflussmengen. Dieses Verfahren wird auch Wirkdruckverfahren genannt. Die Nutzung einer **Lochblende** hat den Vorteil, dass ein großer Messbereich, durch Wechsel bzw. Variation der Lochblende, genauer Durchflussoffnung, realisierbar ist. Für den Betrieb eines differenzdruckbasierten Sensors mittels Lochblende ist zu beachten, dass durch die Drosselung und die dadurch auftretende Dissipation von Energie, hohe Betriebskosten entstehen können. **Venturi-Rohre** haben eine bessere Energiebilanz gegenüber Lochblenden. Des Weiteren weisen Venturi-Rohre gegenüber Lochblenden weniger Verschleiß auf und sind demnach weniger Wartungsintensiv [47, S. 823 ff.].

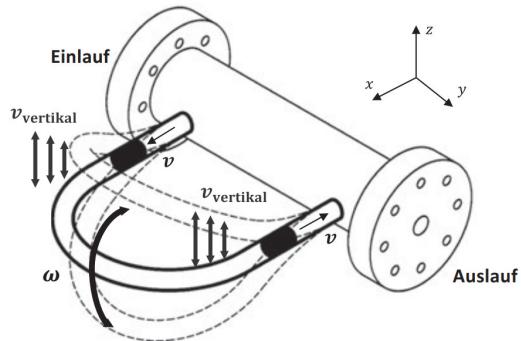
Sensoren auf Basis der Corioliskraft Um den Massenstrom von Fluiden zu ermitteln, kann sich die, aus dem zweiten newtonschen Axiom hergeleitete, Corioliskraft zu nutzen gemacht werden. Die Corioliskraft ist, wie die Zentrifugalkraft, eine Scheinkraft. Die Corioliskraft ($\vec{F}_c = f(\vec{a}_c)$), zur Bestimmung des Massendurchflusses, ist proportional zur Masse des durchströmten Messrohrs ($\vec{F}_c = m \cdot \vec{a}_c$), in der Abbildung 2c und 2d ist es das U-Rohr. Das Messrohr wird elektromagnetisch in Schwingung gesetzt. Wird das Messrohr durchströmt, tritt der Coriolis-Effekt auf und das Messrohr oszilliert um die y-Achse (siehe Abbildung 2d). Für tiefgreifendes Wissen zum Wirkprinzip des Sensors muss auf Fachliteratur verwiesen werden [52, S. 268 ff.].



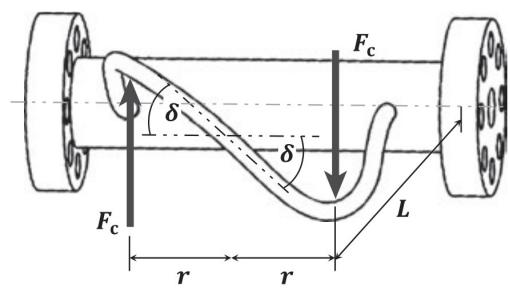
(a) Schwebekörperdurchflussmesser (links)
Federscheibendurchflussmesser (rechts)



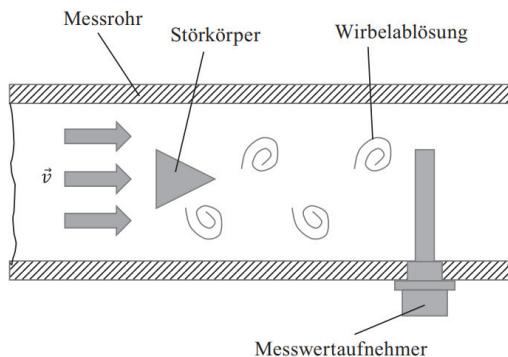
(b) Wirkdruckdrosselgeräte Übersicht nach DIN



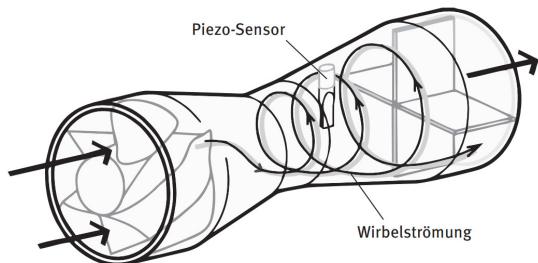
(c) angeregter Coriolissensor ohne Durchsatz



(d) angeregter Coriolissensor mit Durchsatz



(e) Vortexsensor



(f) Drallsensor

Abbildung 2: Übersicht einiger Sensoren, die für die Messung von Gasdurchflüssen geeignet ist: (a) Mechanische Durchflusssensoren (b) Wirkdruckverfahren [47, S. 826], (c, d) Coriolissensor [52, S. 270] (e, f) Vortex- sowie Drallsensor

Vortexdurchflusssensoren Die industrielle Nutzung von Vortexdurchflußsensoren (Wirbelsenso-
ren, siehe Abbildung 2e) begann in den 1960er-Jahren. Das physikalische Grundprinzip ist die Kár-
mánsche Wirbelstraße. Das Wirkprinzip beruht auf der Anregung eines Störkörpers, wie eines Drahtes,
in einer Strömung. Die Schwingfrequenz ist proportional zur Anströmgeschwindigkeit und antiproportio-
nal zur Dicke des Drahtes. Die Ursache der Schwingung ist die Wirbelablösung hinter einem Strö-
mungswiderstand. Die Geometrie des Störkörpers beeinflusst die Wirbelbildung und somit die Schwin-
gung maßgeblich. Bei der Durchflussbestimmung von Gasen ist eine Sensornahre Bestimmung der Dich-
te erforderlich, da durch die Kompressibilität von Gasen, Dichtedifferenzen auftreten können, die sich
maßgeblich auf die gemessene Durchsatzmenge auswirken. Für Gase ist der Vortexsensor, durch die
Kombination mit einer Dichtemessung, somit nur als Massendurchflusssensor geeignet. Eine Weiter-
entwicklung des Vortexdurchflusssensors ist der Drallsensor (siehe Abbildung 2f). Durch den induzier-
ten Drall kann die Einlaufstrecke signifikant verkürzt werden [52, S. 294 ff.].

Thermischer Massendurchflusssensor Die Durchflussmessung von Gasen ist mittels thermischen
Massendurchflussmesser möglich. Zur Messung des Massendurchflusses wird die Wärmeleitfähigkeit
des zu messenden Fluids genutzt. Der prinzipielle Aufbau so eines Sensors ist in Abbildung 3 dargestellt.

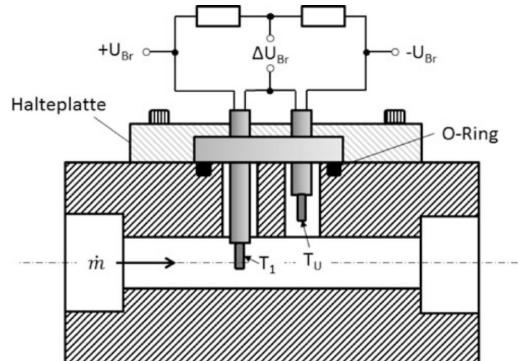


Abbildung 3: Prinzip eines thermischen Massendurchflusssensors [50, S. 656]

Auf das Messprinzip dieses Sensors wird genauer eingegangen, da er im Rahmen dieses Projekts und
in den folgenden Praxisveranstaltungen verwendet wird. Im Inneren des Durchflussmessers sind zwei
Temperatursensoren verbaut. Ein Sensor befindet sich nicht direkt im Strömungskanal (in einem Sack-
loch) und misst die Veränderung der Umgebungstemperatur T_U , als auch die Wärmeübertragung des
Fluids als Referenz. Der zweite Sensor wird beheizt. Die Temperaturdifferenz ΔT wird konstant ge-
halten. Je höher die Fließgeschwindigkeit des Mediums, desto mehr Energie muss dem zweiten Sensor

zugeführt werden, um die Temperaturdifferenz konstant zu halten. Der Massenstrom \dot{m} ist dem Wärmestrom \dot{Q} proportional. Es gilt folgender Zusammenhang zwischen Massen- und Wärmestrom:

$$\dot{Q} = \dot{m} \cdot c_p \cdot \Delta T \quad (3)$$

Der Massenstrom ist der zugeführten Energie in Form von Strom I^2 proportional. Der zweite Sensor hat einen temperaturabhängigen Widerstand $R(T)$.

$$\dot{m} = \frac{R(T) \cdot I^2}{c_p \cdot \Delta T} \quad (4)$$

Beide Sensoren sind zu einer Wheatstone'schen Messbrücke verschaltet, $\sqrt{\dot{m}}$ ist der Ausgangsspannung ΔU_{Br} proportional [50, S. 656].

2.1.2 Druckmessung

In der Industrie ist die Bestimmung von Drücken eine fundamentale Messoperation. Bis in die sechziger Jahren wurde der Druck in der Industrie mittels mechanischen Manometern bestimmt. Diese Technologie wurde daraufhin von den sog. Druckaufnehmern, mittels Dehnungsmessstreifen (DMS) des Werkstoffs Metall oder Drucksensoren, unter der Verwendung von Halbmetallen, abgelöst.

Dehnungsresistive Sensoren Drucksensoren mittels DMS sind sog. *dehnungsresistive* Drucksensoren. Dehnungsresistive Sensoren nutzen elektrisch leitende Werkstoffe bzw. Metalle. Das Messprinzip beruht auf der Änderung des elektrischen Widerstandes R , eines elektrisch leitenden Werkstoffs, als Folge einer mechanischen Verformung. Es existieren drei Drucksensor technologien auf Metallbasis.

- DMS aus Draht
- Metallfolien-DMS
- Dünnfilm-DMS

In den Jahren von 1938 bis zum Ende der 50er Jahre bestand der DMS aus einem Draht, der mäanderförmig auf einen Dehnkörper aufgebracht wurde. Der Draht wurde von Metallfolien und später Dünnfilm-

DMS abgelöst [47, S. 434]. Zur Erläuterung des physikalischen Wirkprinzips wird als Modell ein zylindrischer Körper genutzt.

Zylindermodell Ein zylinderförmiger, elektrisch leitender Werkstoff, charakterisiert durch die Ausgangslänge l_0 , Ausgangsquerschnitt A_0 und dem spezifischem elektrischem Widerstand ρ_{el} , hat einen elektrischen Widerstand R gemäß folgender Gleichung [51, S. 4 ff.]:

$$R = \rho_{\text{el}} \cdot \frac{l_0}{A_0} \quad (5)$$

Von diesem Modell ausgehend wird die relative Widerstandsänderung $\Delta R/R$, durch die Bildung des totalen Differenzials, berechnet. Die Lösung der Differentialgleichung, gemäß [51, S. 7 ff.] (siehe Anhang Gleichung (61), unter der Definition einiger Randbedingungen, führt zur Gleichung (6)).

$$\frac{\Delta R}{R} = \frac{\Delta \rho_{\text{el}}}{\rho_{\text{el}}} + \frac{\Delta l}{l_0} - \frac{\Delta A}{A_0} \quad (6)$$

Die geometrische Verformung wird durch die beiden letzten Terme repräsentiert. Die Längendehnung ψ_L ersetzt $\Delta l/l_0$ und der Term $\Delta A/A_0$ wird durch die *Querkontraktion*, Gleichung (7), mit der *Querkontraktionszahl* ν ($-\psi_L/\psi_Q$), auch Poisson-Konstante genannt, beschrieben. Die mathematische Umformung des Terms $\Delta A/A_0$, unter Anwendung von Gleichung (7), führt zur Gleichung (8).

$$\psi_Q = \frac{\Delta D}{D_0} = -\nu \cdot \psi_L \quad (7)$$

$$\frac{\Delta A}{A_0} = 2\psi_Q = -2 \cdot \nu \cdot \psi_L \quad (8)$$

Durch Einsetzen der Gleichung (8) und der mechanischen Dehnung ψ_L in die Gleichung (6) erhält man:

$$\frac{\Delta R}{R} = \frac{\Delta \rho_{\text{el}}}{\rho_{\text{el}}} + (1 + 2\nu) \cdot \psi_L \quad (9)$$

Der Term $\Delta\rho_{\text{el}}/\rho_{\text{el}}$ ist der sog. *piezoresistive Anteil* (präzisiert in Abschnitt **Halbleiter Drucksensoren**, Gleichung (15)), der bei metallischen Leitern vernachlässigbar ist, jedoch bei Halbleitern einen signifikanten Einfluss hat. Wird nun die sog. *Dehnungsempfindlichkeit* (*engl. gauge-factor*) k eingeführt, erhält man folgende Gleichung (für DMS auf Metallbasis wird der Term $\Delta\rho_{\text{el}}/\rho_{\text{el}}$ als 0 angenommen):

$$\frac{\Delta R}{R} = \underbrace{\left(1 + 2\nu + \frac{\Delta\rho_{\text{el}}}{\rho_{\text{el}}} \cdot \psi_L^{-1}\right)}_{=k} \cdot \psi_L \quad (10)$$

Im allg. ist die Dehnungsempfindlichkeit k nicht konstant, da sie eine Funktion der mechanischen Verformung eines elektrisch leitenden Werkstoffs ist. Für metallische Leiter liegt der Wert von k zwischen 2 und 6 und kann bei vielen DMS als eine Konstante angenommen werden. Bei Halbleiter-DMS kann der k Faktor 50 mal größer sein und weist dadurch, aufgrund der Abhängigkeit von der Dehnung, eine signifikante Nichtlinearität auf. Eine Wheatstone-Brücke wandelt die Widerstandsänderung ΔR in eine proportionale Spannungsänderung um [47, S. 434 ff.].

Quadermodell Der Draht als DMS wurde in den 50er Jahren von den Metallfolien-DMS; mit den Dimensionen Länge L , Breite b , Höhe h ; ersetzt, dessen elektrischer Widerstand sich wie folgt berechnet:

$$R = \rho_{\text{el}} \frac{L}{b \cdot h} \quad (11)$$

$$\check{R}(\psi_L) = \rho_{\text{el}}(\psi_L) \frac{L(1 + \psi_L)}{b \cdot h(1 - \nu \cdot \psi_L)^2} \quad (12)$$

Auf die Herleitung des exakten elektrischen Widerstands \check{R} der Gleichung 12 wird im Rahmen dieser Arbeit verzichtet. Mechanische Dehnungen ψ_L in Längsrichtung sind im technisch relevanten Bereich kleiner 0,1 %. Für diese Dehnungen kann der nichtlineare Term vernachlässigt werden. Für die Linearisierung des spez. elektrischen Widerstands eines Quaders, wird das totale Differenzial des elektrischen Widerstands (Gleichung 11) gebildet. Die Lösung ist Gleichung 13 [35, S. 42 f.].

$$\frac{\Delta R}{R} = \frac{\Delta \rho_{\text{el}}}{\rho_{\text{el}}} - \underbrace{\frac{\Delta b}{b} - \frac{\Delta h}{h}}_{=-2\nu\psi_L} + \underbrace{\frac{\Delta L}{L}}_{=\psi_L} \quad (13)$$

$$= \underbrace{\frac{\Delta \rho_{\text{el}}}{\rho_{\text{el}}}}_{=k} + (1 + 2\nu) \cdot \psi_L \quad (14)$$

Wie zu Beginn dieses Abschnitts erwähnt, existieren neben den DMS aus Metallfolie, ebenfalls Dünnfilm-DMS. Durch die Dünnfilmtechnologie können, gegenüber den Metallfolien-DMS, kleinere Messbereiche erschlossen werden. Der Fertigungsaufwand von Dünnfilm-DMS ist signifikant höher [47, S. 444 ff.]. Für genauere Information muss an Fachliteratur verwiesen werden.

Halbleiter Drucksensoren Neben den metallischen Drucksensoren, existieren Drucksensoren aus Halbleitern. Es existieren die folgenden zwei Drucksensor-technologien auf Halbleiterbasis:

- piezoresistive Drucksensoren
- kapazitive Drucksensoren

Eine Tabelle mit den Kenndaten von *piezoresistiver* und *kapazitiver* Drucksensoren befindet sich im Anhang in der Tabelle 67.

Wie im vorherigen Abschnitt erwähnt, ist die Dehnungsempfindlichkeit k bei Halbleitern nicht als eine Konstante anzunehmen. Der *piezoresistive Anteil* $\Delta \rho_{\text{el}} / \rho_{\text{el}}$ ist eine Funktion der longitudinalen (Index l) und transversalen (Index t) mech. Spannungszuständen $\sigma_{l,t}$ und den *Piezokonstanten* $\pi_{l,t}$ eines elektrisch leitenden Feststoffs, gemäß Gleichung (15):

$$\frac{\Delta \rho_{\text{el}}}{\rho_{\text{el}}} = \sigma_l \cdot \pi_l + \sigma_t \cdot \pi_t \quad (15)$$

2.2 Verfahrenstechnischer Prozess: Durchströmung von Schüttungen

Für die Durchströmung poröser Schichten existieren zwei unterschiedliche Ansätze.

- Das Modell des hydraulischen Durchmessers.
- Das Modell der Einzelkornumströmung.

Eine Partikelschicht wird auch Schüttung oder Festbett genannt. Aus dem Modell des hydraulischen Durchmessers wurden diverse empirische Gleichungen abgeleitet, die sich maßgeblich von Ihrem Gültigkeitsbereich unterscheiden. Mit den empirischen Gleichungen werden strömungsmechanische Effekte nicht mit abgebildet, die in dem Modell der Einzelkornumströmung implementiert sind. Für viele Anwendungen reicht die Approximation, durch Nutzung einer empirischen Gleichung aus. Bei der Auswahl einer empirischen Gleichung sind die Gültigkeitsbereiche und Randbedingungen zu beachten.

Gleichungen zur Beschreibung von Schüttungsdurchströmungen geben den Zusammenhang aus folgenden Variablen an [43, S. 144]:

- eine charakteristische Durchströmungsgeschwindigkeit w , bspw. die mittlere Leerrohrgeschwindigkeit ($\bar{v} = \dot{V}/A_a$)
- eine Druckdifferenz Δp
- Fluideigenschaften, wie die Fluidichte ρ_f und Viskosität η
- und Schichtbeschreibende Geometrien, wie:
 - Feinheitsmerkmal x der Partikel, oftmals der Sauterdurchmesser d_{32}
 - die Porosität ε
 - die Schichthöhe ΔL
 - der Anströmboden oder die Anströmfläche A_a
 - ...

Ein Dimensionsanalyter Ansatz zur Eruierung von Durchströmungsgleichungen führt, unter der Anwendung des Buckingham'schen Π -Theorems, zu den folgenden dimensionslosen Kennzahlen:

$$\Pi_1 \equiv \frac{\Delta p}{\rho \cdot w^2} = \text{Eu} \quad (\text{Eulerzahl}) \quad (16)$$

$$\Pi_2 \equiv \frac{w \cdot x \cdot \rho}{\eta} = \text{Re} \quad (\text{Reynoldszahl}) \quad (17)$$

$$\Pi_3 \equiv \frac{\Delta L}{x} \quad (\text{Längensimplex}) \quad (18)$$

$$\Pi_4 \equiv \varepsilon = \frac{V_H}{V} = \frac{1 - V_{fs}}{V} \quad (\text{Porösität}) \quad (19)$$

Die *Porösität* ε ist das Verhältnis von Hohlraumvolumen V_H zu Gesamtvolume V und kann gemäß Gleichung (19) berechnet werden. Der Feststoff oder Partikel wird im folgenden mit fs indiziert [43, S. 146].

Zur Beschreibung von Partikeln nutzt man Feinheitsmerkmale. Der Sauterdurchmesser d_{32} wird in vielen Gleichungen zur Beschreibung von Partikeln verwendet. Der mittlere Sauterdurchmesser \bar{d}_{32} einer Schüttung kann sich, **für Partikel die keine hohe Porösität (signifikant hohe Partikeloberfläche) aufweisen**, gemäß Gleichung (20) aus einer Partikelgrößenverteilung berechnen lassen. Das Volumen des i -ten Partikelgrößenintervalls ist mit V_i gekennzeichnet [41, S. 1426 ff.].

$$\bar{d}_{32} = \left[\sum_{i=1}^N \left(\frac{V_i}{V} \cdot \frac{1}{d_{32i}} \right) \right]^{-1} \quad (20)$$

Der *hydraulische Durchmesser* d_h steht mit dem Sauterdurchmesser d_{32} und der *spezifischen Oberfläche* S_V im folgendem Zusammenhang [43, S. 144].

$$d_h = 4 \cdot \frac{\varepsilon}{1 - \varepsilon} \cdot \frac{1}{S_V} \quad (21)$$

$$d_h = \frac{2}{3} \cdot \frac{\varepsilon}{1 - \varepsilon} \cdot d_{32} \quad (22)$$

Der *hydraulische Durchmesser* d_h , ist der mittlere Porendurchmesser einer Schüttung. Das Modell appro-

ximiert die Porösität einer Schüttung durch parallele Kanäle, die nicht miteinander in Wechselwirkung stehen [41, S. 1426 ff.].

2.2.1 Darcygleichung

Neben der Dimensionslosenkennzahl ε , ist die Einführung der Reynoldszahl Re zielführend. Die Reynoldszahl ist das Verhältnis der Trägheitskraft zur Reibungskraft einer Strömung und ist für die Umströmung von Partikeln eines Feinheitsmerkmals x wie folgt definiert.

$$Re_x = \frac{\bar{w} \cdot \bar{x} \cdot \rho_f}{\eta} \quad (23)$$

Als charakteristische Geschwindigkeit wird die Leerrohrgeschwindigkeit v gewählt.

In der Strömungslehre, für die Umströmung von **Einzelpartikeln**, wird zwischen drei Strömungszuständen unterschieden, die in der Realität fließend ineinander übergehen. Bei der Durchströmung von Schüttungen wird zwischen zwei Strömungszuständen gemäß Abbildung 4; zähe Durchströmung (I) und zäh-turbulente Durchströmung (II); nach Stieß unterschieden [43, S. 146]. In der Abbildung 4 ist die Widerstandsfunktion $f(Re, \varepsilon)$ gegen Re aufgetragen.

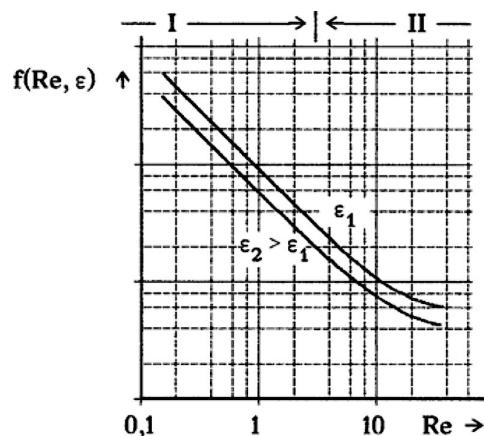


Abbildung 4: Widerstandsfunktion für die Durchströmung von Schüttgütern, gemäß [43, S. 146]

Die Widerstandsfunktion kann gemäß folgender Gleichung berechnet werden:

$$f(\text{Re}_x, \varepsilon) = \frac{\text{Konstante}(\varepsilon)}{\text{Re}_x} \quad (24)$$

Die „Konstante“ zur Berechnung der Widerstandsfunktion, muss von der Porösität der Schüttung abhängen. Gemäß Abbildung 4 ist der Grenzwert, zwischen denen die Strömung unterschieden wird, eine Reynoldszahl des Betrags von ca. 3.

Die Darcygleichung ist für die Berechnung von strömungsinduzierten Druckverlusten Δp , zäh durchströmter Schüttungen (I) anwendbar. Unter der Annahme, dass das Π -Theorem mit den Lösungen (Gleichung (16 – 19)) nach der Eulerzahl auflösbar ist, das Schüttungen *homogen* sind und *Isotropie* innerhalb der Schüttung vorliegt, gilt Gleichung [43, S. 145]:

$$Eu = \frac{\Delta L}{d} \cdot f(\text{Re}_x, \varepsilon) \quad (25)$$

Werden die Gleichungen (24) und (23) in die Eulergleichung (25) eingesetzt, erhält man

$$\frac{\Delta p}{\Delta L} = \frac{\text{Konstante}(\varepsilon)}{\bar{x}^2} \cdot \eta \cdot \bar{v}. \quad (26)$$

Die Schichteigenschaften werden in der Durchlässigkeit B wie folgt zusammengefasst:

$$B \equiv \frac{\bar{x}^2}{\text{Konstante}(\varepsilon)} \quad (27)$$

Die Durchlässigkeit B muss für jede Schüttung empirisch ermittelt werden und wird in [m^2] angegeben. Die Durchlässigkeit charakterisiert die Porenquerschnitte [43, S. 147]. Dadurch vereinfacht sich Gleichung (26) zu folgender Gleichung:

$$\frac{\Delta p}{\Delta L} = \frac{\eta \cdot \bar{v}}{B} \quad (28)$$

2.2.2 Ableitung der Filtergleichung aus der Darcygleichung

Um eine Filtergleichung für den Betriebszustand einer konstanten, irreversiblen Druckdifferenz Δp_{irr} zu erhalten, muss die Ableitung gemäß [42, S.100 ff.] mit folgenden Annahmen erfolgen:

1. Die Massenkonzentration c_m bzw. die Volumenkonzentration c_v der Suspension bleibt zeitlich und örtlich Konstant,
2. Das gewonnene Filtrat ist Feststofffrei,
3. Der Filterkuchen weist eine homogene Isotrope Struktur auf,
4. Filterkuchen sowie Filtermittel werden zäh Durchströmt (I).

Das folgende mathematische Modell, zur Ableitung der Filtergleichung, liegt der Abbildung 5 zugrunde.

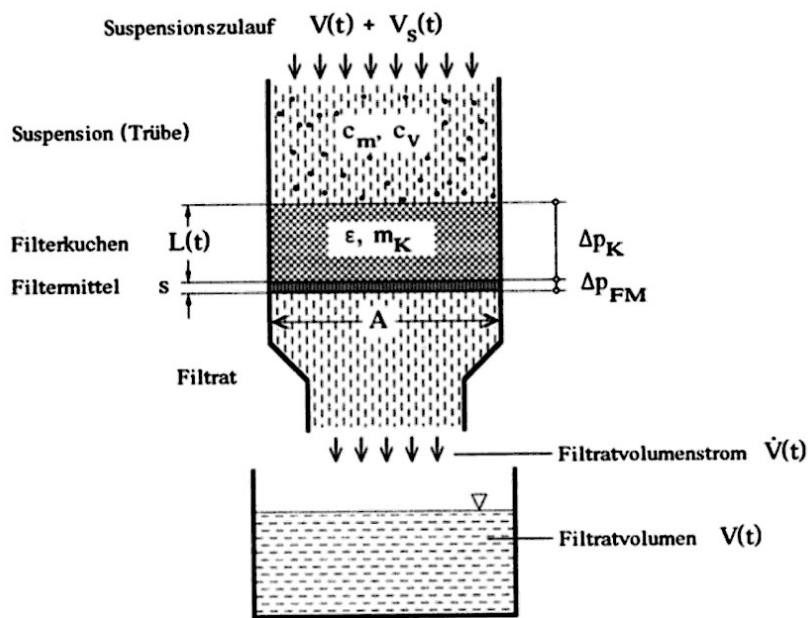


Abbildung 5: Filterkuchenmodell zur Ableitung der Filtergleichung, gemäß [42, S.100 ff.]

Der gesamte irreversible (irr.) Druckverlust Δp_{irr} ist eine Summe aus filterkucheninduziertem Δp_K und filtermittelinduziertem Δp_{FM} Druckverlust.

$$\Delta p_{\text{irr}} = \Delta p_K + \Delta p_{\text{FM}} \quad (29)$$

Der Druckverlust des Filtermittels, mit der Filtermitteldicke s und der Filtermitteldurchlässigkeit B_{FM} , kann gemäß Gleichung (28) wie folgt berechnet werden:

$$\Delta p_{\text{FM}} = \frac{\eta \cdot s}{B_{\text{FM}}} \cdot \frac{\dot{V}}{A} \quad (30)$$

Der Volumenstrom ist im Verlauf der Filtration zeitlich nicht konstant, daher ist das Volumen V über die Zeit zu differenzieren:

$$\Delta p_{\text{FM}}(t) = \frac{\eta \cdot s}{B_{\text{FM}} \cdot A} \cdot \frac{dV}{dt} \quad (31)$$

Analog ist zur Berechnung des Druckverlustes durch den Filterkuchen, mit der zeitlich anwachsende Filterkuchendicke $L(t)$ und dessen Durchlässigkeitskonstante B_{K} (unter der Annahme von *Isotropie* und *Homogenität*), folgender Ansatz zu wählen:

$$\Delta p_K(t) = \frac{\eta \cdot L(t)}{B_{\text{K}} \cdot A} \cdot \frac{dV}{dt} \quad (32)$$

Summiert man Gleichung (31) und Gleichung (32) erhält man Gleichung (33).

$$\Delta p_{\text{irr}}(t) = \frac{\eta}{A} \cdot \left[\frac{L(t)}{B_{\text{K}}} + \frac{s}{B_{\text{FM}}} \right] \cdot \frac{dV}{dt} \quad (33)$$

Durch die Einführung des Filtermittelwiderstandes $\beta = s/B_{\text{FM}}$ und des spezifischen Filterkuchenwiderstands $\alpha_V = 1/B_{\text{K}}$ vereinfacht sich Gleichung (33) zu:

$$\Delta p_{\text{irr}}(t) = \frac{\eta}{A} \cdot \left[\alpha_V \cdot L(t) + \beta \right] \cdot \frac{dV}{dt} \quad (34)$$

Unter der Voraussetzung (1) sowie (3), dass das Kuchenvolumen V_{K} und das Filtratvolumen V in einer direkt proportionalen Beziehung zueinander stehen ($V_{\text{K}} \sim V$), kann die Proportionalitätskonstante κ eingeführt werden.

$$L(t) = \frac{V_{\text{K}}(t)}{A} = \kappa \cdot \frac{V(t)}{A} \quad (35)$$

Ein weiterer Vorteil der Einführung der Proportionalitätskonstante κ , ist die Substitution der schwierig zu messenden Filterkuchendicke $L(t)$, durch das Filtratvolumen $V(t)$, die sich präzise bestimmen lässt.

Setzt man Gleichung (35) in Gleichung (34) ein, dann erhält man eine leicht auswertbare Gleichung folgender Form:

$$\Delta p_{\text{irr}}(t) = \frac{\eta}{A} \cdot \left[\frac{\alpha_V \cdot \kappa}{A} \cdot V(t) + \beta \right] \cdot \frac{dV}{dt} \quad (36)$$

2.2.3 Ableitung der Filtergleichung für die Betriebsweise mit einer konstanten Druckdifferenz

Filtrationen kann man durch unterschiedliche Betriebsweisen realisieren. In der folgenden Auflistung sind die möglichen Varianten aufgelistet:

- I Isobar – $\Delta p = \text{konstant}$,
- II Isochor – $\Delta V = \text{konstant}$,
- III Polytrop – Δp und ΔV variabel.

Für die isobare Filtration, mit einer konstanten Druckdifferenz $\Delta p_{\text{konst,irr}}$ gilt der nun folgende Ansatz.

Durch Trennung der Variablen der Gleichung (36), erhält man Gleichung (37) wodurch man durch Integration Gleichung (38) erhält.

$$dt = \frac{1}{\Delta p_{\text{konst,irr}}} \cdot \left[\frac{\eta \cdot \alpha_V \cdot \kappa}{A^2} \cdot V(t) + \frac{\beta \cdot \eta}{A} \right] \cdot dV \quad (37)$$

$$t = \frac{\eta \cdot \alpha_V \cdot \kappa}{A^2 \cdot 2 \cdot \Delta p_{\text{konst,irr}}} \cdot V^2(t) + \frac{\eta \cdot \beta}{A \cdot \Delta p_{\text{konst,irr}}} \cdot V(t) \quad (38)$$

Löst man Gleichung 38 nach $V(t)$ auf, erhält man Gleichung (39), wodurch die zeitlich zunehmende Filtratmenge berechnet wird. Die grafische Darstellung einer Filterkurve ist der Abbildung 6 zu entnehmen.

$$V(t) = \sqrt{\left(\frac{\beta \cdot A}{\alpha_V \cdot \kappa} \right)^2 + \frac{2 \cdot A^2 \cdot \Delta p_{\text{konst,irr}}}{\eta \cdot \alpha_V \cdot \kappa} \cdot t - \frac{\beta \cdot A}{\alpha_V \cdot \kappa}} \quad (39)$$

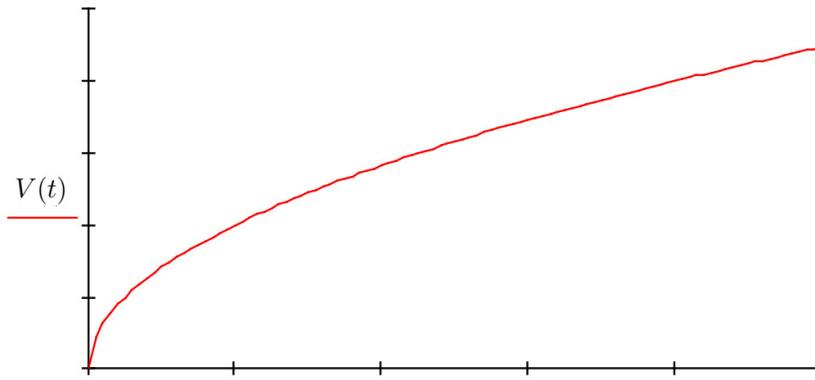


Abbildung 6: Filterkurve bei isobarer Betriebsweise [11, S. 8]

Eine weitere Lösung ist die Linearisierung der Gleichung 38, durch Division durch das Volumen als Funktion der Zeit $V(t)$. Die grafische Darstellung der Gleichung (40) ist der Abbildung 7 zu entnehmen.

$$\frac{t}{V(t)} = \underbrace{\frac{\eta \cdot \alpha_V \cdot \kappa}{A^2 \cdot 2 \cdot \Delta p_{\text{konst,irr}}} \cdot V(t)}_{a_1} + \underbrace{\frac{\eta \cdot \beta}{A \cdot \Delta p_{\text{konst,irr}}}}_{a_0} \quad (40)$$

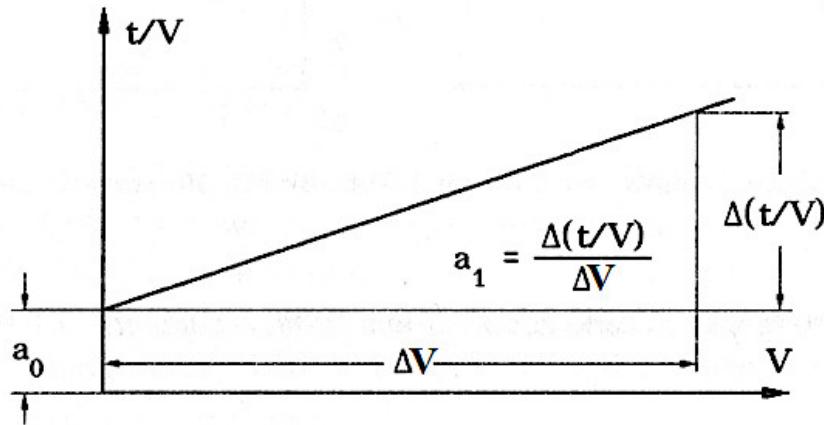


Abbildung 7: Filtergerade für die isobare Betriebsweise [42, S.102 ff.]

Die Ableitungen der Darcy-Gleichung für die isochore (II) oder polytropen (III) Betriebsweise werden im Rahmen dieser Arbeit nicht erläutert.

Anmerkung Neben der Darcy-Gleichung existieren noch weitere empirische Gleichungen, zur Beschreibung von Schüttungen. Die, neben der Darcy-Gleichung, häufig genutzten empirischen Gleichungen werden nachfolgend aufgelistet [43, S. 148 ff.][41, S. 1426 f.]:

- zäh durchströmte Schüttung
 - Carman-Kozeny-Gleichung (höherer Detailgrad)
- zäh *turbulent* durchströmte Schüttung
 - Ergun-Gleichung
 - Brauer-Gleichung

Das Modell der Einzelpartikelumströmung nach *Molerus* kann bspw. dem VDI-Wärmeatlas entnommen werden [41, S. 1427 ff.].

2.3 Verfahrenstechnischer Prozess: Gas-Feststoff Wirbelschicht

Das Wirbelschichtverfahren ist eine verfahrenstechnische Grundoperation (*engl. Unit Operation*). Dieses Verfahren wird im Sprachgebrauch der Wirbelschichttechnik auch Fluidisierung (*engl. fluidization*) genannt. Die Fluidisierung wird für verschiedene Zwecke eingesetzt, wie z.B. zum Fördern, Mischen, Agglomerieren [10], Trocknen, Rösten, aber auch in der Reaktortechnik, wie zum Verbrennen von Schüttgütern [38]. Das Partikelbett (*engl. fixed-bed*) liegt in einem Wirbelschichtapparat auf Anströmböden A_a . Im Labormaßstab entspricht das einem Filtergewebe.

2.3.1 Festbett-/Wirbelschichtparameter

Ein Schüttgutpartikelbett, welches keine Anströmung durch ein Fluid erfährt oder keinen fluidähnlichen Verhalten hat, wird als Festbett (*engl. fixed-bed*) bezeichnet. Es kennzeichnet sich durch folgende Parameter aus [41, S. 1561 ff.]:

- das Schüttgutvolumen V_{fb}
- der Masse m_{fs}
- die Dichte der Partikel ρ_{fs}
- der Festbettporosität ε_{fb} , der Partikelgrößenverteilung; die dem Feinheitsmerkmal der Partikel, dem sog. Sauterdurchmesser d_{32} ($d_{32} = 6/S_V$) inhärent ist
- und der apparatabhängigen Anströmfläche A_a
- Die mittlere Leerrohrgeschwindigkeit \bar{v}

In der Wirbelschichttechnik verwendet man die **Leerrohrgeschwindigkeit**. Die Leerrohrgeschwindigkeit \bar{v} ist auf die leere Wirbelschichtquerschnittsfläche A_a bezogene Fluidgeschwindigkeit.

Des Weiteren werden für die vollständige Berechnung der Wirbelschicht weitere dimensionslose Kennzahlen benötigt, auf die im Rahmen dieser Arbeit nicht weiter eingegangen wird. Nähere Informationen können bspw. dem VDI-Wärmeatlas entnommen werden [41, S. 1561 ff.].

2.3.2 Klassifizierung von Schüttgütern nach Geldart

Nicht alle Schüttgüter sind für Wirbelschichten gleichermaßen geeignet. Geldart hat eine Klassifizierung durchgeführt (siehe Abbildung 8). Im sog. Geldart-Diagramm ist auf der Ordinate die Differenz zwischen Fluid und Partikel dargestellt. Auf der Abszisse ist der mittlere Partikeldurchmesser \bar{d}_{32} (gemäß Grafik d_p) aufgetragen. In den Abbildungen sind vier Cluster (C, A, B, D), zwei breite und ein schmaler Übergangsbereich zu erkennen. Das Verhalten der Schüttgüttypen in einer Wirbelschicht unterscheidet sich z.T. signifikant [41, S. 1566].

Gruppe C Schüttgüter die nach Klasse C klassifiziert werden, haben kleine mittlere Partikeldurchmesser und weisen größere Haftkräfte auf, als die durch die Strömung induzierte Kraft. Es bilden sich Kanäle aus. Eine Ausbildung einer Wirbelschicht ist nur durch apparative Hilfsmittel oder der Zusatz durch Fließmittel möglich [41, S. 1566].

Gruppe A Oberhalb der Lockerungsgeschwindigkeit bildet sich eine homogene WIrbelSchicht aus. Wird die Leerrohrgeschwindigkeit weiter erhöht Bilden sich Blasen aus, die sich beim emporsteigen zusammenschließen (Koaleszenz). Die Blasenaufstiegsgeschwindigkeit ist höher als die des Zwischenraumgas und die Blasengröße ist ab erreichen einer bestimmten Betthöhe begrenzt. Wird die Anströmung unterbrochen, kollabiert das Partikelbett langsam [41, S. 1566].

Gruppe B Unmittelbar nach Überschreiten der Lockerungsgeschwindigkeit bilden sich Blasen aus. Die Blasenaufstiegsgeschwindigkeit ist höher als die des Zwischenraumgas. Die Blasengröße ist nicht begrenzt. Die Bettausdehnung ist geringer als die der Schüttgüter der Gruppe A. Wird die Anströmung unterbrochen kollabiert das Bett schlagartig [41, S. 1566].

Gruppe D Die Partikel dieser Gruppe sind groß und weisen eine große Dichte auf. Die Blasen haben eine geringere Aufstiegsgeschwindigkeiten als die des Zwischenraumgases. Nutzt man keinen breiten Anströmboden sondern nur eine Zentrale Bohrung, dann bildet sich bei hinreichend großer Anströmgeschwindigkeit eine Strahlschicht aus [41, S. 1566].

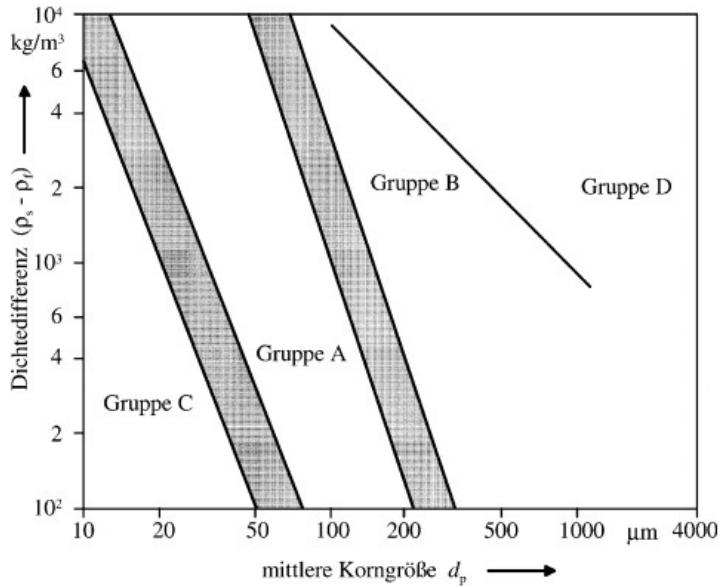


Abbildung 8: Klassifizierung von Schüttgütern nach Geldart [41, S. 1566 ff.]

2.3.3 Wirbelschichtzustände eines für die Fluidisierung optimalen Schüttguts

Um die Funktionsweise der Wirbelschicht zu verstehen, werden vier Zustände der Wirbelschicht; anhand eines für die sog. *Fluidisierung* optimalen Schüttguts (Klassifizierung nach Geldart A), gemäß der Abbildung 9; beschrieben. In der Abbildung 9 sind Wirbelschichtzustände als Funktion der Anströmgeschwindigkeit schematisch dargestellt [41, S. 1561 f.].

Abbildung 9a zeigt ein Schüttgutfestbett, welches mit einem \dot{V} Volumenstrom beaufschlagt wird, welcher kleiner ist als der minimale Fluidiservolumenstrom \dot{V}_{mf} . Die Mehrheit der Partikel befindet sich in Kontakt.

Sobald die Anströmgeschwindigkeit der minimalen Fluidisiergeschwindigkeit entspricht, ist der sog. Lockerungspunkt erreicht (**Abbildung 9b**). Das Schüttgut nimmt fluidähnliches Verhalten an. Das gesamte, nun Fluidbettbett ist in einer homogenen Bewegung. Es liegt der Zustand der Minimalfluidisation vor.

Wird die Strömungsgeschwindigkeit weiter erhöht tritt eine Blasenbildung auf. Welche Form der Blasenbildung entsteht ist unter anderem abhängig von der Geometrie des Apparates (vgl. **Abbildungen 9c**

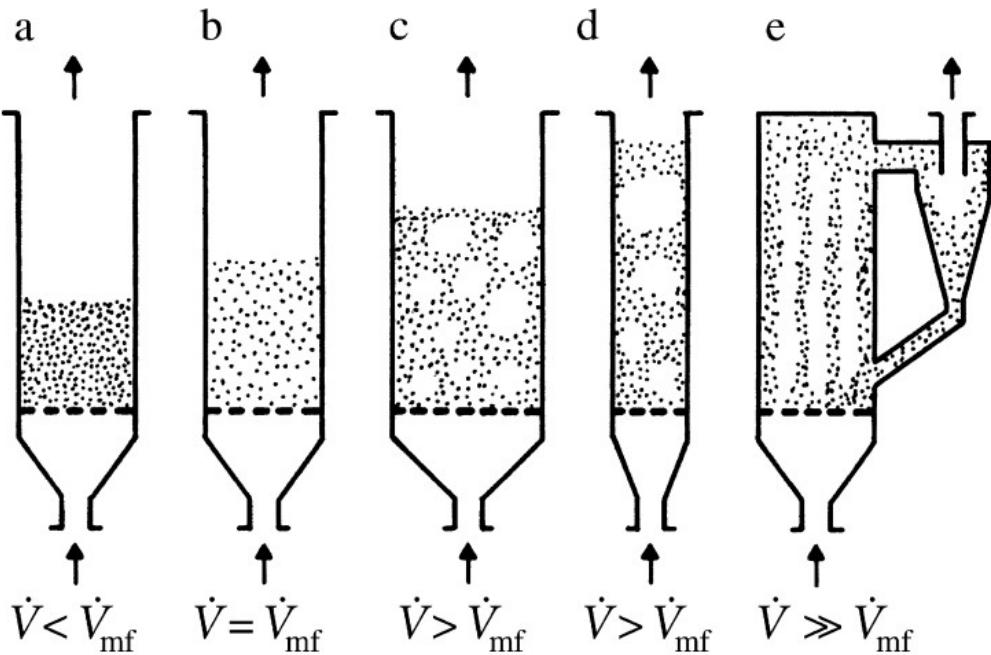


Abbildung 9: Wirbelschichtzustände als Funktion der Anströmgeschwindigkeit [41, S. 1562]

und 9d). Beim Aufsteigen der vereinzelten Gasblasen verschmelzen diese. Dieser Vorgang wird Koaleszenz genannt.

Ist die Anströmgeschwindigkeit viel größer als die minimale Fluidisiergeschwindigkeit und erreicht bzw. übersteigt die Sinkgeschwindigkeit der Partikel (Einzelkornsinkgeschwindigkeit), dann bricht die Blasenbildung auf und der Feststoff wird ausgetragen (siehe Abbildung 9e). Wie in der Abbildung 9e zu erkennen, bilden sich Feststoffsträhnen, dadurch ist die Aufrechterhaltung der Wirbelschicht möglich. Wenn das Ziel der Wirbelschicht nicht der Partikelförderung entspricht, dann ist eine Rückführung der Partikel, bspw. durch einen nachgeschalteten Zyklon, gemäß Abbildung 9e zu realisieren.

Anmerkung Schüttgüter der Geldartklassifizierung C und D benötigen andere apparative Lösungen. Es wird auf Fachliteratur wie bspw. den VDI-Wärmeatlas hingewiesen.

2.3.4 Druckverlust beim Betrieb einer Wirbelschicht

Die Wirbelschicht wird dadurch ausgezeichnet, dass sich das Partikelbett in einem definierten Volumenstromintervall wie ein Fluid verhält. Das Partikelbett hat in dem Intervall zwischen der **Lockerungsgeschwindigkeit** \bar{v}_{mf} und der Partikelaustragsgeschwindigkeit fluidähnliche Eigenschaften. Die Partikelaustragsgeschwindigkeit kann mit der **Einzelkornsinkgeschwindigkeit** \bar{w}_{fs} abgeschätzt werden. Das Leerrohrgeschwindigkeitsintervall der Fluidisierung ist somit $[\bar{v}_{mf}; \bar{v}_{fs}]$ [41, S. 1562 ff.].

Der Wirbelschicht liegt ein Kräftegleichgewicht zugrunde. Im stationären Zustand befinden sich die Partikel in der Schwebew. In der Abbildung 10 ist das Kräftegleichgewicht eines Wirbelschichtelements skizziert. Das Kräftegleichgewicht eines Wirbelschichtelements der Länge ΔL und der Anströmfläche A_a lässt sich wie folgt berechnen.

$$\Delta P_{\text{Schicht}} \cdot A_a = G_{fs} + G_f \quad (41)$$

Das Feststoffgewicht G_{fs} ergibt sich für eine Schicht mit der Feststoffdichte ρ_{fs} durch

$$G_{fs} = \rho_{fs} (1 - \varepsilon) g A_a \Delta L \quad (42)$$

und das Gewicht des Fluids, mit der Fluiddichte ρ_f durch

$$G_f = \rho_f \varepsilon g A_a \Delta L. \quad (43)$$

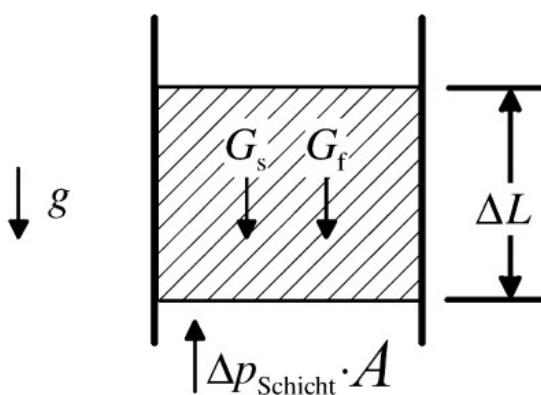


Abbildung 10: Wirbelschichtkräftegleichgewicht an einem Wirbelschichtelement der Länge ΔL und Anströmfläche A , angelehnt an [41, S. 1562]

Der Druckabfall einer Schicht berechnet sich mit den Gleichungen 42 und 43 aus Gleichung 41 wie folgt:

$$\Delta P_{\text{Schicht}} = \underbrace{(\rho_{fs} - \rho_f)(1 - \varepsilon) g \Delta L}_{=\Delta p_{\text{irr}}} + \underbrace{\rho_f g \Delta L}_{=\Delta p_{\text{reversibel}}} \quad (44)$$

Der erste Term repräsentiert den irreversiblen Anteil des Druckabfalls. Um die Partikel in der Schwebe zu halten wird Energie benötigt, die sich anhand des ersten Terms berechnen lässt. Der irreversible Anteil des Druckabfalls setzt sich aus der Gewichtskraft pro Fläche (Gleichung 45), abzüglich der volumenstrominduzierten Auftriebskraft pro Fläche (Gleichung 46) zusammen [41, S. 1562 ff.].

$$p_{fs} = \rho_{fs} (1 - \varepsilon) g \Delta L \quad (45)$$

$$p_f = -\rho_f (1 - \varepsilon) g \Delta L \quad (46)$$

Der zweite Term der Gleichung (44) repräsentiert die potentielle Energie des geförderten Fluids, der Höhe ΔL . Diese Energie kann im Falle einer Flüssig-Feststoffwirbelschicht, bspw. durch die Nutzung einer Turbine, zurückgewonnen werden [41, S. 1562 ff.].

Die Leerohrgeschwindigkeit der Minimalfluidisation v_{mf} , auch Lockerungsgeschwindigkeit genannt, kann empirisch ermittelt werden. Wird der dimensionslose Druck gegen die Leerohrgeschwindigkeit aufgetragen, ist die Lockerungsgeschwindigkeit mit dem erreichen des dimensionslosen Drucks mit dem Betrag von 1 erreicht (siehe Abbildung 11). Ab dem Überschreiten der Leerohrgeschwindigkeit ist der Druckverlust, relativ zur Umgebung Δp ebenfalls konstant, dazu wird eine doppelt logarithmische Auftragung empfohlen. Simultan zum Anstieg der Leerohrgeschwindigkeit, ab der Lockerungsgeschwindigkeit v_{mf} , ist eine Expansion der Wirbelschicht messbar. Die Druckkonstanz ab dem Lockerungspunkt ist damit zu erklären, dass die erhöhte Energiezufuhr für die Expansion der Wirbelschicht genutzt wird. Ab einer bestimmten Expansion schließen sich Hohlräume zusammen und es kommt zur Blassenbildung (siehe Abbildung 9). Dieser Effekt liegt der Koalesenz zugrunde.

2.3.5 Anmerkung

Für tiefgreifendes Wissen zur Berechnung von Wirbelschichten wird auf Fachliteratur, wie dem VDI Wärmeatlas [41, S. 1561 ff.], verwiesen.

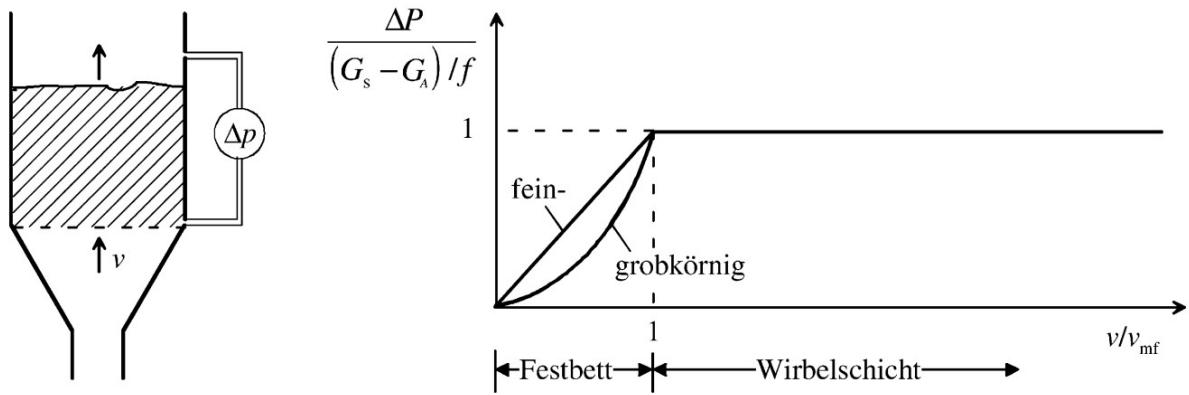


Abbildung 11: Konstanz des dimensionslosen Drucks, ab dem überschreiten der Lockerungsgeschwindigkeit v_{mf} , in Anlehnung an [41, S. 1563]

2.4 Signalverarbeitung

In diesem Abschnitt werden die Grundlagen der Signalverarbeitung erläutert. Im ersten Abschnitt werden nochmals die Grundlagen der Rechnerkommunikation erläutert, gefolgt von dem Unterschied zwischen analogen und digitalen Signalen. Es werden einige Kabelgebundene Schnittstellen, aber auch Schnittstellen die via Funk funktionieren, erläutert.

2.4.1 Vom Bit bis zur „vom Menschen lesbaren“ Alphanumeric

Sobald man von digitalen Daten spricht, ist von Bit encodierten, binären Daten die Rede. PC's sind in der Lage, diese Form der Signale direkt zu verarbeiten. In der Abbildung 12 sind verschiedene Datengrößen binärer Daten aufgezeigt. In der Abbildung variiert die Spanne der Datengrößenlänge von einem Bit bis zu einem Wort bestehend aus 16-Bit. Binäre digitale Daten sind diskret. Ein Bit hat zwei mögliche Zustände, an oder aus, die durch die binären Werte 0 oder 1 sowie den booleschen Ausdrücken `true` oder `false` repräsentiert werden können. Die Anzahl, wie viele Bit ein Zeichen (*engl. character*) repräsentiert, kann sich unterscheiden. Häufig wird ein Byte bestehend aus 7 oder 8-Bit verwendet. In der Abbildung 12 ist eine ByteLänge von 8-Bit, häufig auch Oktett genannt, dargestellt. Ein Byte, welches aus 8-Bit besteht, hat 256 mögliche Zustände. Die Hälfte eines (8-Bit) Byte wird Nibble genannt [17, S. 3]. Sind Signale ASCII encodiert, beträgt die Byte Länge 7 Bit. Der ASCII Zeichensatz beinhaltet somit 128 (0 - 127) Zeichen. ASCII-Tabellen können alle Zeichen der 7-Bit ASCII Codierung sowie

die Zuordnung der dezimalen und hexadezimalen Werte entnommen werden.

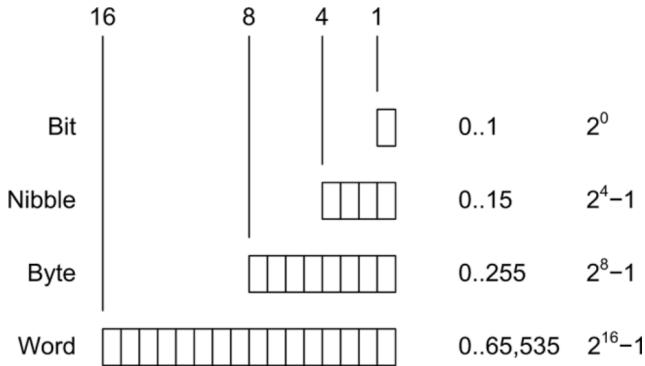


Abbildung 12: Binary Data Sizes [17, S. 3]

Hexadezimal Zeichen können neben der binärer und der dezimalen Darstellung auch Hexadezimal dargestellt werden. Die Hexadezimale Skala geht von 0 bis 9 gefolgt von A bis F. Die Skala hat somit eine dezimale Ziffernlänge von 16. Ein Nibble kann somit durch eine hexadezimale Ziffer dargestellt werden.

2.4.2 Analoge und digitale Signale

Möchte man Signale verarbeiten, dann ist zwischen zwei Signaltypen zu unterscheiden. Die reale Welt lässt sich durch ein Kontinuum von Zuständen beschreiben. Elektrische Messgeräte erfassen diese Zustände und erzeugen bspw. eine Spannung. Bei einer kontinuierlichen Messung würden bei einer definierten Frequenz, Signale in Form von Spannung generiert werden. Als Beispiel werfen wir einen Blick auf das Messen einer Temperatur. Bei diesem Beispiel nehmen wir einmal an, dass wir einen Temperaturfühler verwenden, der eine Messgenauigkeit von 0.0001 °C besitzt. Bei der Messung der Temperatur von Wasser könnte man eine Gleitkommazahl (floating point number oder kurz float) von 19.2334 °C angezeigt bekommen. Bis zur zweiten Messung könnte ein Temperaturausgleich zwischen Umgebung und dem Wasser erfolgt sein, was zu einem anderen Temperatormesswert von bspw. 12.240 °C führen könnte. Analoge Signale sind ein Kontinuum von Signalen.

PC's sind in der Lage digitale Signale sequenziell zu verarbeiten. Ein digitales Signale repräsentiert ein physikalisches Signal, Strom oder Spannung. In der Abbildung 13 sind typische Methoden abgebildet,

wie Daten eines Messobjekts akquiriert und an einen PC übermittelt werden. Oben im Bild sind vier Schalter zu erkennen, die einen Stromkreis schließen können. Diese Form der Signalübertragung wird diskret genannt. Auf dem Bild sind alle Schalter geöffnet, daher wäre pro Leiter ein binär Wert von 0 zu erwarten. In der Mitte der Abbildung ist ein Bit-serieller Datenstrom, anhand der Rechteckfunktion und den binär Werten, zu erkennen. Im unteren Teil der Abbildung ist ein analoges Signal zu erkennen, welches mittels eines analog zu digital Wandlers die Signale in eines von einem PC interpretierbares Signal wandelt. Die umgewandelten Daten werden mittels paralleler Schnittstelle an einen PC sendet.

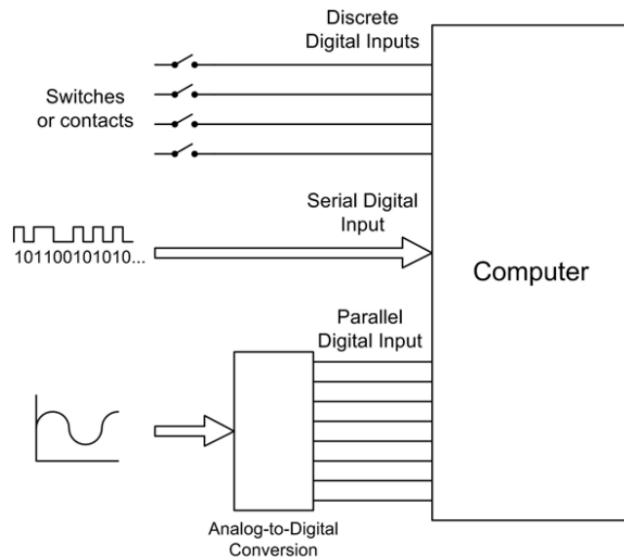


Abbildung 13: Digital and analog data inputs [17]

2.4.3 Schnittstellen

Messgeräte verfügen über Schnittstellen, über die Daten- und Steuersignale an andere Komponenten oder einen Messgerätebus erfolgen. An einen Messgerätebus werden Messgeräte (Slaves) und Rechner (Master) angeschlossen. Die Datenübertragung ist entweder Bit- oder Byte-seriell. Der Anschluss eines Messgeräts an einen PC kann über zwei Methoden erfolgen. Messgeräte können über analog Signalausgänge verfügen. Diese Signale müssen in einem analog/digital Wandler in ein digitales Signal gewandelt werden. Das Messgerät kann einen integrierten analog/digital Wandler und digitale Schnittstelle verfügen, über die Signale direkt an einen PC gesendet werden können. Die zwei Methoden können der Abbildung 14 entnommen werden [7, S. 479]. Diese analogen Signale müssen mittels einer Messkarte in ein digitales Signal umgewandelt werden, die im Anschluss in einem PC in einer Software weiter

verarbeitet werden können. Im folgendem werden zwei Kabelgebundene (USB, RS-232) und zwei Kabelfreie (Bluetooth und W-LAN) Schnittstellen erläutert.

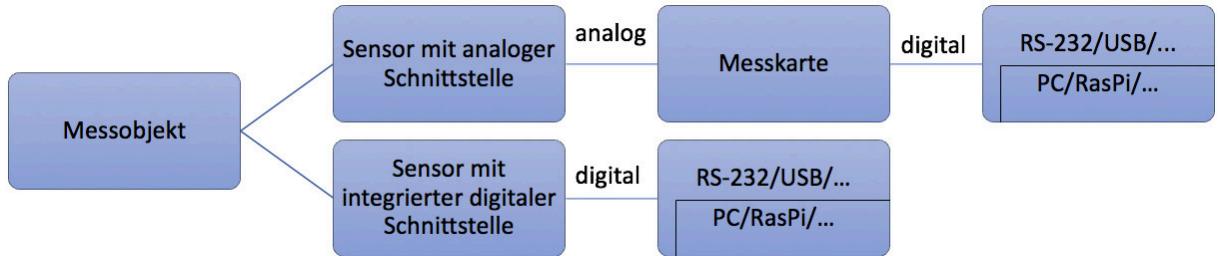


Abbildung 14: Anschluss eines Messgerätes an den PC, angelehnt an [7, S. 479])
oben mit Messkarte,
unten mit digitaler Schnittstelle

2.4.4 Verbinder

Um eine Kommunikation zwischen den Geräten zu ermöglichen, bedarf es physikalischer Schnittstellen. Im Laborumfeld werden die USB Schnittstelle und die DB-9 Schnittstelle besonders häufig angetroffen. In der Abbildung 15 ist ein männlicher DB-9 Stecker abgebildet.

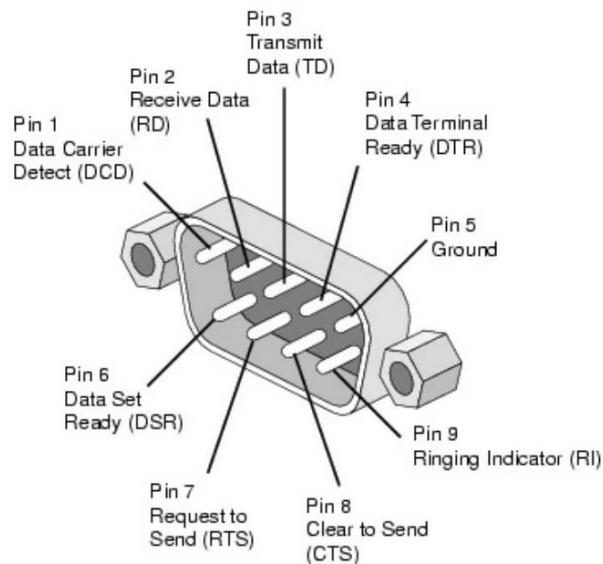


Abbildung 15: DB-9 pin and socket numbering [49]

2.4.5 Bitserielle Datenübertragung

Es gibt verschiedene Möglichkeiten Signale, bzw. Daten zu übertragen. Eine Möglichkeit ist die serielle Datenübertragung. Eine genauere Bezeichnung wäre Bit-serielle Datenübertragung, denn genau genommen gibt es eine Reihe von Bit-seriellen Datenübertragungssystemen, unter anderem jedoch auch die Byte-serielle Datenübertragung.

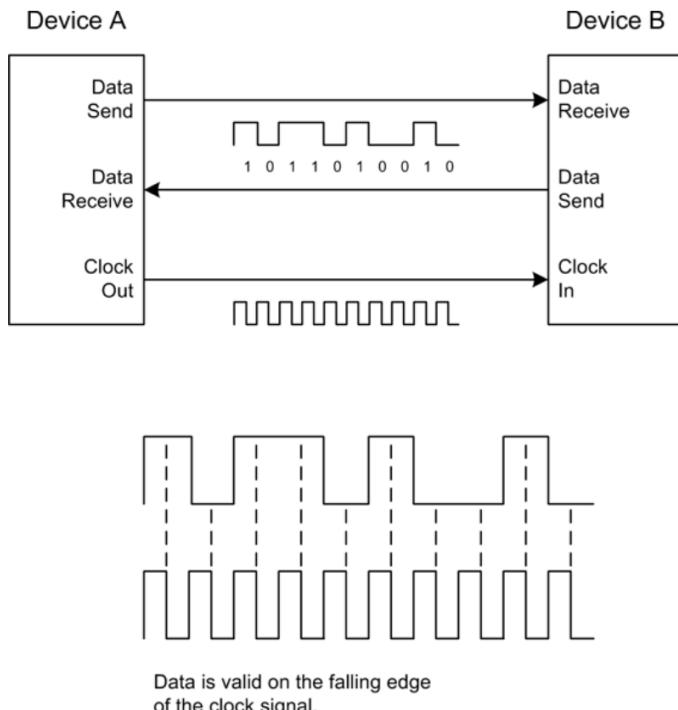


Abbildung 16: Synchronous serial data communication [17, S. 52]

Des Weiteren gibt es zwei Formen der seriellen Datenübertragung, die synchrone und die asynchrone Datenübertragung. In Abbildung 16 ist eine synchrone Datenübertragung dargestellt. An dem Ausgang *Clock Out* ist zu erkennen, dass das Gerät A den Takt und somit die Austauschrate der Schnittstelle vorgibt. Daten sind nur zum Zeitpunkt der fallenden Flanke des Taktsignals valide. Es ist zu erkennen, dass der Takt stets die Mitte der Bit Position trifft, um mögliche Fehler vorzubeugen. Die „reale“ Rechteckfunktion ist eine Superposition von Funktion verschiedener Frequenzen, was dazu führt, dass bei aufsteigender Flanke und beim fallen der Flanke ein Einschwingen der Funktion stattfindet, bis sich ein konstanter Wert einstellt (siehe Abbildung 17). In der Praxis wird eine asynchrone Bit-serielle Daten-

übertragung häufiger genutzt. Eine fehlerfreie Kommunikation soll die sogenannte Datenflußsteuerung (*engl. flow control*) übernehmen. Bei asynchroner Datenübertragung existiert somit kein Taktsignal. Bei einer asynchronen Datenübertragung kann der Bit Datenstrom neben der „gewünschten“ Informationen, z.B. von einer Messung, noch Datenflußsignale enthalten. Datenflußsignale können physikalische Signale sein. Diese Form der Flußsteuerung nennt sich *Hardware-Handshake*. Eine andere Möglichkeit der Datenflußsteuerung ist der sog. *Software-Handshake*. Die Datenflußsignale sind **Startbit**, **Stopbit** und **Paritätsbit**, die ein Zeichen (*engl. Character*) von einem anderen Zeichen abgrenzt. Zeichen können Buchstaben, Zahlen, Vorzeichen etc. sein, die binär versendet werden (0 oder 1) und beispielsweise ASCII encodiert sind.

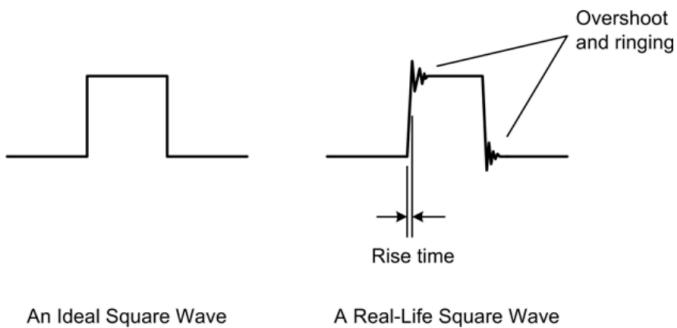


Abbildung 17: Ideal versus real square waves [17, S. 5]

2.4.6 RS-232

Die RS-232 Schnittstelle wurde in den 1960 Jahren von der Electronic Industrial Alliance (EIA) standardisiert. Jedes zu übertragende Zeichen (Zahl, Buchstabe, Sonderzeichen) wird zwischen 5 und 9 Bit codiert, wobei 7 und 8 Bit am gebräuchlichsten sind. Die Trennung von benachbarten Zeichen werden mit *Start-* und *Stopbit* realisiert. Die Übertragungsrate kann zwischen 0,3 Bd und 115,2 kBd betragen. Die Entfernung, die erzielt werden kann, ist stark vom verwendetem Kabel und der Übertragungsgeschwindigkeit abhängig. Es sind Entfernungen bis zu einem Kilometer realisierbar. Die Übertragungsgeschwindigkeit wird *Baudrate* genannt. Ein Bd entspricht 1 Zeichen pro Sekunde. Ein *Symbol/Character* kann aus einem oder mehreren Bits besteht. Die Übertragungsrate ist somit im Vergleich zu anderen Schnittstellen mit, im Normalfall von einem Bit pro Zeichen, der Übertragungsgeschwindigkeit von 0,3 bis 115,2 kbit/s verhältnismäßig gering, reicht aber für die meisten Messtechnischen Anwendungen aus. Für die Kommunikation mittels RS-232 benötigt man eine sog. Flusssteuerung (*engl. flow control*). Der

Sender und der Empfänger müssen sich gegenseitig mitteilen, ob sie sende- oder empfangsbereit sind, um Datenverluste zu vermeiden. Die Signale, die man dafür benötigt, werden *Handshake-Signale* genannt. Der *Handshake* lässt sich mit physischen Signalen über Steuerleitungen (*Hardware-Handshake*) oder über digitale Signale (*Software-Handshake*) realisieren. Die Software-Handshake Signale werden über die TxD Signalleitungen versendet. Für den *Software-Handshake* ist die XON/XOFF und für den *Hardware Handshake* die Datenflußsteuerung mittels CTS/RTS eine gängige Methode [7, S. 480 f.].

RS-232 beschreibt Spannungspegel und Timing zwischen der Datenübertragungseinrichtung (DÜE) und Datenendeinrichtung (DEE). Nicht Bestandteil des Standards ist das eigentliche Übertragungsprotokoll, wie die Signale kodiert sind (z.B. ASCII), Fehlererkennung usw.. Solche Parameter müssen zwischen der Datenübertragungseinrichtung und der Datenendeinrichtung ausgehandelt werden [25]. Üblicherweise werden für die Übertragung 9-Polige D-Sub Kabel verwendet, aber RS-232 ist nicht an diese Kabel und Stecker Kombination gebunden. In der Abbildung Abbildung 15 sind alle Signale und die Definition der Signale der DB-9 Schnittstelle aufgelistet. Bei der Verwendung eines Software-Handshake sind nur 3 Signalleitungen notwendig, da die Steuersignale über die Datenleitungen gesendet wird. Für den Datentransfer mittels 9-Poligen D-Sub Kabel (siehe Abbildung 15), sind somit lediglich die Pins 2 (RxD), 3 (TxD) und 5 (GND) notwendig. Bei der Verwendung eines Hardware-Handshakes werden die Datenleitungen über die Pins 7 (RTS) und 8 (CTS) notwendig [12]. RS-232 ist eine Volt-basierte Schnittstelle. Der logische Wert wahr (`true`) ist dem negativen Volt-Level zugeordnet und der logische Wert falsch (`false`) ist dem positiven Volt-Level zugeordnet. Die Spanne der Volt-Spannung ist in RS-232 von +/- 3 V bis +/- 25 V spezifiziert.

2.4.7 RS-485

RS-485 ist eine Weiterentwicklung des RS-232 Standards. RS-485 ist ein Bus System, welche bis zu 32 Datensender und Empfänger verbinden kann und wird in der Prozessmess- und Prozessleittechnik angewandt. Es sind Übertragungsgeschwindigkeiten von bis zu 35 MBit/s mit der Verwendung eines 10 Meter Kabels und 100 KBit/s mit einer Kabellänge von 1200 Meter möglich. Die RS-485 Schnittstelle verfügt über 2 Signalkabel. Die Polarität der beiden Kabel ist gegensätzlich (vergleiche Abbildung 18). Beide Leitungen können Signale in beide Richtungen senden, jedoch nicht zur gleichen Zeit. Der Signalabgriff ist somit differentiell. Das Signal wird somit nicht gegen Null gemessen, sondern zwischen den zwei Leitern gegensätzlicher Polarität. In Abbildung 18 ist eine 8-Bit asynchrone Datenübertragung via RS-485 zu erkennen. Am Anfang und dem Ende der 8-Bit befindet sich ein Start- und ein Stopbit

[17, S. 222 f.].

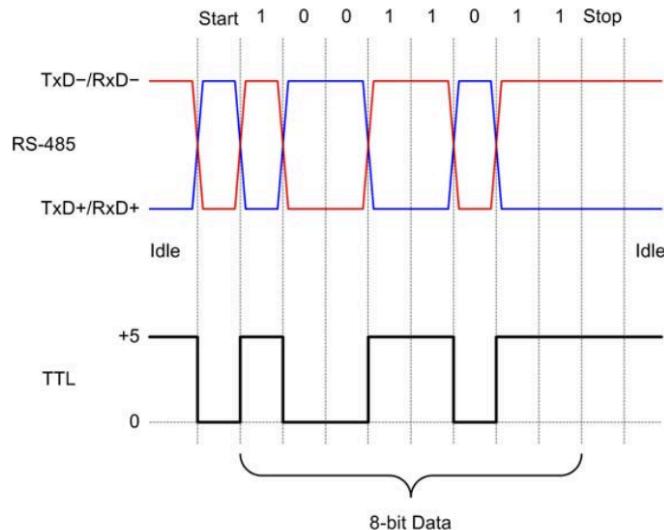


Abbildung 18: RS-485 signal levels [17, S. 226]

2.4.8 USB

Die überbrückbare Länge via USB liegt unter 5 m und liegt somit weit unter der Entfernung, die mit RS-232 möglich sind. USB wurde zwecks Schaffung einer einheitlichen Schnittstelle für alle Arten von PC's Laptops von INTEL entwickelt. USB-Kabel haben vier Leitungen, zwei für die *Bit-serielle* bidirektionale Datenübertragung, eine Leitung für die +5V Versorgungsspannung und eine für das Massepotenzial. An einen USB-Anschluss eines PC's lassen sich theoretisch bis zu 128 Geräte anschließen, da die angeschlossenen Messgeräte durch den USB-Controller im PC mit einer 7-Bit-Adresse adressiert werden. Daraus folgen $2^7 = 128$ mögliche Messgeräte. Praktisch sind es 127, da die Adresse „Null“ (000 0000) für die Geräte-Identifizierung genutzt wird. USB verfügt über zwei Eigenschaften, dem „Hot-Plugging“ und dem „Plug-and-Play“. Dank dem Hot-Plugging ist es vor dem Verbinden eines Geräts mit dem PC nicht notwendig diesen auszuschalten. Dank Plug-and-Play konfiguriert sich die Verbindung selbst. USB 1.0 hat eine Übertragungsgeschwindigkeit von 1,5 bzw. 12 MBit/s, USB 2.0 bis zu 480 MBit/s und USB 3.0 hat eine Highspeedübertragungsrate von 5 GBit/s. Es ist je nach verwendeten Geräten möglich, diese an einer USB-Schnittstelle zu betreiben, die eine geringere Übertragungsrate besitzen als die USB-Schnittstelle des PC's.

2.4.9 Bluetooth

Bluetooth ist eine kabellose Datenübertragungsmethode via Funk. Der Frequenzbereich ist zwischen 2,402 bis 2,480 GHz. Die Übertragungsdistanz beträgt ca. 10 m mit einer Übertragungsgeschwindigkeit von 2,1 MBit/s für die Spezifikation *Bluetooth 2.0 + EDR* (Enhanced Data Rate) [7, S. 481 f.]. Bluetooth 2.0 + EDR ist 2004 auf dem Markt erschienen. Bluetooth 4.2 Smart ist 2014 mit Einführungen von wichtigen Funktionen für das Internet der Dinge (IoT) erschienen. Bluetooth 5 (2017) ist die aktuellste Version und hat IoT im Fokus. Bluetooth 5 unterstützt Übertragungsraten von 2 MBit/s und einer Reichweite von maximal 250 m [8].

Drahtlose IoT- Lösungen nehmen in der produzierenden Industrie an Bedeutung zu. Da kabellose Netzwerke beim Aufbau von neuen Fertigungslinien flexibler sind als kabelgebundene Netzwerke, ist Bluetooth im Rahmen von Industrie 4.0 eine bedeutende Schnittstelle. Bluetooth Mesh Networking wurde unter anderem für industrielle drahtlose Sensornetzwerke konzipiert (WSN). Mit diesem Standard ist es möglich viele Geräte und Sensoren in einem Bluetoothnetzwerk kommunizieren zu lassen. Jedes Gerät wird „Knoten“ genannt [3].

Sicherheit: Mesh-Vernetzung per Bluetooth [3] Die Netzwerksicherheit bei produzierenden Betrieben ist nicht verhandelbar. Die Sicherheit bei der Verwendung von Bluetooth, soll durch die sog. Mesh-Vernetzung gewährleistet werden. Die Informationssicherheit soll durch ein Schichtenmodell erreicht werden, dass auf voneinander getrennten Sicherheitsschlüssel basiert:

Netzwerkschlüssel (NetKeys) gelten für alle Nachrichten im Netzwerk, damit die Knoten sicher miteinander kommunizieren.

Anwendungsschlüssel (AppKeys) schützen Nachrichten zu bestimmten Anwendungen wie Klimaanlage, Beleuchtung oder physische Sicherheit.

Geräteschlüssel (DevKey) ermöglichen das Einrichten und Konfigurieren eines Knotens, um neue Geräte zum Netzwerk hinzuzufügen.

WLAN

WLAN ermöglicht die Datenübertragung via Funk, ermöglicht jedoch eine signifikant höhere Übertragungsgeschwindigkeit und Übertagungsreichweite als Bluetooth. Mit der technischen Lösung 802.11 ist eine Gruppe von Standards für Funknetzwerke des Institute of Electrical and Electronics Engineers (IEEE) entwickelt worden.

2.4.10 Verarbeitung digitaler Signale in Labview

Jeff Kodosky et al. hat in Zusammenarbeit mit der University of Texas die Programmiersprache G und die dazugehörige Entwicklungsumgebung LabVIEW entwickelt.

LabVIEW stellt zum umsetzen spezieller Anwendung spezifische Werkzeuge und Modelle bereit. Das Ziel war es ein Tool zu entwickeln, wodurch Menschen mit unterschiedlichem know-how im Programmieren befähigt werden maßgeschneiderte Programme grafisch generieren zu können, da sich im Laborumfeld Versuchsaufbauten und somit auch Messungsaufgaben und Prozessabläufe schnell ändern. Diese Programme nennen sich *Virtual Instruments* (VI) und emulieren physische Instrumente. Diese Programmiersprache setzt sich aus zwei Programmiermethoden zusammen, dem strukturorientiertem Programmieren und der datenstromorientierten Programmierung [23]. Zu den strukturorientierten textbasierten Programmiersprachen gehören unter anderem C++, Java und Python. Programmiersprachen, die strukturorientiert arbeiten, befolgen auf der untersten Funktionsebene folgende drei Kontrollstrukturen [9]:

1. Sequenzielle Abarbeitung von Programmanweisungen
2. Verzweigung innerhalb Programmabschnitte (if/when/else)
3. Iterationen/Schleifen

Viele Messobjekte generieren kontinuierliche Datenströme. Die datenstromorientierte Programmiermethode ist bei dieser Art von Messobjekt die Methode der Wahl, daher liegt der Programmiersprache G im wesentlichen das Datenflusskonzept zugrunde. G ist eine effiziente Programmiersprache, die die Kommunikation mit den Geräten, die Visualisierung von Daten und deren Analyse ermöglicht. Mittels LabVIEW lassen sich Programme grafisch mittels drag and drop entwickeln. LabVIEW Programme setzen sich aus interaktiven Front Panels und dem Programm in Form von Blockdiagrammen zusammen [23]. Das Front Panel ist die Programmoberfläche des Programms, womit das Programm angewendet wird. In Abbildung 19 sind Frontpanel und Blockdiagramm eines einfachen Programms dargestellt.

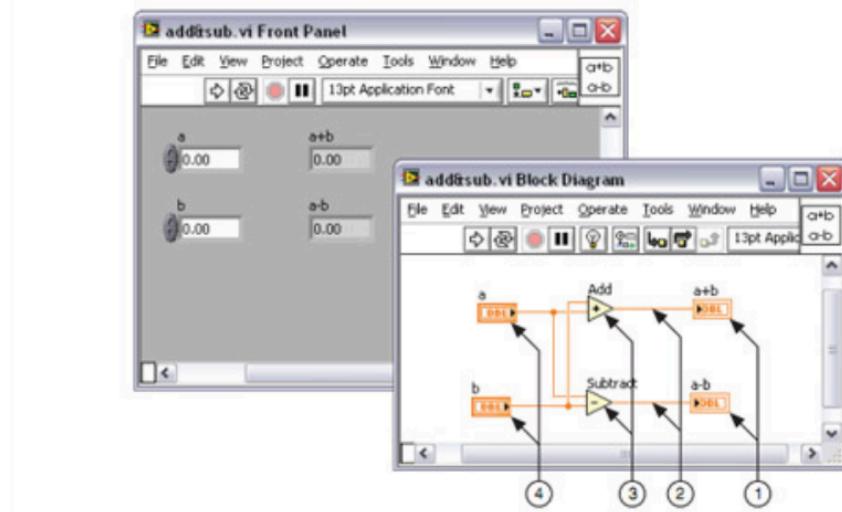


Abbildung 19: Beispiel für ein Blockdiagramm und das zugehörige Frontpanel [18]

2.5 Datenverwaltung

Die analogen Signale, die von einem Messobjekt akquiriert (*engl. acquisition*), in digitale Signale umgewandelt und in der Signalverarbeitung aufbereitet werden, sollen in den meistens Fällen weiterverwendet werden. Die Verwendung dieser Daten kann je nach Anwendungsfeld in Echtzeit oder zu einem beliebigen Zeitpunkt erfolgen. Der Zeitpunkt der Wiederverwendung ist bei der Wahl der Technologie zur Verwaltung der Daten zu beachten. Welche Art von Daten gespeichert werden sollen determiniert ebenfalls die Wahl der Lösung. Sollen z.B. nur Zahlen oder Strings abgespeichert werden oder ist auch die Speicherung von Audio, Bild oder Videos erwünscht oder in Betracht zu ziehen. Des Weiteren unterscheiden sich Softwarelösungen darin, ob die Daten von einem Anwender oder von einer Software autonom abgerufen und verwendet werden sollen [31, S. 2 ff.].

In diesem Abschnitt werden Informationsmanagementsysteme, das PERA Modell zur Strukturierung von Automatisierungsnetzen, welches Teil des ISA-95 Standards ist sowie ISA-S95 erläutert.

2.5.1 Informationsmanagementsysteme

Für strukturierte, leicht abrufbare Daten bieten sich Informationsmanagementsysteme (IMS) an. Im Rahmen großer Datenmengen in der Privatwirtschaft ist der Begriff Datawarehouse zu nennen, darauf wird im Rahmen dieser Arbeit nicht weiter eingegangen. IMS können verschiedenste Informationen miteinander Verknüpfen. Die automatisierte Erhebung von Kennzahlen ist damit einfach zu erreichen.

Des Weiteren wird eine schnelle Verfügbarkeit der Informationen (die sich im IMS befinden) für Anwender und Software durch gut programmierbare Abfragen (*engl. Querys*) ermöglicht. Der Nachteil besteht darin, dass die Daten, die das IMS erhebt bzw. dem IMS zugeführt werden sollen im Vorwege definiert werden müssen. Folglich kann daraus ein Daten bzw. Informationsverlust entstehen [31, S. 2 ff.]. Die Daten und Informationen die heute noch als unwichtig gelten, können morgen eine signifikante Relevanz in Bezug auf die Wettbewerbsfähigkeit eines Unternehmens haben [27]. IMS müssen von Administratoren (Spezialisten) instand gehalten werden, wodurch Kosten entstehen. Nutzt man einen Dienstleister, fallen i.d.R nur gut kalkulierbare Fixkosten an. Wird sich für ein betriebsinternes IMS entschieden, entstehen Fixkosten und variable Kosten (sobald Updates oder Modifikationen oder der gleichen an der IMS Struktur anfallen). Der Trend von Unternehmen in Bezug auf IMS geht weg von betriebsinterner Datenverwaltung in Richtung der Nutzung von Dienstleistern wie Amazon und co.. Vor- und Nachteile von Informationsmanagementsystemen können der Tabelle 4 entnommen werden.

Tabelle 4: Vor- und Nachteile von Informationsmanagementsystemen [31, S. 2 ff.]

Vorteile	Nachteile
<ul style="list-style-type: none"> • schnelle Verfügbarkeit von Information für Software und Anwender • Kennzahlengenerierung • Informationsverknüpfung 	<ul style="list-style-type: none"> • potentieller Informationsverlust von vermeintlich unwichtigen Daten • Wartungs-/Instandhaltungskosten

Informationen sind in vielen Branchen entscheidend für die Konkurrenzfähigkeit eines Unternehmens. Informationen können sich aus Daten destillieren lassen und sind in dem derzeitigen digitalem Zeitalter von großer Bedeutung. Neben den konventionellen Produktionsfaktoren, Arbeit, Boden und Kapital könnten Informationen bzw. Wissen somit hinzugezählt werden. Produktionsfaktoren sind Wirtschaftsgüter die genutzt werden, um Leistungen erbringen zu können. Produktionsfaktoren sind Ressourcen die geplant, gesteuert, überwacht und kontrolliert werden müssen. Wird sich dafür entschieden Wissen bzw. Daten als einen Produktionsfaktor zu betrachten, wird das Datenmanagement zur Führungsaufgabe. Die Entscheidung Informationen als Produktionsfaktor anzuerkennen wirkt sich wie folgt aus [31, S. 2 f.]:

- **Entscheidungsgrundlage:** Für viele Entscheidungen dienen Daten als Grundlage, somit sind sie für alle Organisationsfunktionen von Bedeutung.
- **Qualitätsanspruch:** Verfügbarkeit, Korrektheit und Vollständigkeit sind maßgeblich für die Qualität der Daten.
- **Investitionsbedarf:** Die Speicherung, Verwaltung, und ggf. Aufbereitung von Daten erzeugen Kosten und erfordern demnach Kapital.
- **Integrationsgrad:** Der Integrationsgrad der Daten bestimmt die Verknüpfbarkeit von Informationen zwischen den Aufgabengebieten und -träger jeder Organisation.

In der Abbildung 20 ist das Schema eines Informationsmanagementsystems dargestellt, welches mit einem Kommunikationsnetz bzw. mit dem Internet verbunden ist. Informationssysteme können eine Wissensbank, Methodenbank und/oder Datenbank enthalten. Auf die Methoden- und Wissensbank wird im Rahmen dieser Arbeit nicht weiter eingegangen [31, S. 2 f.].

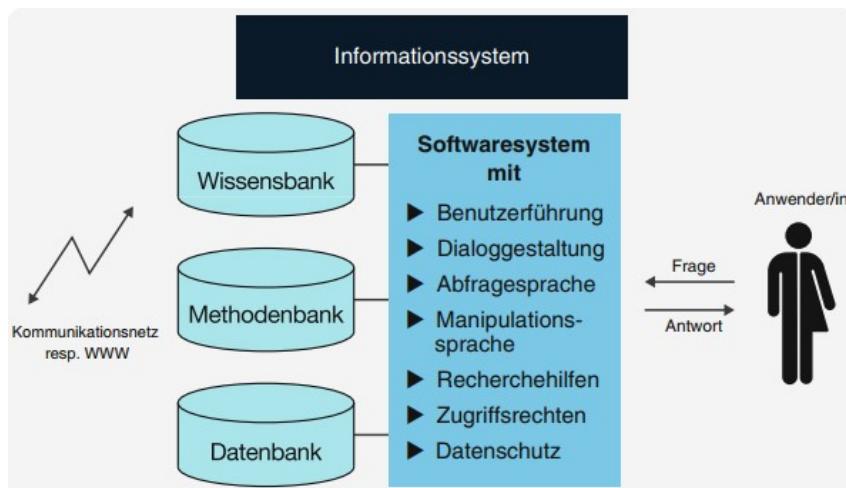


Abbildung 20: Architektur und Komponenten eines Informationssystems [31, S. 3]

Datenbankmanagementsysteme Ein Datenbanksystem (*engl. database management system, DBMS*) dient der Speicherung und Abfrage von Daten und besteht aus einer Speicher- und Verwaltungskomponente. Die Speicherungskomponente umfasst alle Daten, welche in einer organisierten Form abgespeichert werden sowie deren Beschreibung. Um die Daten und Informationen verändern und auswerten zu

können bedarf es einer Abfrage- und Manipulationssprache, die die Verwaltungskomponente darstellt. Die Zugriff- und Bearbeitungsrechte werden ebenfalls von der Verwaltungskomponente verwaltet. In der Praxis kommen häufig SQL-Datenbanken (*Structured Query Language*) zum Einsatz. Sind Datenbestände heterogener Natur und müssen in Echtzeit verarbeitet werden, dann sind SQL-Datenbanken ungeeignet. Für solche Anwendungsfälle bieten sich Lösung im Rahmen von Big Data oder Hybridlösungen an. Datenverwaltungslösungen im Rahmen von Big Data sind No-SQL Datenbanken (not only SQL-Datenbanken) [31, S. 2 f.].

SQL-Datenbanken SQL-Datenbanken sammeln Daten bzw. Informationen in Tabellen. Diese Tabellen können miteinander in Beziehung stehen. Aus diesem Grunde werden SQL-Datenbanken auch relationale Datenbanken genannt. In der Abbildung 21a ist eine Tabelle abgebildet, die den Namen MITARBEITER trägt. Tabellen bestehen aus Schlüsselmerkmalen und Attributen. Schlüsselmerkmale dienen der **eindeutigen** Identifikation. Ein Merkmal oder Attribut ordnet jedem Eintrag einer Tabelle einen bestimmten Datenwert aus einem vordefinierten Wertebereich (*engl. domain*) zu. Somit kann jeder Mitarbeiter, der eindeutig mit der Mitarbeiter ID identifiziert werden kann, mit Namen und Ort eingetragen und in der Tabelle gefunden werden. Als Schlüsselmerkmal bietet sich demnach die Mitarbeiter ID ($M\#$) an. Die minimale Schlüsselkombination wird in diesem Beispiel durch den **Identifikationsschlüssel** $M\#$ repräsentiert. Schlüsselkombinationen haben einen **Minimalitätsanspruch**. In diesem Fall darf kein Schlüssel oder Kombination gestrichen werden, ohne dass die eindeutige Identifikation verloren geht. Ein Schlüssel ist mit den Forderungen der **Minimalität** und **Eindeutigkeit** vollständig charakterisiert. Datenwerte dürfen mehrfach vorkommen, siehe Abbildung 21b. Eine komplette Zeile einer Tabelle wird Datensatz bzw. Tupel genannt und beschreibt ein Objekt in einer Tabelle (ID, Name, Adresse usw.) [31, S. 3 ff.].

Die Anzahl an Spalten und Zeilen ist unbegrenzt. Die Benennung von Tabellen und Merkmalen sowie der Identifikationsschlüssel bzw. Schlüsselkombination muss eindeutig sein. Mathematisch ist eine Relation R, die diesen Anforderung erfüllt, eine Teilmenge aus dem kartesischen Produkt von Wertebereichen $R \subseteq D_1 \cdot D_2 \cdot \dots \cdot D_n$ mit D_i als Wertbereich des i-ten Merkmals [31, S. 3 ff.].

Ein Tupel r ist demnach ($r = d_1, d_2, \dots, d_n$). Tupelredundanzen sind aufgrund des Teilmengenbegriffs nicht erlaubt, d.h. $R = \{r_1, r_2, \dots, r_n\}$

Anmerkung: Beim programmieren sind Tupel unveränderbare, schreibgeschützte Arrays!

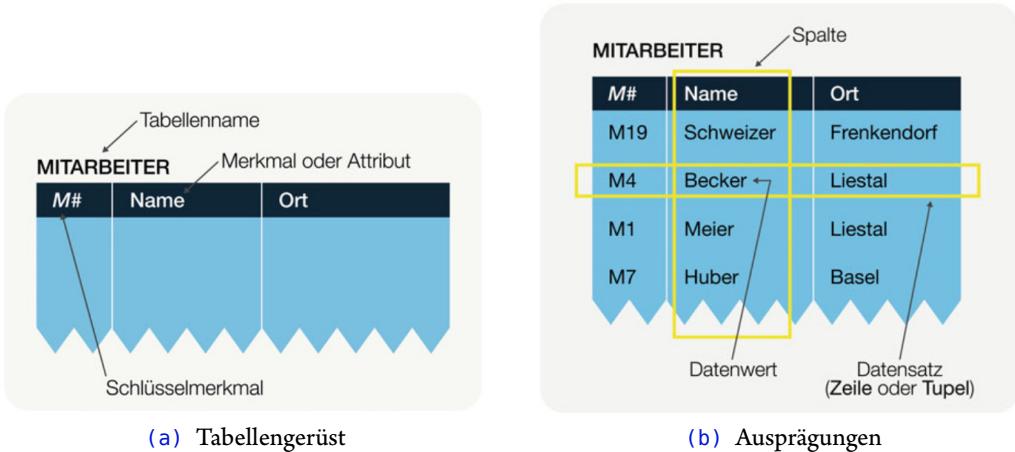


Abbildung 21: Relationenmodell: Tabellengerüst und Ausprägung einer am Beispiel einer MITARBEITER Tabelle

Strukturierte Abfragesprache SQL SQL (*Structured Query Language*) ist eine *deskriptive* von ANSI und ISO genormte Abfrage- und Manipulationssprache für Tabellen. In der Abbildung 22 ist ein Beispiel einer Abfrage unter der Tabelle MITARBEITER ist die Abfrage in natürlicher Sprache («Selektiere den ... wohnen») zu sehen. Das Resultat der Abfrage sollen alle Namen/Personen die den Wohnort Liestal angegeben haben sein. Die Formulierung einer Abfrage genügt dem Schema SELECT-FROM-WHERE. Eine Formulierung in SQL sieht wie auf Abbildung 22 zu erkennen folgt aus [31, S. 6 ff.]:

```
SELECT Name  
      FROM MITARBEITER  
 WHERE Ort = 'Liestal'
```

Für tiefgreifendes Wissen bezüglich SQL-Datenbanken muss auf Fachliteratur wie [31] verwiesen werden.

Big Data Datenbestände die mit den herkömmlichen technischen Lösungen (SQL-Datenbanken) aufgrund vom sehr hohem Datenvolumen, einer heterogenität von Daten und der Notwendigkeit diese in Echtzeit bei einer hinreichender Geschwindigkeit zu verarbeiten nicht mehr zu handhaben sind, werden unter dem Begriff von *Big Data* zusammengefasst. Der Begriff *Big Data* ist nicht klar definiert. Datenspezialisten wie die Gartner Group definieren Big Data in ihrem „*Information Technology Glossary*“ mit dem Vorhandensein von mindestens drei V's [31, S. 11]:



Abbildung 22: Formulierung einer SQL Abfrage

„**Big data** is high-*volume*, high-*velocity* and/or high-*variety* information assets that demand cost-effective, innovative forms of information processing that enable enhanced insight, decision making, and process automation.“, [13].

Wie bereits in Kapitel 2.5.1 erläutert sind Informationen entscheidend für die Konkurrenzfähigkeit von Unternehmen in vielen Branchen. Die Gartner Group bezeichnet Informationen sogar als *Vermögenswert* oder *Informationskapital* (*engl. information asset*).

Eine Präzisierung der drei *V*'s laut[31] ist folgende:

- **Velocity:** Datenströme sollen in Echtzeit analysiert und ausgewertet werden können,
- **Volume:** Umfangreiche Datenbestände bis zum Zettabytebereich (1 Zettabyte $+10^{12}$ GB),
- **Variety:** Strukturierte, semi-strukturierte und unstrukturierte Multimedia-Daten (Texte, Bilder, Grafiken, Audio Videos) sollen gespeichert werden können.

Zuletzt bleibt noch die Information als *Vermögenswert* (*engl. value*) und die Aussagekraft bzw. Wahrhaftigkeit (*engl. veracity*) der Daten bzw. Auswertungsergebnisse. Da die Daten im Rahmen von *Big Data*

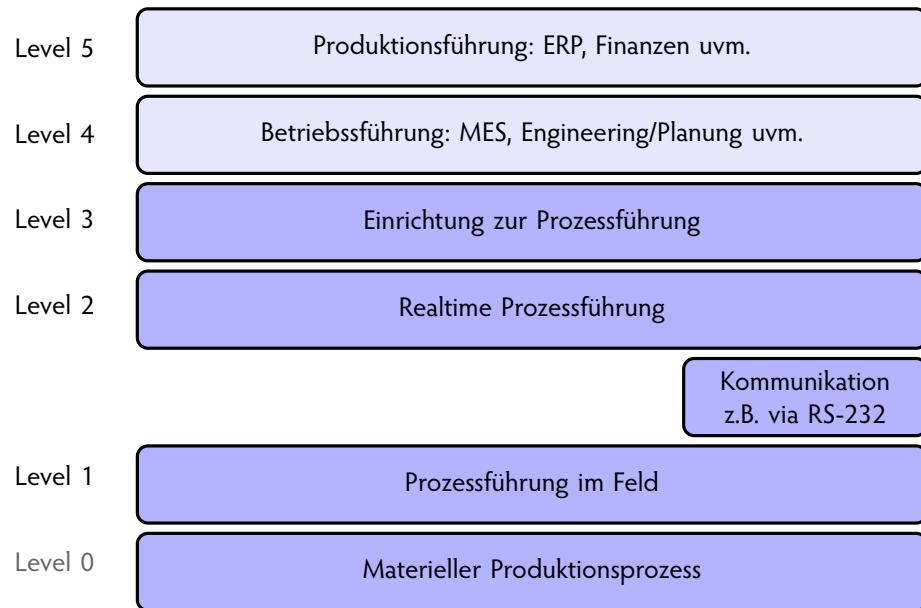


Abbildung 23: ICS-PERA Model, angelehnt an der Grafik des Bundesamtes für Sicherheit in der Informationstechnik des ICS-Kompendiums (vgl. Abbildung 68 im Anhang ; [40, S. 18])

heterogener Natur in Art und Präzision sind, müssen spezielle Algorithmen für hinreichende Ergebnisse angewendet werden.

2.5.2 PERA-ICS Modell

Die Purdue Enterprise Reference Architecture (PERA) wurde Anfang 1990 entworfen, um Automatisierungsnetze zu beschreiben und zu unterteilen. Des Weiteren dient es unter anderem dazu, den Schutz von industriellen Steuerungs- und Automatisierungsnetzen, vor Hackern, zu gewährleisten. Klassisch wird das ICS-Modell (Industrial Control System) in fünf Ebenen (*engl. Level*; siehe Abbildung 23) unterteilt. Die oberen zwei Hierarchieebenen sind, gemäß PERA-ICS, von Level 3 bis Level 0, aus sicherheitstechnischen Gründen, physisch voneinander getrennt [40, S. 15 ff.]. Auf die Funktionen und Komponenten, wie z.B. HMI/BuB (Human-Machine-Interface = Bedienung und Beobachtung), *Manufacturing Execution System* (MES) oder *Manufacturing Operations Management* (MOM) und *Enterprise Ressource Planing* (ERP) Systeme usw., die auf den Leveln vorhanden sind, wird im Rahmen dieser Arbeit nicht eingegangen.

Level 0 Gemäß des ICS-Kompendiums des Bundesamtes für Sicherheit in der Informationstechnik, befindet sich vor dem Level 1 der **materielle Produktionsprozess** [40, S. 18]. In dem ICS-Kompendium ist es nicht gelabelt, doch in vielen anderen Quellen wird dieses mit Level 0 deklariert.

Level 1 Level 1 ist die **Prozessführung im Feld** (*engl. Basic Control*). Daten und Signale werden auf dieser Ebene in Realtime generiert und erfasst. Die Ausführung der Steuer- und Regelung physischer Prozesse geschieht auf dieser Ebene. Als Beispiele, für die Infrastruktur dieser Ebene, können Remote I/O (ggf. mit Signalvorverarbeitung, dann spricht man von RTU), Interface-Bausteine zur Signalkonditionierung, Switches bei Verwendung von Feldbus Lösungen genannt werden [40, S. 18].

Level 2 Komponenten der **Realtime Signalverarbeitung**, im Sinne der Darstellung der automatisierten Funktionen sind dem Level 2 zugeordnet. Beispiele können Zustände der Komponenten der Feldebene sein. Als Beispiele können Füllstände, Motoren (an/aus; Leistungsabnahme), Ventilstellungen uvm. genannt werden. Stellvertretend für die Kommunikation zwischen diesem und Level 1 ist die **RS-232-Schnittstelle** aufgelistet [40, S. 19].

Level 3 **Einrichtungen zur Prozessführung** (*engl. Supervisory Control*) sind dem Level 3 zugeordnet. Diese Einrichtungen sind für die Prozessführung notwendig, jedoch werden keine Daten in Echtzeit verarbeitet. Als Beispiele können *HMI/BUB*, produktbezogene Engineering- und Wartungsstationen, Messwert- und Prozessdatenarchivserver genannt werden. Diese Komponenten sind wichtig, jedoch in Bezug auf das Zeitverhalten oder die Verfügbarkeit unkritischer als Komponenten der Level 1 und 2. Softwareupdates von Komponenten die dieser Ebene zugeordnet sind, werden restriktiv behandelt, da diese die Schnittstelle für die Komponenten der Level 1 und 2 sind, dessen Verfügbarkeit in Realtime zwingend erforderlich ist [40, S. 19 f.].

Level 4 Es ist zu erkennen, dass das 4. Level der **Betriebsführung** entspricht. Betriebsführung wird oftmals auch **operatives Management** genannt. Level 4 sind Funktionen und Komponenten wie, MES oder MOM, Engineering/Planung und lokale Office IT, zuzuordnen [40, S. 20].

Level 5 Hinter Level 5 verbirgt sich die **Produktionsführung**, mit Funktionen wie ERP Anbindung (interagiert oftmals mit MES Systeme (Level 4) via XML/B2MML, siehe Abschnitt 2.5.3, Inter-/Intranet Zugang, Remote Access Einrichtungen (zur Fernwartung) [40, S. 20 f.].

2.5.3 ISA-S95

Unternehmen sind sozio-ökonomische Systeme. Um global wettbewerbsfähig zu bleiben, ist Flexibilität und das Anwenden effizienterer Methoden, als Unternehmen aus „Billiglohnländern“, ein muss. Um die Wettbewerbsfähigkeit zu gewährleisten, ist ein System notwendig, welches die gesamte Wertschöpfungskette abbilden kann. Ein Austausch von Daten, bspw. zwischen der Supply Chain und der Produktion wird dadurch ermöglicht.

Unternehmen können hierarchisch, nach Level, unterteilt werden. Die International Society of Automation, kurz ISA, hat 1995 einen Standard etabliert, der Terminologie und abstrakte Modelle für den effizienten Datenaustausch zwischen Systemen verschiedener Hierarchiestufen ermöglicht. Ein effizienter Datenfluss zwischen verschiedenen Unternehmenssystemen, wie ERP (Level 5) oder MES (Level 4), wird durch den ISA-95 Standard ermöglicht. ISA-S95 definiert Terminologie und einheitliche Modelle, um die Kommunikation zwischen sämtlichen Kontroll- und Unternehmenssystemen zu verbessern. Die Aktivitäten innerhalb eines Unternehmens werden nach dem PERA-ICS Modell unterteilt. Die Level 0, 1 und 2 decken damit die Aktivitäten der Produktion ab. **Demnach enthält der Standard alles, was Unternehmen benötigen, um sich ihr MES entwickeln zu können [37].**

ISA-S95 definiert keine Datentypen. Die Definition, in welcher Form Daten für die Datenkommunikation vorzuliegen haben, übernimmt B2MML - Business To Manufacturing Markup Language. In XML-Schema Definition (XSD) wird die Datenstruktur festgelegt [37]. Die XSD-Dateien dienen somit als Referenz, zur Validierung der XML-Dateien, mit denen im Unternehmen operiert wird und können sich demnach, je nach Unternehmen und Unternehmensfunktion, unterscheiden. Auf der Seite des ISA-S95 gibt es diverse B2MML XML-Schemata frei zum downloaden [21]. Auf XML wird im Rahmen dieser Arbeit nicht weiter eingegangen. ISA-S95 besteht aus den folgenden fünf Parts.

Part 1 Der erste Part, „Models and Terminology (veröffentlicht 2000)“, kann als Leitfaden dienen, um Schnittstellen zwischen Prozess- und Produktions-/ Leitsystemen zu definieren [21].

Part 2 In Part zwei, „Object Model Attributes (veröffentlicht 2001)“, werden in Kombination mit Part 1 die Inhalte der Schnittstellen zwischen den Steuerungsfunktionen in der Produktion und der Unternehmensführung definiert [21].

Part 3 Der dritte Part, „Models of Manufacturing Operations“, konzentriert sich auf die Produktions- und MES Ebene (Level 4). Der Fokus liegt auf den Aktivitäten und Funktionen von Produktion, Wartung, Lagerhaltung und Qualitätskontrolle [21].

Part 4 „Object Models and Attributes of Manufacturing Operations Management“ dient als Basis für das Design und die Implementierung von Schnittstellenstandards [21].

Part 5 Part 5, „Business to manufacturing transactions“, nutzt die abstrakten Modelle aus Part 1 sowie 2 und definiert Transaktionsmodelle für den Informations-/Datenaustausch [21].

2.5.4 B2MML

Das World Batch Forum (WBF) hat, in Bezug auf ISA-S95, die Business To Manufacturing Markup Language (B2MML) etabliert. WBF ist ein Teil der Manufacturing Enterprise Solutions Association (MESA). ISA-S95 gibt die Terminologie und die abstrakten Modelle vor und das WBF hat mit B2MML einen Datentyp bereit gestellt. B2MML und XML sind textbasierte Formate und lassen sich somit von Menschen lesen (vgl. Ausschnitt der ProcessSegment XSD in der Abbildung 69 im Anhang). B2MML ist eine abgewandelte Form des ursprünglichen XML Formats. Für Batch und kontinuierliche Prozesse hat das WBF neben dem B2MML ebenfalls BatchML etabliert. Um die Datenstruktur besser zu verstehen, wird ein fiktiver (siehe Abbildung 24), jedoch plausibler Prozess und die Prozesssegment XSD (siehe Abbildung 25), in visualisierter Form, des *Methods Artikel* [14] verwendet. Im Rahmen dieser Erläuterung, soll diese Prozesskette einen Wirbelschichtreaktionsprozess darstellen.

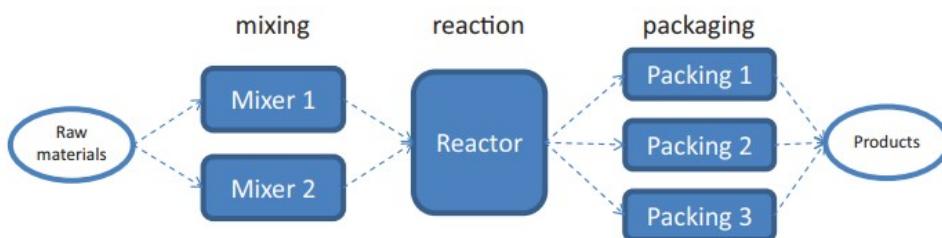


Abbildung 24: Fiktiver Prozess des Artikels [14, S. 3]

In der Abbildung 24 ist eine Schema einer Prozesskette eines Batchprozess; vom Rohmaterial, bis zum fertigen Produkt; abgebildet. Die verfahrenstechnischen Schritte sind das homogenisieren mittels Mixer (zwei verfügbar), der Reaktor, in dem die chemische Reaktion stattfindet und die Verpackung, wofür drei Packstationen oder Abfüllungen zur Verfügung stehen. In der Abbildung 25 ist das Datenformat

der Prozesssegment XSD, in visualisierter Form, dargestellt. Es ist eine hierarchische Struktur zu erkennen. Der Hierarchieursprung wird **Wurzelement** genannt und ist in dieser XSD Visualisierung *ProcessSegmentInformation*. Jede B2MML oder BatchML Datei hat einen identischen Aufbau. *ProcessSegmentInformation* ID (1) könnte z.B. „Wirbelschichtreaktionsprozess“ heißen. Demnach gäbe es drei *ProcessSegment* ID's (2); für „mixing“, „reaction“, „packaging“. Da sich Equipment des gleichen Typs, z.B. Mixer, in ihrer Funktion unterscheiden können, empfiehlt es sich ggf. *EquipmentClassID*'s (3) zu vergeben. Des Weiteren können Abhängigkeiten (*engl. Dependency*) zwischen den *ProcessSegments* existieren, die unter dem Reiter *SegmentDependency* (4 - 6) genau definiert werden können.

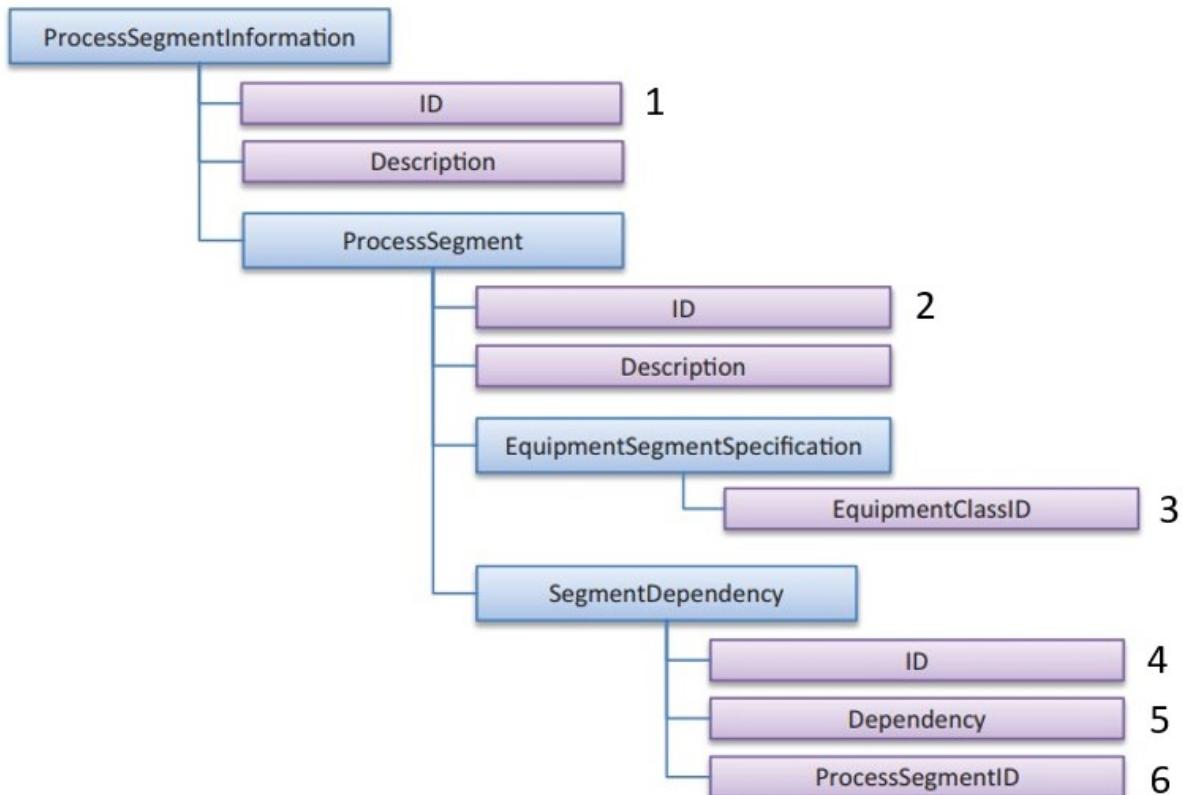


Abbildung 25: Prozesssegment Informationen [14]

In der folgenden Liste ist ein Teil der vordefinierten, der auf der MESA frei erhältlichen XSD-Files, nach Erstellung eines Accounts, aufgelistet [32]:

- ProcessSegmentInformation (B2MML-V0600-ProcessSegment.xsd)

- EquipmentInformation (B2MML-V0600-Equipment.xsd)
- MaterialInformation (B2MML-V0600-Material.xsd)
- PersonnelInformation (B2MML-V0600-Personnel.xsd)
- OperationsCapability (B2MML-V0600-OperationsCapability.xsd)
- OperationsDefinitionInformation (B2MML-V0600-OperationsDefinition.xsd)
- OperationsSchedule (B2MML-V0600-OperationsSchedule.xsd)
- OperationsResponse (B2MML-V0600-OperationsPerformance.xsd)
- BatchML-V0600-BatchInformation.xsd
- BatchML-V0600-BatchProductionRecord.xsd
- BatchML-V0600-GeneralRecipe.xsd

3 Konzeptentwicklung für die Labordigitalisierung

In diesem Abschnitt werden die Konzepte erläutert, die im Verlauf des Projekts erarbeitet werden. Es werden zwei Konzepte entwickelt, die auf das gesamte verfahrenstechnische Labor angewendet werden können. Die Konzepte schließen sich nicht gegenseitig aus, sondern bauen aufeinander auf. Als erstes wird das Konzept erläutert, welches auf die Verwendung einer Datenbank verzichtet. Als zweites wird ein Konzept erläutert, in dessen Zentrum sich eine Datenbank befindet. Die Datenübertragungswege sind durchnummieriert. Des Weiteren gibt es pro vorgestelltem Entwurf alternative Datenübertragungswege oder Lösungen die jedoch nicht betrachtet werden.

Es wurden **Anforderungen** für die Konzeptionierung definiert, die nachfolgend aufgelistet sind:

- Das Konzept soll für das verfahrenstechnische Labor allgemeingültig sein.
- Die „digitalen Kompetenzen“ der Studierenden sollen maximiert werden, ohne die fachlichen Kompetenzen signifikant zu reduzieren, durch
 - Unter Automatisierung kann z.B. die Manipulation von Daten, wie die Detektion von Ausreißer und dessen Entfernung, genaue Leitfäden in der Handhabung, die keine Fehler mehr zulassen o.ä., verstanden werden.
- Der monetäre Aufwand soll minimal sein.
- Möglichkeiten der Datenerfassung/Signalverarbeitung sollen eruiert werden.
- Die inkrementelle Implementation von Applikationen im Rahmen von Industrie 4.0 und Big Data (KI, Digital Twin etc.) soll tendenziell möglich sein.
- Cloud Computing soll in Betracht gezogen werden.
 - Erster Schritt der Umsetzung: Speicherung der Rohdaten auf der HAW Cloud
 - Zweiter Schritt der Umsetzung: Cloudnutzung, gemäß des Impulsvortrags zum Digitalisierungsfond ⇒ Amazon, Google, Microsoft Azure
- Die gesamte Protokollierung der Studierenden soll in Zukunft digital sein.

3.1 Konzeptentwurf 3.0 (ohne Datenbank)

In der Abbildung 26 ist ein Digitalisierungsentwurf ohne die Verwendung einer Datenbank dargestellt. Die Feldebene auf der linken Seite der Abbildung und die Softwareebene sind durch eine **Strich-Doppelpunkt Linie** voneinander getrennt. Der Feldebene sind Feldgerätedaten und Daten, die manuell eingegeben werden müssen, zugeordnet. Die Datenerfassung und das Schreiben von Messwertdaten in CSV-Dateien, die Auswertung und die Protokollierung sind der Softwareebene zugeordnet. CSV-Dateien sind Textdateien, die in ein Tabellenkalkulationsprogramm importiert werden können, deren Spalten durch ein frei wählbares Trennzeichen getrennt sind. In den folgenden sechs Absätzen werden die Elemente der Feld- und Softwareebene erläutert.

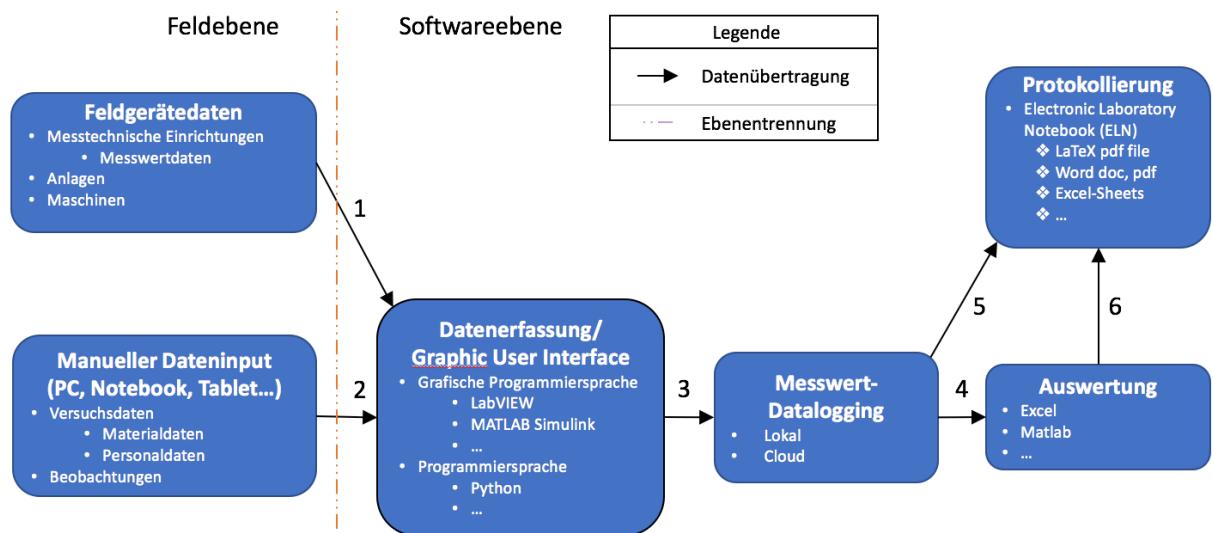


Abbildung 26: Digitalisierungsentwurf ohne Datenbank

Feldgerätedaten Die Feldgerätedaten, wie z.B. welche Anlage, Maschine, Messeinrichtung und deren Messwertdaten sind der Feldebene zugeordnet. Auch die Signale oder Daten, die von einem Gerät erfasst werden, sind der Feldgeräteebene zugeordnet.

Manueller Dateninput Dem manuellen Dateninput sind alle Daten zugeordnet, die dokumentiert werden müssen. Unter den zu dokumentierenden Daten können Versuchsparameter wie die Partikel-

dichte, die Schüttgutmasse etc. sein. Organisatorische Daten sind ebenfalls zu dokumentieren und können bspw. die Gruppen ID, Anwesenheitsinfomationen o.ä. sein.

Datenerfassung Für die Datenerfassung, auch Datenakquisition (DAQ, engl. *data acquisition*) bieten sich Lösungen mit einer grafischen Benutzeroberfläche (engl. *graphic user interface, GUI*) an. Die Datenerfassung kann mittels einer grafischen Programmiersprache wie z.B. LabVIEW realisiert werden oder mit einer textbasierten Programmiersprache wie bspw. Python [30]. Des Weiteren ist eine Datenerfassungssoftware der Gerätehersteller ebenfalls denkbar, (z.B. Laserbeugung-Partikelgrößenanalyse). Datenmanipulationen während der Datenerfassungen müssen dokumentiert und nachvollzogen werden können.

Messwert-Datalogging Die diskreten oder kontinuierlichen Daten von Messeinrichtungen können in eine Datei geschrieben werden. Prinzipiell sind zwei Datenformate möglich, das schreiben der Daten im ASCII-Format oder binär. Das entscheidende Auswahlkriterium ist der Speicherbedarf (vgl. Abbildung 70 im Anhang [15, S. 44]). Eine etablierte Dateiform für Messungsdaten ist die Speicherung in Text-Dateien, genannt CSV-Datei. CSV steht für comma separated value. Die Spalteneinträge sind durch ein Spaltentrennzeichen (engl. *delimiter*) getrennt. Als Spaltentrennzeichen kann neben dem Komma ein beliebiges Zeichen gewählt werden. Die Lösung des Dataloggings in CSV-Dateien hat einen geringen Speicherbedarf (z.B. im vgl. zu Excels *.xlsx-Dateien), da nur „Nutzdaten“ sowie delimiter, jedoch keine Formatierungsinformation oder der Gleichen vorhanden sind [34, S. 319].

Auswertung Die Auswertung kann je nach Vorgabe mit entsprechenden Softwarelösungen (Excel, Matlab, Python etc.) realisiert werden. Die Rohdaten (engl. *raw data*) aus den CSV-Dateien können bspw. in Excel importiert und weiter verarbeitet werden.

Protokollierung Das Digitalisierungskonzept welches ohne die Nutzung einer Datenbank auskommt, nutzt zur Datenarchivierung und Protokollierung ein sog. Elektronisches Labor Notebook (ELN), auch Labor Journal genannt. In ELN's können Eingabemasken für diskrete Daten (Versuchsparameter, organisatorische Informationen...) generiert werden. Des Weiteren lassen sich gängige Dateiformate per drag and drop zur Dokumentation in ein ELN importieren [28, S.44 f.].

Die durch die Sensoren erfassten Messwerte werden, z.B. über eine RS-232 Schnittstelle, an die Datenerfassungssoftware übertragen (1). Als Datenerfassungssoftware könnte beispielsweise LabVIEW

genutzt werden. Die Versuchssinformationen, Zeitreihendaten und Beobachtung während der Versuche können in der Datenerfassungssoftware aufgenommen werden (2). Die erfassten Daten und ggf. Beobachtungen, werden in eine Textdatei (CSV) geschrieben (3). Im englischen Sprachgebrauch wird ein Punkt als Dezimaltrennzeichen verwendet, woraus die Verwendung des Kommas als delimiter keinem Konflikt gegenübersteht. Im deutschsprachigem Raum ist das Komma jedoch nicht als delimiter zum empfehlen [34, Vorwort]. Als delimiter für die Spalten in der Textdatei bietet sich das [Tabualtor](#) Zeichen an. Es ist gefordert, dass die generierten CSV-Dateien schreibgeschützt sein sollen. Eine Speicherung der Rohdaten soll auf die HAW Cloud erfolgen. Die Rohdaten der Versuche können für die Auswertung bspw. in Excel importiert (4) und sollen dem Protokoll als Anlage angehängt werden (5). Alle zweckmäßigen Datenmanipulationen, die auf den Pfaden 3 bis 6 durchgeführt werden, müssen bekannt und nachvollziehbar dokumentiert werden. Zweckmäßige Datenmanipulationen könnten Glättungsoperationen der Daten, wie der gleitender Mittelwert, die Entfernung von Ausreißern, das Logarithmieren etc. sein.

3.2 Konzeptentwurf 4.0 (mit Datenbank und Softwarevernetzungsmöglichkeiten)

In der Abbildung 27 ist eine Erweiterung des zuvor dargestellten Konzepts dargestellt. Im Zentrum dieses Entwurfs steht die Verwendung einer [Datenbank](#). Der Softwareebene wurden bei diesem Konzept drei Elemente hinzugefügt. Nachfolgend werden die drei Elemente erläutert.

Datenbank Es könnte eine lokale oder eine cloudbasierte Datenbank zum Einsatz kommen. Auf dem Markt gibt es Dienstleister, die cloudbasierte Lösungen anbieten. Deren Geschäftsmodelle werden X-as-a-Service genannt. Das X dient als Platzhalter für Plattform, Database, Infrastructure, Software etc. [26]. Die Vorteile und Nachteile von Plattform- oder Database-as-a-Service sind in der Tabelle 5 aufgelistet [33, 16, 26], werden im Rahmen dieser Arbeit jedoch nicht näher erläutert.

Digital Twin Es existieren verschiedene Formen des sog. „digitalen Zwillings“ (*engl. Digital Twins*). Mittels *Digital Twins* können Produkte, Produktionsabläufe oder ein gesamter Prozess, durch die Kopp lung von Produktions- und Produkt Twin, abgebildet und somit optimiert werden. Mittels *Digital Twins* lassen sich sogenannte Soft-Sensoren integrieren (angewandte numerische Simulation). In Produktionsprozessabschnitten, in denen die Bedingungen es nicht erlauben einen realen Langzeitsensor zu implementieren, können, mittels *Digital Twins*, Sensoren in Simulationssoftware generiert werden, die mit

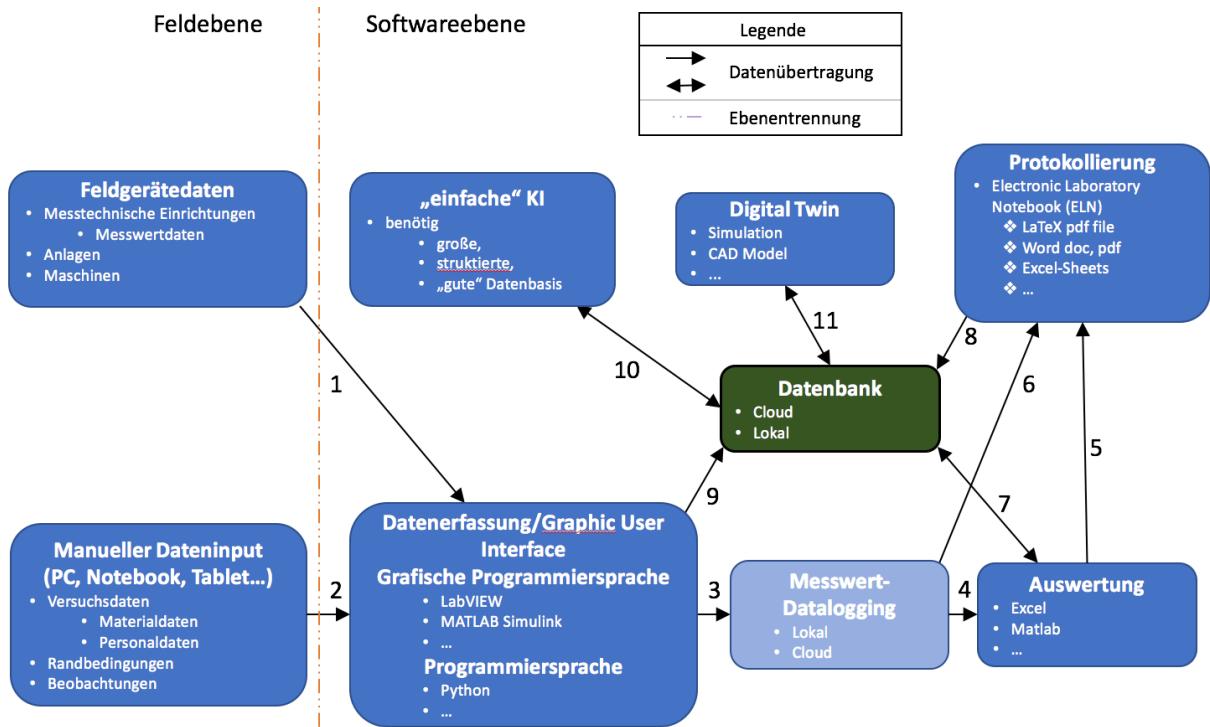


Abbildung 27: Digitalisierungsentwurf mit Datenbank

einem Prozess synchronisiert sind, um Totzeiten zu verringern oder zu eliminieren, bis schlechte Produktionsbedingungen detektiert werden können. Dadurch kann die Effizienz eines Prozesses signifikant gesteigert werden kann. Für die Simulation mittels Digital Twins werden Prozess-, Rohstoff- und geometrische Daten benötigt. Die Form der Randbedingungen können vielfältig sein und unterscheiden sich je nach Simulationsobjekt. Randbedingungen könnten Parameter der Umgebung betreffend, materialabhängige Parameter oder Parameter, die der Wertschöpfungskette zugehörig sind, sein. Als Beispiel für Parameter der Wertschöpfungskette kann ein tendenziell auftretender Rohstoffengpass durch Lieferverzug seitens des Zulieferers genannt werden [36]. Weitere Informationen ist weiterführenden Fachliteratur zu entnehmen.

Künstliche Intelligenz Ein Teilgebiet der Informatik ist das Themengebiet der „künstlichen Intelligenz“ (KI, engl. *artificial intelligence, AI*). Teilgebiete der Künstlichen Intelligenz sind das *Machine Learning* (ergänzung im Anhang), *Robotik*, sog. *Expertensysteme*, Mustererkennung, die Verarbeitung natür-

Tabelle 5: Vor- und Nachteile der Nutzung von Database- oder Plattform-as-a-Service, gemäß der Quellen [33, 16, 26]

Vorteile	Nachteile
+ Kostensparnis <ul style="list-style-type: none"> • Desaster Recovery Maßnahmen • Updates, Wartung, Instandhaltung • Infrastruktur • Software • Personalkosten 	- Abhängigkeit vom Dienstleister <ul style="list-style-type: none"> • Unzureichende Kapazitäten • Insolvenz • Kundenservice
+ Infrastruktur <ul style="list-style-type: none"> • Rechenleistung • Speicherkapazität • Netzwerkkapazität 	- Datenschutz <ul style="list-style-type: none"> • Servicestandort oft nicht in EU/Dt.
+ Anpassbar- und Skalierbarkeit	- Datensicherheit (Datenverlust)
+ Gut kalkulierbare Fixkosten	- eigene IT-Kompetenz
+ Sicherheit in Bezug.. <ul style="list-style-type: none"> • auf Mitarbeiter • Dezentralität • Schutz vor lokale Hackerangriffe • social hacking Prävention 	- Infrastruktur <ul style="list-style-type: none"> • Latenzzzeiten • lokale Ausfälle • lokale Strom- oder Netzwerk Ausfälle
+ Portfoliosoftware Kompatibilität	

licher Sprache oder maschinelles Übersetzen (deepl.com) [29].

Die potenziellen Fähigkeiten von KI Applikationen sind das Lernen, Planen und Problemlösen. Die Ansätze sind jedoch humanzentriert, zur Unterstützung der Menschen, bei ihren Tätigkeiten. Präzisiert bedeutet das [1, S. 5]:

- | | |
|---|---|
| <ul style="list-style-type: none"> • Wissenserwerb | <ul style="list-style-type: none"> • kognitives Erfassen und Automatisieren logischer Schlussfolgerungen |
| <ul style="list-style-type: none"> • Maschinenlernen | <ul style="list-style-type: none"> • Planung und Ausführung industrieller Automatisierungsprozesse |
| <ul style="list-style-type: none"> • Bild- und Spracherkennung | <ul style="list-style-type: none"> • Mustererkennung |

Der Entwurf aus dem vorherigen Abschnitt wird um die Verwendung einer Datenbank erweitert. Dadurch wird die Nutzung neuer Software für die Erschließung neuer Prozesse, Prozessoptimierung un-

d/oder Validierung ermöglicht. Die Feldgerätedaten werden, wie bei dem vorherigen Konzept, an eine Datenerfassungsoftware übertragen (1). Die diskreten und/oder kontinuierlichen Daten werden durch die Datenerfassungssoftware direkt in eine Datenbank geschrieben (9). Nach der Einrichtung einer Datenbank ist es theoretisch möglich auf das [Messwert-Datalogging](#) in Form von CSV-Dateien zu verzichten (3), jedoch könnte eine redundante Datensicherung sinnhaft sein. Durch die Verwendung einer Datenbank ergeben sich neue Datenübertragungswege (7 - 11). Eine Software Schnittstelle wird API (application programming interface) genannt. Sollte, auf Basis dieser Arbeit, eine Machbarkeitsstudie dieses Konzepts erfolgen, dann ist die API Kompatibilität jeglicher potenziell erwünschter Software zu prüfen. Durch die Verwendung einer Datenbank ist es möglich Daten strukturiert zu archivieren (7 - 11) und mittels sog. Abfragen (*Querys*) mit geringem Aufwand wieder abzurufen (7, 10, 11). Durch die Verwendung einer Datenbank kann die Nutzung von „einfachen“ KI's (10) und Digital Twins ermöglicht (11) sowie die Ergebnisse der Berechnungen in der Auswertung diskutiert werden (7).

Dem Professor für Umformtechnik sowie stellvertretenden Leiter des Maschinenbau und Produktions-technik Departments Dr. E. Stöver, bei einem zweistündigem offenem Dialog mit Prof. Dr. C Frank, Prof. Dr. K. Freudenthal, Prof. Dr. Geweke, Dipl.-Ing. M. Hannappel, Prof. E. Stöver, Dipl.-Ing. S. Wittkowski nach einem Vortrag von mir, D. Ludwig, Beginn 9 Uhr, zum Ziel der digitalen Transformation des verfahrenstechnischem Labors, mit dem Schwerpunkt dieser Masterthesis, am **26.08.2020**, von 9 bis 12 Uhr, entnehmen, dass die Vision des Konzepts dieser Masterthesis „in die richtige Richtung zeigt“. Prof. Dr. Enno Stöver ist dabei, die Idee des „Lernorts Digitale Umformtechnik“ umzusetzen. Dabei verfolgt er auch die Vision, dass Unternehmen und Studierende in eine Kooperation am Ort der Hochschule kommen. Dabei muss der Professor nicht der Initiator sein, sondern vielmehr der Ort als Treffpunkt die Kooperation auf niedrigem Level begünstigen. Die Vision von Prof. Dr. E. Stöver ist es, den „digitalen Lernort Umformtechnik“ an der HAW, am Berliner Tor zu etablieren. Lean Runden der Studierenden sind gelebter Alltag und die Vision ist folgende,

„Der „Lernort Digitale Umformtechnik“ stellt einen physischen Raum und eine virtuelle Plattform für die Lehrenden und die Studierenden der HAW Hamburg, sowie Unternehmen der Metropolregion Hamburg zum Themenbereich Industrie 4.0 und Digitalisierung im Bereich der Umformtechnik dar. Anwendungsbezogene Lösungen werden hier ausprobiert und stehen zum gemeinsamen kompetenzorientierten, forschungsbasierten und digital unterstütztem Lernen bereit. Neue Lösungen werden gemeinsam im Zusammenspiel von Praxis und Lehre entwickelt [44].“

Bei einer Begehung der mechanischen verfahrenstechnischen Einrichtung der TUHH mit Dr. S. Pietsch, am **14.09.2020**, ab **14** Uhr, hat sich herausgestellt, dass ein ganzheitliches Konzept, unter der Verwendung einer Datenbank, **im akademischen Rahmen** „innovativ sein könnte“. Die TUHH hat ein Digital Twin Konzept im Projektbacklog, welches innerhalb eines Zeitrahmens von ca. zwei Jahren umgesetzt, bzw. integriert werden soll. Die Implementation eines Datenbankmanagementsystems ist nicht ange- dacht. Eine Kooperation, mit gegenseitigem Nutzen, könnte möglich sein.

Im Verlauf einer ZOOM Präsentation meinerseits, am **26.10.2020** von **16:15** Uhr bis **17:45** Uhr, mit den Teilnehmern Prof. Dr. Geweke, Prof. Dr. Hölling, Prof. Dr. Sievers konnte ich von Prof. Sievers entnehmen, dass ein Vorteil des datenbankorientierten Konzepts, die vereinfachte Investigationsmöglichkeit bei vermeintlich konstanten Betriebsparametern, jedoch unterschiedlichen Ergebnissen, ist. Daraus ließen sich schneller studentische Projekte ableiten. Dem Professor der mech. Verfahrenstechnik M. Geweke, konnte ich im Verlauf dieses Meetings entnehmen, dass die Nutzung eines Datenbank orientierten Konzepts einen weiteren Nutzen hat. Die Covid-19 Situation hat Schwächen im konventionellen System aufgezeigt. Eine unstrukturierte, multimediale Archivierung hat einen großen Aufwand erzeugt, um alternative Lehrinhalte zum nicht durchführbaren Verfahrenstechnik sowie Lebensmitteltechnik- praktikum zu generieren.

S. Schwemm, B. Sc. Student der Verfahrenstechnik, hat mir in einem kurzem Interview seine Schwierigkeit mit dem daraus resultierenden, „alternativem Lehrinhalt“ nahegelegt. Die Hypothese die man daraus schlussfolgern könnte ist die folgende,

„Die Energie der Professoren, die in die Suche von geeigneten, aufeinander abgestimmten Daten dissipiert ist, hätte und kann in Zukunft vermieden werden. Die somit verfügbare Energie der Professoren, kann zum Optimieren der Lehrinhalte genutzt werden. Zu vermuten, dass der Worst Case eingetroffen und überstanden ist sowie ähnliche Situationen „unwahrscheinlich“ sind, wäre fahrlässig.“

3.3 Fazit

Das verfahrenstechnische Labor soll, zeitgemäß, digital transformiert werden. An der Stelle muss nochmals erwähnt werden, dass eine digitale Transformation ein Prozess und kein Projekt ist. Die beiden vorgestellten Konzepte schließen sich nicht gegenseitig aus. Das Konzept ohne Datenbank (DB) kann

als Vorstufe des Konzept mit einer DB betrachtet werden. Das Konzept 3.0 ist eine Mindestanforderung für die digitale Transformation des verfahrenstechnischem Labors. Ein Ziel dieses Projekts ist es, Messwerte digital verfügbar zu machen und diese zu dokumentieren. Ein weiteres Ziel des Projekts ist es einen Wandel von der Versuchsdokumentation in Hardcopy, zu einer digitalen Dokumentation einzuleiten. Es ist anzumerken, dass dieses Konzept die **absolute Mindestanforderung** für den ersten Schritt in die digitale Transformation darstellt.

Im folgenden werden die Vorteile des Konzepts ohne eine Datenbank aufgelistet:

- Die Verfügbarkeit von digitalen Messwerten wird ermöglicht,
- Die Dokumentation der Versuchsrandbedingung, eingesetzte Rohstoffe etc. kann digital durchgeführt werden,
- Die Verfügbarkeit der Versuchsdaten und Ergebnisse wird durch die Einführung eines elektronischen Labor Notebooks verbessert,
- Versuchsbeobachtungen können direkt dem Zeitstempel zugeordnet werden.

Eine Mindestanforderung sollte jedoch kein Maßstab für eine Hochschule darstellen! Meines Erachtens nach unterliegt der Aufwand dem Nutzen der Konzeptrealisierung unter der Verwendung einer Datenbank im hohen Maße, aus den folgenden Gründen.

Im akademischen Rahmen unterliegt der Aufwand, der Realisierung des zweiten Konzepts mit Datenbank, einen **großen** Mehrwert in Bezug auf das Verständnis und Lernerfolg der Studierenden. Die Datenbank könnte auch für den Laborbetrieb des Departments genutzt werden, wodurch sich möglicherweise organisatorische Vereinfachungen, wie z.B. Chemie- und Verfahrenstechniklagerverwaltung o.ä. realisieren lassen. Des Weiteren könnte die Datenbank für die verfahrenstechnische Forschung, Projekte im Rahmen der angewandten numerische Simulation uvm. genutzt werden.. Die **größte Hürde** für die Konzept 4.0 Realisierung könnte der **Changeprozess** sein.

In unserer agitierten Welt ist weder Platz für schwere Unternehmenstanker noch für träge Universitäten oder Hochschulen. In der Zeit der Globalisierung und digitalen Transformation ist Flexibilität und Vernetzung nicht nur eine hinreichende Forderung, sondern absolut notwendig. Unternehmen sollten sich Ihre Spitzenträger schaffen. Der Spruch, „von der Berufsausbildung oder dem Studium braucht man am Ende des Tages maximal 15 %“, ist nicht tolerierbar sowie zu verantworten und doch ist es aus eigenen Erfahrungen Realität. Wie lange Überlebt ein Unternehmen welches 85 % Ausschuss hat? Wenn durch

diese Art des Paradigmenwechsels eine Ausschussreduktion auf bspw. 70 % erfolgt, ist das ein immenser Fortschritt! In der Tabelle 6 sind die Vor- und Nachteile des Konzepts 4.0 aufgelistet.

Tabelle 6: Vor- und Nachteile des Konzeptentwurfs 4.0

Vorteile	Nachteile
<ul style="list-style-type: none"> + Stand der Technik in der Industrie / Zeitgemäß + Nutzbar für Tätigkeiten außerhalb vom VT1 und VT2 Praktikum (Forschung, Organisatorisches wie + Lagerverwaltung, angewandte numerische Simulation uvm.) + wenn Cloudservicenutzung, dann Portfolio-software Kompatibilität des Anbieters + Image/Professionalität (u.a. auch Anmerkungen von Bachelorabsolventen Ende 2020) + ggf. mehr Kooperationsprojekte mit Unternehmen und Universitäten + Kompetenzsteigerung der Hochschule und Absolventen + interdisziplinäre Vernetzung/Projekte (Industrie 4.0 ≡ Vernetzung) + gute Investigationsmöglichkeiten bei Produkt-abweichung bei konstanten Parametern → Studentenprojekte 	<ul style="list-style-type: none"> - Changeprozesse - notwendiges Know-How erforderlich - Aufwand der Erstellung der Datenbank - Einrichtung automatischer Einspeisung der Sensordaten in die Datenbank - Kosten könnten entstehen <ul style="list-style-type: none"> • möglicherweise Sponsoring/Kooperation (als Gegenleistung Projekte o.ä) da Hochschule • geringe Datenmengen im vgl. zu Unternehmen → möglicherweise keine oder geringe Kosten

3.4 Strukturkonzept einer relationalen Datenbank

Im Rahmen dieser Projektarbeit wurde eine Datenstruktur für eine relationale Datenbank entworfen. Das Konzept wurde der ISA-S95 Struktur nachempfunden. ISA-S95 gliedert Unternehmen hierarchisch auf. Eine Datenstruktur gibt ISA-S95 jedoch nicht vor. *Abteilungen* und *Rubriken* werden demnach hierarchisch aufgegliedert. Eine Rubrik könnte bspw. alle Informationen (100_Informations) sein, dazu später mehr. Die Unterteilung erfolgt demnach von einem *Wurzelement* (engl. *rootelement*) in *Abzweigungen* (engl. *branchelements*) und *Dokumentreferenzen* (engl. *sheetelements*; siehe Abbildung 28). An der Bezeichnung *Dokumentreferenzen* lässt sich erkennen, dass die strukturvorgabe Dokumentenbasiert erfolgt. Eine präzisierung erfolgt an anderer Stelle.

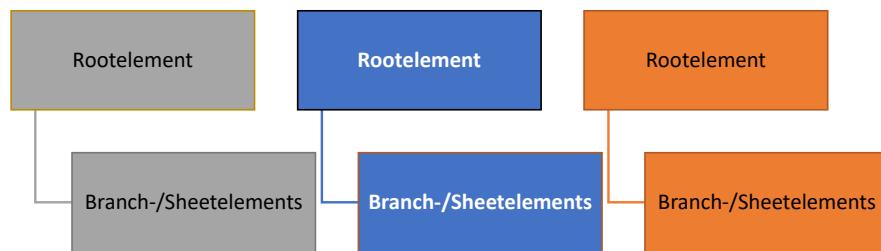


Abbildung 28: Root-/Branchelements, gemäß ISA-S95

Anmerkung: Die Grafiken dieses Abschnitts und denen im Anhang erheben keinen Anspruch auf Vollständigkeit und Konsistenz. Die Erstellung der Grafiken wurden manuell in PowerPoint und keinem professionellem Tool erstellt, Tabellen und Bezeichnungen haben demnach keine Referenzfunktion. Zur Darlegung der konzeptionellen Idee sollte die Güte der Qualität hinreichend sein.
Für die Verfahrenstechnik wurde im Rahmen dieses Projekts eine hierarchische Gliederung durchgeführt (siehe Abbildung 29). Es wurden drei *rootelements* definiert. Die unterschiedlichen Farben dienen im Verlauf dieses Abschnitts der visuellen Differenzierung der *Rubriken*.

- 100_Informations
- 200_ProcessEngineering
- 300_Trials

Dem *rootelement* 100_Informations sind alle *branch-* und *sheetelements* untergeordnet, die allgemeine Informationen enthalten. Analog ist das für die definierten *rootelemente* 200_ProcessEngineering und

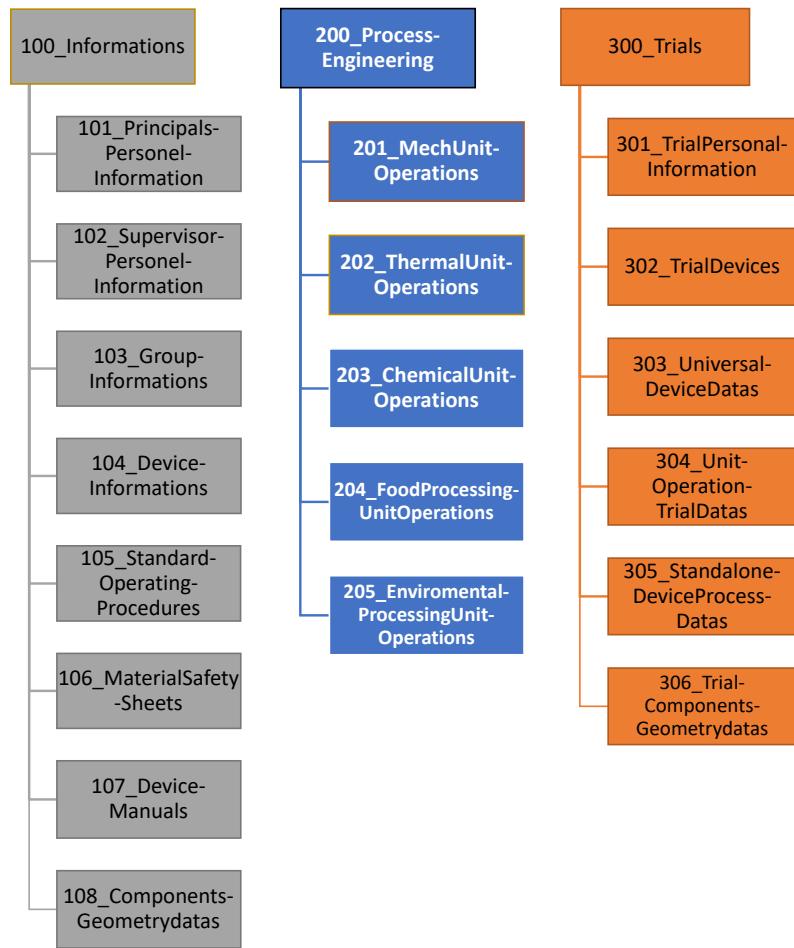


Abbildung 29: Labordatenstrukturdarstellung, angelehnt an ISA-S95

300_Trials geschehen. In den sheetelements, die dem rootelement 300_Trials untergeordnet sind, sind alle diskreten und kontinuierlichen Versuchsdaten und Parameter; wie bspw. Schüttgutspezifikationen, Druck und Volumenstrom sowie deren Kommentare am jeweiligen Zeitstempel, Geräte ID, etc.; hinterlegt. Allgemeine Informationen sind der Abbildung 29 zu entnehmen. Von dieser Grafik ist eine tabellarische Auflistung im Anhang vorhanden (siehe Tabelle 17).

An dieser Stelle wird exemplarisch die visualisierte *Rubrik* 100_Informationen erläutert (siehe Abbildung 30). 100_Informationen könnte in allgemeine Informationen (*engl. general informations*) und Laborinformationen (*engl. laboratory informations*) unterteilt werden. Zu den *GeneralInformations* gehören unter anderem personelle Informationen (101 - 103). Weitere Unterrubriken wurden nicht weiter

aufgelistet. *LaboratoryInformations* sind Informationen wie Versuchsanweisungen (105), Sicherheitsdatenblätter (106) etc. untergeordnet.

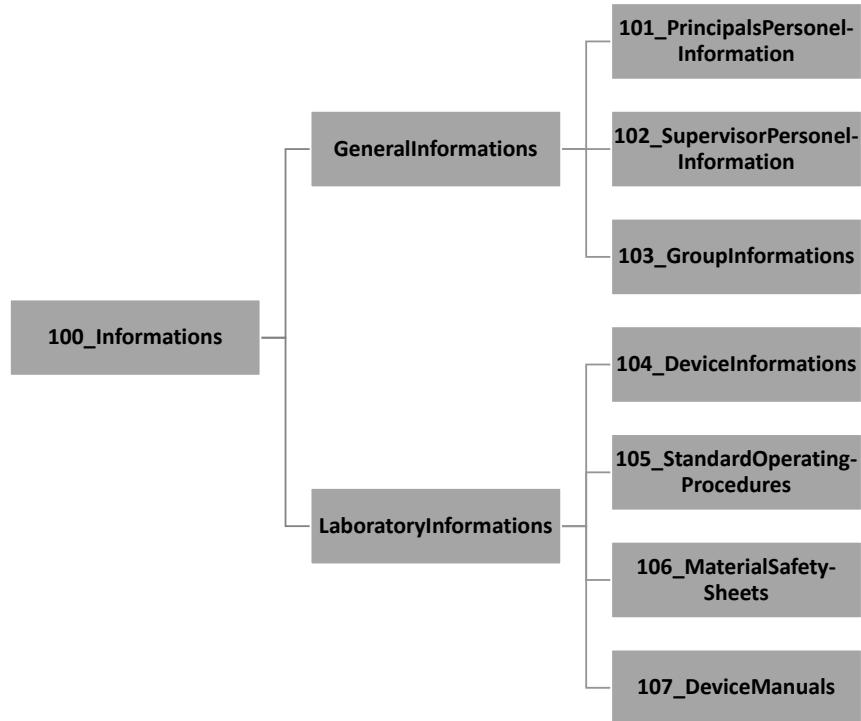


Abbildung 30: 100_Informations, angelehnt an ISA-S95

Den Abbildungen 71 und 72 im Anhang kann die hierarchische Struktur der Rubriken 200_ProcessEngineering und 300_Trials entnommen werden.

Das *World Batch Forum* hat eine Datenstruktur, mit dem Bezeichnung Business To Manufacturing Markup Language (B2MML) entwickelt, auf der Basis von XML. XML-Dateien sind hierarchisch aufgegliedert. Jede XML- bzw. XSD-Datei (Dokument) besteht aus einem Wurzelement (*engl. rootelement*). An dieser Stelle ist anzumerken, dass XML-Datenbanken dokumentbasiert funktionieren und hierarchische Strukturen aufweisen [48]. Ein B2MML und BatchML Dokument (auf der Basis von XML) einer *Rubrik* enthält demnach alle Informationen für eine bestimmte *Rubrik*. B2MML bildet Geschäftsprozesse ab. Mit den B2MML und BatchML Vorlagen sind Unternehmen in der Lage ein individuelles Manufacturing Execution System (MES) zu erstellen.

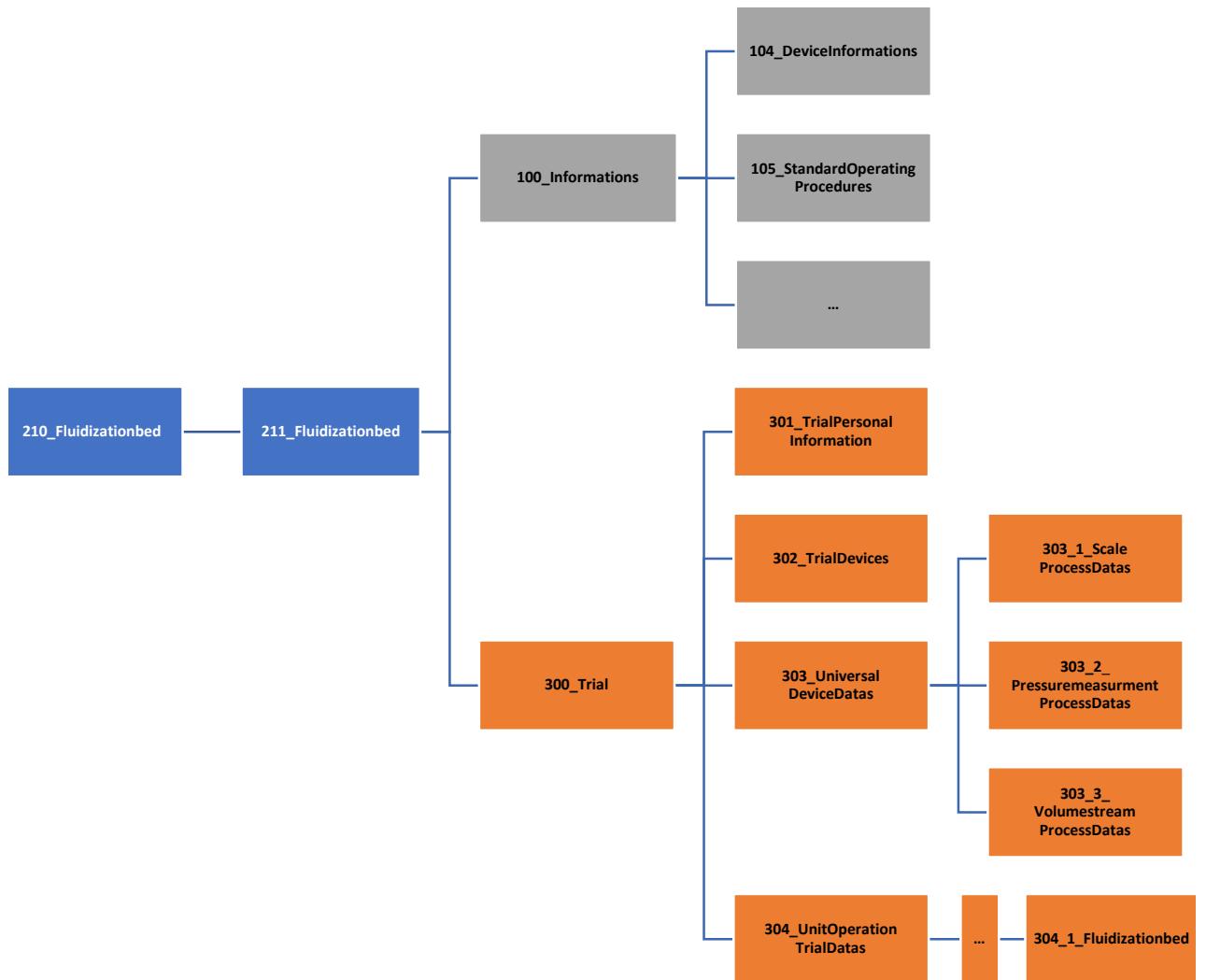


Abbildung 31: Potentielle hierarchische Datenstruktur der Wirbelschicht (*engl. fluidization bed*) Unit Operation, angelehnt an ISA-S95 und B2MML

Wird das hierarchische Konzept auf das verfahrenstechnische Labor transferiert, dann könnte eine *Rubrik* die Unit Operation Wirbelschicht (*engl. fluidization bed*) sein. Die Struktur eines B2MML-Dokuments, zur Beschreibung eines **Wirbelschichtversuchsdurchlaufs**, könnte eine ähnliche Datenstruktur wie in Abbildung 31 dargestellt aufweisen. Daraus folgt, dass jeder Versuchstag bzw. Versuchsdurchlauf ein eigenes B2MML- bzw. XML-Dokument erhält. Gemäß Abbildung 31 würde die Unit Operation Wir-

belschicht (210) weitere *branchelements* aufweisen, wie bspw. Wirbelschichttrocknen, Wirbelschichttagglomeration etc.. Das Labor der mechanischen Verfahrenstechnik der HAW Life Sciences hat nur eine Variation der Wirbelschicht, daher ist die erste und einzige Abzweigung (210). An dieser Stelle kommen nun alle Elemente der obersten Hierarchiestufe (100, 300). *Branchelemente* der Rubrik (100) sollten jedoch nur Dokumentreferenzen sein, denn eine Redundanz dieser Daten hätte keinen Nutzen, sondern nur Aufwand zur Folge. Die nächste Hierarchieebene differenziert sich weiter auf und wird spezifischer. Auf der untersten eben wird es explizit. Im Rahmen dieses Beispiels würde die unterste Hierarchieebene die folgenden Elemente labeln:

- 301 Personenbezogenedaten des Versuchstags
- 302 Daten der notwendigen Geräte
- 303 Erfasst alle zu erfassenen Daten. Darunter gehören diskrete sowie kontinuierliche Daten.
 - 303_1 Wiegedaten
 - 303_2 Druckmessungsdaten
 - 303_3 Volumenstommessungsdaten
- 304 Hier werden die prozessspezifischen Parameter dokumentiert
 - 304_1 In diesem Fall wären es die prozessspezifischen Parameter des Wirbelschichtversuchs an dem jeweiligen Versuchsstag

Die Operation mit relationalen Datenbanken, auch SQL DB genannt, gestaltet sich in vielen Anwendungsfällen einfacher. Des Weiteren sind SQL DB am weitesten verbreitet und Programmschnittstellen sind in vielen Fällen für relationale Datenbanken (DB) ausgelegt. Transferiert man das hierarchische Konzept von ISA-S95 und B2MML auf eine SQL DB, dann bildet jedes einzelne Element eine eigenständige Tabelle ab. Eine dokumentenbasierte Datenbank würde beispielsweise eine ganze Unit Operation in einem Dokument zusammenfassen (vgl. Abbildung 31). In einer SQL DB beschreiben demnach viele Tabellen eine Unit Operations bzw. einen Versuchsdurchlauf. Folglich ist der Aufwand der Tabellenerstellung möglicherweise höher, jedoch ist die Erstellung der abfragen via SQL einfach zu realisieren. Daher wird das ISA-S95, B2MML Konzept auf eine relationale Struktur transferiert (vgl. 32). Die Bezeichnungen in diesem Konzeptabschnitt wurden ins englische überführt.

In der Abbildung 32 ist eine potentielle Datenstruktur für eine SQL Datenbank abgebildet. Es ist zu erkennen, dass Tabellen der unterschiedlichsten Rubriken; 100_Informations, 200_ProcessEngineering,

300_Trials; in einer 1:N Beziehung zu einander stehen. Diese Struktur könnte den Wirbelschichtversuch hinreichend abbilden, denn:

100_ Informations stehen in relation mit der Unit Operation 211_Fluidizationbed

- ◊ Personelle Informationen werden erfasst.
 - 101 Auftraggeber (Professoren und Dozenten)
 - 102 Wissenschaftliche Mitarbeiter
 - 103 Gruppeninformationen
- ◊ Allgemeine Informationen werden erfasst. Dateipfadreferenzen können hinterlegt werden. An dieser Stelle könnte eine dokumentenbasierte Datenbank Ihre Anwendung finden und die Archivierungsaufgabe besser lösen als eine SQL DB
 - 104 Geräteinformationen könnten sich auch Dokumentenbasiert abbilden lassen
 - 105 Versuchsanweisung (Standard Operating Procedures)
 - 106 Materialdatenblätter
 - 107 Betriebsanweisungen von Geräten und Anlagen
 - 108 Geometrische Daten von Versuchsanlagen und Komponenten könnten Dokumentenbasiert möglicherweise effizienter archiviert werden. Eine Möglichkeit via SQL ist es für jedes Komponent eine Tabelle zu generieren. Dokumentenbasiert würde ein Bauteil ein Dokument erhalten und alle Komponenten enthalten.

300_ Trials werden via diverser Tabellen erfasst

300 könnte organisatorische Daten die versuchstagbezogen sind, wie bspw. das Datum oder der wievielte Versuch (sollte eine Gruppe mal gescheitert sein), aber auch das Hochladedatum sowie eine Speicherortreferenz erfassen. Für die Speicherung der Protokolle könnte ebenfalls ein dokumentenbasierte DB Lösung zum Einsatz kommen.

301 Personenbezogene Informationen pro Versuchstag werden erfasst

302 könnte bei redundanten Apparaten, Geräten, Aggregaten relevant sein

303 Erfasst alle zu erfassenen Daten. Darunter gehören diskrete so wie kontinuierliche Daten.

303_1 Wiegendaten

303_2 Druckmessungsdaten

303_3 Volumenstommessungsdaten

- An der Stelle können diverser Kommentarfelder eingefügt werden, die einen Versuch beschreiben wie Subjektive Beobachtungen, die Wirbelschichthöhte und die Wirbelschichtporösität.

304_1 Hier werden die prozessspezifischen Parameter dokumentiert

Im Anhang befindet sich eine Grafik, die verschiedene Abstraktionsebenen vermischt darstellt (siehe Abbildung 74). Auf der untersten Abstraktionsebene wären die expliziten Tabellen aufzufinden. Eine beispielhafte Zusammenstellung ist dem Anhang, der Abbildung 75 zu entnehmen.

3.4.1 Fazit

Im Rahmen der Anwendungsfälle im Hochschulalltag könnten SQL Datenbanken den Anforderungen genügen. Das Know-How mit dieser standardisierten Datenbank Technologie ist auf andere Softwaretechnologien transferierbar. Es sollte auch möglich sein, die Erstellung einer Datenbank, im Rahmen studentischer Arbeiten, zu delegieren. Des Weiteren ist auch im Verlauf des *Digitalisierungsprozess* eine Hybridlösung denkbar.

3 Konzeptentwicklung für die Labordigitalisierung

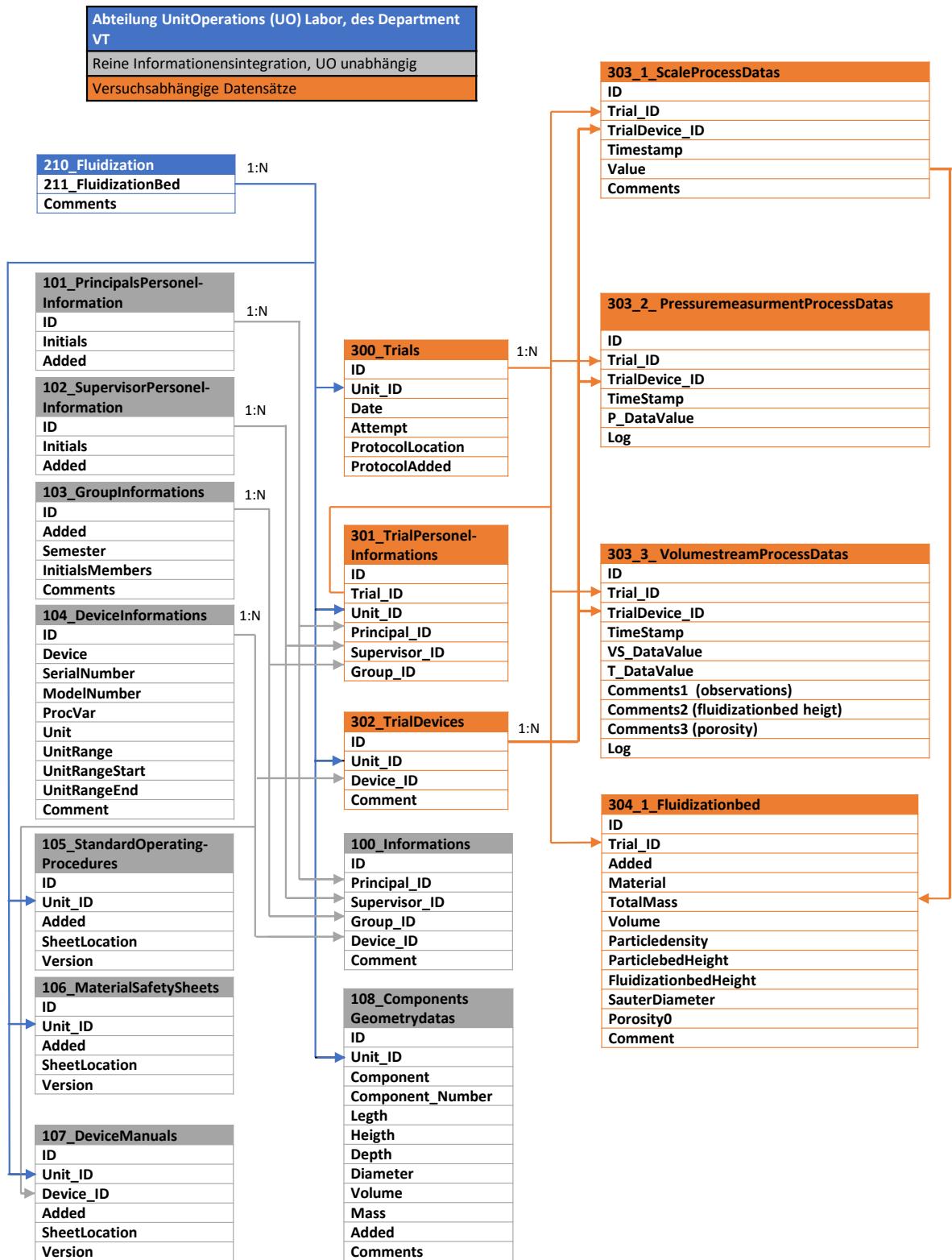


Abbildung 32: Datenstrukturentwurf einer relationalen Datenbank zur Beschreibung der Wirbelschicht Unit Operation, angelehnt an ISA-S95 und B2MML

4 Signal- und Datenverarbeitung von Messeinrichtungen

Datenerfassung oder auch Datenakquisition (*engl. data acquisition*) wird häufig DAQ abgekürzt. In den folgenden Abschnitten wird die DAQ mittels

integrierter Messdatenerfassung via **RS-232** Schnittstelle,

analog Signalumwandlung mittels **NI-USB 6001** Messkarte via **USB** Schnittstelle

unter der Verwendung von LabVIEW (2018, 2020 und 2021) erläutert. Für die Umsetzung wurden Anforderungen definiert, die wie folgt aufgelistet sind:

- Die Applikationen sollen eine graphische Oberfläche haben.
- Der Aufwand für die Applikationsgenerierung soll minimal sein.
- Die Applikationen sollen nur von ausgewählten Personal („leicht“) anpassbar sein.
- Die verwendete Software soll in der Lage sein, Daten der verschiedensten Versuchsstände und Geräte unterschiedlicher Hersteller abzubilden.
- Es sollen Echtzeitgraphen generiert werden können.
- Die verwendete Applikation soll kontinuierliche und diskrete Daten verarbeiten können.
- Die Möglichkeit, die Daten in Zukunft direkt in eine Datenbank schreiben zu können, ist wünschenswert.

Es werden Module programmiert, mit denen kontinuierliche und diskrete Signalströme und Daten verarbeiten werden können. Somit ist es möglich, für alle Unit Operations Programme mittels LabVIEW zu schreiben. Diese diskreten Daten können in einer separaten Datei oder als Header einer z.B. Tabulator getrennten „kont iMessungs.txt“ Datei gespeichert werden. Man ist nicht daran gebunden als Spaltentrennzeichen (*engl. delimiter*) Tabulator zu verwenden, auch die Verwendung eines Komma (*.csv Datei), Semikolon oder auch ein frei wählbares Trennzeichen ist möglich. Aufgrund der internationalen Konventionsunterschiede in Bezug auf das Dezimaltrennzeichen (ASCII ‘.’, in Deutschland ‘,’ etc.) ist von der Verwendung eines Kommas jedoch abzuraten.

RS-232 zu USB Adapter Wenn ein PC keine RS-232 Schnittstelle besitzt, kann ein RS-232 zu USB Adapter verwendet werden. Ein Mikroprozessor im Kabel emuliert eine RS-232 Schnittstelle.

4.1 Serielle Schnittstellenkonfiguration und Signalverarbeitung von Messeinrichtungen

In diesem Abschnitt werden virtuelle Instrumente (VI) zur Verarbeitung kontinuierlicher und diskreter Daten im ASCII Format (ASCII encoded) programmiert, die über eine RS-232 Schnittstelle empfangen werden und die Daten verarbeitet. In der Abbildung 33 ist zu erkennen, wie die hardwareseitige Datenverarbeitung im Falle eines Messgeräts mit integrierter digitaler Schnittstelle aussieht. Das Messobjekt, dass könnte bspw.. die gravimetrische Bestimmung eines Syntheseprodukts sein, wird messtechnisch mittels Sensor erfasst und transmittiert, dass im Messgerät bereits umgewandelte Signal, an einen Microcontroller (Arduino, RasPi ...), PC oder ähnliches.



Abbildung 33: DAQ von Messeinrichtungen mit integrierter digitaler Schnittstelle

Es werden die Schritte erklärt, wie man mit LabVIEW einen kontinuierlichen Datenstrom verarbeitet, der über eine RS-232 Schnittstelle empfangen wird. Es wird ein Windows 10 PC sowie ein Mac Book Pro und LabVIEW 2018 sowie LabVIEW 2020 verwendet, um die VI's zu programmieren und die Signale eines Messinstruments zu verarbeiten. Als Demonstrationsmesssystem wird eine Digitalwaage des Modells 440-47N des Herstellers KERN ausgelesen und die Daten interpretiert, die in eine Tabulator getrennten Text-Datei (*.txt) geschrieben werden. Die Waage hat einen integrierten analog/digital Wandler und besitzt eine RS-232 Schnittstelle. Das Programm wird aus mehreren Sub VI's und dem Haupt VI bestehen.

In einem VI wird der Datenstrom, der von der Waage übermittelt wird, akquiriert. Das Sub VI wird folgend **Serial Sub VI** oder in der Kurzform **SSVI** genannt. Das **Haupt VI** wird die Daten, die es vom Sub VI erhält verarbeiten/interpretieren und in eine *.txt Datei loggen. Des Weiteren soll das Haupt

VI einen live plotter besitzen. Für einen modularen Header werden Sub VI's geschrieben, welcher Daten wie Gruppennummer, lokaler (des jeweiligen PC's) Datum- und Zeitstempel (modifizierbar), die VT Praktikums Nr. o.ä. enthält.

Um die korrekte Konfiguration Ihres Geräts vorzunehmen zu können, ist in dem Handbuch des Geräts nach folgenden Parametern zu schauen:

- Zeichen Codierung, z.B. 7-Bit 'ASCII'
- Baudrate
- Paritätsbit
- Stopbit
- Flow Control (*engl. dt. Flusssteuerung*)

Die Digitalwaage sendet ihre digitalen Signale über eine RS-232 Schnittstelle, die 8-Bit ASCII-encodiert sind. Um einen **kontinuierlichen Datenstrom** zu erhalten und somit **kontinuierlich** zu wiegen, wird in der Einstellung der Waage „automatisches Senden“ eingestellt (das Gewicht soll gesendet werden, auch wenn der Wert nicht stabil ist). Die seriellen Konfigurationsparameter der Waage und des PC's bzw. des verarbeitenden Programms wie z.B. LabVIEW müssen identisch sein, um decodierbare Daten empfangen zu können. Diese Einstellungen sind über das Menü der Waage vorzunehmen. In der Tabelle 7 werden die verwendeten seriellen Konfigurationsparameter der Waage 440-47N des Herstellers Kern aufgelistet (*engl. shorthand notation*)).

Tabelle 7: RS-232 Konfigurationsparameter der Digitalwaage Kern 440-47 N in shorthand notation 8-N-1

Baudrate in s ⁻¹	1200
Bytelänge in Bit	8
Paritätsbit	None (N)
Stopbit	1
Encoded	ASCII
Flow-Control	Software-Handshake

Anmerkung: Für die Programmierung wurde eine Digitalwaage des Herstellers Kern des Typs 440-47 N gewählt. Das Programm wird auf andere Geräte übertragbar sein. Die einzige Anforderung ist, dass die RS-232 Parameter aufeinander abgestimmt werden.

4.2 Serielle Schnittstelle im Betriebssystem finden und Debuggen

Die für den Benutzer sichtbare Benennung des RS-232 Ports unterscheidet sich zwischen den Betriebssystemen. Im Verlauf des Projekts wurde mit einem Unix (Mac Book Pro) und Windows 10 Betriebssystem gearbeitet.

Microsoft Windows Bei der Verwendung eines Windows PC's wird der Port der seriellen Schnittstelle Com Port genannt und wird im Gerätewizard und in Programmen, wie z.B. in LabVIEW, COM 1-N aufgelistet.

Mac Unix Bei der Verwendung eines MACOS/Unix ist die serielle Schnittstelle im Terminal des Mac unter dev/tty.* und dev/cu.* zu finden und wird in LabVIEW ASLR 1-N : : INSTR genannt.

Um die richtigen Parameter zu identifizieren oder ob generell Signale, und wenn in welcher Form, empfangen werden kann mit einigen Tools ermittelt werden.

Tool zur Identifikation und Debuggen serieller Schnittstellen Um herauszufinden ob die Konfigurationsparameter stimmen; ein Gerät Daten sendet, wie das Datenformat aussieht und ob die Daten korrekt übermittelt werden; kann man unter Windows das kostenlose Tool namens hTerm verwenden. Ein etwas weniger komfortables, jedoch vergleichbares open source Tool unter Mac oder Linux ist CoolTerm.

4.3 Kontinuierliche Daten via RS-232 und LabVIEW verarbeiten

Die grafische Programmiersprache G mit der Entwicklungsumgebung LabVIEW ist in der Lage Daten via RS-232 Schnittstelle zu akquirieren und zu interpretieren. Um eine serielle Schnittstelle in LabVIEW auslesen zu können, wird der NI-VISA Treiber benötigt.

Anmerkung: Es werden unter MACOS einige Funktionen nicht unterstützt, wie unter anderem der LabVIEW Paketmanager und einige Pakete wie bspw. DAQmx für den Mac zu erhalten und zu implementieren hat „Potential“.

4.3.1 Anpassung der LabVIEW-Einstellungen

Damit Texteingaben in Eingabefenstern (*Control Fenster*) mit der Betätigung von Enter quittiert werden, ist in den LabVIEW Einstellungen folgender Haken zu setzen:

Tools > Options ... > Category > Environment > End text entry with Enter key [aktivieren]

Wenn ein serieller Port geöffnet wird ist es wichtig, dass bei dem Beenden des Programms der serielle Port geschlossen wird! Wenn der Abbruch Button (*engl. abort button*) von LabVIEW genutzt wird, wird der Port nicht geschlossen, daher soll das VI nur über den *Stop Button* des Frontpanels geschlossen werden. Um den *Abort Button* für das gerade verwendete VI (Programm) zu entfernen, gehen Sie bitte wie folgt vor:

File > VI properties > Window Appearance > Customize ... > Show Abort button [deaktivieren]

Tabelle 8: Benötigte LabVIEW Konfigurationsfunktionen Teil 1

Nr.	Funktion	Lokalisation der Funktion in LabVIEW
1	For Loop	BD/Functions/Programming/Structures
2	Case Structures	BD/Functions/Programming/Structures
3	First Call?	BD/Functions/Programming/Synchronization
4	Configure Port	BD/Functions/Instruments I/O/Serial
5	Bytes at port	BD/Functions/Instruments I/O/Serial
6	Read	BD/Functions/Instruments I/O/Serial
7	Write	BD/Functions/Instruments I/O/Serial
8	Close	BD/Functions/Instruments I/O/Serial
9	Flush Buffer	BD/Functions/Instruments I/O/Serial
10	String Konstante	BD/rM auf Funktionseingang/Create Constant
11	Numerische Konstante	BD/rM auf Funktionseingang/Create Constant
12	String Indicator	FP/Modern/String & Path
13	VISA ressource	FP/Modern/I/O
14	Stop Button	FP/Boolean
15	VI Object I/O Definition	obere rechte Ecke des LabVIEW Fensters
16	Serial Sub VI	drag and drop (siehe Abbildung 37)
17	Concatenate Strings	BD/Functions/Programming/Strings
18	Iterationszähler	siehe Abbildung 38
19	Shift Register	siehe Abbildung 38
20	While Loop Abbruchbedingung	siehe Abbildung 38
21	String Indicator/Anzeige	FP/Modern/String & Path/String Indicator oder rM auf Datenquelle/Create Indicator
22	String Control/Eingabe	FP/Modern/String & Path/String Indicator oder rM auf Datenquelle/Create Control
23	Search/Split String	BD/Functions/Programming/String/Additional String Functions
24	Case Selektor	siehe Abbildung 41
25	String Subset	BD/Functions/Programming/String
26	Equal?	BD/Function/Programming/Comparison
27	Not	BD/Function/Programming/boolean
28	Regular Expression Match	BD/Functions/Programming/String
29	Local variable	BD/Functions/Structures

4.3.2 Serielle Sub VI programmierung

Zu Beginn wird ein VI oder wie auch nachfolgend *Objekt* genannt programmiert (in LabVIEW auf englisch *Function*). Dieses VI wird im folgenden Abschnitt in das Haupt VI als Sub VI mit dem Namen *Serial Sub VI* integriert. Zu Beginn öffnet man ein neues leeres VI. Das *Serial Sub VI*, im folgenden *SSVI*, wird für die Konfiguration einer seriellen Schnittstelle verwendet. Die von LabVIEW verwendeten *Objekte* können Sie den Tabellen 8 und 9 entnehmen. In den Tabellen 8 und 9 (die sich ebenfalls im Anhang befinden), sind alle Objekte aufgelistet die benötigt werden, um die *VI's* zu programmieren. In den Tabellen 8 und 9 wird das Blockdiagramm mit **BD**, das Front Panel mit **FP** und die rechte Maustaste mit **rM** abgekürzt. In den folgenden Programmierabbildungen sind die Funktionen wie in den Tabellen 8 und 9 nummeriert.

Erster Schritt: Serielle Objekte Nach dem Erstellen eines neuen VI's fügt man alle Objekte ein, die man für den Datentransfer mit serieller Schnittstelle benötigt (4 - 9 in Tabelle 8) (siehe Abbildung 34).

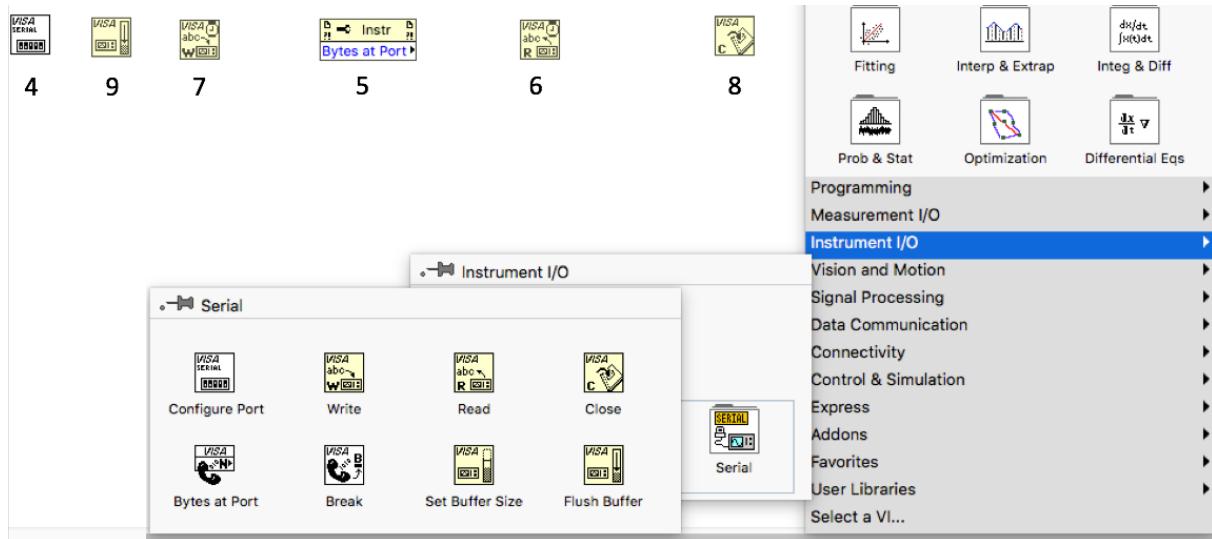


Abbildung 34: Einfügen aller seriellen Funktionen

Zweiter Schritt: Port Initialisierungs- und Abbruchskonfiguration An dem Objekt *Configure Port* sind Konstanten mit den Konfigurationswerten hinzuzufügen. An den Objekteingängen auf der

linken Seite des Objekts, erstellt man Konstanten (rechte Maustaste/ *Create Constant* (11), siehe Abbildung 35a), die die seriellen Konfigurationsparameter enthalten (siehe Tabelle 7). Um den Anwender dieses VI's den gesamten Kombinationsraum der Parameter zu Verfügung zu stellen, ist es auch möglich anstelle der Konstanten, *Controls* zu verwenden. Anschließend zieht man eine *Case-Structur* (2a) um das *Configure Port* (zzgl. Konstanten) sowie das *Flush Buffer* (9) Objekt und eine weitere *Case-Structur* (2b) um das *Close Port* Objekt. Wird die *Case-Structure* (29) mittels eines Wahrheitswerts (*boolean variable*, *true/ false*) getriggert, wird somit der Port konfiguriert und der I/O Buffer *flushed*. Die *Case Structure* (2b), auf der rechten Seite wird mittels eines *Stop Button*s auf *true* gestellt, um den Port mittels des *Close* (8) Objekts zu schließen.

Den *Stop Button* (14), die *VISA ressource name* (13) zur Auswahl des Ports und den *String Indicator* (12) (siehe Tabelle 8, 12 - 14) fügt man z.B. über das Front Panel hinzu und positioniert die dazugehörigen Objekte, die automatisch im Blockdiagramm erscheinen, an der richtigen Positionen. Der *Stop Button* wird mit dem *Case Selektor* der zweiten *Case-Structure* (2b) verbunden. Wird das Programm durch diesen *Stop Button* beendet, wird somit der Port geschlossen. **Aufgrund dessen ist es wichtig, den VI Abbruch Button zu Deaktivieren:** *File > VI properties... > Window Apperance > Customize ... > [Markierung entfernen] Show Abort button*. Ein *Indicator* zum Anzeigen und zum weiterreichen des, durch das *Read* Objekt (6) ausgelesenen Strings, wird an das *Read* Objekt (6) angeschlossen. Die erste *Case Structure* (2a) wird durch ein *First Call?* (3) Objekt getriggert.

Dritter Schritt: N = 1 For Loop, XON flow control Nun zieht man eine *For Loop* um alle Objekte, bis auf die *VISA Resource* und das *Flush Buffer* Objekt. Die *For Loop* wird in diesem Fall mit der Konstante 1 initialisiert (siehe Abbildung 35b, oben links). Nach dem hinzufügen von zwei Schieberegistern (*engl. Shift Register*), sind alle Objekte wie in Abbildung 35b zu verdrahten. Durch das Hinzufügen eines *Shift Registers* an einem Ende einer *For Loop* oder *While Loop*, werden automatisch die *Shift Register* auf der anderen Seite des Loops generiert. Alle Funktionen müssen nun, wie in Abbildung 35b, verbunden werden (die blaue Leitung zwischen *Bytes at Port* und *Read Port* ist ebenfalls zu verbinden). Da die „Verkabelung“ im *true Case* beider *Case Structures* verkabelt sind, muss selbiges noch im *false Case* beider *Case Structures* (2ab) geschehen (siehe Abbildung 36a). Das *Write* Objekt (7) ist mit der String Konstante (10) mit dem Wert 11 zu verknüpfen. Die String Konstante 11 beinhaltet das ASCII flow control (siehe Abschnitt RS-232) Signal XON und

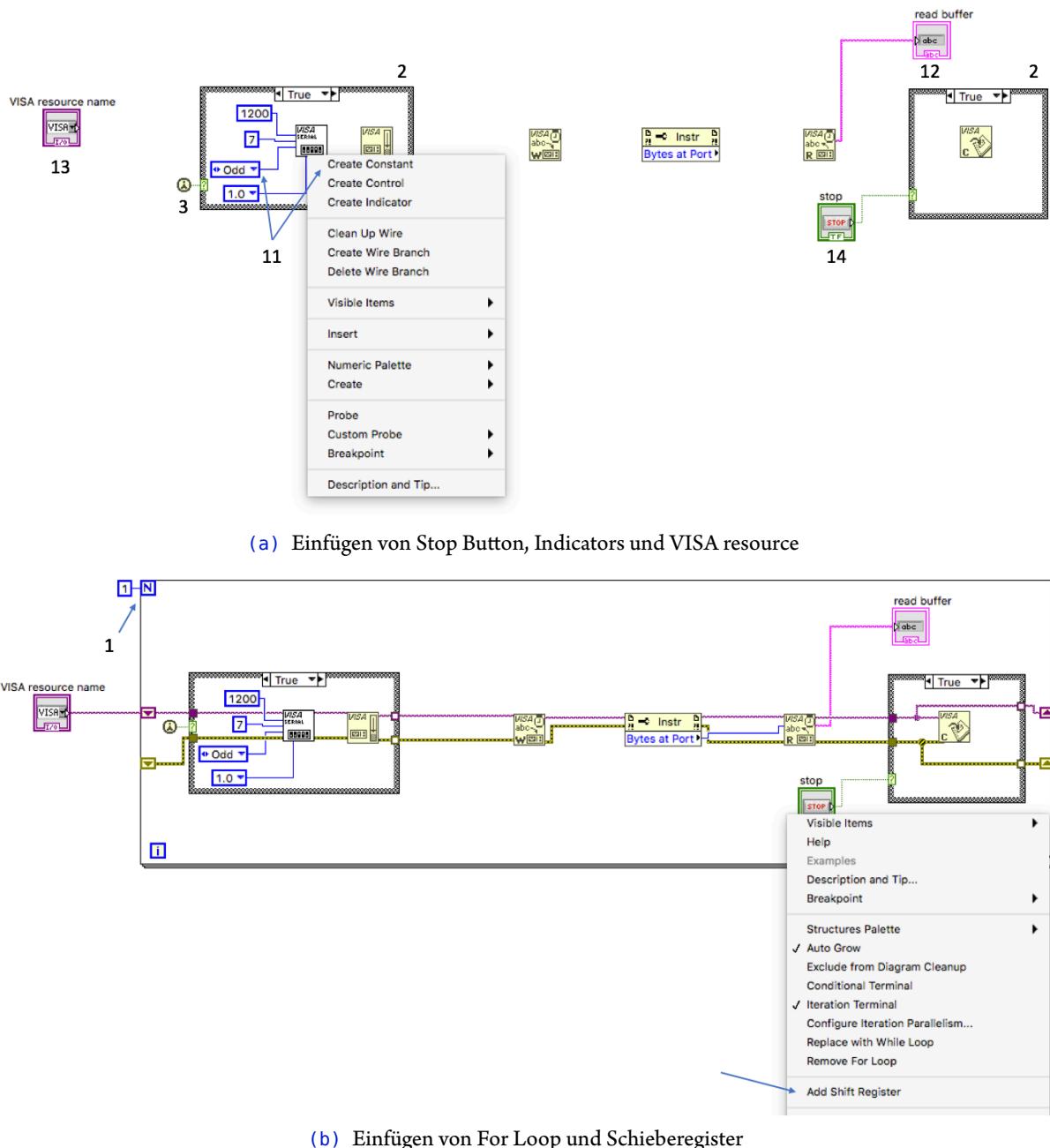
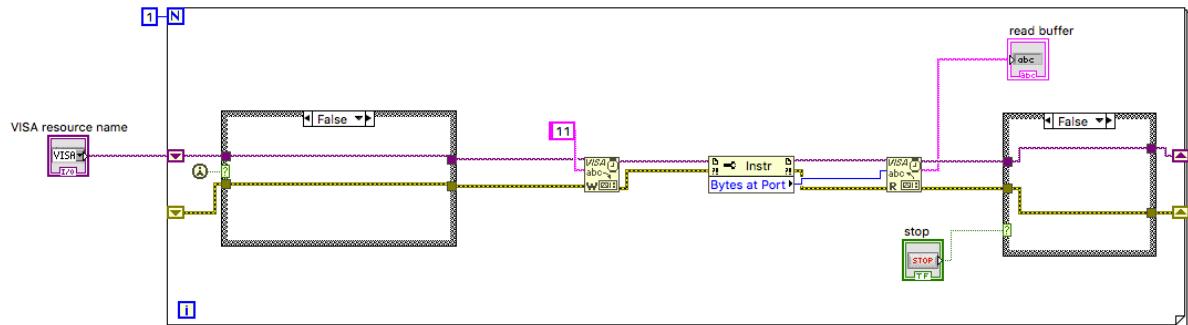
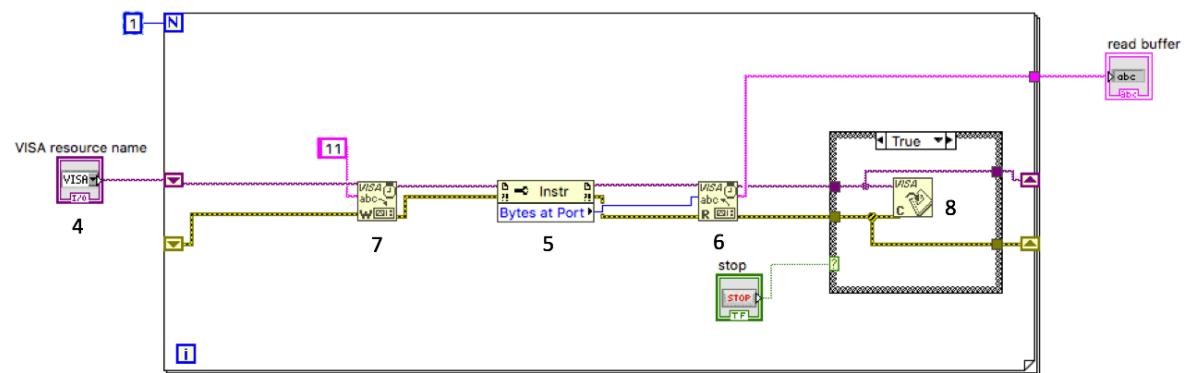


Abbildung 35: Einfügen von Stop Button, Indicators, VISA resource, For Loops und Schieberegister

signalisiert der Waage, dass der PC bereit ist Daten zu empfangen, da die Waage als `flow control` den Software-Handshake verwendet. Technisch wird an die Waage der Hexadezimale *character* 11 geschickt, also `0x11`. `0x11` ist das ASCII Hexadezimalzeichen für XON. Dieses Objekt in dieser Form ist nun alleinstehend fähig den seriellen Port zu konfigurieren, der Waage das `flow control` Signal XON zu senden, die Bytes auszulesen und darzustellen oder dem Haupt VI zu übergeben sowie den seriellen Port zu schließen, wenn das Programm über den Stop Button beendet wird.



(a) „Verdrahtung“ der **false** Cases



(b) Optimiertes SSVI

Abbildung 36: „Verdrahtung“ der **false** Cases und Optimiertes SSVI

Vierter Schritt: Optimierung Das Programm in Abbildung 35b und 36a konfiguriert bei jeder Iteration den Port und öffnet ihn. Das öffnen eines seriellen Ports, der bereits geöffnet ist, kann zu Störungen führen, daher sollte die Konfiguration des Ports über das Haupt-VI erfolgen. Der I/O-Buffer soll

auch nur bei der ersten Iteration des Hauptprogramms gelöscht werden, der wird ebenfalls ins Haupt VI verlagert. In Abbildung 36b ist das Objekt nach der Optimierung abgebildet. Um den ausgesehenen String nach Abschluss der Iteration an das Haupt VI übergeben zu können, wird der read Buffer rechts aus der For Loop geleitet. Das Objekt **Configure Port** hat für den Inputbuffer per default 4096 Bytes (Anmerkung: Sollte Ein I/O Buffer overflow o.ä. auftreten, dann könnte die Pause zwischen Iterationen zu lang sein). Mit Abschluss dieser Schritte ist das VI Objekt einsatzbereit und kann in mehrere Programmen, implementiert werden. Dafür müssen in der oberen rechten Ecke des Frontpanels die Ein- und Ausgänge des VI festgelegt werden (siehe Abbildung 37, Nr. 16). Dazu klickt man auf ein Feld des Icons in der oberen rechten Ecke und auf eine Funktion, die sich im Frontpanel befindet. In diesem Beispiel wurde die obere rechte Ecke des mini Icons angewählt und mit einem Klick auf **read buffer**, dem ausgelesenen String zugeordnet. Die obere linke Ecke des mini Icons wird mit **VISA resource name** und die Ecke unten links mit dem **Stop Button** belegt.

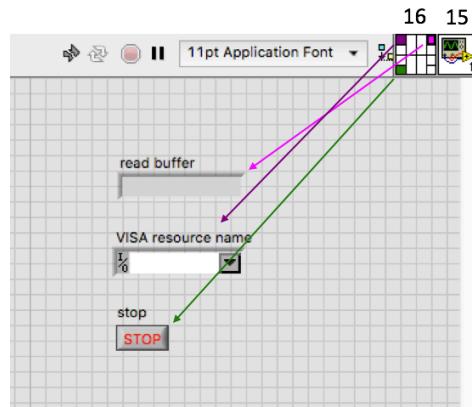


Abbildung 37: Definition der VI Objekt Ein- und Ausgänge

4.3.3 Kontinuierliche Messdatenerfassung RS-232 im Haupt VI

Die Daten, die man via RS-232 empfängt, müssen interpretiert und in einer *.txt Datei gelogged werden. Die empfangenen Daten hängen von der verwendeten Messtechnik ab. Das Untersuchungsobjekt im Rahmen dieses Abschnitts ist eine Digitalwaage (KERN 440-47N). Das Datenformat dieser Waage ist wie folgt:

1. Es können beliebig viele Leerzeichen zu Beginn des Strings vorhanden sein (' \s ')
2. Es kann die Vorzeichen + oder - enthalten

3. Es können beliebig viele '\s' folgen
4. Es können beliebig viele Ziffern folgen
5. Es wird ein dezimal Trennungszeichen folgen (ASCII encodiert wäre es der Punkt '.')
6. Es können wieder unbekannt viele Ziffern folgen
7. Es können beliebig viele '\s' folgen
8. Es kann die Maßeinheit 'g' folgen, wenn der Messwert stabil ist
9. Es können beliebig viele '\s' folgen
10. Das Zeilenende kann je nach System variieren. Es kann '\r', '\n', '\r\n' oder '\r\n\r\n' sein

Erster Schritt: Port Konfiguration und String verketten/anzeigen lassen Nach dem Erstellen eines neuen, leeren VI's, wird das VI des vorherigen Abschnitts per drag and drop in das leere VI gezogen. Dazu zieht man das Symbol 16 des SSVI (siehe Abbildung 37) in das neue VI. Als nächstes setzt man eine While Loop um dieses VI Objekt und verknüpft einen Stop Button (14) mit dem Stop Eingang des SSVI sowie dem Abbruchkriterium der While Loop (20). Außerhalb des Loops wird die Portkonfiguration (siehe Tabelle 7) und das Objekt, welches den I/O Buffer (9) löscht, platziert. Um die Daten der Waage in Form eines Strings aus dem SSVI auszulesen, wird die Funktion zum Verketten von Strings *engl. Concatenate Strings* (17) in den Loop platziert. Man benötigt Schieberegister (*engl. Shift Register*), erweitert das *Concatenate Strings* (17) Objekt auf zwei Eingänge und verbindet die *Shift Register* mit der *Concatenate Strings* (17) Funktion (links oberer Eingang des *Concatenate Strings* (17) Objekts). Damit das *Shift Register* bei der ersten Iteration (Index = 0, da LabVIEW, wie viele andere Programmiersprachen, 0 indexiert arbeitet) leer ist, wird das linke *Shift Register* mit einer leeren String konstante gelöscht. In Abbildung 38 sind die beschriebenen Funktionen abgebildet. Um zu sehen ob und was empfangen und verkettet wird, fügen wir einen *String Indicator* (21) ein. Da der Nutzer des Programms VISA Ressource und *String Indicator* sehen wird, ist es empfehlenswert es im FP in etwas eindeutiges umzubenennen. Um eindeutig erkennen zu können, welche Zeichen empfangen werden, kann man im Front Panel per rechts-klick in das entsprechende Fenster, die *String Indicator* Anzeige in '\' umstellen (siehe Abbildung 39).

Zweiter Schritt: „String Slicing“ Das VI verkettet den String bis zum Beenden des Programms. Um eine einzige abgeschlossene Zeichenkette (von String Beginn bis '\n') zu erhalten, müssen wir

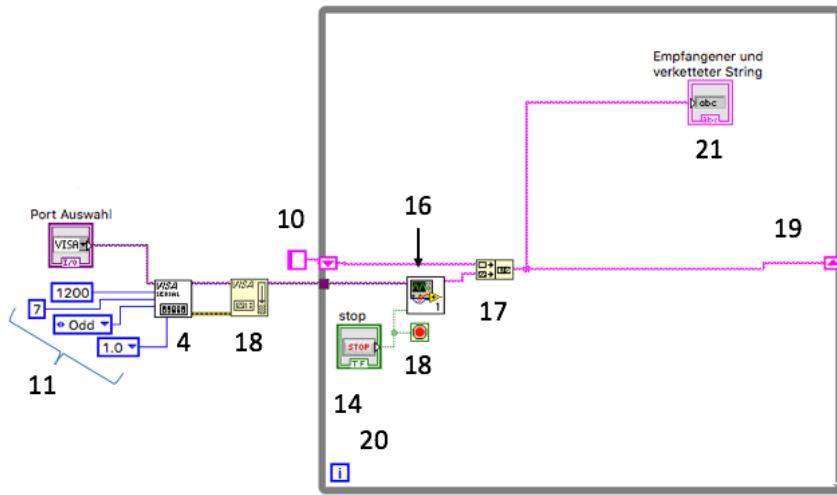


Abbildung 38: BD Haupt VI: SSVI Integration, Port Konfiguration, String auslesen und verketten

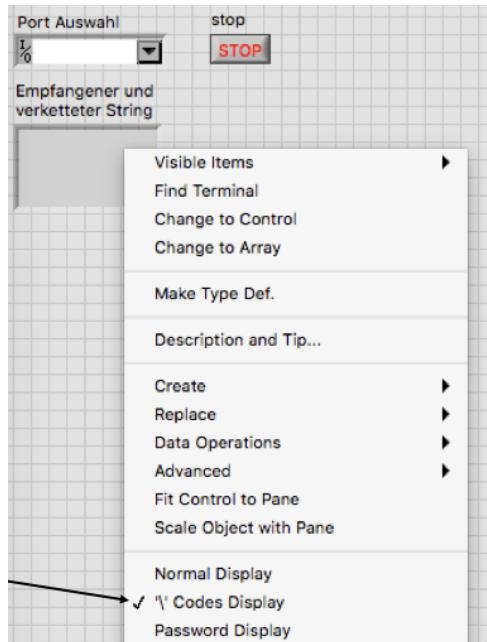


Abbildung 39: FP Haupt VI: SSVI Integration, Port Konfiguration, String auslesen und verketten

den String kürzen (*engl. slicing*). Dazu wird eine **Case Structure** (2c) programmiert, die den String verkettet, bis die **Search/Split String** (23) Funktion das von uns gesuchte Zeichen, das ASCII Steuerzeichen '\n' liest. Es können zwei Zustände vorliegen, **true** und **false**.

Anmerkung: Bei der Funktion **Configure Port** (4) ist per default **Termination Character**

auf `true` gesetzt, was dazu führt, dass `Read (6)` des SSVI beim decodieren der bytes, die sich im Input Buffer des Serial Ports befinden, beim decodieren des Steuerzeichens '`\n`', die Zeichenkette terminiert und alle weiteren Bytes, die sich im Input Buffer befinden, in der nächsten Iteration bis maximal zu einem '`\n`' gelesen und decodiert werden.

false Die `Search/Split String` Funktion soll nach einem '`\n`' Suchen. Der *offset of match* Ausgang der Funktion gibt den Index des Matches im String an. Wenn die `Search/Split String` Funktion das zu suchende Zeichen (in dieser Situation '`\n`') nicht findet, gibt *offset of match* -1 wieder. Diese Information wird genutzt, um den `Case Selektor` auf `false` zu setzen.

Wenn der String kein '`\n`' enthält, gibt die Funktion als `offset of match` den Index -1 zurück, demnach ist das boolsche Signal `true` aus der `Equal? (26)` Funktion. Dieses Signal wird mit dem boolschen Operator `Not (27)` negiert. Der `Case Selector (24)` ist somit auf `false` gesetzt.

Ist der `Case Selector (24)` auf `false`, wird der Draht, der den verketteten String (`Shift Register` und `SSVI`) enthält, in die `Case Structure (2c)` geleitet und mit dem rechten `Shift Register` verbunden.

An dieser Stelle (siehe Abbildung 40) wird nun eine **Sicherheit** eingebaut. Der String soll erst interpretiert werden, wenn die Länge des Strings größer einer Vorgabe ist. Die Länge einer Zeile kann dem Handbuch des jeweiligen Geräts entnommen werden oder sie wird durch das manuelle *Debugging* z.B. hTerm oder trial and error eruiert.

true Wenn der String '`\n`' enthält, gibt *offset of match* dem `Search/Split String (23)` Objekt den Index an, wo das Zeichen (in diesem Fall '`\n`') im String gefunden wurde. Der ist ungleich -1, der boolsche Zustand ist somit `false` und wird mit der `Not (27)` Funktion wieder in ein `true` umgewandelt. Der `Case Structure Selector (24)` ist damit auf `true` gesetzt. Nun soll der Folgeiteration der Teil des Strings ab dem gesuchten Zeichen übergeben werden. Das kann mithilfe des Objekts `String Subset` erreicht werden, indem man den Rest des Strings, des Objekts `Search/Split String (23)`, im `true` Case, dem Objekt `String Subset (25)` übergibt. Das Objekt `String Subset (25)` wird die Konstante 1 der Funktion `offset` zugewiesen. Durch diese Konfiguration der Funktionen wird der Rest des Strings **nach** dem gesuchten Zeichen ('`\n`') dem `Shift Register (19)` übergeben. Ist der PC und das Programm schneller als die empfangenen Daten, dann ist der übergebene String leer.

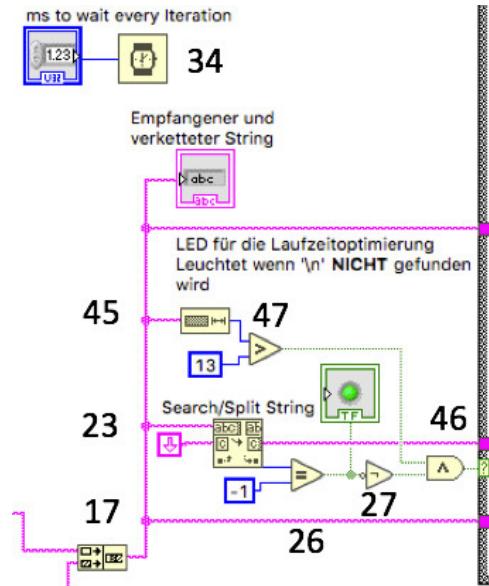


Abbildung 40: Slicing Bedingung

Dritter Schritt: „Dateninterpretation“ Sobald eine vollständige Zeichenkette gelesen wurde, müssen die Daten in ihre Bestandteile zerlegt werden (siehe Abbildung 41). In der Case Structure (2c) wird im true Case mit dem Objekt Regular Expression Match (28), aus dem gesamten vorgelegten String alle gewünschten und potentiell vorkommenden Ausdrücke (*engl. expressions*) separat extrahieren. In der folgenden Liste sind die Ausdrücke (*engl. Submatches*) aufgelistet, die der String enthält, bzw. enthalten könnte.

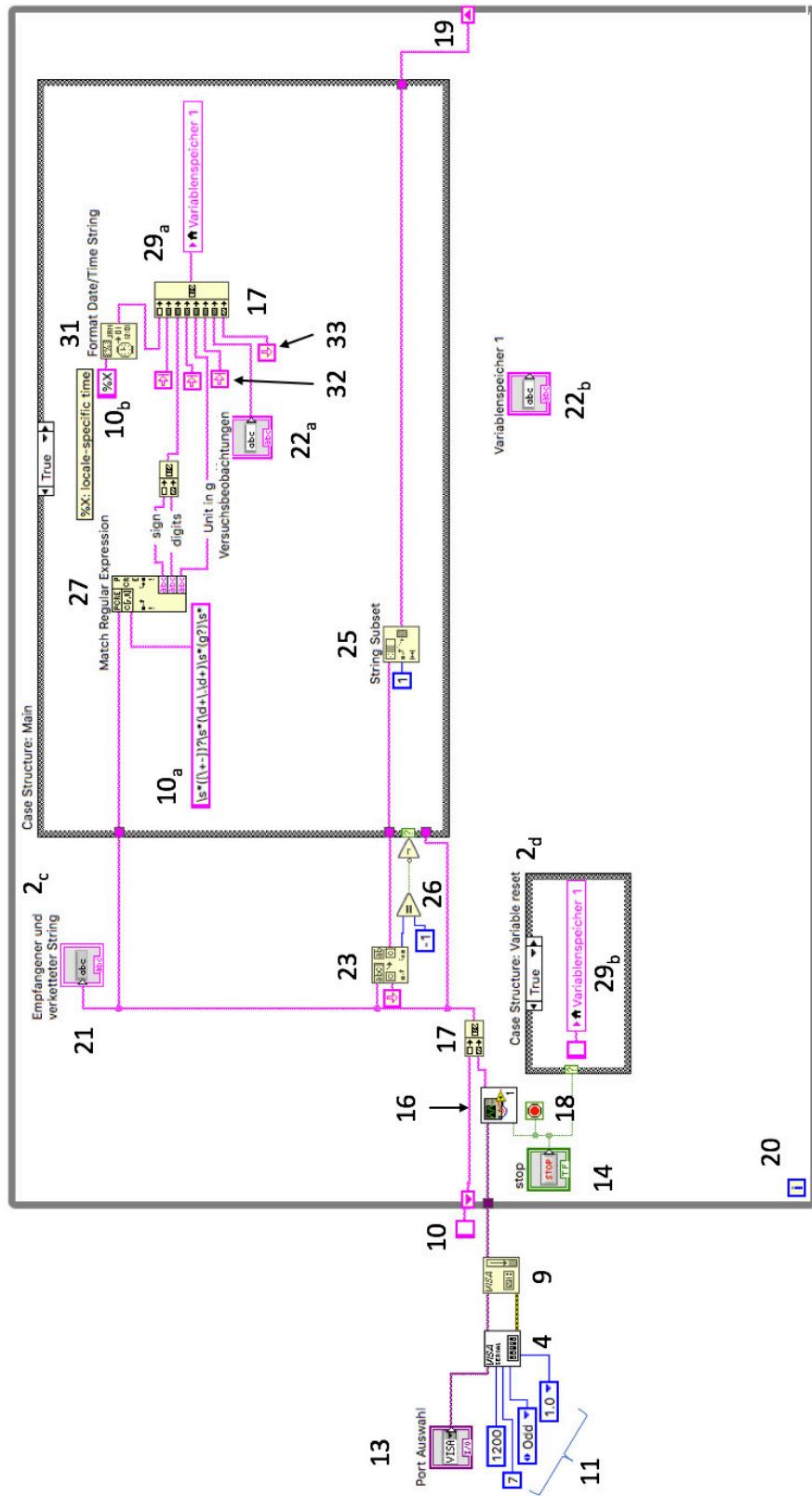


Abbildung 41: „String Slicing“: Dateninterpretation

- Es kann ein '+' oder '-' enthalten
- Beliebig viele '\s' (Leerzeichen)
- Beliebig viele Ziffern
- Dezimaltrennzeichen (unter ASCII '.')
- Beliebig viele Ziffern
- Beliebig viele '\s'
- Es kann ein 'g' enthalten sein
- Beliebig viele '\s'

In der folgenden Liste werden die verwendeten *Regular Expression* Operatoren erklärt.

- **(x)** generiert ein Submatch des Ausdrucks x
- **x?** der Ausdruck x kann null oder einmal vorkommen
- **x*** der Ausdruck x kann null oder viele male vorkommen
- **\d+** es können beliebig viele Ziffern (*engl. digits*) vorkommen
- **[...]** erstellt eine Zeichenklasse, die Angabe von **[xyz]** bedeutet, es kann x, y oder z vorkommen
 - sollen Elemente der Zeichenklasse mehrfach vorkommen dürfen, dann ist [...] ein + nachzustellen, daraus folgt **[xyz]+**

Dem Objekt *Regular Expression Match* müssen wir folgenden Ausdruck in einer String Konstanten mitteilen, die drei Submatches generiert.

`\s*([\+-])?\s*(\d+\.\d+)\s*(g?)\s*`

Submatch 1 `([\+-])?`

Submatch 2 `(\d+\.\d+)`

Submatch 3 `(g?)`

Nun ist eine Stringzeile zu konstruieren (siehe Abbildung 41). Das Ziel ist eine *.txt Datei, bei der die Spalten durch das Tabulator Zeichen ('\\t') (32) getrennt werden. **Beim Import einer, mit diesem Programm generierten, Tabulator-getrennt.txt-Datei in, z.B. ein Tabellenkalkulationsprogramm, wie Excel, ist als Spaltentrennzeichen Tabulator anzugeben.** Für die Konstruktion (Verkettung von Strings) einer Zeile verwendet man das `Concatenate` (17) Objekt. Ein String wird in einer textbasierten Programmiersprache wie Python mit Anführungsstrichen (") deklariert. Das Objekt `Concatenate` verkettet Strings.

`Concatenate` (17) bewirkt folgendes: ('ich bin' + 'ein String' = 'ich bin ein String')

LabVIEW's `Read` (6) Objekt liest Bit-seriell vom Port, versendet nach dem Lesen Byte seriell. Eine von der Waage gesendete Zeile könnte wie folgt aussehen:

```
'\s''\+\''\s''\s''\s''\s''12''\.\.'8''\s''g''\s''\r''\n'
```

Bei dem `Configure Port` Objekt (4) ist per default *Termination Character* auf `true` gesetzt, d.h. `Read` (6) liest bis zu einem '\n'. Der Rest verbleibt für die folgenden Iterationen im Input Buffer. Eine vollständig von `Read` (6) gelesene Zeile ergibt somit folgenden String:

```
'\s\+\s\s\s\s12\.8\sg\s\r\n'
```

Nun sollen die drei potentiellen Bestandteile ('+' , '12.8' , 'g') extrahiert werden. Das Vorzeichen und der Character 'g' können, müssen aber nicht vorkommen. Die drei Submatches sind somit:

```
'+' , '12.8' , 'g'
```

Mit den drei Elementen soll eine Datalogger Zeile generiert werden. Die Zeile soll wie folgt aufgebaut sein.

```
Zeitstempel \t Messwert \t Einheit in g \t Versuchsbeobachtung \n
```

Vierter Schritt: Zeilenkonstruktion Das Programm wird Zeile für Zeile den Inhalt, der in die Tabgetrennte.txt Datei geschrieben werden soll, vorerst als String verketteten. Dieser String wird nach dem Stoppen des Programms in die *.txt Datei geschrieben. Für die Konstruktion einer String Zeile wird

das *Concatenate* (17) Objekt genutzt (siehe Abbildung 42). Eine Dataloggerzeile wird in dieser Konfiguration (17, 22a, 31, 10b, 32, 33 und 29a), wie im vorherigen Abschnitt gefordert, konstruiert. Die Versuchsbeobachtungen werden mittels *Control* (22a) dem jeweiligen Zeitstempel zugeordnet. Nach dem Schreiben in die *.txt Datei muss die *Control* (22a) mit einem leeren String gelöscht werden (siehe Abschnitt 4.3.3 Abbildung 44). Die verkettete Zeile ist einem *Control* mittels einer lokalen Variable dem *Variablen Speicher 1* (29a) zu übermitteln. Mit der Information, die im *Variablen Speicher 1* hinterlegt wird, ist das kontinuierliche Loggen in eine *.txt, mit '\t' als delimiter möglich.

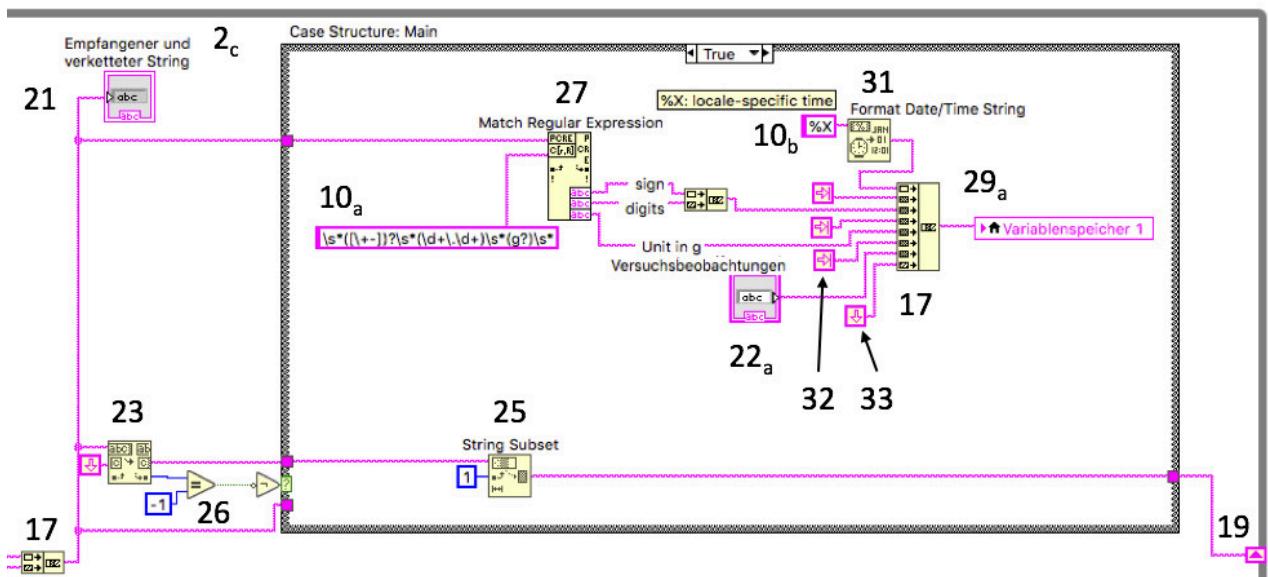


Abbildung 42: String-/Zeilenkonstruktion

Fünfter Schritt: Zeitstempelprogrammierung Der Zeitstempel (*engl. Timestamp*) kann durch die folgenden zwei Methoden implementiert werden. Bei der ersten Methode wird mittels des Objekts *Format Date/Time String* (siehe Abbildung 42 (31) und (10b)) eine frei formatierbare Zeitreihe möglich (für genauere Infos schauen Sie bitte in [19]).

Um als Zeitstempel die verstrichene Zeit in ms zu bekommen, ist das Objekt wie in der Abbildung 43 zu Programmieren (VTP Elapsed Time Sub VI) (44). Durch diese Methode ist es möglich bei Aktivierung dieses VI's eine Startzeit zu ermitteln. Die Startzeit wird für den Header und für den Start des Datalogs,

nach Betätigung der Start Datalogging in Tabgetrennt.txt Datei, benötigt. Aus diesem Grund sind zwei Kopien dieses VI's nötig!

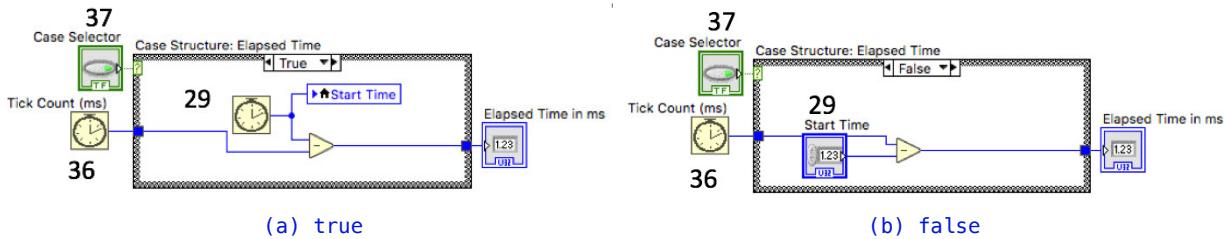


Abbildung 43: Methode 2 „VTP Elapsed Time in ms“ Sub VI (37)

Tabelle 9: Benötigte LabVIEW Konfigurationsfunktionen Teil 2

Nr.	Funktion	Lokalisation der Funktion in LabVIEW
30	LED	BD/rM auf Objektausgang oder Datenleitung/Create Indicator
31	Format Date/Time String	BD/Functions/Programming/Timing
32	Tabulator	BD/Functions/Programming/String
33	Linefeed/Newline	BD/Functions/Programming/String
34	wait (ms)	BD/Functions/Programming/Timing
35	wait until next ms multiple	BD/Functions/Programming/Timing
36	Tick Count (ms)	BD/Functions/Programming/Timing
37	Boolean Control	rM/Create Control
38	Write to Text	BD/Functions/Programming/File I/O
39	Set File Position	BD/Functions/Programming/ File I/O/Adv File Func
40	Build Path	BD/Functions/Programming/File I/O
41	Close File	BD/Functions/Programming/File I/O
42	Open/Create/Replace File	BD/Functions/Programming/File I/O
43	Elapsed Time	BD/Functions/Programming/Timing
44	VTP Elapsed Time	Abbildung 43
45	String Length	BD/Functions/Programming/String
46	Or	BD/Function/Programming/Boolean
47	Greater?	BD/Function/Programming/Comparison

Sechster Schritt: Versuchsbeobachtungen Es ist eine Kommentarfunktion gewünscht. Das Kommentar soll eingegeben werden können und nach der Bestätigung der Eingabe mit `Enter`, soll der Kommentar zum Zeitpunkt der `Enter` Betätigung dem jeweiligen Zeitstempel zugeordnet werden. Dafür ist die Einstellung gemäß Abschnitt 4.3.1 vorzunehmen. Ein **Control** Objekt (22) kann als Variablen speicher interpretiert werden. Nach der Übergabe eines Wertes an das **Control** Objekt des Kommentar muss diese nach der Ausgabe wieder gewiped werden. Zum „löschen“ der letzten Eingabe in **Control** Objekten, ist diese mit einer leeren Stringkonstanten zu überschreiben (siehe Abbildung 44) 29a. Der Programmablauf wäre dann wie folgt:

- Versuchsbeobachtung wird eingegeben (22a)
- *Case Selektor* wird getriggert durch Programmstart oder wenn die Dauer für ein Messwertaufnahmeverstrichen ist
- sobald der Datalogger Case Selektor getriggert wird, schaltet die *Case Structure* auf `true`
- Löschen der *Control Versuchsbeobachtung* durch die lokale Variable (298)

Siebter Schritt: Dataloggerprogrammierung Die Messwerte sollen in eine *Tabulator* getrennte *.txt geschrieben werden. Als nächstes ist ein Datalogger zu programmieren (siehe Abbildung 44). Mit den Objekten 38 bis 42 kann ein Datalogger programmiert werden, der jede `While` Loop Iteration die Daten in ein *.txt schreibt, die er in der jeweiligen Iteration übergeben bekommt. Während einer Iteration, kann er beliebig viele Zeilen in die *.txt Datei schreiben. Im `true` Case wird der String aus *Variablen speicher 1* in die *.txt Datei geschrieben, die in 40b, vor Programmausführung zu benennen ist (`test_aller_Funktionen_Beispiel.txt`). Mit dem *Elapsed Time* Objekt (35) kann ein **Control** Objekt programmiert werden, mit dem das Wertaufnahmeverstrichen in s eingestellt werden kann. Diese wird mit der Funktion *Time Target (s)* des *Elapsed Time* (43) Objekts verbunden. Über diese **Control** kann man das Messwertaufnahmeverstrichen variiieren, auch während des Programmablaufs. Der *Time has Elapsed* boolsche Ausgang ist mit einem `Or` (46) zu verbinden *Case Selector*. Wenn der Iterations Index 0 oder die angegebene Zeit verstrichen ist, wird ein Messwert gelogged, bzw. alle Zeile in die *.txt Datei geschrieben, die dem *Write to Text File* (38) Objekt übergeben wird. Im `false` Case soll nix passieren, daher werden die Tunnel (insgesamt 2 x 2) beider *Case Structure* Seiten mit einander verbunden. Der *Variablen speicher 1* wird mit einem leeren String gelöscht (*engl. wiped*, umgangssprachlich gewiped). Damit ist der *Variablen speicher 1* bis zur nächsten Eingabe geleert.

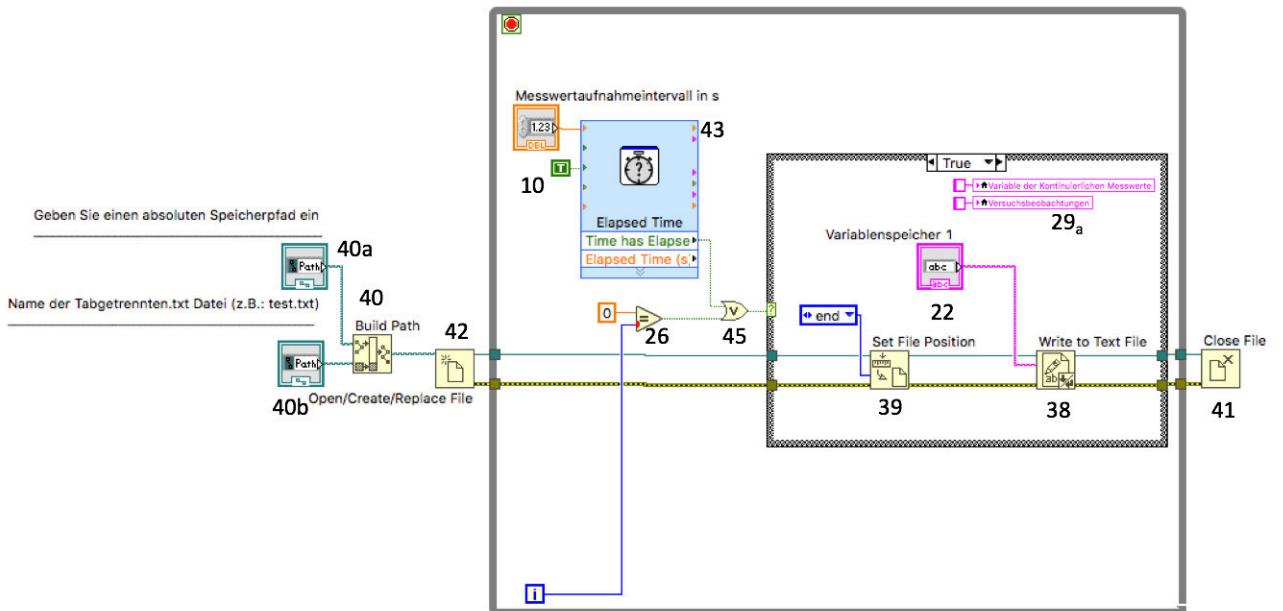


Abbildung 44: Datalogger

Siebter Schritt ergänzung gemäß neuster Applikationen: Graph mit zwei Plots Anmerkung: Aus zeitlichen Gründen wird auf eine genaue Bezeichnung verzichtet. Die Initialisierung gemäß Abbildung 44 ist zu erweitern, damit der Datalog per Knopfdruck geschehen kann. Die *case structure* hat eine *and* Bedingung erhalten, damit das Datalogging per Knopfdruck getriggert wird (siehe Abbildung 45a). In Abbildung 45a und 45b ist das VTP_ElapsedTime.vi zu erkennen. Damit bekommt der Data-logger eine Startzeit. In der Abbildung 45c ist die Erstellung des Graphen mit zu erkennen. Der Graph ist zum Plotten der Arrays Volumenstrommesswerte_arr_input und Druckmesswerte_arr_input. Die Arrays werden mit der Build Array Methode pro Iteration erweitert. Damit im Verlauf der Messung auch Zahlen im Blockdiagramm erkennbar sind wurde jeweils ein Array Indicator dem Draht hinzugefügt und die Anzeige gespiegelt. Die Spiegel dient dazu, dass der neuste Wert oben im Array angezeigt wird. Der Graph bekommt seine Zeitreihenwerte vom VTP_ElapsedTime2.vi (siehe Abbildung 45a). Die Zeitreihen und Messwert Arrays werden mittels Cluster Bundle zusammengeführt. Die erstellten Cluster werden mittel der Build Array Methode zusammengeführt und dem Graph als Dateninput geliefert. Das Graph VI kann aus einem Build XY Graph extrahiert werden (Doppelklick Strg+E etc.). Für den *false* case werden alle Drähte mit dem zugehörigen shift register verbunden.

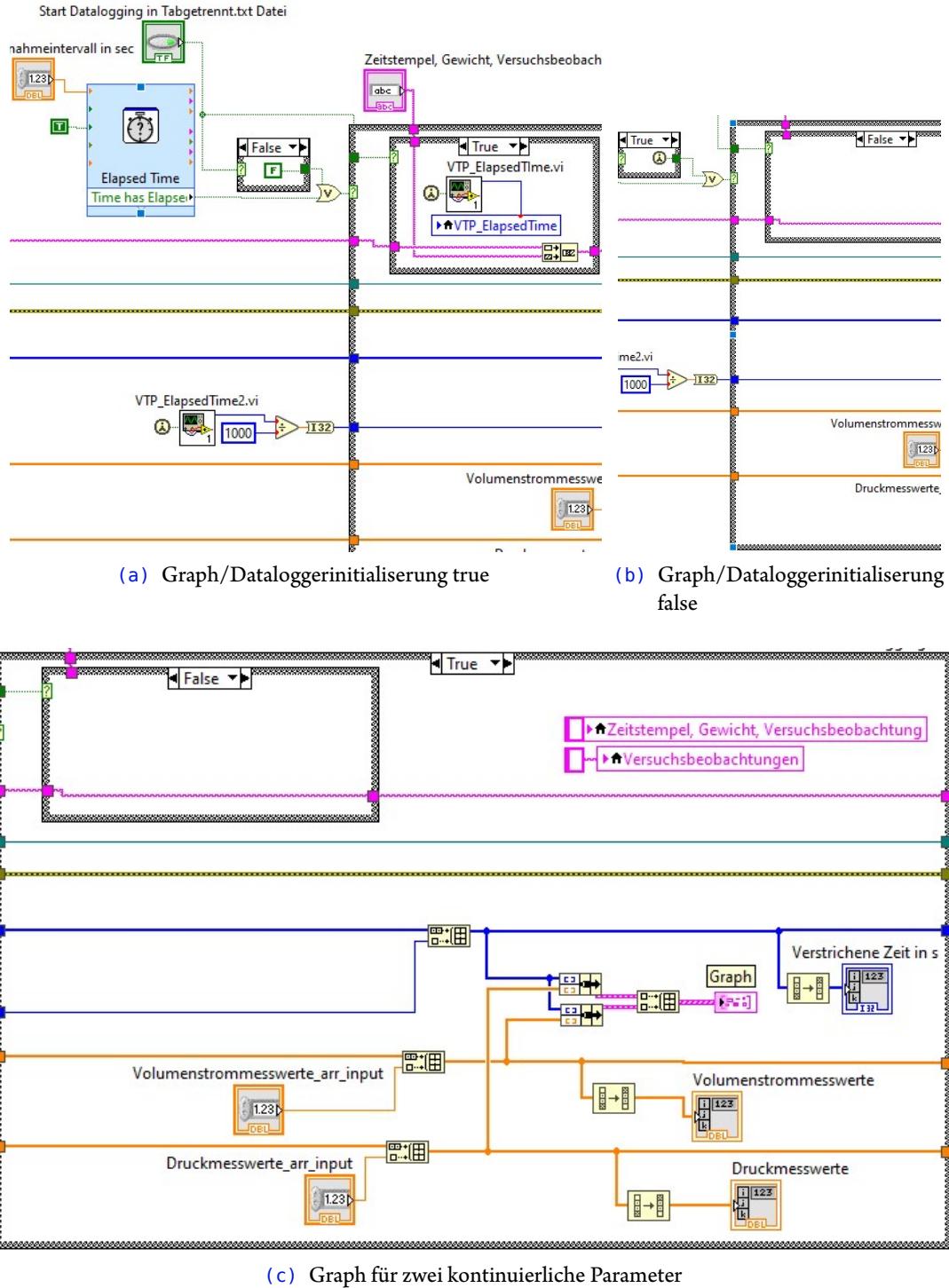


Abbildung 45: Graph für zwei kontinuierliche Parameter

Achter Schritt: Diskrete Daten am Beispiel des Datalog Headers In der Abbildung 46 ist exemplarisch die erste Headerzeile (a) und die Methode (b), wie man diese mittels *Concatenate* (17) verkettet abgebildet. Um eine schnelle und einfache Zusammenstellung der Headerkomponenten sowie die Übersichtlichkeit gewährleisten zu können, wird aus jedem Headereintrag ein Sub VI erstellt (siehe Abbildung 46 (b)). Der Datalogger Header ist somit einfach zu modifizieren.

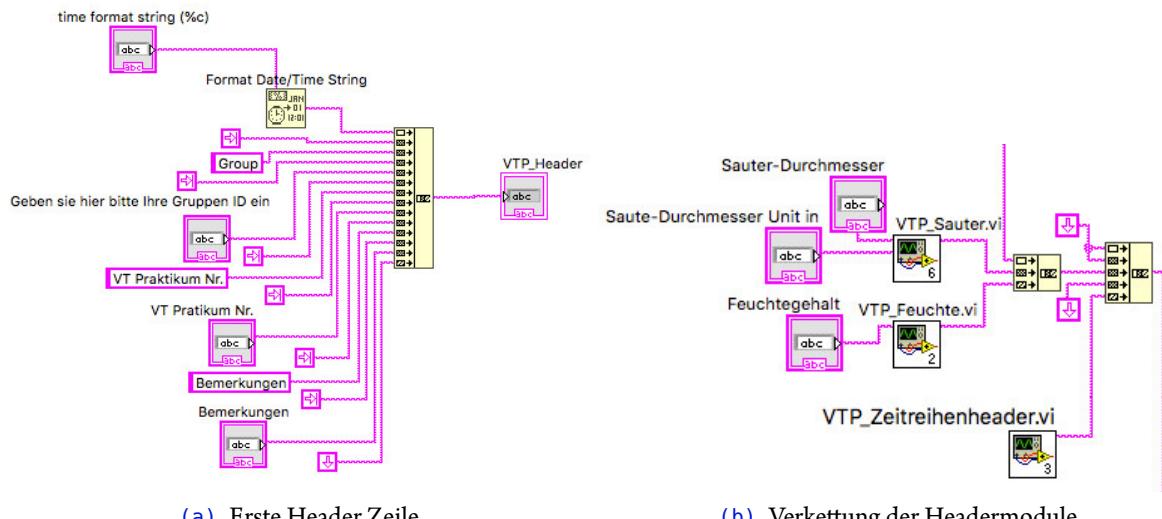


Abbildung 46: Modularer Header Sub VI's

Neunter Schritt: Auslastungsoptimierung Um die optimale Taktrate für das Programm empirisch zu ermitteln, kann man eine LED (30) zwischen den Objekten Equal? (26) und Not (27) platzieren. Nun variiert man die Taktrate der *While Loop* des *Haupt VI's*, bis die LED für die Optimierung blinkt, um Daten *Just in Time* zu erhalten. Mit dem Objekt *wait (ms)* (34) kann man eine Wartedauer vor jeder Iteration einstellen. Mit dem Objekt *wait until next ms multiple* (35) kann man eine Taktzeit für einen Iteration einprogrammieren. Die Dauer der Pause, bzw. die Taktrate jeder Iteration sollte geringer sein als die Dauer die benötigt wird, um die Daten von dem seriellen Port auszulesen. Das Blinken signalisiert, dass der (Read Buffer) der Datenquelle nicht überfüllt (engl. *buffer overflow*) wird. Damit ist gewährleistet, dass für das Ausführen des Programms **nur soviel Rechenkapazität verwendet wird wie nötig, jedoch genug, damit keine Daten verloren gehen oder eine Totzeit entsteht!** Um die Taktrate rechnerisch zu ermitteln, benötigt man die zu erwartenden Bytes, bzw. character pro Zeile. Die Baudrate (Bd) gibt an, wie viele **Zeichen** pro Sekunde versendet werden. In vielen Fällen wird für 1 **Zeichen**, 1 Bit benötigt, daraus folgt, dass die Baudrate in diesen Fällen Bit pro s bedeutet.

Um ein 7 Bit encodiertes Byte zu versenden, werden 3 weitere Bit auf der Datenleitung versendet. In der Tabelle 10 sind die Konfigurationsmöglichkeiten bei der Nutzung von 6, 7 oder 8 Bit Encodierung und RS-232.

Tabelle 10: 6, 7, 8 Bit-Encodierung via RS-232

	6 Bit	7 Bit	8 Bit
Startbit	1	1	1
Symbol-/Characterbit	6	7	8
Paritätsbit	Odd/Even/Mark/Space		none
Stoppbit	2	1	0
Baudrate in Bd	diverse Möglichkeiten		
Handshake	Software/Hardware		

Die Dauer zum empfangen einer Zeile ('+' oder '-' + floatnumber + potentiell 'g'), die 7 oder 8 bit ASCII encodiert ist und via RS-232 Schnittstelle versendet wird, lässt sich wie folgt berechnen (Annahmen: Bd = 1200 $\frac{\text{Byte}}{\text{s}}$, 15 $\frac{\text{Byte}}{\text{Zeile}}$, 10 $\frac{\text{Bit}}{\text{Byte}}$ 70).

$$1 \text{ Bd} = 1 \text{ Zeichen/Bit pro Sekunde} \quad (47)$$

$$1 \text{ Byte} = 10 \text{ Bit} \quad (48)$$

$$15 \text{ Byte} = 1 \text{ Zeile} \quad (49)$$

$$\frac{1}{1200} \frac{\text{s}}{\text{Bit}} \cdot 10 \frac{\text{Bit}}{\text{Byte}} \cdot 15 \frac{\text{Byte}}{\text{Zeile}} = 0,125 \text{ s pro Zeile} \quad (50)$$

$$\equiv 125 \text{ ms pro Zeile} \quad (51)$$

4.3.4 NI DAQmx Programmierung am Beispiel der Wirbelschichtanlage

Neben Geräten, die einen integrierten analog digital Wandler haben, gibt es Sensoren, die lediglich das analoge Signal transmittieren. Folglich muss das analoge Signal in ein digitales Signal umgewandelt werden. Für den Fall das Sensoren keinen integrierten analog/digital Wandler besitzen, ist eine Messkarte zwischen Sensor und PC zu implementieren (siehe Abbildung 47).

Die NI Messkarte des Typs USB-6001 nennt sich in LabVIEW *DAQmx*. Im folgenden Abschnitt wird die DAQ eines Volumenstromsensors und eines Drucksensors mittels USB-6001 Messkarte von National Instruments erläutert. In der Abbildung 48a ist die Initialisierungssequenz abgebildet, um ana-

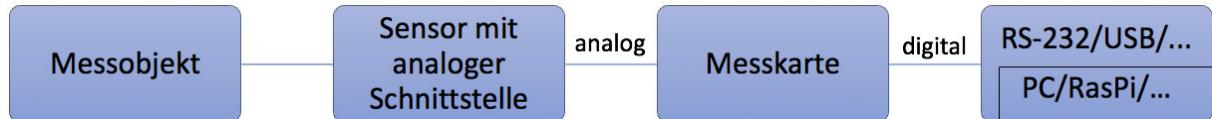
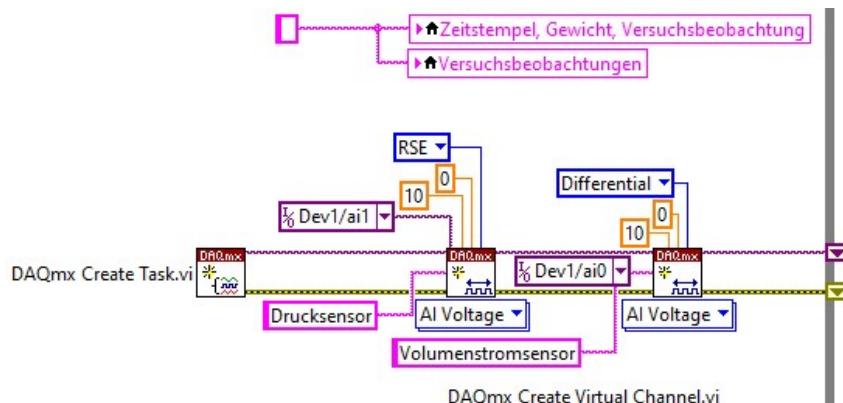
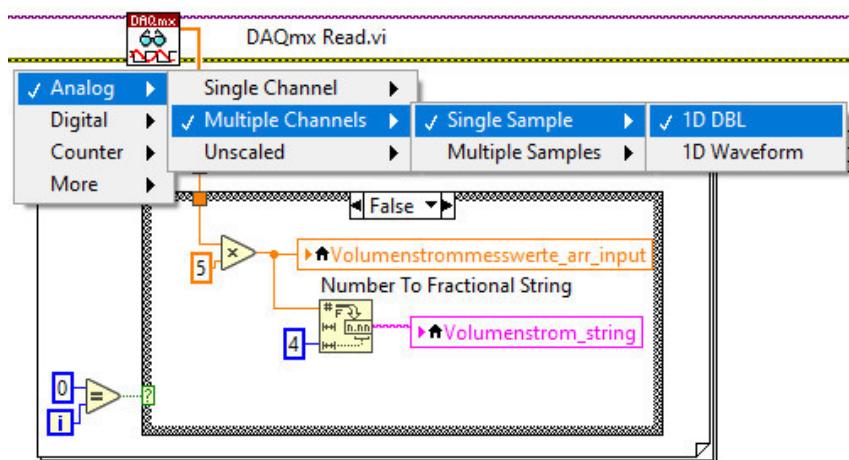


Abbildung 47: DAQ von Messeinrichtungen mit analoger Schnittstelle



(a) Druck- und Volumenstromsensor Initialisierung der Wirbelschicht



(b) DAQ read options

Abbildung 48: DAQ Programmierung, Initialisierung und read.vi

loge Signale von zwei Sensoren zu akquirieren. Es ist zu erkennen, dass zwei Variablen (Zeitstempel,... ; Versuchsbeobachtungen), mittels lokaler Variable durch einem leeren String initialisiert werden. Diese Form der Initialisierung hat eine Löschung der Werte aus der vorherigen Iteration zur Folge. Darunter ist die Initialisierung der DAQ Aufgabe (*engl. task*), gefolgt von der Erstellung von zwei virtuellen Kanälen (*engl. channel*) zu erkennen. Die Reihenfolge wurde zufällig gewählt. Es ist zu erkennen, dass der erste `channel` den Drucksensor abfragt. Für den Drucksensor ist der Modus Operandi (Betriebsmodus) single ended (RSE) auszuwählen. Die minimalen und maximalen Signal- bzw. Spannungswerte entsprechen 0 und 10 V. Der Drucksensors ist an Slot `ai1` angeklemmt. Das DAQmx Gerät hat im System die Kennung `Dev1` erhalten, daraus folgt das der Slot `Dev1/ai1` auszuwählen ist. Für den Volumenstromsensor ist analog vorzugehen. Gemäß der elektrotechnischen Verschaltung (siehe Abbildung 64 im Abschnitt 5.2.2) ist das Signal des Volumenstromsensors nach dem Strom-/Spannugnswandler differenziell (double ended), daher ist als Modus `Differential` auszuwählen. Die Signalspanne ist analog des Drucksensors von 0 bis 10 V. Der Draht mit den Messignalen und der Errordraht betreten die While-Loop via `Shift Register`. Das Auslesen wird mit dem `DAQmx Read.vi`, mit den Optionen gemäß 48b durchgeführt.

An der Stelle muss die serielle Abfrage Programmiert werden. In der Abbildung 51a ist zu erkennen, dass der `for loop tunnel` (ist am N zu erkennen) ein weißer Kasten mit orangen Klammern ist. Das bedeutet, dass die `for loop` betretenden Werte indexiert sind. Während einer `while loop` Iteration erfolgen **zwei for loop Iterationen**. Zur Erinnerung wird an dieser Stelle noch mal erwähnt, dass LabVIEW alles (Arrays, Schleifen etc.) mit dem Index null initiiert. Pro `while loop` Iteration werden die Werte, die in die `for loop` eintreten also immer mit 0 und 1 indexiert. Der case structure selector muss mit dem `equal?` Vergleichsoperator abfragen, ob der Wert 0 oder 1 ist. Das Signal des Drucksensors hat den Index 0, da es der erste `channel` ist, der initialisiert wird (vgl. Abbildung 48a). An dieser Stelle ist nun ein Gleichungssystem zu lösen. Der Sensor hat eine Messspanne von -1 bis 1 bar. Das elektrische Signal geht von 0 bis 10 V. Gemäß Abschnitt 2.1.2 ist die Approximation durch eine Gerade hinreichend. Daraus folgt, dass für den Fall, dass die Spannung am DAQ Gleich 0 ist folgende Gleichung:

$$p = a \cdot x + b \quad (52)$$

$$-1 \text{ bar} = a \cdot 0 \text{ V} + b \quad (53)$$

$$-1 \text{ bar} = b \quad (54)$$

Für den Fall, dass die Messkarte 10 V anzeigt gilt in Kombination mit Gleichung 54

$$1 \text{ bar} = a \cdot 10 \text{ V} - 1 \text{ bar} \quad (55)$$

$$a = \frac{1 \text{ bar} + 1 \text{ bar}}{10 \text{ V}} \quad (56)$$

$$a = \frac{2 \text{ bar}}{10 \text{ V}} \quad (57)$$

$$p = 0,2 \frac{\text{bar}}{\text{V}} - 1 \text{ bar} \quad (58)$$

Da ein linearer Zusammenhang besteht und der Druck relativ zur Atmosphäre angegeben wird kann der Offset durch die Addition des atmosphärischen Drucks 1 bar eliminiert werden. Der Multiplikationsfaktor $0,2 \frac{\text{bar}}{\text{V}}$ ist somit der gesuchte Wert für die Signalumrechnung.

In der Abbildung 49 ist der Vergleich der Analog- und Digitaldruckmessung als Funktion des analog gemessenen Durchflusses zu erkennen. Des Weiteren wurde von beiden Messwertreihen eine lineare Approximationsfunktion erstellt. Der Drucksensor hat zum Zeitpunkt dieser Messung in der LabVIEW Applikation (als Konstante) zwei signifikante Stellen. Eine Wiederholung dieser Messung wird empfohlen, daher wird auf eine Anzeige der Approximationssfunktion zur Berechnung der Geraden verzichtet. In der LabVIEW Applikation ist für den Drucksensor die Anzahl der signifikanten Nachkommastellen auf vier korrigiert worden (siehe Abbildung 48).

In einem thermischen Massendurchflusssensor ist eine Wheatstone'sche Brückenschaltung als elektrotechnische Komponente verbaut. Das elektrische Ausgangssignal des Sensors ist demnach ebenfalls linear. Für den Volumenstromsensor ist folglich analog vorzugehen. Der Parameter a des Volumenstromsensors ist somit $5 \frac{1/\text{min}}{\text{V}}$, mit $b = 0 \frac{1}{\text{min}}$.

In der Abbildung 50 ist eine Volumenstrommessung, analog sowie digital, sowie der Betrag der absoluten Differenz beider Messwertreihen als Funktion der Manometerdruckdifferenz in mm abgebildet. Bei beiden Messwertreihen ist ein linearer Zusammenhang zu erkennen. In Folge dessen wurden lineare Approximationssfunktionen erstellt. Des Weiteren ist zu erkennen, dass beide Messwertreihen divergieren. Die größte Abweichung hat einen Betrag von $6,7 \frac{1}{\text{min}}$. Da der „Wahre“ Wert unbekannt ist wird keine

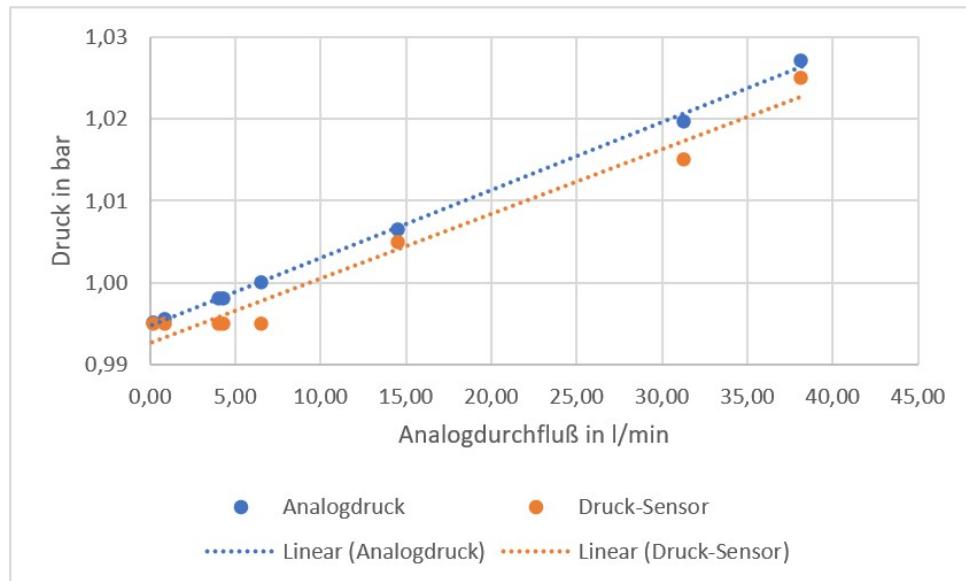


Abbildung 49: Analog-, Digitaldruckmessung als Funktion der analogen Volumenstrommesswerte

relative Abweichung sowie Standardabweichung ermittelt.

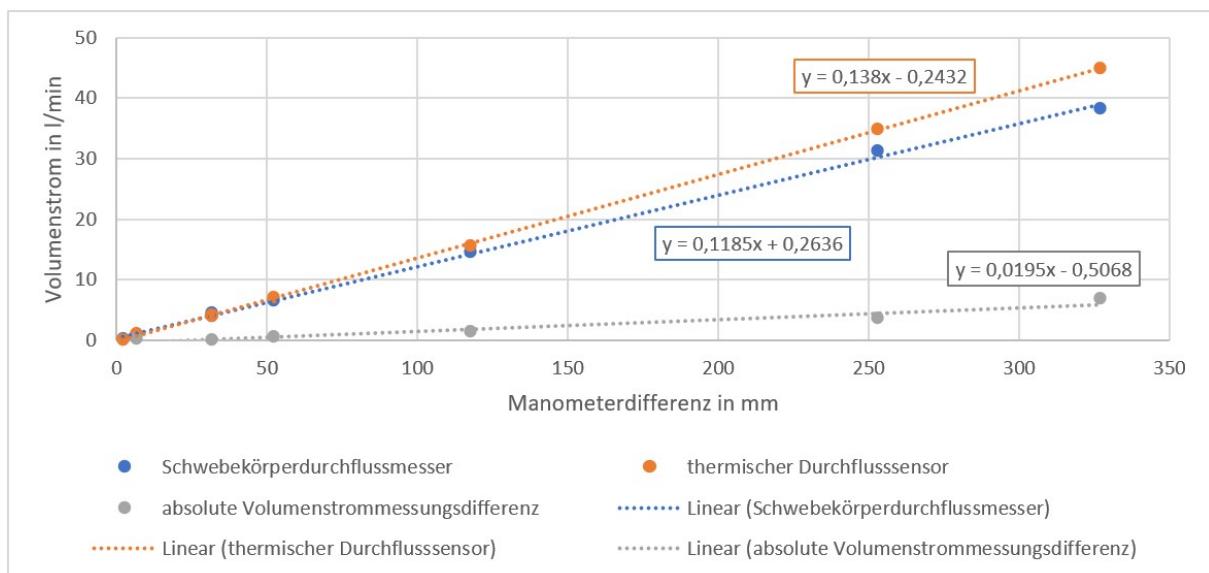


Abbildung 50: Analog-/Digitalvolumenstrommessung sowie absolute Abweichung in l/min, als Funktion der Manometerdifferenz in mm

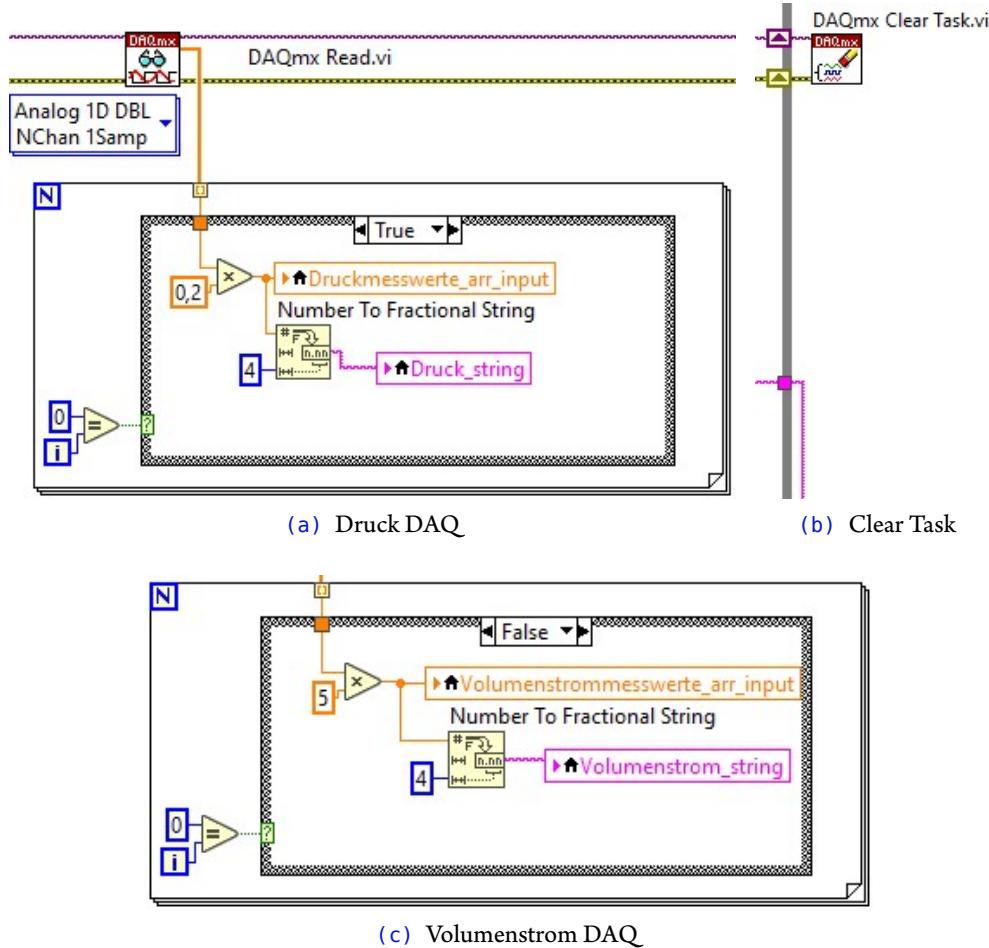


Abbildung 51: DAQ und interpretation beider kontinuierlichen Parameter, Druck und Volumenstrom

Bei der Beendigung des Programms musst der DAQmx task terminiert werden, dafür ist das DAQmx Clear Task.vi außerhalb der *while loop* zu platzieren und mit dem Signal sowie error shift register zu verbinden (siehe Abbildung 51b).

Haupt Programm Dataloggerinitialisierung am Beispiel der Filterkuchenversuchsanlage

Im Verlauf des Projekts ist der Wunsch geäußert worden, dass die generierten Protokolle unveränderlich (Schreibschutz) sein sollen. Der Dateidialog (siehe Abbildung 82 im Anhang) wird vor der Schreibschutzabfrage platziert (siehe Abbildung 52) und, im Falle einer versuchten Öffnung einer Datei mit Schreibschutzes, in der case structure (**No Error**) entfernt. Im **Error** case werden die Tunnel

beider Seiten verbunden (siehe Abbildung 52). Auf weitere Erläuterungen wird im Rahmen dieser Arbeit verzichtet

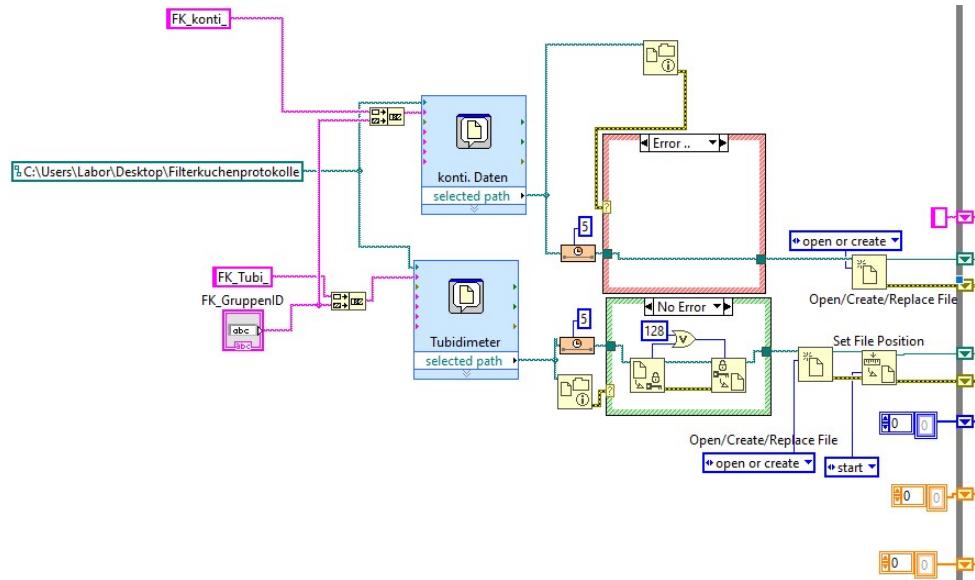


Abbildung 52: Dataloggerinitialisierung

Haupt Programm Dataloggerheader am Beispiel der Filterkuchenversuchsanlage An der Abbildung 53 lässt sich erahnen, dass die Erstellung diskreter Daten in LabVIEW eine tiefgreifende Verschachtelungen mittels eigens erstellter Sub VI's benötigt. Auf eine Erläuterungen wird im Rahmen dieser Arbeit verzichtet. Es ist zu erkennen, dass der verkettete String aus der `while` loop geführt wird.

Im Verlauf des Projekts wurde der Wunsch geäußert eine weitere Datei, mit sogenannten Tubidimeter Daten, generieren zu lassen. Analog des Headers ist eine verschachtelte Programmierung der Tubidimeter Eingaben erfolgt. Auf eine Erläuterungen wird im Rahmen dieser Arbeit verzichtet. Diese Daten werden während der Versuchsdurchführung, nach der Beendigung (Stop-Button im Frontpanel siehe Abbildung 56) eines Messungsdurchlaufs eingefügt. Es werden laut Praktikumsanweisung pro Versuchstag hintereinander drei Messungen durchgeführt. Das Schreiben, in die Datei mit den kontinuierlichen Daten und in die Datei mit den Tubidimeter Daten, muss sich an der Stelle unterscheiden. Für die Tubidimeterdatei muss die gesamte Datei Überschrieben werden und nicht wie im Fall der kontinuierlichen Datei am Ende angehängt. Set File Position.vi erhält an dieser Stelle die Konstante start (siehe Abbildung 52).

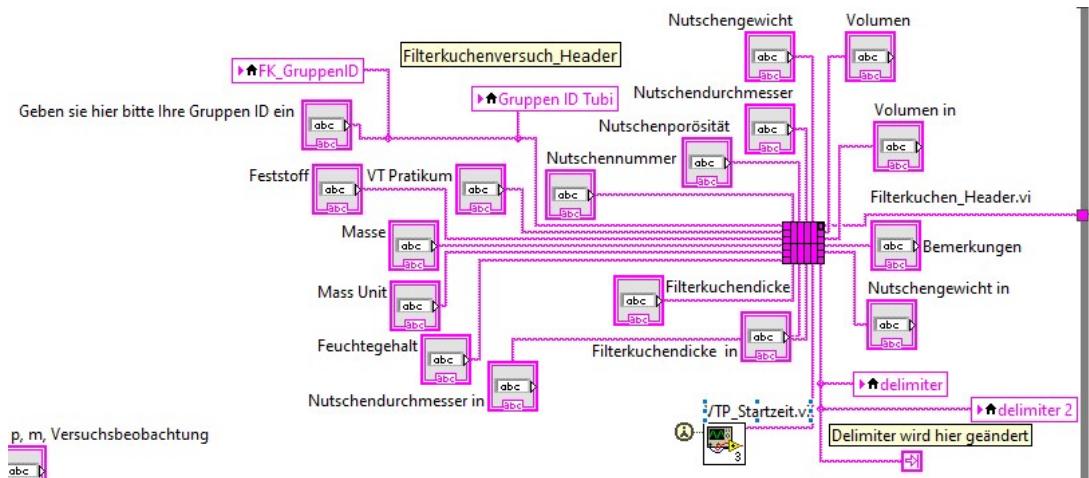


Abbildung 53: Filterkuchenheader

Haupt Programm Endsequenz Wird das Hauptprogramm via Stop-Button beendet, dann erfolgt die Endsequenz gemäß Abbildung 54. Die Daten werden an dieser Stelle in die geöffnete bzw. erstellte Datei geschrieben, diese Datei geschlossen und mit einem Schreibschutz versehen.

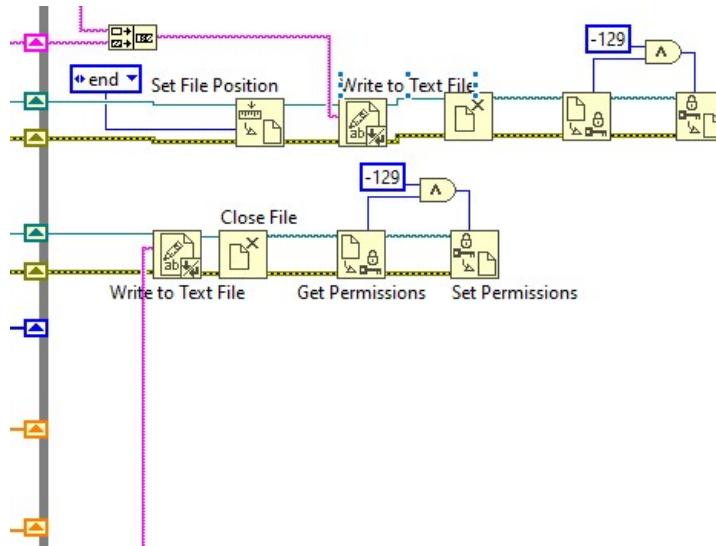


Abbildung 54: Endsequenz des Hauptprogramms

4.4 Verwendung beider Programme

Damit die Kommentarfunktion wie gewünscht funktioniert, ist vor der Verwendung des Programms in den LabVIEW Einstellungen unter Environment End Text entry with Enter key einzustellen (siehe Abbildung 55). Ist diese Option nicht eingestellt, muss nach jeder Betätigung der Enter-Taste, die Eingabe ein weiteres mal mittels eines Mausklicks auf einen schwarzen Haken, der neben dem VI Ausführen Pfeil (oben links im Front Panel) erscheint, quittiert werden.

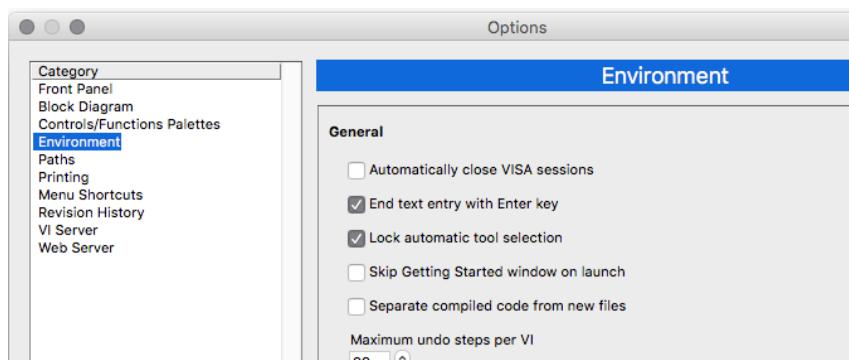


Abbildung 55: Einstellung „End text entry with Enter key“

Es wurde eine Tab Control verwendet, um verschiedene Registerfenster zu generieren, siehe folgende Liste:

- Versuchseingaben
- Während der Messung
- Tubidimeter
- Konfiguration
- Debugging

Wie der Liste (Markierung) und der Abbildung 56 zu entnehmen ist, sind die Tabs Konfiguration und Debugging nicht sichtbar. Zum erreichen dieser Tabs ist das Frontpanel mit einem Passwort zu entsperren (z.B. durch die Betätigung von `Strg+E`) und gemäß Abbildung 83 im Anhang fortzufahren.

In der Abbildung 56 ist das Frontpanel für den Filterkuchenversuchstand abgebildet. Oben links im Bild ist das Programmstarticon ⇒ zu erkennen. Für den Wirbelschichtversuchsstand sieht das Frontpanel

identisch aus.

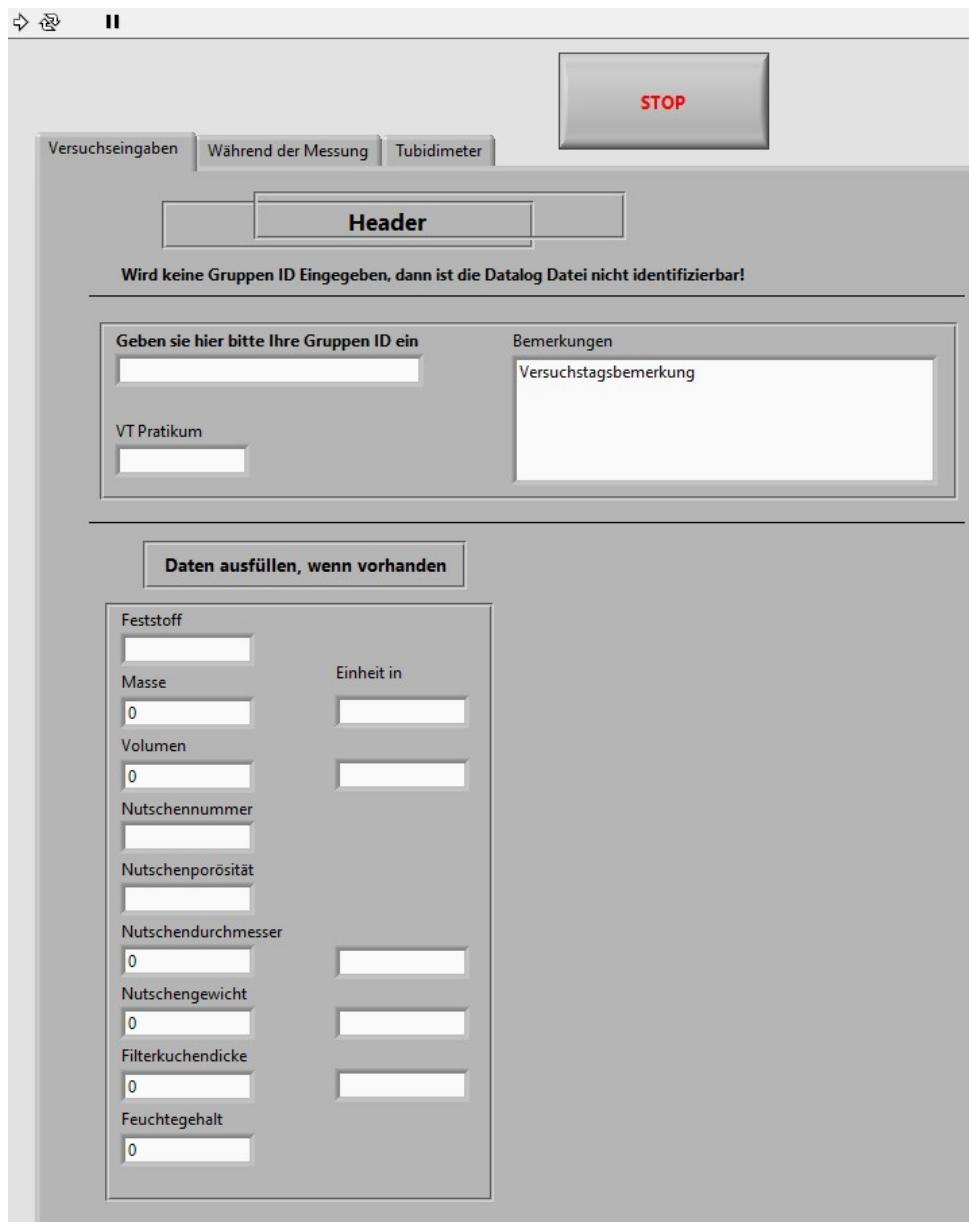


Abbildung 56: Filterkuchen Frontpanel: Versuchseingaben

Die Frontpanel Tabs für die kontinuierlichen Messungen sehen für beide Versuchsstände identisch aus (siehe Abbildung 57). Unter dem Plotter ist links ein Eingabefeld für Versuchsbeobachten zu erkennen.

Die Eingaben werden dem Zeitstempel, zum Zeitpunkt der Enterbetätigung, angehängt. Rechts von dem Eingabefeld für die Versuchsbeobachtungen sind die Arrays abgebildet, die oben den aktuellsten Messwert anzeigen. Das Schreiben der kontinuierlichen Daten (in einen String) erfolgt erst nach Betätigung des Buttons `Start Datalogging` in `Tabgetrennt.txt Datei`. Die Endsequenz, in der die Daten der kontinuierlichen Messung sowie die der Tubidimetereingaben (siehe Abbildung 58), in die geöffnete oder generierten Datei geschrieben werden, erfolgt nach dem Betätigen des Stop-Buttons (siehe Abbildung 56).

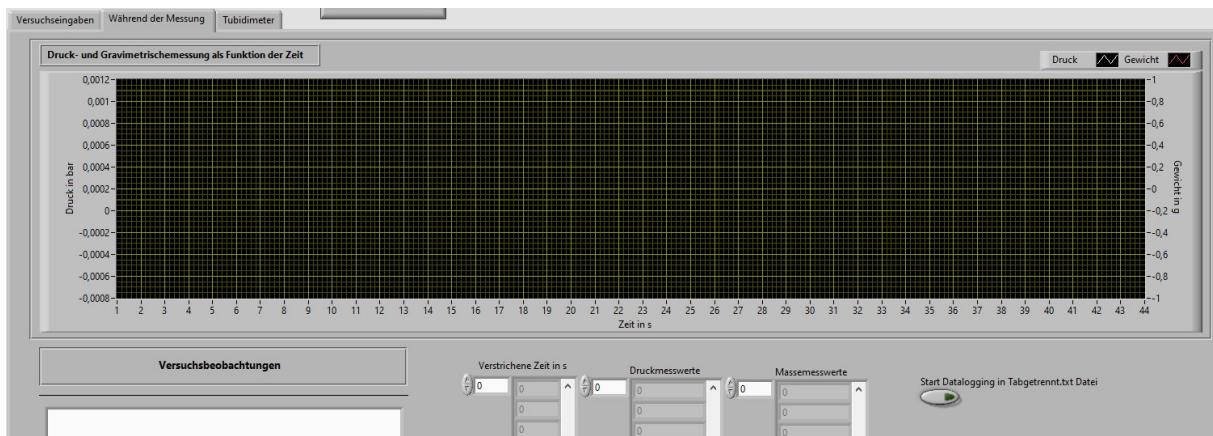


Abbildung 57: Tab der kontinuierlichen Messung am Beispiel der Filterkuchenapplikation

In der Abbildung 59a ist ein Datalog eines Wirbelschichtestdurchlaufs zu erkennen. Diese .txt Datei wurde in Excel Importiert. Es ist zu erkennen, dass diese Messwerte vier signifikante Stellen aufweisen. Es kann vorkommen, dass die automatische Datenimportfunktion das Spaltentrennzeichen nicht erkennt, dann ist gemäß Microsofts Problembehandlungshinweis fortzufahren (siehe Teilausschnitt in Abbildung 59b). Im Falle der Tubidimeterdatei ist dies der Fall gewesen, daher wird der Import nicht automatisch eingefärbt (siehe Abbildung 59c). Des Weiteren ist zu erkennen, dass die Eingaben keine Restriktionen enthalten (`int8, 16, 32, 64; float8, ...;` die Ziffer steht für die Bytelänge). Alle Eingaben sind `Strings`, das bedeutet, es ist im Textformat. Bei der Weiterverarbeitung ist eine Umwandlung in ein `int` oder `float` zu erfolgen (in Excel Zelle formatieren...).

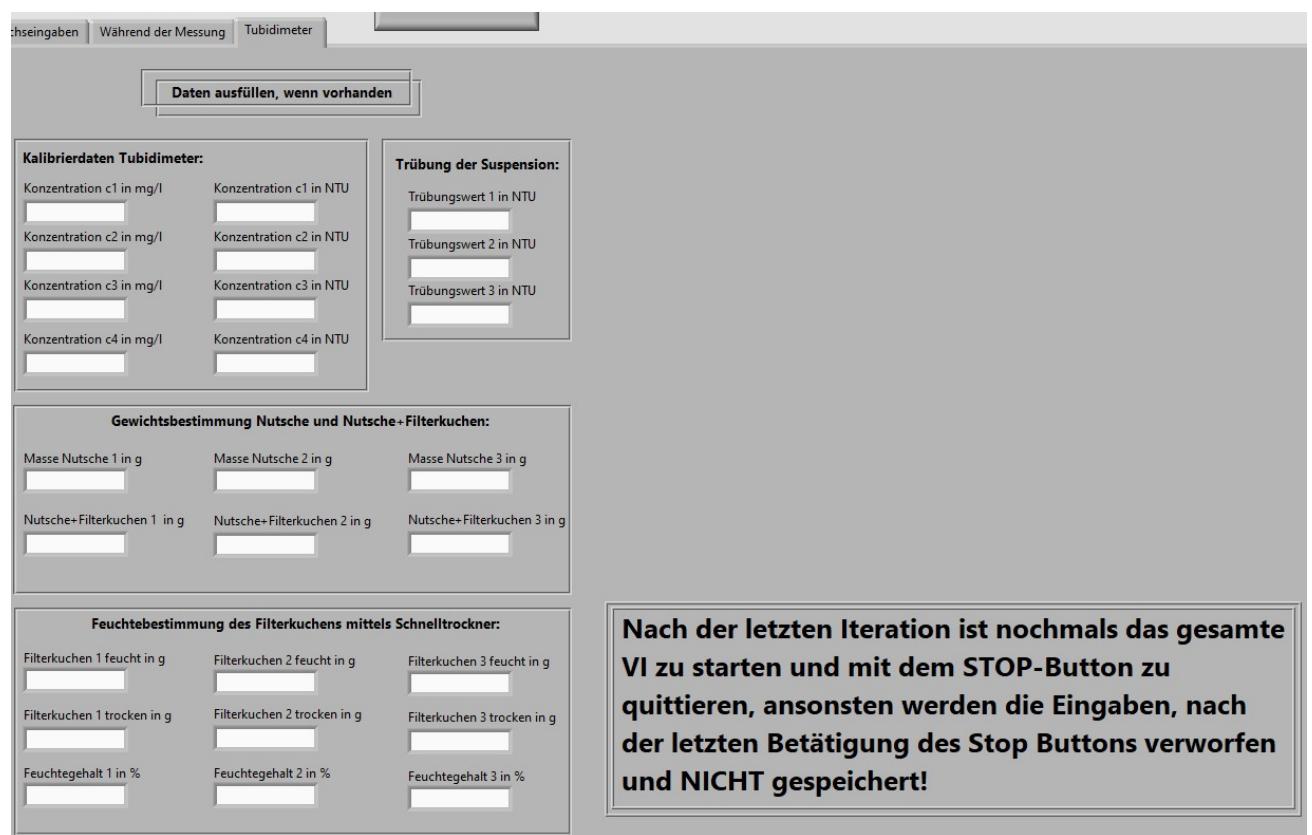


Abbildung 58: Filterkuchen Frontpanel: Tab der Tubidimetereingaben

4 Signal- und Datenverarbeitung von Messeinrichtungen

1	Column1	Column2	Column3	Column4	Column5	Column6
2						
3						
4	Datum	Startzeit	Endzeit	Gruppe	VT Praktikum Nr.	Bemerkungen
5	24.02.2021		12:22:36	Daniel_Validierung	Versuchstagsbemerkung	
6						
7	Rohstoff					
8	Rohstoffdichte					
9	Schüttdichte					
10	Masse					
11	Volumen					
12	Sauter-Durchmesser					
13	Festbett-Porösität					
14						
15	Zeitstempel	Druck in bar	Volumenstrom in l/min	Versuchsbeobachtung		
52	12:22:00	1,0019	35,5418			
53	12:22:01	1,0029	36,8504			
54	12:22:02	1,0042	38,7907			
55	12:22:03	1,0050	40,1121			
56	12:22:04	1,0065	41,6915			
57	12:22:05	1,0076	43,3546			
58	12:22:06	1,0089	45,3464			
59	12:22:07	1,0101	47,3254			
60	12:22:08	1,0132	49,2270			
61	12:22:09	1,0138	50,0457			
62	12:22:10	1,0171	50,0522			

(a) Datalogimport der kontinuierlichen Daten in Excel, am Beispiel der Wirbelschicht Daten

Hinweise: Der Text Import-Assistent ist ein Legacy Feature, das möglicherweise aktiviert werden muss. Wenn Sie dies noch nicht getan haben:

1. Klicken Sie auf **Datei > Optionen > Daten**.
2. Wählen Sie unter **Assistenten für Legacy-Datenimport anzeigen** die Option **aus Text (Legacy) aus**.

(b) Legacy Import, gemäß Microsoft

A	B	C	D	E
3 Datum	27.02.2021		Gruppen ID	
4				
5 Konzentration c1 in mg/l	c1	Konzentration c1 in NTU	ntu1	
6 Konzentration c2 in mg/l	c3	Konzentration c2 in NTU	ntu2	
7 Konzentration c3 in mg/l	c3	Konzentration c3 in NTU	ntu3	
8 Konzentration c4 in mg/l	c4	Konzentration c4 in NTU	ntu4	
9				
10 Trübung der Suspension in NTU				
11				
12 Trübungswert 1 in NTU	ntu11			
13				
14 Trübungswert 2 in NTU	ntu22			
15				
16 Trübungswert 3 in NTU	ntu33			
17				
18				
19 Massebestimmung der Nutsche und Filterkuchen				
20				
21 Masse Nutsche 1 in g (tara)	mn1			
22 Nutsche+Filterkuchen 1 in g	nf1			
23				
24 Masse Nutsche 2 in g (tara)	mn2			
25 Nutsche+Filterkuchen 2 in g	nf2			
26				

(c) Turbidimeter Datalogimport in Excel

Abbildung 59: Datalogs und Excel troubleshooting

4.4.1 Fazit

Die implementation eines Cloudservice (HAW Cloud) ist im Verlauf des Projekts nicht erfolgt. Die HAW Richtlinien scheinen es **derzeit** nicht zu gestatten, dass Kennungen an nicht humane Entitäten bzw. nicht natürliche Personen vergeben werden. Nicht humane Entitäten könnten die folgenden sein:

- Geräte
 - HMI's
 - Sensoren
- Aggregate
- Maschinen

Die Idee, dass jeder Laboraccount eine eigene Kennung erhält, wodurch die Laborgeräte wie HMI's ein Cloudverzeichnis auf der HAW Cloud erhalten würden, ist derzeit nicht realisierbar. Die Idee, dass nicht humane Entitäten eine Kennung erhalten können, sollte in höheren Instanzen der HAW diskutiert werden, damit zukünftige Cloudlösungen HAW global realisiert werden können. Die Hochschule sollte das Ziel haben die Privatwirtschaft abzubilden. Das nicht humane Entitäten ein eigenes Kommunikationsnetzwerk erhalten ist bereits Stand der Technik (siehe Abschnitt 2.4.9).

Es ist anzumerken, dass die Controls für die diskreten Eingaben keine Eingaberestriktionen besitzen und alles im Textformat gespeichert wird.

5 Praktische Teilumsetzung des Konzepts 3.0 an ausgewählten Versuchsständen

Gemäß der Aufgabenstellung wurde zu Beginn des Projekts eine Analyse durchgeführt, um herauszufinden wie viele Versuchsstände digitale Schnittstellen besitzen. In Folge dessen wurde der *Grad der Digitalisierung* definiert (Schnittstellen vorhanden, „effektive“ Schnittstelle; gemäß Tabelle 2) und ermittelt, wie hoch der prozentuale Anteil der MVT Versuchsstände ist, bei denen bereits Daten akquiriert werden können. Unter anderem war ein Ergebnis der Analyse, dass zwei Versuchsstände (Wirbelschicht und Filterkuchen), mit digitaler Messtechnik bzw. Sensoren aufgerüstet werden sollen. In den folgenden Abschnitten wird die praktische Modifikation der Versuchsstände erläutert.

5.1 Filterkuchenversuchsanlage

Im Rahmen dieser Arbeit wurde die Filterkuchenversuchsanlage (siehe Abbildung 60a) aufgerüstet. Das Fließschema der Anlage zu Projektbeginn ist der Abbildung 60b zu entnehmen. Bei diesem Versuchsstand gibt es zwei Absperrhähne. Der erste Absperrhahn (H1) verriegelt das Leitungssystem gegen die Vacuumpumpe (A), somit kann ein Unterdruck während Versuchsvorbereitungen oder umbauten aufgebaut werden. Das Manometer (B) ist in der Vacuumpumpe integriert. Der zweite Absperrhahn (H2) verriegelt den Trübebehälter (J), indem ein Rührer (K) das Aussedimentieren der Suspension unterbindet. Unter dem Trübebehälter befinden sich Kuststoffklemmen (G) und Gummiringe, um die Filternutschen (I) zu fixieren und abzudichten. Die Filternutschen werden mit dem Filtratauffangbehälter (D) verbunden, welcher sich bei Versuchsstillstand auf einer Schutzplatte (E) befindet, um die Digitalwaage (F) vor einer Dauerlast zu bewahren. Des Weiteren schützt die Schutzplatte die Waage bei Versuchsumbauten. Vor dem Starten eines Versuchsdurchlaufs ist die Schutzplatte (E) zu entfernen und der Filtratauffangbehälter (D) auf der Waage zu positionieren. Zwischen der Vacuumpumpe (Richtung A) und dem Filtratauffangbehälter (D) befindet sich ein Pufferbehälter, um dem Pufferbehälter vor einer potentiellen Suspensionskontamination zu bewahren. Die Filterkuchenversuchsanlage wird im Rahmen der Präsenzveranstaltungen isobar betrieben. Die Gleichungen zur Berechnung der Filtrationsanlage sind dem Abschnitt 2.2.1 zu entnehmen. Die linearisierte Filtrationsgleichung, auf der Grundlage der Darcy-Gleichung, ist nachfolgend aufgeführt:

$$\frac{t}{V(t)} = \underbrace{\frac{\eta \cdot \alpha_V \cdot \kappa}{A^2 \cdot 2 \cdot \Delta p_{\text{konst,irr}}}}_{a_1} \cdot V(t) + \underbrace{\frac{\eta \cdot \beta}{A \cdot \Delta p_{\text{konst,irr}}}}_{a_0} \quad (59)$$

Die zu bestimmenden Parameter sind der folgenden Auflistung zu entnehmen:

- **diskret**

- a0
 - die Viskosität η
 - der Filtermittelwiderstand β
 - die Filtrationsfläche A
- a1
 - der Filterkuchenwiderstand α_V
 - die Proportionalitätskonstante κ
 - die Filtrationsfläche A
 - die Viskosität η
- die Fluidehdichte ρ_f
- die Feststoffdichte ρ_{fs}

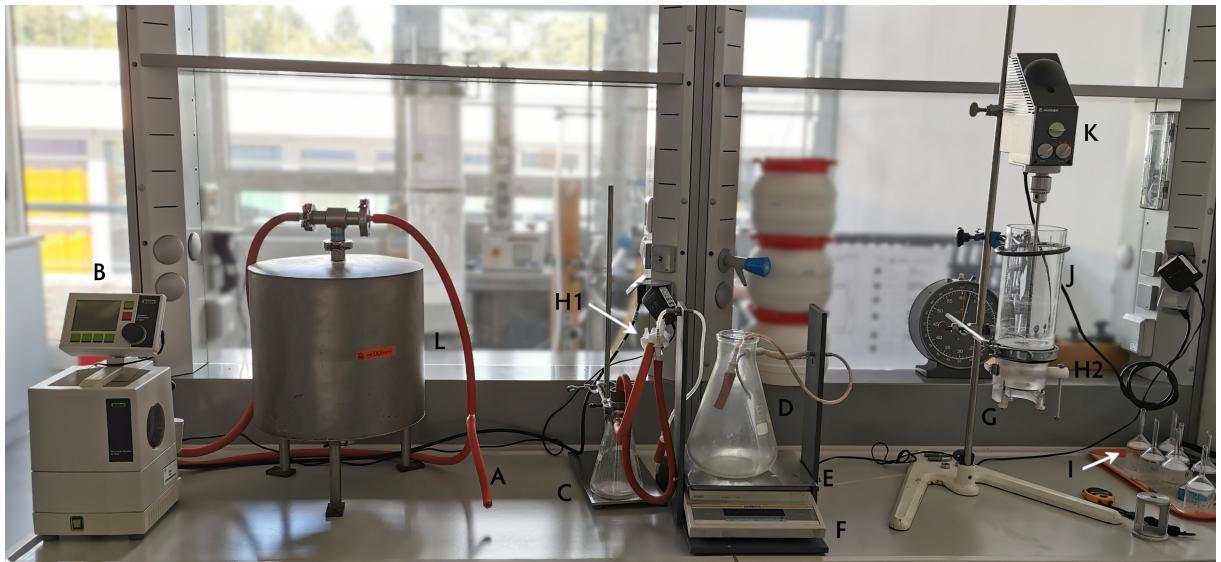
- **kontinuierlich**

- der irreversible Druckverlust $\Delta p_{\text{konst,irr}}$
- das Volumen als Funktion der Zeit $V(t) = \frac{m_{\text{Filtrat}}(t)}{\rho_f}$

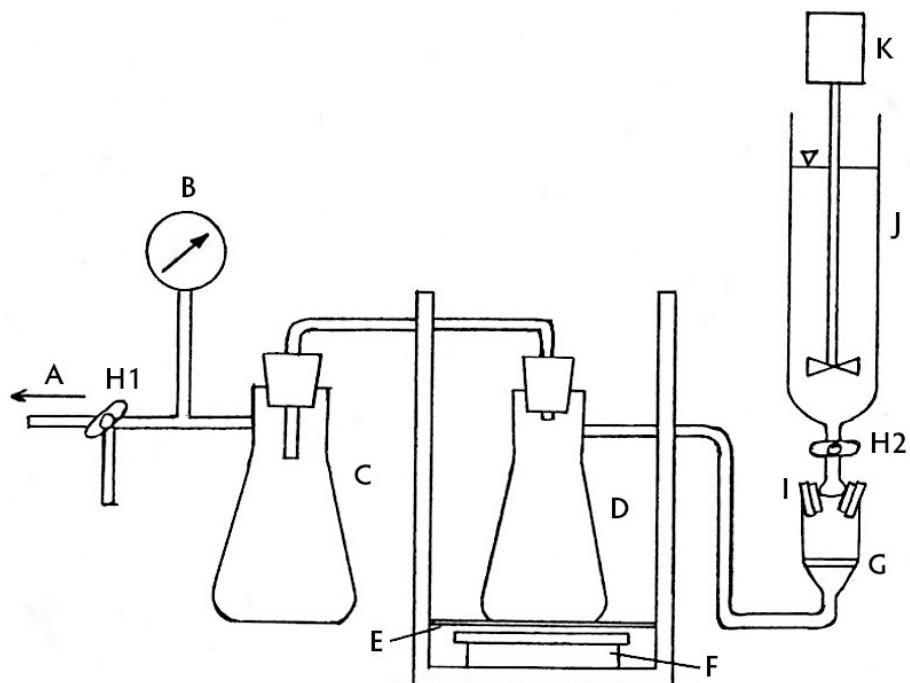
Das Filtrationsvolumen, als Funktion der Zeit, wird über die zusammengesetzte, extensive Zustandsgröße – der Filtratmasse $m_{\text{Filtrat}}(t)$ – bestimmt.

5.1.1 Sensorauswahl

Im Betrieb der Filterkuchenversuchsanlage sind die Parameter, Volumenstrom \dot{V} und Druck p , kontinuierlich zu erfassen. Die Masse des Filtrats wird von einer Laborwaage des Unternehmens Sartorius AG des Typs Practum S101-1S detektiert. Die Waage hat eine USB Schnittstelle, die mit einem USB Kabel



(a) Filterkuchenversuchsstand vor den Modifikationen



(b) Fließschema in Form einer schematischen Skizze der Filterkuchenversuchsanlage vor den Modifikationen [11]

Abbildung 60: Filterkuchenversuchsanlage nach den Modifikationen

eine RS-232 Schnittstelle emuliert. Dafür sind, gemäß der Bedienungsanweisung der Waage, die Einstellung mit den gewünschten Schnittstellenparametern vorzunehmen. Im Rahmen dieser Arbeit werden die Schnittstellenparameter gewählt, die in der Tabelle 11 **schwarz** hinterlegt sind. Die Auswahl der Parameter erfolgt willkürlich. Die Übertragungsgeschwindigkeit (1200 Bd) wurde hinreichend niedrig gewählt.

Es haben sich zwei Drucksensor technologien etabliert, der dehnungs resistive Drucksensor und der piezoresistive Drucksensor. Im Anhang ist eine Vergleichstabelle (siehe Tabelle 78), die beide Sensortypen verschiedener Hersteller und deren Spezifikationen auflistet. Gemäß der Tabelle ist zu entnehmen, dass piezoresistive Sensoren kleinere Dehnungen genauer Druckdifferenzen erfassen können. Die zu erwartenden Druckdifferenzen am Versuchsstand sind gering, daher wird ein piezoresistiver Relativdrucksensor des Unternehmens B+B Thermo-Technik GmbH des Typs DRTR-AL-10V-RV1 gewählt. Der Sensor ist im verfahrenstechnischem Labor ein häufig genutztes Messinstrument und ist im Verlauf dieser Arbeit bereits vorrätig gewesen, daher findet ein finanzieller Aufwandsvergleich im Rahmen dieser Arbeit nicht statt. Die Messspanne dieses Sensors ist von -1 bis 1 bar. Die Spanne des analogen Ausgangssignals ist von 0 bis 10 V. Die technischen Daten sind der Tabelle 77 im Anhang zu entnehmen.

Tabelle 11: RS-232 Schnittstellenparameter der Practum 5101-1S Waage

	6 Bit	7 Bit	8 Bit
Startbit	1	1	1
Symbol-/Characterbit	6	7	8
Paritätsbit	Odd/Even/Mark/Space		none
Stoppbit	2	1	0
Baudrate	1200		

5.1.2 Elektrotechnik der Filterkuchenversuchsanlage

Die Sensorik ist elektrotechnisch zu verschalten, die genutzten Komponenten sind:

- als Spannungsquelle ein Transformator des Typs Voltcraft TOPS-3205
 - Conrad Electronic AG
- ein Drucksensor des Typs DRTR-AL-10V-RV1
 - B+B Thermo-Technik GmbH
- eine Laborwaage des Typs Practum 5101-1S
 - Unternehmens Sartorius AG
- eine DAQ Messkarte (*engl. data acquisition*) des Typs USB-6001
 - National Instruments (NI)

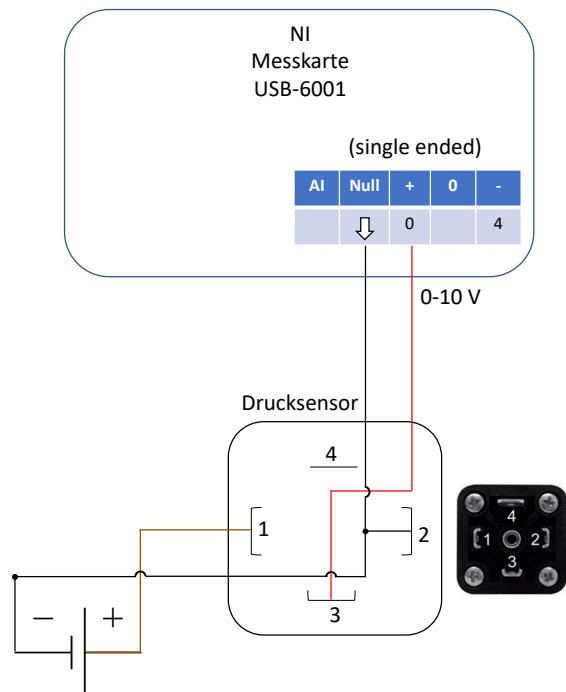


Abbildung 61: Elektrotechnische Verschaltung der Komponenten der Filterkuchenversuchsanlage

Die Elektrotechnische Verschaltung des Drucksensors ist der Abbildung 61 zu entnehmen. Die Laborwaage des Typs Practum S101-1S wird, via mitgelieferten USB-Y zu RS-232 Schnittstellenemulationsprozessorstecker Kabel, mit dem Desktop PC verbunden.

5.1.3 Filterkuchenversuchsanlage nach den Modifikationen

In der Abbildung 62a ist die gesamte Filterkuchenversuchsanlage, nach dem Aufrüsten mit digitaler Sensorik zu erkennen. Die Grafik des Schlauchleitungs- und Gerätefließschemas ist der Abbildung 62b zu entnehmen. Der Versuchsaufbau wurde um eine Spannungsquelle (N) für den Drucksensor (M), einen Drucksensor (M), sowie ein Human-Machine-Interface (HMI, O) erweitert. Als HMI dient ein Desktop PC. Das Manometer zur Druckmessung bildet mit der Vacuumpumpe (B) eine Einheit.

5.1.4 Workflow der Filterkuchenversuchsanlagen

In diesem Abschnitt wird der neue Workflow beschrieben. Die Datalogs werden derzeit in dem Data-Logordner auf dem Desktop gespeichert. Der Tabelle 12 sind die Bezeichnungen der Komponenten zu entnehmen.

1. Starten des HMI

- Inbetriebnahme der Spannungsquelle
 - i Hauptschalter auf der Rückseite betätigen
 - ii Spannungshöhe zwischen 15 und 24 V einstellen
 - iii On-button auf der Vorderseite betätigen ⇒ Spannung stellt sich auf den gewählten Betrag ein
- Waagensignalleitung ist mit der RS-232 Schnittstelle des HMI Towers zu verbinden.
- DAQ ist mit einem beliebigen USB Slot des HMI Towers zu verbinden
- Starten der LabVIEW Applikation `Filterkuchenversuch_V01DL.vi`
- Dateinamen eingeben: `FK_GruppenID.txt`
 - Der Dateiname kann über den Versuchstag identisch bleiben, die Daten werden an das Dateiende angehängt.
- Tubidimeterdateinamen eingeben: `FK_Tubi_GruppenID.txt`

- Der Dateiname kann über den Versuchstag identisch bleiben, die Daten werden pro Versuchsiteration überschrieben.
- Versuchseingaben tätigen
- wenn alle Vorbereitung der Filtrationseinrichtung **abgeschlossen** sind, dann
 - I Start des Programms: Das ⇒ Icon oben links
 - II öffnen des Absperrhahns H2
 - III wenn Messung abgeschlossen, dann
 - Stop betätigen
 - Filtratmenge bestimmen
 - Filterkuchendicke bestimmen
 - die trockene Filterkuchenmasse bestimmen
 - Tubidimeter eingaben tätigen
 - wenn abgeschlossen, starten der nächsten Iteration bei I
- IV **Nach der letzten Tubidimetereingabe ist das Programm nochmals zu starten und der Stop-button zu betätigen, damit die aktuellsten Tubidimeterdaten in der FK_Tubi_GroupenID.txt gespeichert werden!**

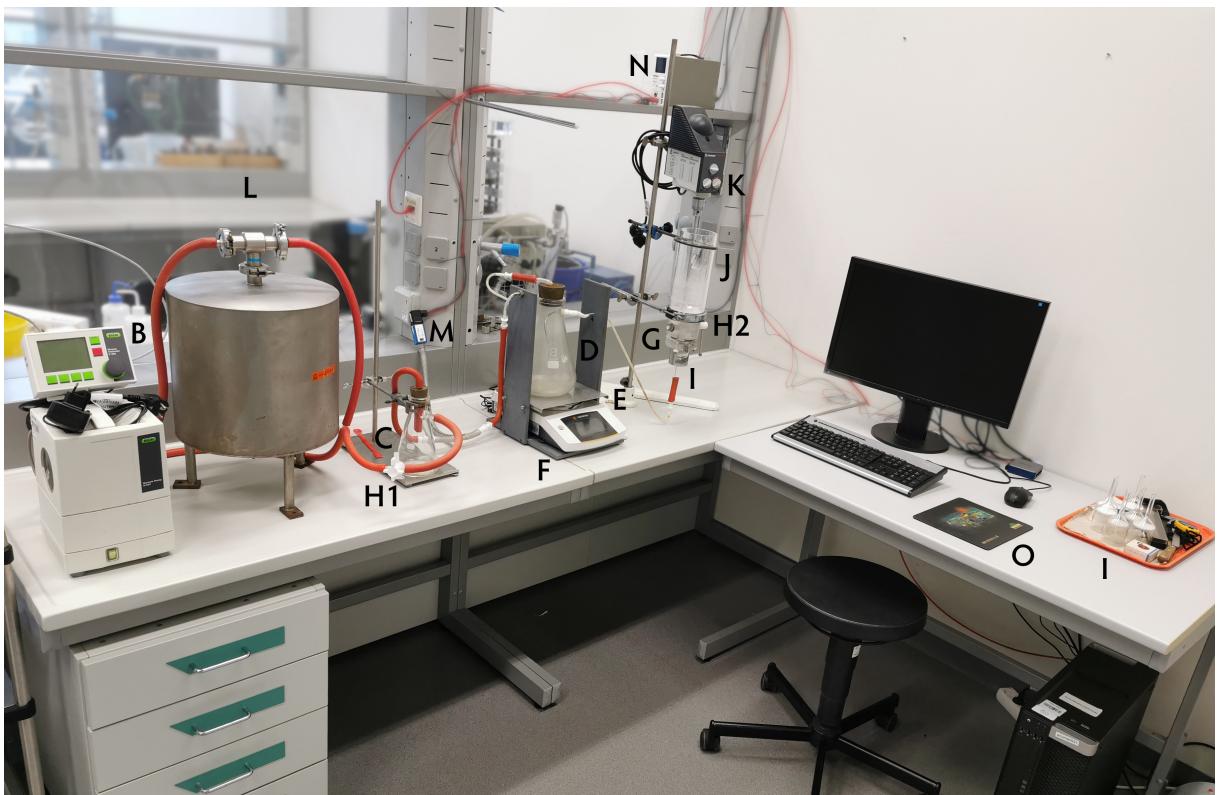
2. Vorbereitung der Filtrationseinrichtung

- a Absperrhahn H1 schließen
- b Filtrationsanlage evakuieren, dazu ist die *manuell* Taste des Vakuumcontrollers zu betätigen. Bei erreichen des Solldrucks wird der isobare Zustand der Anlage erhalten.
- c Gereinigte Filternutsche (l) in der Kunststoffklemmvorrichtung (G) fixieren.
- d Es ist darauf zu achten, dass der Metallsteg der Nutschenhalterung nach vorn zeigt
- e Absperrhahn H2 schließen (Hebelstellung: waagerecht).
- f Schläuche sachgemäß montieren.
- g Schutzplatte der Waage entfernen und daraufhin einschalten.
- h Einfüllen der Suspension und Inbetriebnahme des Rührers.

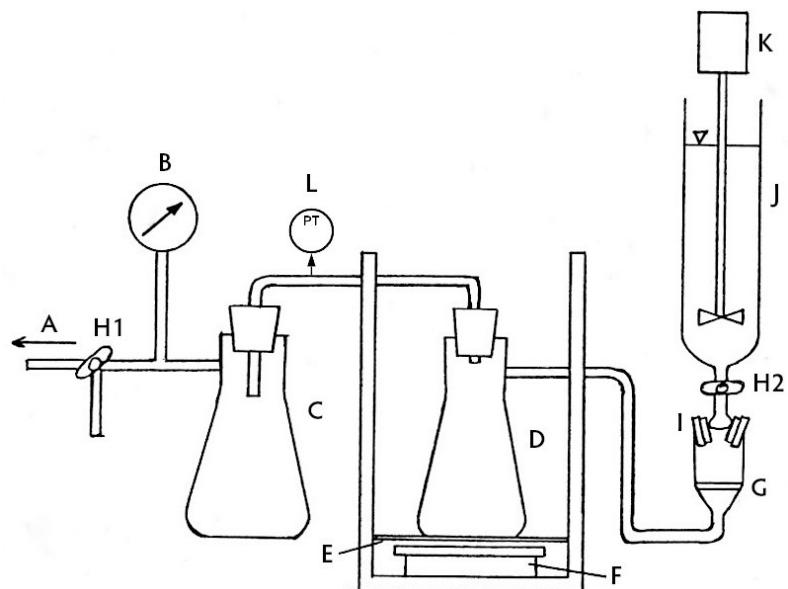
- i Öffnen des Dreiwegehahns H1
- j Wenn alle Vorbereitungen abgeschlossen sind und die angezeigte Masse Konstant ist, ist die Waage zu tarieren.

Tabelle 12: Bezeichnungen der Filterkuchenversuchsanlage

A	Verbindung zur Vacuumpumpe	H2	Absperrhahn
B	Vacuummanometer	I	Filternutsche
C	Rezipient und Wasserabscheider	J	Trübebehälter
D	Filtratauffangbehälter	K	Rührer
E	Schutzplatte für Waage	L	Pufferbehälter
F	Waage mit digitaler Schnittstelle	M	Drucksensor
G	Kunststoffklemmvorrichtung – Gummidichtung	N	Spannungsquelle
H1	Dreiwegehahn	O	HMI (Human-Machine-Interface)



(a) Filterkuchenversuchsanlage nach den Modifikationen



(b) Fließschema in Form einer schematischen Skizze der Filterkuchenversuchsanlage nach den Modifikationen [11]

Abbildung 62: Filterkuchenversuchsanlage nach dem Aufrüsten mit digitaler Sensorik

5.2 Wirbelschichtversuchsanlage

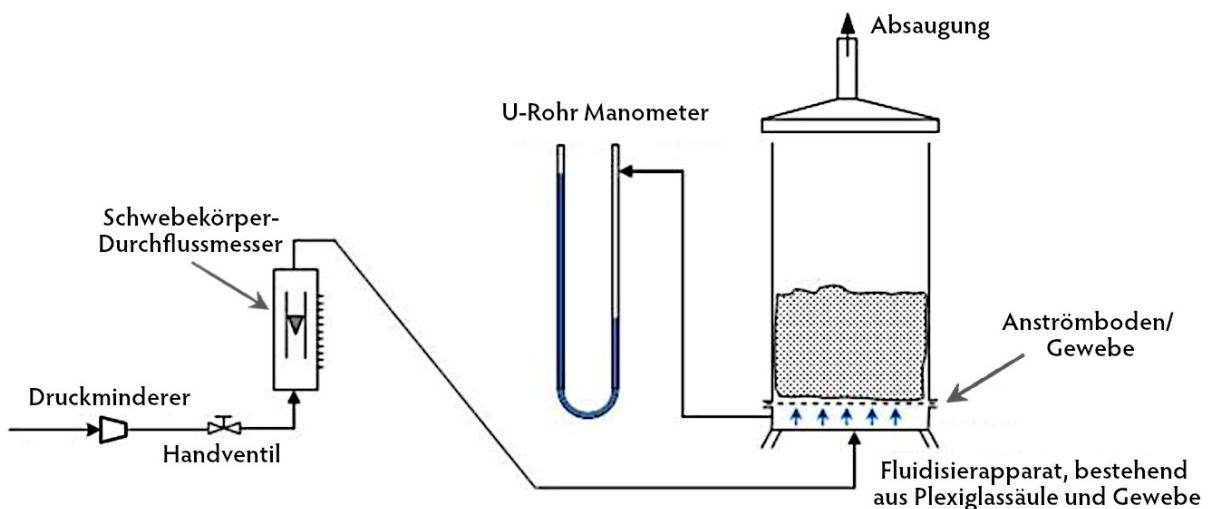
Das verfahrenstechnische Labor der HAW-Hamburg Life Sciences besitzt eine Wirbelschichtversuchsanlage im Labormaßstab (siehe Abbildung 63a). In der Abbildung 63b ist eine Versuchsskizze des Versuchsstands dargestellt. Der gesamten Anlage ist ein Druckminderer vorgeschaltet (A). Um den Volumenstrom zu Steuern, befindet sich nach dem Druckminderer ein Handventil (B). Zur Detektion des Volumenstroms sind zwei Schwebekörperdurchflussmesser (SKDM) (C) in Reihe nachgeschaltet, um einen Volumenstrom von bis zu 50 l/min messen zu können. Mit dem SKDM auf der rechten Seite kann der Messbereich von 0,185 bis 5 l/min gemessen werden. Mit dem SKDM auf der linken Seite sind Volumenströme 1,86 bis 50 l/min messbar. Vor dem Anströmboden muss ein Druck gemessen werden, dafür befindet sich am Boden des Fluidisierapparats (E) ein kleiner Flansch, mit dem der Druck relativ zur Umgebung mittels U-Rohr Manometer (D) gemessen werden kann. Der Fluidisierapparat besteht aus einer Plexiglassäule und dem Boden, in dem ein Gewebefilter eingespannt ist. Da Partikel mit dem Volumenstrom ausgetragen werden können, befindet sich über dem Fluidisierapparat eine Absaugung (F).

Bei diesem Versuch sind im Verlauf der Versuchsdurchführung die folgenden Parameter zu erfassen:

- diskret
 - Festbettbetthöhe h_{fb}
 - Festbettporosität ε_{fb}
 - Schüttgutmasse $m_{Schüttgut}$
 - Partikeldichte ρ_{fs}
 - Sauterdurchmesser
- kontinuierlich
 - \dot{V}_{Luft}
 - $\Delta p(t)$
- kontinuierlich/diskret
 - Wirbelschichthöhe $h_{ws} = f(p)$
 - Wirbelschichtporosität $\varepsilon_{ws} = f(p)$



(a) Wirbelschichtversuchsanlage vor den Modifikationen



(b) Schematische Skizze des Wirbelschichtversuchs vor den Modifikationen

Abbildung 63: Wirbelschichtversuchsanlage der HAW Hamburg der Fakultät Life Sciences, vor den Modifikationen

Es ist hinreichend, die Wirbelschichthöhe und die daraus resultierende Porösität diskret zu erfassen. Im Rahmen dieser Arbeit wird die diskrete, manuelle Erfassung gewählt. Im Anhang befindet sich ein Diagramme für den SKDM der Messspanne von 0,185 bis 5 l/min (Abbildung 84) und ein Diagramm für den SKDM der Messspanne von 1,86 bis 50 l/min (siehe Abbildung 85). Beide Diagramme enthalten Kennlinien der SKDM zzgl. jeweils zwei Approximationsfunktionen. Des Weiteren befindet sich eine Manometerkennlinie, siehe Abbildung 86.

5.2.1 Sensorauswahl

Im Betrieb der Anlage sind die kontinuierlichen Parameter Volumenstrom \dot{V} und Druck p messtechnisch zu erfassen. Die zu erfassenen Druckdifferenzen bei diesem Versuchsstand sind ebenfalls gering, daher wird ein piezoresistiver Relativdrucksensor des Unternehmens B+B Thermo-Technik GmbH des Typs DRTR-AL-10V-RV1 gewählt. Die Messspanne dieses Sensors ist von -1 - 1 bar. Die Spanne des analogen Ausgangssignals ist von 0 - 10 V. Die Technischen Daten sind der Tabelle 77 im Anhang zu entnehmen.

Gemäß Abschnitt 2.1.1 sind diverse Durchflussensoren geeignet, um den Luftvolumenstrom zu erfassen. Differenzdrucksensoren sind für kleine Messbereiche nicht geeignet. Gemäß der Randbedingung des Versuchs, sind die folgenden Drucksensoren potentiell möglich.

- Vortexsensor
- Drallsensor
- Coriolissensor
- thermischer Massendurchflussensor

Da vor einer Kostenanalyse der Sensoren eine Anschaffung eines thermischen Massendurchflusssensor des Typs VA-525 des Unternehmens CS Instruments GmbH & Co. KG mit einer Messspanne von 0,02 - 50 l/min erfolgt ist, wird ein Vergleich der Randbedingungen und des finanziellen Aufwands im Rahmen dieser Arbeit nicht erfolgen. Das Kalibrierzertifikat des Durchflusssensors ist dem Anhang zu entnehmen (siehe Abbildung 79).

5.2.2 Elektrotechnik der Wirbelschichtversuchsanlage

Die Sensorik ist elektrotechnisch zu verschalten, die genutzten Komponenten sind:

- als Spannungsquelle ein Transformator des Typs Voltcraft TOPS-3205
 - Conrad Electronic AG
- ein Strom- Spannungswandler des Typs WAA 7-0541
 - Friedrich Lütze GmbH & Co. KG
- ein Drucksensor des Typs DRTR-AL-10V-RV1
 - B+B Thermo-Technik GmbH
- ein thermischer Massendurchflusssensor des Typs VA-525
 - CS Instruments GmbH & Co. KG
- eine DAQ Messkarte (*engl. data acquisition*) des Typs USB-6001
 - National Instruments (NI)

In der Abbildung 64 ist ein Schema der elektrotechnischen Verschaltung dargestellt. Der Volumenstromsensor generiert analoge Messwertsignale der Spanne 4 bis 20 mA. Die DAQ Messkarte kann analoge Signale der Spanne 0 bis 10 V in ein digitales Signal umwandeln, daher ist zwischen dem Durchflusssensor und dem DAQ ein Strom- Spannungswandler zu implementieren. Der Stromspannungswandler benötigt eine Spannungsquelle (Klemmen 1 und 6). Die analogen Signale des Volumenstromsensors (1) und (3) werden am Stromspannungswandler an (2) und (3) angeklemmt. Die umgewandelten Signale treten aus den Klemmen (4) und (5) des Wandler aus. **Dieses Signal ist an einem DAQ nun differenziell zu verschalten.** Das minus (-) Signal wird an der Messkarte demnach nicht über Null (*engl. Ground*), sondern an **minus** angeklemmt.

Der Drucksensor generiert analoge Messsignale der Spanne 0 bis 10 V. Die Ausgänge des Sensors (2) und (3) sind mit Null und plus (+) zu verschalten. Es ist anzumerken, dass die Slots im DAQ willkürlich so gewählt wurden. Sollten die Slots gewechselt werden, dann muss es im Blockdiagramm des Hauptprogramms angepasst werden.

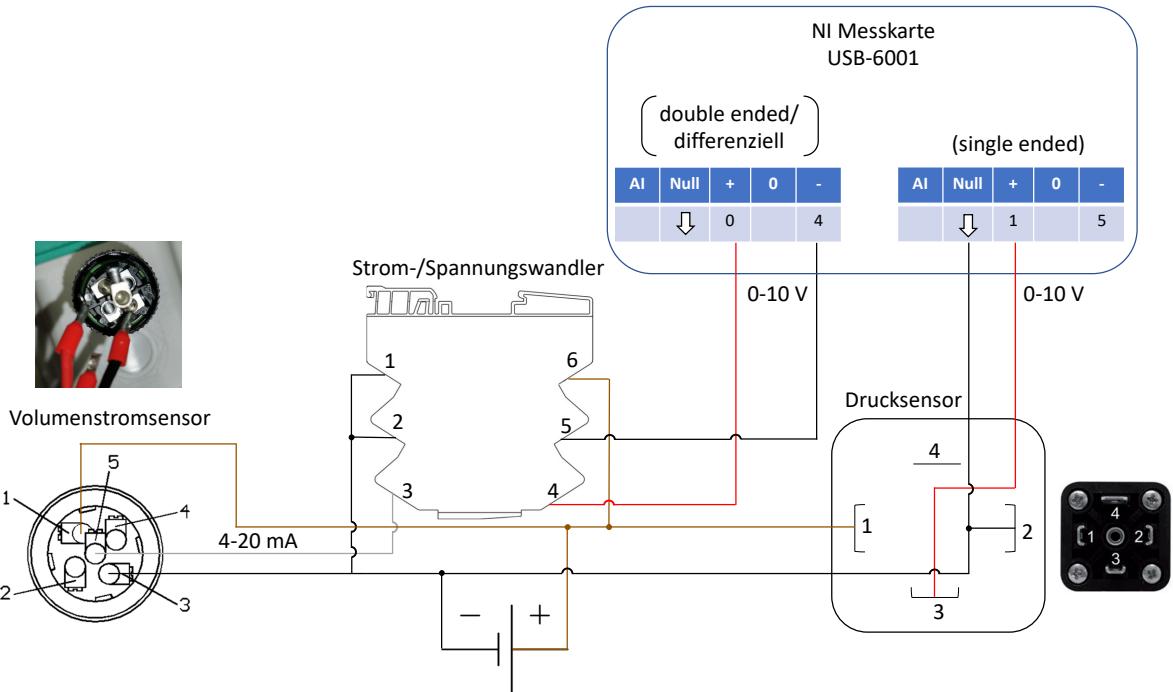


Abbildung 64: Elektrotechnische Verschaltung der Komponenten der Wirbelschichtversuchsanlage

5.2.3 Wirbelschichtversuchsanlage nach den Modifikationen

In der Abbildung 65a ist die Wirbelschichtversuchsanlage nach der Implementation des Druck und Volumenstromsensors zu erkennen. In der Abbildung 65b ist das Fließschema der aufgerüsteten Wirbelschichtversuchsanlage abgebildet. Der Tabelle 13 sind die Bezeichnungen der Komponenten, der Wirbelschichtanlage nach den Modifikationen, zu entnehmen. Dem Druckminderer (A) ist das Handventil (B), zur manuellen Volumenstromsteuerung, nachgeschaltet. Nach dem SKDM (C) ist der Volumenstromsensor (D) des Typs VA-525 nachgeschaltet. Nach dem Volumenstromsensor wurde ein Drucksensor (E) des Typs DRTR-AL-10V-RV1, wie auch der Volumenstromsensor, als diversitäre Redundanz implementiert. Der Drucksensor, wie auch das Manometer (F), misst den Druck vor dem Anströmboden. Über dem Fluidisertopf (H) ist eine Absaugung (I) positioniert. Wie der Abbildung 65a zu entnehmen, ist es möglich eine weitere Absaugung über dem HMI (L) zu positionieren. Des Weiteren sind weitere messtechnische Komponenten hinzugekommen. Als DAQ Messkarte wird ein Gerät des Unternehmens National Instruments des Typs USB-6001 (K) verwendet. Der Strom-/Spannungswandler (G) ist an der Trennwand befestigt. Als Spannungsversorgung für den Volumenstromsensor (D), dem Drucksensor (E), dem Strom-/Spannungswandler (G) und dem DAQ (K), wird ein Transformator (J) des Typs TOPS-3205 von Voltcraft verwendet.

Tabelle 13: Bezeichnungen der Wirbelschichtversuchsanlage nach den Modifikationen

A	Druckminderer	H	Fluidisiertopf
B	Handventil		– Plexiglassäule
C	Schwebekörperdurchflussmesser		– Gewebefilter als Anströmboden
D	thermischer Durchflusssensor	I	Absaugungen
E	piezoresistiver Drucksensor	J	Spannungsquelle
F	U-Rohr Manometer	K	NI DAQ_USB-6001
G	Strom-/Spannungswandler	L	HMI (Human-Machine-Interface)

5.2.4 Workflow der Wirbelschichtanlage

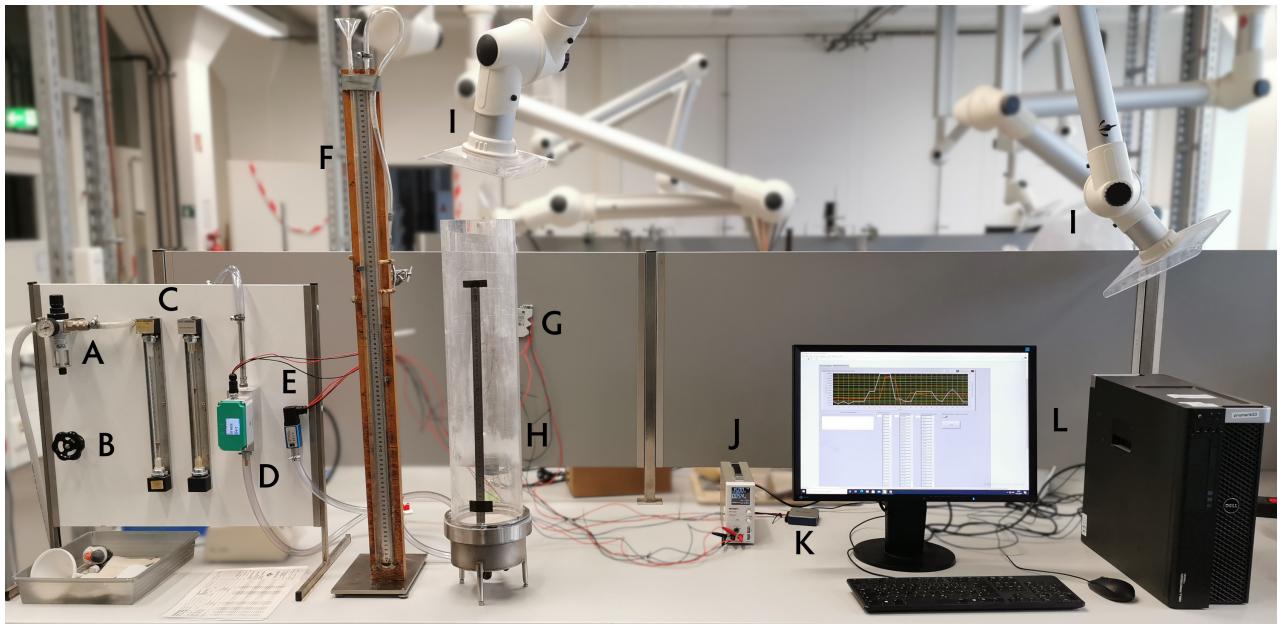
In diesem Abschnitt wird der neue Workflow der Wirbelschichtanlage beschrieben. Die Datalog-Datei wird derzeit ebenfalls in dem Datalogordner auf dem Desktop gespeichert.

1. Starten des HMI

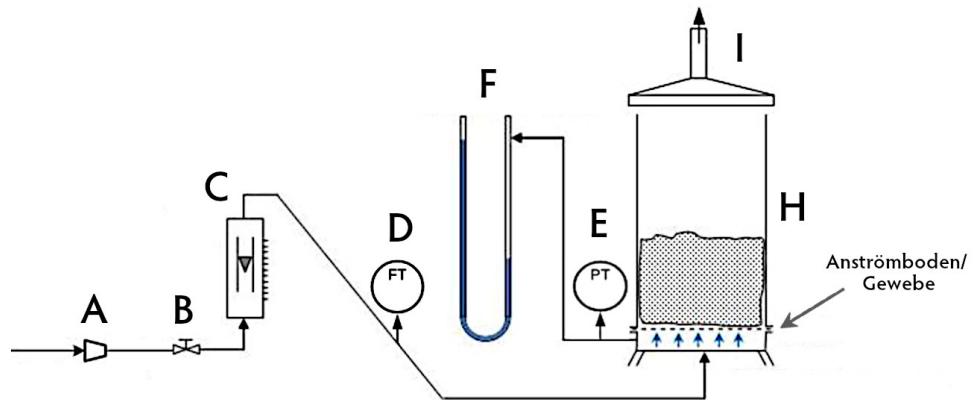
- I Inbetriebnahme der Spannungsquelle
 - i Hauptschalter auf der Rückseite betätigen
 - ii Spannungshöhe zwischen 15 und 24 V einstellen
 - iii On-button auf der Vorderseite betätigen ⇒ Spannung stellt sich auf den gewählten Betrag ein
- II DAQ ist mit einem beliebigen USB Slot des HMI Towers zu verbinden.
- III Starten der LabVIEW Applikation `Wirbelschichtversuch_V01DL.vi`
- IV Datalognamen eingeben: `WS_GroupenID.txt`
 - Der Dateiname kann über den Versuchstag identisch bleiben, die Daten werden an das Dateiende angehängt
- V Versuchseingaben tätigen
- VI Start des Programms: Das ⇒ Icon oben links

2. Praktische Versuchsdurchführung

- Druckleitung öffnen
- Handventil zur manuellen Steuerung des Volumenstroms nutzen



(a) Wirbelschichtversuchsanlage nach den Modifikationen



(b) Fließschema in Form einer schematischen Skizze der Wirbelschichtversuchsanlage nach den Modifikationen

Abbildung 65: Wirbelschichtversuchsanlage nach dem Aufrüsten mit digitaler Sensorik

5.2.5 Fazit

Der Versuchsstand wurde mit geeigneter Sensorik aufgerüstet, um die kontinuierlichen Parameter, Druck p und Volumenstrom \dot{V} , zu erfassen. Die Signale des Drucksensors sind von 0 bis 10 V und die des thermischen Massendurchflusssensors sind von 4 bis 20 mA. Eine digitale Erfassung der Festbett- bzw.

Wirbelschichthöhe ist denkbar. Bei bedarf könnten bspw. geeignete optische, akustische, mechanische Entfernungsmesser, im Rahmen einer studentischen Projektarbeit, recherchiert und umgesetzt werden. Des Weiteren könnte die Steuerung des Volumenstroms, derzeit per Handventil, durch eine digitale Steuerung ersetzt werden. Der derzeitige Volumenstromsensor (VA-525) könnte entfernt und für andere Versuchsaufbauten verwendet werden.

6 Fazit und Vision

Im Verlauf des Projekts wurden Informationen aus den verschiedensten Bereichen zusammengetragen und in diesem Dokument destilliert. In den folgenden Abschnitten werden nochmals die Schlussbemerkungen eines jeden Abschnitts zusammengefasst.

6.1 Bilanz in Bezug auf die Ist-Analyse zu Beginn des Projekts

Zum Ende dieses Projekts soll diese Arbeit anhand der generierten Kennzahlen Tabelle 2, aus dem Abschnitt 1.1, Bilanziert werden. Dafür wurde diese Tabelle aktualisiert. Das Resultat ist die Tabelle 14. Zu Beginn des Projekts hatte die Tabellenhälfte mit den MVT Versuchsständen, in der Spalte **priorisierte Einrichtungen**, einen prozentualen Anteil, relativ zu allen Spalteneinträgen, von 50 %. Beim Projektabschluss wurde dieser Wert auf 100 % angehoben. Folglich haben sich auch die prozentualen Anteile der Spalten **Sensorik vorhanden**; von 29 auf 43 %; und „**effektive Schnittstelle**“; von 50 auf 67 %; verändert. Die untere Tabellenhälfte, mit den MVT Analyseeinrichtungen, hat keine Veränderungen von sich getragen.

Tabelle 14: „Grad der Digitalisierung“ der MVT Geräte und Anlagen

MVT Versuchsstände	Sensorik zweckmäßig	Sensorik vorhanden	„effektive“ Schnittstelle	priorisierte Einrichtungen
Sichten				
Kuchenfiltration	○	✓	✓	✓
Blaine				
Staubabscheidung	○	✓	✓	✓
Wirbelschicht	○	✓	✓	✓
Backenbrecher				
Planetenkugelmühle	○	✓	✗	
Prallmühle				
Walzenmühle				
Schneidmühle				
Hochdruckhomogenisator				
Intensivmischer		✓	✗	
Conchiermaschine				
Rührversuch	○	✓	✓	✓
Summe aller ✓, ✓ oder ○	5	4	4	4
Summe aller ✗ oder ✗			2	2
prozentualer Anteil, relativ zu allen Einrichtungen Spalteneinträgen	36 %	43 %	67 %	100 %
MVT Analyseeinrichtungen	Sensorik zweckmäßig	Sensorik vorhanden	„effektive“ Schnittstelle	priorisierte Einrichtungen
Zugversuch	○	✓	✓	
Siebturmanalyse zzgl. digital Waage	○	✓	✗	✗
Luftstrahlsiebung	○	✓	✓	✓
Scherzelle	○	✓	✗	✗
Stampfdichte				
Laserbeugungs- Partikelgrößenanalyse	○	✓	✓	✓
Summe aller ✓ oder ○	5	4	2	2
Summe aller ✗ oder ✗			2	2
prozentualer Anteil, relativ zu allen Einrichtungen Spalteneinträgen	83 %	83 %	60 %	50 %

6.2 Fazit und Vision des Kapitels Konzeptentwicklung

Das verfahrenstechnische Labor soll zeitgemäß, digital transformiert werden. An der Stelle muss nochmals erwähnt werden, dass eine digitale Transformation ein Prozess und kein Projekt ist. Das Konzept 3.0, ohne Datenbankmanagementsystem (DBMS), sollte als Vorstufe des Konzept 4.0, mit DBMS, betrachtet werden. Das Konzept 3.0 ist eine Mindestanforderung für die digitale Transformation des verfahrenstechnischen Labors. Im folgenden werden die Wünsche und Anforderung aus dem Kapitel 3 Konzeptentwicklung für die Labordigitalisierung aufgelistet. Die erfüllten Anforderungen werden mit einem Haken (✓) und die Anforderungen, die im Rahmen dieses Projekts nicht umgesetzt wurden, mit einem Kreuz (✗) markiert. Stichpunkte, die Anmerkungen enthalten, werden mit einem ◇ gelabelt.

- ✓ Das Konzept soll für das verfahrenstechnische Labor allgemeingültig sein.
- ✓ Die „digitalen Kompetenzen“ der Studierenden sollen maximiert werden, ohne die fachlichen Kompetenzen signifikant zu reduzieren, durch
 - ✓ ein Minimum an Automatisierung
 - ✓ Unter Automatisierung kann z.B. die Manipulation von Daten, wie die Detektion von Ausreißer und dessen Entfernung, genaue Leitfäden in der Handhabung, die keine Fehler mehr zulassen o.ä., verstanden werden.
- ✓ Der monetäre Aufwand soll minimal sein.
- ✓ Möglichkeiten der Datenerfassung/Signalverarbeitung sollen eruiert werden.
- ✓ Die inkrementelle Implementation von Applikationen im Rahmen von Industrie 4.0 und Big Data (KI, Digital Twin etc.) soll tendenziell möglich sein.
 - ◇ Es wurden keine Schritte eingeleitet, die eine inkrementelle Implementation von Applikationen ausschließen.
- ✓ Cloud Computing soll in Betracht gezogen werden.
 - ◇ Die Implementation eines Cloudservice (HAW Cloud) ist im Verlauf des Projekts nicht erfolgt. Im Verlauf eines Telefonats mit einem ITSC-Mitarbeiter am 08.02.21, um ca. 11 Uhr, scheinen die HAW Richtlinien es **derzeit** nicht zu gestatten, dass Kennungen an nicht humane Entitäten bzw. nicht natürliche Personen vergeben werden (siehe Protokoll im Anhang 80 und 81). Die HAW-Cloud funktioniert nutzerbezogen, wofür Kennungen (des Typs axxXXX) erforderlich sind. Nicht humane Entitäten könnten die folgenden sein:

- Aggregate
- Maschinen
- Geräte
 - HMI's
 - Sensoren

✗ Erster Schritt der Umsetzung: Speicherung der Rohdaten auf der HAW Cloud

- ◊ Da die Nutzung der HAW Cloud möglich wäre, der Aufwand dem Nutzen jedoch nicht gerecht wird, wurde eine alternative Lösung entwickelt (siehe Anhang 80,81). Als Alternative wurde ein lokaler Server an der HAW Life Sciences, am Standort Bergedorf vorgeschlagen.

✗ Zweiter Schritt der Umsetzung: Cloudnutzung, gemäß des Impulsvortrags zum Digitalisierungsfond ⇒ Amazon, Google, Microsoft Azure

✗ Die gesamte Protokollierung der Studierenden soll in Zukunft digital sein.

- ◊ Der Fortschritt des Teilprojekts, unter der Leitung von Dipl.-Ing. M. Hannappel und Dipl.-Ing. S. Wittkowski, *Electronic Labor Notebook* hat noch keine vollständige Validierung des Konzepts 3.0 zugelassen.

Der *digitale Transformationsprozess* des verfahrenstechnischen Labors wurde mit dem Konzept 3.0 initialisiert. Die **Vision** sollte die inkrementelle Umsetzung des **Konzepts 4.0** sein (siehe Abbildung 27). Meines Erachtens nach unterliegt der Aufwand dem Nutzen der Realisierung des Konzepts 4.0 im hohen Maße, aus den folgenden Gründen.

Im akademischen Rahmen unterliegt der Aufwand, der Realisierung des Konzept 4.0, einen großen Mehrwert, in Bezug auf das Verständnis und Lernerfolg der Studierenden. Die Datenbank könnte auch für den Laborbetrieb des Departments genutzt werden, wodurch sich möglicherweise organisatorische Vereinfachungen, wie z.B. Chemie- und Verfahrenstechniklagerverwaltung o.ä. realisieren lassen. Des Weiteren könnte die Datenbank für die verfahrenstechnische Forschung, Projekte im Rahmen der angewandten numerische Simulation, uvm. genutzt werden. Die **größte Hürde** für die Konzept 4.0 Realisierung könnte der **Changeprozess** sein, daher wurde der Punkt Changeprozess, in der Tabelle 15 mit einem Dicken **Minus** (-) versehen.

In unserer agitierten Welt ist weder Platz für schwere Unternehmenstanker noch für träge Universitäten oder Hochschulen. In der Zeit der Globalisierung und digitalen Transformation ist Flexibilität und Vernetzung nicht nur eine hinreichende Forderung, sondern absolut notwendig. Unternehmen sollten sich Ihre Spitzenträger schaffen. Der Spruch, „von der Berufsausbildung oder dem Studium braucht man am Ende des Tages maximal 15 %“, ist nicht tolerierbar sowie zu verantworten und doch ist es aus eigenen Erfahrungen Realität. Wie lange Überlebt ein Unternehmen welches 85 % Ausschuss hat? Wenn durch diese Art des Paradigmenwechsels eine Ausschussreduktion auf bspw. 70 % erfolgt, ist das ein immenser Fortschritt! In der Tabelle 15 sind die Vor- und Nachteile des Konzepts 4.0 nochmals aufgelistet.

Im Rahmen der Anwendungsfälle im Hochschulalltag könnten SQL Datenbanken den Anforderungen genügen. Das Know-How mit dieser standardisierten Datenbanktechnologie ist auf andere Softwaretechnologien transferierbar. Es sollte auch möglich sein, die Erstellung einer Datenbank, im Rahmen studentischer Arbeiten, zu delegieren. Des Weiteren ist auch im Verlauf des *Digitalisierungsprozess* eine Hybridlösung denkbar. Datenbankmanagement könnte Morgen, in diesem *digitalen Zeitalter*, eine von Ingenieuren geforderte **Kernkompetenz** sein.

Tabelle 15: Vor- und Nachteile des Konzeptentwurfs 4.0

Vorteile	Nachteile
<ul style="list-style-type: none"> + Stand der Technik in der Industrie / Zeitgemäß + Nutzbar für Tätigkeiten außerhalb vom VT1 und VT2 Praktikum (Forschung, Organisatorisches wie) + Lagerverwaltung, angewandte numerische Simulation uvm.) + wenn Cloudservicenutzung, dann Portfolio-software Kompatibilität des Anbieters + Image/Professionalität (u.a. auch Anmerkungen von Bachelorabsolventen Ende 2020) + ggf. mehr Kooperationsprojekte mit Unternehmen und Universitäten + Kompetenzsteigerung der Hochschule und Absolventen + interdisziplinäre Vernetzung/Projekte (Industrie 4.0 ≡ Vernetzung) + gute Investigationsmöglichkeiten bei Produkt-abweichung bei konstanten Parametern → Studentenprojekte 	<ul style="list-style-type: none"> - Changeprozesse - notwendiges Know-How erforderlich - Aufwand der Erstellung der Datenbank - Einrichtung automatischer Einspeisung der Sensordaten in die Datenbank - Kosten könnten entstehen <ul style="list-style-type: none"> • möglicherweise Sponsoring/Kooperation (als Gegenleistung Projekte o.ä) da Hochschule • geringe Datenmengen im vgl. zu Unternehmen → möglicherweise keine oder geringe Kosten

6.3 Fazit und Vision des Kapitels Signalverarbeitung

Zu Beginn des Projekts wurden Anforderungen an die Signalverarbeitung definiert. Erfüllte Anforderungen werden ebenfalls mit einem Haken (✓) und nicht erfüllte mit einem (✗) versehen.

- ✓ Die Applikationen sollen eine graphische Oberfläche haben.
- ✓ Der Aufwand für die Applikationsgenerierung soll minimal sein.

- ✓ Die Applikationen sollen nur von ausgewählten Personal („leicht“) anpassbar sein.
 - ◊ Den wissenschaftlichen Mitarbeitern ist das Passwort bekannt (Entsperrungssequenz bspw. durch Druck der Kombination Strg+E zu erreichen)
- ✓ Die verwendete Software soll in der Lage sein, Daten der verschiedensten Versuchsstände und Geräte unterschiedlicher Hersteller abzubilden.
- ✓ Es sollen Echtzeitgraphen generiert werden können.
- ✓ Die verwendete Applikation soll kontinuierliche und diskrete Daten verarbeiten können.
- ✓ Die Möglichkeit, die Daten in Zukunft direkt in eine Datenbank schreiben zu können, ist wünschenswert.
 - ◊ Es ist möglich die Daten via LabVIEW direkt in eine Datenbank schreiben zu lassen.
- ✓ Im Verlauf des Projekts wurde ein Schreibschutz der Dateien nach Abschluss eines Versuchsdurchgangs erwünscht. Dieser Wunsch wurde erfüllt.
 - ◊ Das Kopieren der Daten aus der Datei und Löschen der Datei ist weiterhin möglich. Eine Manipulation kann derzeit durch den Speicherungszeitpunkt der **neuen** Datei und ob ein Schreibschutz bei der **neuen** Datei aktiviert wurde ermittelt werden.

Es wurde ein Programm mittels Python geschrieben und ist als .py Datei dem Anhang beigefügt. Es ist in der Lage eine RS-232 Waage auszulesen, die Werte zu interpretieren und einen Graphen zu plotten (siehe Abbildung 66).

Da die Implementation eines Cloud oder lokalen Servers nicht erfolgt ist, werden die Datalogs derzeit auf dem jeweiligen Laborrechner auf dem Desktop gespeichert. Der Dateipfad ist bei nachträglicher Implementation von Serverlösungen in den VI's, in der Initialisierung anzupassen (siehe Abbildung 52 im Abschnitt 4.3.4).

Die Idee, dass jeder Laboraccount eine eigene Kennung erhält, wodurch die Laborgeräte wie HMI's ein Cloudverzeichnis auf der HAW Cloud erhalten würden, ist derzeit nicht realisierbar. Die Idee, dass nicht humanoide Entitäten eine Kennung erhalten können, sollte in höheren Instanzen der HAW diskutiert werden, damit zukünftige Cloudlösungen HAW global realisiert werden können. Die Hochschule

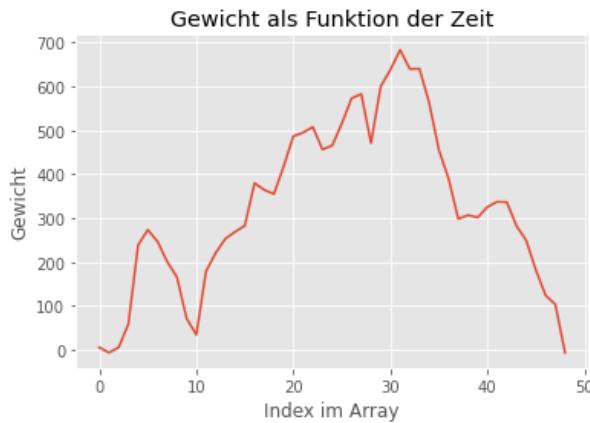


Abbildung 66: Gravimetrische Messung in Python

sollte das Ziel haben die Privatwirtschaft abzubilden. Das nicht humane Entitäten ein eigenes Kommunikationsnetzwerk erhalten ist bereits Stand der Technik (siehe Abschnitt 2.4.9).

Es ist anzumerken, dass die in LabVIEW geschriebenen Programme alle Daten in einer *.txt Datei als Strings speichern. Bei einem Import ist eine Umwandlung zu erfolgen (in bspw. Excel Zellen formattieren...). Es wird empfohlen den Studenten dass **nicht** mitzuteilen, damit ein autodidaktischer Lernprozess stattfinden kann.

6.4 Fazit und Vision der Teilumsetzung an den ausgewählten Versuchsanlagen

Die Filterkuchen- sowie die Wirbelschichtversuchsanlage wurde mit geeigneter Sensorik aufgerüstet, um die kontinuierlichen Parameter, Druck p und Volumenstrom \dot{V} , zu erfassen. Des Weiteren wurde neben dem Datalogging kontinuierlicher Parameter, das Datalogging diskreter Daten ermöglicht. Die Messspanne der Drucksensoren ist von -1 bis 1 bar relativ zur Atmosphäre; repräsentiert von der elektrischen Signalspanne von 0 bis 10 V; und die des thermischen Massendurchflusssensors an der Wirbelschichtversuchsanlage ist von 0 bis 50 l/min; repräsentiert durch die elektrische Signalspanne von 4 bis 20 mA.

Eine digitale Erfassung der Festbett- bzw. Wirbelschichthöhe an der **Wirbelschichtversuchsanlage** ist denkbar. Bei Bedarf könnten bspw. geeignete optische, akustische, mechanische Entfernungsmesser, im

Rahmen einer studentischen Projektarbeit, recherchiert und ggf. umgesetzt werden. Eine Implementation in die im Rahmen dieser Arbeit programmierten Applikation sollte komplikationslos möglich sein. Des Weiteren könnte die Steuerung des Volumenstroms, derzeit per Handventil, durch eine digitale Steuerung ersetzt werden. Der derzeitige Volumenstromsensor (VA-525) könnte dadurch substituiert und für andere Versuchsaufbauten verwendet werden.

Die **Filterkuchenversuchsanlage** hat eine umfangreichere Funktion der diskreten Datenaufnahme erhalten. Die Feststoffbestimmung im Filtrat kann manuell im LabVIEW Programm dokumentiert werden. Diese Messverfahren ist eine Vergleichsmessung mit der Bezeichnung *Tubidimetermessung*. Neben der Datalogdatei der kontinuierlichen Messung, die ebenfalls den Header enthält, analog der Wirbelschichtversuchapplikation, wird eine zweite Datei generiert. Die Funktionsweise der Datalogger werden in den folgenden Paragraphen nochmals zusammengefasst.

Kontinuierlicher Datalogger Bei beiden Anlagen werden kontinuierliche Dateien des Typs `FK_<Gruppen_ID>.txt` und `WS_<Gruppen_ID>.txt` generiert. Bei Starten einer Messung in der jeweiligen LabVIEW Applikation ist das Starticon ⇒ oben links im Front Panel zu betätigen. Es folgt ein Dialogfenster, wo der Dateiname des Typs, wie oben Angegeben, eingegeben werden soll. Ist vor dem Applikationsstart bereits die `Gruppen_ID` eingegeben worden, dann ist nur das suffix `.txt` zu ergänzen. Ist vor dem Applikationsstart bereit die `Gruppen_ID` eingegeben worden, dann ist nur das suffix `*.txt` zu ergänzen. Bei beiden Dataloggern wird im Falle einer redundanten Namensvergabe der Inhalt der nächsten Versuchsiteration unten dem Dokument angehängt. Ein Datenverlust ist ausgeschlossen. Im Verlauf des Projekts wurde sich ein Schreibschutz der Datalog-Datei gewünscht. Die LabVIEW Applikation kann die Verschlüsselung wieder öffnen das Logging am Dokumentende starten und am Ende wieder Verschlüsseln.

Diskreter Datalogger Die Applikation der Filterkuchenversuchsanlage generiert zwei Dateien. Bei Starten einer Messung in der Applikation `Filterkuchenversuch_V01DL.vi`, durch das Betätigen des StartIcons (⇒) oben links im Front Panel, folgen zwei Dialogfenster, wo der Dateiname des Typs, wie oben Angeben, eingegeben werden soll. Beim zweiten Dialogfenster, für die Tubidimeter Datei ist eine analoge Dateinamenvergabe zu vollziehen (`FK_Tubi_<Gruppen_ID>.txt`). **Bei dieser Datei wird die Datei Überschrieben!** Der Grund liegt an der Tatsache, dass bei der Filterkuchenversuchsanlage drei Versuchsdurchläufe durchgeführt werden. Nach dem Beenden einer kontinuierlichen Messung werden die diskreten Tubidimeterspezifischen Daten eingegeben. Daraufhin wird bis zu dreimal der Versuch

wiederholt. Die Tubidimeter Datei wird bei jeder Iteration weiter vervollständigt. Ein Anhängen der neuen Daten hätte sinnlose redundante Datengenerierung zur Folge. **Nach der letzten Iteration ist das Programm nochmals zu starten und mit Stop zu quittieren, da sonst die Tubidimeterdaten der letzten Eingabe nicht erfasst werden.**

Hinweis zum Schrebschutz Der gewünschte Schrebschutz der Dateien wurde implementiert, somit sind diese originalen Dateien nicht modifizierbar, jedoch ist das Kopieren und das Einfügen des Inhalts in eine andere Datei weiterhin möglich. Des Weiteren beinhaltet der Schrebschutz nicht das nicht Löschen der Datei. Manipulationsversuche können somit nur damit detektiert werden, wann die Datei erstellt wurde und ob ggf. ob kein Schrebschutz bei einer duplizierten und manipulierten Datei vorhanden ist oder nicht.

Abschließend kann gesagt werden, dass der erste Schritt der *digitalen Transformation* getätigigt wurde. Es war mir eine Ehre den ersten Schritt beitragen zu dürfen. Ich wünsche der HAW Life Sciences viel Erfolg für die nächsten Schritte.

Literatur

- [1] Plattform Industrie 4.0. „Technologieszenario „Künstliche Intelligenz in der Industrie 4.0““. In: *Bundesministerium für Wirtschaft und Energie (BMWi)* (2019). URL: https://www.plattform-i40.de/PI40/Redaktion/DE/Downloads/Publikation/KI-industrie-40.pdf?__blob=publicationFile&v=10 (besucht am 29.01.2021).
- [2] Accenture. „Labor 4.0 – digitale Transformation erfasst Forschung und Entwicklung“. In: *CHE-Manager* (12. Feb. 2019). URL: <https://www.chemanager-online.com/themen/strategie/labor-4-0-digitale-transformation-erfasst-forschung-und-entwicklung> (besucht am 26.09.2020).
- [3] Gillhuber Andrea. *Bluetooth in Industrie 4.0*. 30. Mai 2019. URL: <https://www.industrial-production.de/ki---datenanalyse/bluetooth-in-industrie-4-0.htm> (besucht am 10.10.2020).
- [4] B+B Thermo-Technik GmbH. *Technische Daten des DRTR-AL-10V-RV1 Drucksensors*. 21. Feb. 2021. URL: https://b2c.bb-sensors.com/out/media/Datenblatt_Relativdrucktransmitter_Aluminium.pdf (besucht am 21.02.2021).
- [5] Jürgen Böhm. „Digitale Transformation – Trend oder industrielle Revolution?“ In: *Computerwoche* (23. Juni 2016). URL: <https://www.computerwoche.de/a/digitale-transformation-trend-oder-industrielle-revolution,3312556> (besucht am 05.02.2021).
- [6] Andriy Burkov. *Machine Learning Kompakt*. 1. Frechen: mitp Verlag, 2019. ISBN: 978-3-9-5845-9.
- [7] Rudolf Busch. *Elektrotechnik und Elektronik - Für Maschinenbauer und Verfahrenstechniker*. Berlin Heidelberg New York: Springer-Verlag, 2015. ISBN: 978-3-658-09675-5.
- [8] Jason Cross. „Bluetooth-Standard: Mehr Reichweite, mehr Speed“. In: *Computerwoche Voice of Digital* (21. März 2018). URL: <https://www.computerwoche.de/a/mehr-reichweite-mehr-speed,3544575> (besucht am 10.10.2020).
- [9] O. J. Dahl, E. W. Dijkstra und C. A. R. Hoare, Hrsg. *Structured Programming*. GBR: Academic Press Ltd., 1972. ISBN: 0122005503.
- [10] Gudrun Ding. „Die Wirbelschichttechnologie und ihre Möglichkeiten“. In: *Pharma+Food* (9. Apr. 2010). URL: <https://www.pharma-food.de/ausuestung/feststofftechnik/die-wirbelschichttechnologie-und-ihre-moeglichkeiten-1.html>.

- [11] Martin Geweke. *Versuchsanweisung für die Kuchenfiltrationsanlage*. 2020.
- [12] Infotip Service GmbH. RS232. 10. Okt. 2020. URL: <https://kompendium.infotip.de/rs-232-die-serielle-schnittstelle.html>.
- [13] Gartner Group. *Big Data*. 6. Jan. 2021. URL: <https://www.gartner.com/en/information-technology/glossary/big-data>.
- [14] Iiro Harjunkoski und Reinhard Bauer. „Sharing data for production scheduling using the ISA-95 standard. METHODS ARTICLE“. In: *Frontiers of Energy Research: Process and Energy Systems Engineering* (21. Okt. 2014). URL: <https://www.frontiersin.org/articles/10.3389/fenrg.2014.00044/full> (besucht am 18. 10. 2020).
- [15] Nils Heinrich. „Development of a Portable Process Data Acquisition System“. Magisterarb. HAW der Fakultät Technik und Informatik, 25. Juni 2010. URL: https://deposit.haw-hamburg.de/bitstream/20.500.12738/5198/1/Nils_Heinrich_1836842.pdf.
- [16] Hosting-Technik. „Datenbankmanagementsystem (DBMS) erklärt“. In: *IONOS* (18. Mai 2020). URL: <https://www.ionos.de/digitalguide/hosting/hosting-technik/datenbankmanagementsystem-dbms-erklärt/> (besucht am 25. 08. 2020).
- [17] J.M. Hughes. *Real World Instrumentation with Python: Automated Data Acquisition and Control Systems*. O'Reilly Media, 2010. ISBN: 9781449396633. URL: <https://books.google.de/books?id=MAdNTvLNrJsC>.
- [18] National Instruments. *Grundlagen zur LabVIEW-Umgebung*. 13. Okt. 2020. URL: <https://www.ni.com/getting-started/labview-basics/d/environment> (besucht am 13. 10. 2020).
- [19] National Instruments. *Zeitreihenformattierung*. 24. Dez. 2020. URL: https://zone.ni.com/reference/en-XX/help/371361R-01/glang/codes_for_time_format_str/ (besucht am 17. 11. 2020).
- [20] Intellipaat. *Supervised Learning vs Unsupervised Learning vs Reinforcement Learning*. 27. Jan. 2021. URL: <https://intellipaat.com/blog/supervised-learning-vs-unsupervised-learning-vs-reinforcement-learning/>.
- [21] ISA-S95. *ISA-S95*. 23. Nov. 2020. URL: <https://isa-95.com/isa-95-enterprise-control-systems/> (besucht am 23. 11. 2020).

- [22] K. G. Jumo GmbH Co. „Smarte Sensoren als Türöffner der Industrie 4.0“. In: *CHEManager* (28. Juni 2017). URL: <https://www.chemanager-online.com/themen/mess-automatisierungstechnik-smarte-sensoren-als-tueroeffner-der-industrie-40> (besucht am 04.02.2021).
- [23] J. Kodosky, J. MacCrisken und G. Rymar. „Visual programming using structured data flow“. In: *Proceedings 1991 IEEE Workshop on Visual Languages*. Los Alamitos, CA, USA: IEEE Computer Society, Okt. 1991, S. 34, 35, 36, 37, 38, 39. DOI: 10.1109/WVL.1991.238853. URL: <https://doi.ieee.org/10.1109/WVL.1991.238853>.
- [24] Christian Lemke. „Mit Reinforcement Learning auf dem Weg zur Allgemeinen KI“. In: *Digitale Welt* (18. Okt. 2019). URL: <https://digitaleweltmagazin.de/2019/10/18/mit-reinforcement-learning-auf-dem-weg-zur-allgemeinen-ki/>.
- [25] Stefan Luber und Andreas Donner. *Was ist RS-232?* 21. März 2019. URL: <https://www.ip-insider.de/was-ist-rs-232-a-807547/> (besucht am 10.10.2020).
- [26] Stefan Luber und Florian Karlstetter. „Was ist Database as a Service (DBaaS)?“ In: *Cloudcomputing Insider* (16. März 2018). URL: <https://www.cloudcomputing-insider.de/was-ist-database-as-a-service-dbaas-a-692502/> (besucht am 25.08.2020).
- [27] Stefan Luber und Nico Litzel. „Was ist ein Datalake?“ In: *BigData Insider* (15. Feb. 2018). URL: <https://www.bigdata-insider.de/was-ist-ein-data-lake-a-686778/>.
- [28] Alexander Mäder. „Überflüssiger Papierkram“. In: *Laborjournal* (März 2018), S. 44–45. URL: http://www.laborjournal-archiv.de/epaper/LJ_18_03/44/ (besucht am 24.01.2021).
- [29] Klaus Manhart. „Was Sie über Maschinelles Lernen wissen müssen“. In: *Computerwoche* (3. Sep. 2020). URL: <https://www.computerwoche.de/a/was-sie ueber-maschinelles-lernen-wissen-muessen,3329560> (besucht am 29.01.2021).
- [30] F.J. Ferrero Martín u. a. „Low-cost open-source multifunction data acquisition system for accurate measurements“. In: *Science Direct* (24. Mai 2014). URL: <https://www.sciencedirect.com/science/article/abs/pii/S0263224114002206>.
- [31] Andreas Meier und Michael Kaufmann. *SQL- und NoSQL-Datenbanken*. Berlin Heidelberg New York: Springer-Verlag, 2016. ISBN: 978-3-662-47664-2.
- [32] MESA. *Business to Manufacturing Markup Language - B2MML*. 8. Sep. 2020. URL: <http://www.mesa.org/en/B2MML.asp> (besucht am 08.09.2020).

- [33] Sebastian Minnich. „Vor- und Nachteile des Cloudcomputing“. In: *heise* (11. Mai 2017). URL: <https://www.heise.de/download/blog/Die-Vorteile-und-Nachteile-des-Cloud-Computing-3713041> (besucht am 25.08.2020).
- [34] Bernward Mütterlein. *Handbuch für die Programmierung mit LabVIEW - mit Studentenversion LabVIEW 2009*. Berlin Heidelberg New York: Springer-Verlag, 2009. ISBN: 978-3-827-42338-2.
- [35] Ulrich Nolten. „Entwicklung und Charakterisierung dehnungsbasierter Kraft- und Momentensensoren für medizinische Anwendungen“. Diss. RWTH Aachen, 19. Apr. 2013. URL: <http://publications.rwth-aachen.de/record/229156/files/4667.pdf>.
- [36] B. Markus Pfeiffer u. a. „Digitale Zwillinge und Prozessmodellierung – wie Sie davon profitieren können“. In: *Process Vogel* (15. Jan. 2020). URL: <https://www.process.vogel.de/digitale-zwillinge-und-prozessmodellierung-wie-sie-davon-profitieren-koennen-a-896481/> (besucht am 20.08.2020).
- [37] PROCESS. „ISA-95 schließt letzte Lücke zwischen Geschäfts- und Produktions-Prozessen“. In: *Process Vogel* (18. Sep. 2006). URL: <https://www.process.vogel.de/isa-95-schliesst-letzte-luecke-zwischen-geschaefts-und-produktions-prozessen-a-60348/> (besucht am 24.11.2020).
- [38] Silke Reuter. „Partikeldynamik in Wirbelschichtsystemen - Geschwindigkeit und Zirkulation granularer Medien untersucht mit bildgebender und geschwindigkeitskodierender NMR“. Diss. RWTH Aachen, 2011. URL: <http://publications.rwth-aachen.de/record/82677/files/3846.pdf> Korngroessenreduktionsmechanismen %20in %20Wirbelschichten %20bei%20unterschiedlichen%20Betriebsbedingungen.pdf.
- [39] Dorian Selz. „Entscheidungsrelevantes Wissen aus unstrukturierten Daten produzieren“. In: *Digitale Welt* (16. Okt. 2018). URL: <https://digitaleweltmagazin.de/2018/10/16/entscheidungsrelevantes-wissen-aus-unstrukturierten-daten-produzieren/> (besucht am 28.01.2021).
- [40] Bundesamt für Sicherheit in der Informationstechnik. „ICS-Security-Kompendium“. In: (2013). URL: https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/ICS/ICS-Security_kompendium_pdf.pdf?__blob=publicationFile (besucht am 08.11.2020).

- [41] Peter Stephan u. a. *VDI-Wärmeatlas - Fachlicher Träger VDI-Gesellschaft Verfahrenstechnik und Chemieingenieurwesen*. Wiesbaden: Springer Berlin Heidelberg, 2019. ISBN: 978-3-662-52988-1.
- [42] Matthias Stiess. *Mechanische Verfahrenstechnik - Band 2*. Berlin Heidelberg New York: Springer-Verlag, 2013. ISBN: 978-3-662-08599-8.
- [43] Matthias Stiess. *Mechanische Verfahrenstechnik - Partikeltechnologie 1* -. Berlin Heidelberg New York: Springer-Verlag, 2008. ISBN: 978-3-540-32552-9.
- [44] Enno Stöver. *Lernort digitale Umformtechnik*. 21. Feb. 2021. URL: <https://www.haw-hamburg.de/hochschule/technik-und-informatik/departments/maschinenbau-und-produktion/institute-und-einrichtungen/ipt/schwerpunkte/lernort-digitale-umformtechnik/>.
- [45] Technikum-Wien-Academy. *Was ist digitale Transformation?* 25. Aug. 2020. URL: <https://academy.technikum-wien.at/ratgeber/digitale-transformation-was-ist-das/> (besucht am 25.08.2020).
- [46] TensorFlow. *Das neuronale strukturierte Lern-Framework*. 27. Jan. 2021. URL: https://www.tensorflow.org/neural_structured_learning/framework (besucht am 27.01.2021).
- [47] Hans-Rolf Tränkler und Leonhard M. Reindl. *Sensortechnik - Handbuch für Praxis und Wissenschaft*. Berlin Heidelberg New York: Springer-Verlag, 2015. ISBN: 978-3-642-29942-1.
- [48] Datenbanken Verstehen. *XML-Datenbank*. 18. Nov. 2020. URL: <https://www.datenbanken-verstehen.de/lexikon/xml-datenbank/#:~:text=Man%20unterscheidet%20dabei%20zwischen%20nativen%20und%20relationalen%20XML%E2%80%93Datenbanken:&text=Dabei%20liegen%20die%20Inhalte%20in,dem%20SQL%E2%80%93XML%E2%80%93Standard..>
- [49] Olga Weiß. *Pinbelegung und Verwendung des RS-232 DB9*. 23. Juni 2020. URL: <https://www.eltima.com/de/article/9-pin-serial-port.html>.
- [50] Gerhard Wiegleb. *Gasmesstechnik in Theorie und Praxis - Messgeräte, Sensoren, Anwendungen*. Berlin Heidelberg New York: Springer-Verlag, 2016. ISBN: 978-3-658-10687-4.
- [51] Marcus Wolff. *Sensor-Technologien - Band 1: Position, Entfernung, Verschiebung, Schichtdicke*. Berlin: Walter de Gruyter GmbH & Co KG, 2016. ISBN: 978-3-110-46095-7.
- [52] Marcus Wolff. *Sensor-Technologien: Band 2: Geschwindigkeit, Durchfluss, Strömungsfeld*. Berlin, Boston: De Gruyter Oldenbourg, 2017. ISBN: 978-3-11-047784-9. DOI: <https://doi.org/10.1515/9783110477849>. URL: <https://www.degruyter.com/view/title/521081>.

Abbildungsverzeichnis

1	Übersicht verschiedener Durchfluss- und Mengenmessverfahren [52, S. 794]	12
2	Übersicht einiger Sensoren, die für die Messung von Gasdurchflüssen geeignet ist	14
3	Prinzip eines thermischen Massendurchflusssensors	15
5	Filterkuchenmodell zur Ableitung der Filtergleichung	24
7	Filtergerade für die isobare Betriebsweise	27
8	Klassifizierung von Schüttgütern nach Geldart	31
9	Wirbelschichtzustände als Funktion der Anströmgeschwindigkeit	32
11	Konstantz des dimensionslosen Drucks, ab dem überschreiten der Lockerungsgeschwindigkeit	35
12	Binary data sizes	36
13	Digital and analog data inputs [17]	37
14	Digitale Schnittstelle oder über Messkarte	38
15	DB-9 pin and socket numbering	38
16	Synchronous serial data communication	39
17	Ideal versus real square waves	40
18	RS-485 signal levels [17, S. 226]	42
19	Beispiel für ein Blockdiagramm und das zugehörige Frontpanel	45
20	Informationssystem	47
21	Relationenmodell: Tabellengerüst und Ausprägung einer SQL-Tabelle	49
26	Digitalisierungsentwurf ohne Datenbank	58
27	Digitalisierungsentwurf mit Datenbank	61
28	Root-/Branchelements, gemäß ISA-S95	67
29	Labordatenstrukturdarstellung, angelehnt an ISA-S95	68
30	100_Informations, angelehnt an ISA-S95 und B2MML	69
31	Potentielle hierarchische Datenstruktur der Wirbelschicht (<i>engl. fluidizationbed</i>) Unit Operation, angelehnt an ISA-S95 und B2MML	70
32	Datenstruktrentwurf einer relationalen Datenbank zur Beschreibung der Wirbelschicht Unit Operation, angelehnt an ISA-S95 und B2MML	74
33	DAQ von Messeinrichtungen mit integrierter digitaler Schnittstelle	76
34	Einfügen aller seriellen Funktionen	81
37	Definition der VI Objekt Ein- und Ausgänge	85
38	BD Haupt VI: SSVI Integration/Port Konfiguration/Stringoperationen	87
39	FP Haupt VI: SSVI Integration/Port Konfiguration/Stringoperationen	87
40	Slicing Bedingung	89
41	„String Slicing“: Dateninterpretation	90
42	String-/Zeilenkonstruktion	93
44	Datalogger	96

45	Graph für zwei kontinuierliche Parameter	97
46	Modulare Header Sub VI's	98
47	DAQ von Messeinrichtungen mit analoger Schnittstelle	100
48	DAQ Programmierung, Initialisierung und read.vi	100
49	Analog-, Digitaldruckmessung als Funktion der analogen Volumenstrommesswerte .	103
50	Analog-/Digitalvolumenstrommessung sowie absolute Abweichung in l/min, als Funktion der Manometerdifferenz in mm	103
51	DAQ und interpretation beider kontinuierlichen Paramter, Druck und Volumenstrom	104
52	Dataloggerinitialisierung	105
53	Filterkuchenheader	106
54	Endsequenz des Hauptprogramms	106
55	Einstellung „End text entry with Enter key“	107
56	Filterkuchen Frontpanel: Versucheingaben	108
57	Filterkuchen Frontpanel: Tab der kontinuierlichen Messung am Beispiel der Filterkuchenapplikation	109
58	Filterkuchen Frontpanel: Tab der Tubidimetereingaben	110
59	Datalogs und Excel troubleshooting	111
60	Filterkuchenversuchsanlage nach den Modifikationen	115
63	Wirbelschichtversuchsanlage der HAW Hamburg der Fakultät Life Sciences, vor den Modifikationen	123
66	Gravimetrische Messung in Python	137
67	Kenndaten von Kapazitiven und piezoresistenten Drucksensoren [47, S. 472 ff.]	v
68	Hierarchische Gliederung eines ICS, gemäß ICS-Kompendium des Bundesamtes für Sicherheit in der Informationstechnik [40, S. 18]	vi
69	Ausschnitt der <i>ProcessSegment</i> XSD der B2MML-V0600-ProcessSegment.xsd	vii
70	Speicherbedarfsvergleich ASCII/binär [15, S. 44]	viii
71	200ProcessEngineering, angelehnt an ISA-S95 und B2MML	ix
72	300_Trials, angelehnt an ISA-S95 und B2MML	x
73	Abstrakte Dataflow Darstellung einer potentiellen SQL DB	xi
74	potentieller SQL Dataflow (Abstraktionsebenen gemix)	xii
75	Unterste Abstraktionsebene der SQL Datenstruktur	xiii
77	Technische Daten des DRTR-AL-10V-RV1 Drucksensors	xvi
79	VA-525 Kalibrierzertifikat	xviii
80	Cloudnutzungsalternative S. 2	xix
81	Cloudnutzungsalternative S. 2	xx
82	Dateidialog	xxi
83	Go to hidden tabs	xxii
84	Schwebekörperdurchflussmesser ($m=0,6585\text{ g}$) mit einer Messspanne von 0,185 bis 51/-min, zzgl. Approximationsfunktionen der Kennlinie	xxii

85	Schwebekörperdurchflussmesser ($m=2,225\text{ g}$) mit einer Messspanne von $1,861/\text{min}$ bis $501/\text{min}$, zzgl. Approximationsfunktionen der Kennlinie	xxiv
86	Manometerkennline; Druckdifferenz zzgl. 0,99 bar, laut digitalem Drucksensor am 17.12.20, um 13:02 Uhr	xxiv

Tabellenverzeichnis

1	Mechanische Unit Operations, deren Versuchsstände und messtechnische Einrichtungen	3
2	„Grad der Digitalisierung“ der MVT Geräte und Anlagen	4
3	Schnittstellenverteilung im MVT-Labor	5
4	Vor- und Nachteile von Informationsmanagementsystemen [31, S. 2 ff.]	46
5	Vor- und Nachteile der Nutzung von Database- oder Plattform-as-a-Service, gemäß der Quellen [33, 16, 26]	62
6	Vor- und Nachteile des Konzeptentwurfs 4.0	66
7	RS-232 Konfigurationsparameter der Digitalwaage Kern 440-47 N in shorthand notation 8-N-1	77
8	Benötigte LabVIEW Konfigurationsfunktionen Teil 1	80
9	Benötigte LabVIEW Konfigurationsfunktionen Teil 2	94
10	6, 7, 8 Bit-Encodierung via RS-232	99
11	RS-232 Schnittstellenparameter der Practum S101-1S Waage	116
12	Bezeichnungen der Filterkuchenversuchsanlage	120
13	Bezeichnungen der Wirbelschichtversuchsanlage nach den Modifikationen	127
14	„Grad der Digitalisierung“ der MVT Geräte und Anlagen	131
15	Vor- und Nachteile des Konzeptentwurfs 4.0	135
16	Versuchspараметр der MVT Geräte und Anlagen	ii
17	Labordatenstrukturdarstellung, angelehnt an ISA-S95 und B2MML	viii
18	Tabelle mit den Werten aller Sensordiagramme	xxiii
19	Tabelle mit den Werten Aller Sensordiagramme	xxiii

Anhang

Einleitung

Tabelle 16: Versuchsparameter der MVT Geräte und Anlagen

MVT Versuchsstände	diskret	konti.	MVT Versuchsstände	diskret	konti.
Kuchenfiltration			Hochdruckhomogenisator (HDH)		
Gewicht		✓	Fluid1	✓	
Druck		✓	Fluidvolumen1	✓	
Suspensionskonzentration	✓		Fluid2	✓	
Suspensionsvolumen	✓		Fluidvolumen2	✓	
Filterkuchendicke	✓		Disperer RPM	✓	
Filterkuchendurchmesser	✓		Dispersionsdauer	✓	
Filterkuchenvolumen	✓		Druck	✓	
Filterkuchenfeuchtegehalt	✓		HDH Dauer	✓	
Filtrernutschen Nr.	✓		Temperatur	✓	
Filterporösität	✓		Partikelgrößenverteilung	✓	
Gas-Feststoff Wirlbelschicht			ZickZack Sichten		
Volumenstrom _{Luft}		✓	Feststoff	✓	
Druck		✓	Gesamtmasse	✓	
Festbetthöhe	✓		Masse Grobgut	✓	
Schüttgutmasse	✓		Masse Feingut		
Partikeldichte	✓		Vibrationsintensität der		
Wirlbelschichthöhe = f(p)	✓	✓	Vibrationsförderung	✓	
Wirlbelschichtporösität = f(p)	✓	✓	Volumenstrom	✓	
Sauterdurchmesser	✓		Förderklappenstellung	✓	
Festbettporösität	✓				
Rührversuch			Walzenmühle		
Fluid	✓		Feststoff	✓	
Fluiddichte	✓		Masse	✓	
dynamische Viskosität	✓		Walzengeometrie (Profil)	✓	
Rührer	✓		Walzenspalt	✓	
Rührereintauchtiefe	✓		Umfangsgeschwindigkeit	✓	
Strombrecher	✓				
RPM	✓				
Drehmoment		✓			

Ergänzung des Abschnitts Dehnungsresistive Sensoren

an dieser Stelle ist anzumerken, dass die Variablenbezeichnung nicht vollständig mit dem Hauptdokument übereinstimmen und auch nicht den DIN Normen entsprechen.

Die Drucksensoren mittels DMS sind dehnungsresistive Sensoren. Das Messprinzip beruht auf die Änderung des elektrischen Widerstandes R , eines elektrisch leitenden Werkstoffs, als Folge einer mechanischen Verformung. Ein zylinderförmiger Werkstoff, charakterisiert durch die Ausgangslänge l_0 , Ausgangsquerschnitt A_0 und dem spezifischem elektrischem Widerstand ρ_{el} , hat einen elektrischen Widerstand R gemäß folgender Gleichung [51, S. 4 ff.]:

$$R = \rho_{el} \cdot \frac{l_0}{A_0} \quad (60)$$

Um die relative Widerstandsänderung $\Delta R/R$ zu berechnen wird das totale Differenzial (Gleichung 61) aus Gleichung 5 gebildet.

$$dR = \frac{\partial R}{\partial \rho_{el}} d\rho_{el} + \frac{\partial R}{\partial l_0} dl_0 + \frac{\partial R}{\partial A} dA \quad (61)$$

$$= \frac{l_0}{A} d\rho_{el} + \frac{\rho_{el}}{A} dl_0 - \frac{\rho_{el} \cdot l_0}{A^2} dA \quad (62)$$

$$= R \cdot \left(\frac{d\rho_{el}}{\rho_{el}} + \frac{dl_0}{l_0} - \frac{dA}{A} \right) \quad (63)$$

Ersetzt man das Infinitesimal d durch eine größere Schrittweite Δ , unter der Bedingung dass $\Delta\rho \ll \rho$, $\Delta l_0 \ll l_0$, $\Delta A \ll A$, dann erhält man Gleichung 6.

$$\frac{\Delta R}{R} = \frac{\Delta \rho_{el}}{\rho_{el}} + \frac{\Delta l}{l_0} - \frac{\Delta A}{A_0} \quad (64)$$

Die geometrische Verformung wird durch die beiden letzten Terme repräsentiert. Die Dehnung ε ersetzt $\Delta l/l_0$ und der Term $\Delta A/A$ wird durch die Querkontraktion ($\varepsilon_Q = \frac{\Delta D}{D_0} = -\nu \cdot \varepsilon$), mit der Querkontraktionszahl (Poisson-Konstante) ν , beschrieben und führt zu der Lösung der Gleichung (7)

$$\frac{\Delta A}{A} = \frac{\frac{\pi}{4} \cdot ((D + \Delta D)^2 - D^2)}{\frac{\pi}{4} \cdot D^2} \quad (65)$$

$$= \frac{2D \cdot \Delta D + (\Delta D)^2}{D^2} \quad (66)$$

$$\approx \frac{2\Delta D}{D} \quad (67)$$

$$= 2\varepsilon_Q = -2 \cdot \nu \cdot \varepsilon \quad (68)$$

Durch Einsetzen der Gleichung (7) und der Dehnung ε in die Gleichung (6) erhält man:

$$\frac{\Delta R}{R} = (1 + 2 \cdot \nu) \cdot \varepsilon + \frac{\Delta \rho_{el}}{\rho_{el}} \quad (69)$$

Der Term $\frac{\Delta \rho_{el}}{\rho_{el}}$ ist der sog. „piezoresistive Effekt“, der bei metallischen Leitern vernachlässigbar ist, jedoch bei Halbleitern überwiegt.

Abbildung 67: Kenndaten von Kapazitiven und piezoresistenten Drucksensoren [47, S. 472 ff.]

Messprinzip	piezoresistiv	kapazitiv
Nenndrücke [kPa]	10 100 1000	10 100 1000
Spanne [mV/V]	8 10 40	0,5 pF 20 pF
Linearitätsfehler [%] ^a	$\pm 0,3$ $\pm 0,3$ $\pm 0,3$	nichtlinear
Offset [mV/V]	± 2	20 pF
Offset TK [%/K]	$\pm 0,02$	$\pm 0,2$
Empfindlichkeit TK [%/K]	-0,15	-0,01
Hysterese [%]	$\pm 0,4$ (bei 1 Mio. Lastwechsel)	$\pm 0,5 \pm 0,2$
Gesamtfehler kompensiert [%]	$\pm 0,5$	$\pm 2,0$
Überlastbarkeit	$5 \times P_N$ $2 \times P_N$ $1,5 \times P_N$	$100 \times P_N$
Trimmverfahren	Lasertrimmen, resistiver Abgleich	Digitaler Abgleich
Temperaturbereich	-40 °C bis +100 °C	-40 °C bis +100 °C
Technologie	VMM ^b , Anodic Bonding	VMM, OMM ^c , Anodic Bonding, Silicon Direct Bonding
Chipgröße [mm ²]	5 x 5 4 x 4 4 x 4	6 x 6 4 x 4
Eigenfrequenz [kHz]	10	>20
Speisespannung [V]	5	
Geeignet für Absolutdruck	sehr gut	sehr gut
Relativdruck	sehr gut	gut
Differenzdruck	bedingt	gut

^a bei Druckbeaufschlagung von oben,

^b VMM: Volumenmikromechanik,

^c OMM: Oberflächenmikromechanik

Anhang des Abschnitts Datenverwaltung im Stand der Technik

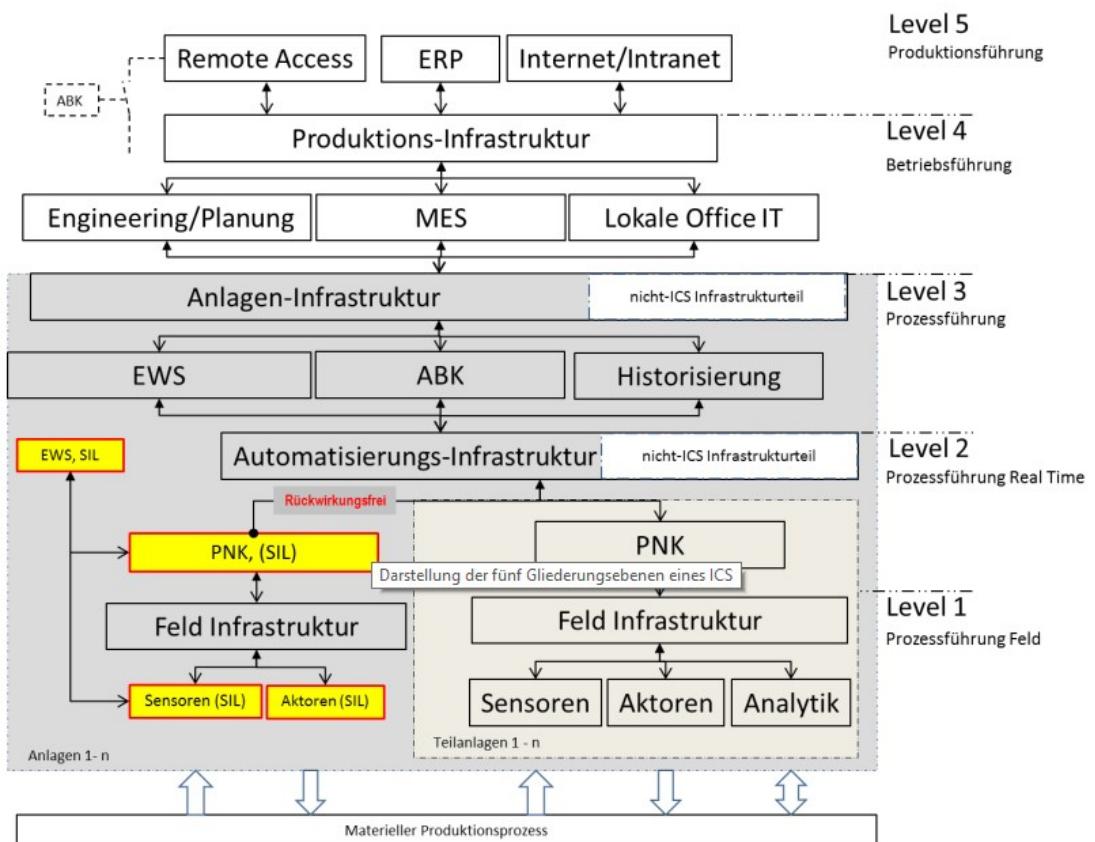


Abbildung 68: Hierarchische Gliederung eines ICS, gemäß ICS-Kompendium des Bundesamtes für Sicherheit in der Informationstechnik [40, S. 18]

```
<!-- - - - - - -->
<!-- ProcessSegmentInformation Transaction Types -->
<!-- - - - - - -->

<xsd:complexType name = "GetProcessSegmentInformationType">
  <xsd:sequence>
    <xsd:element name = "ApplicationArea" type = "TransApplicationAreaType"/>
    <xsd:element name = "DataArea">
      <xsd:complexType>
        <xsd:sequence>
          <xsd:element name = "Get" type = "TransGetType"/>
          <xsd:element name = "ProcessSegmentInformation" type = "ProcessSegmentInformationType"
            minOccurs = "1"
            maxOccurs = "unbounded"/>
        </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
  </xsd:sequence>
  <xsd:attribute name = "releaseID" type="xsd:normalizedString" use="required"/>
  <xsd:attribute name = "versionID" type="xsd:normalizedString" use="optional"/>
</xsd:complexType>

<xsd:complexType name = "ShowProcessSegmentInformationType">
  <xsd:sequence>
    <xsd:element name = "ApplicationArea" type = "TransApplicationAreaType"/>
    <xsd:element name = "DataArea">
      <xsd:complexType>
        <xsd:sequence>
          <xsd:element name = "Show" type = "TransShowType"/>
          <xsd:element name = "ProcessSegmentInformation" type = "ProcessSegmentInformationType"
            minOccurs = "1"
            maxOccurs = "unbounded"/>
        </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
  </xsd:sequence>
</xsd:complexType>
```

Abbildung 69: Ausschnitt der *ProcessSegment* XSD der B2MML-V0600-*ProcessSegment.xsd*

Anhang der Konzeptentwicklung

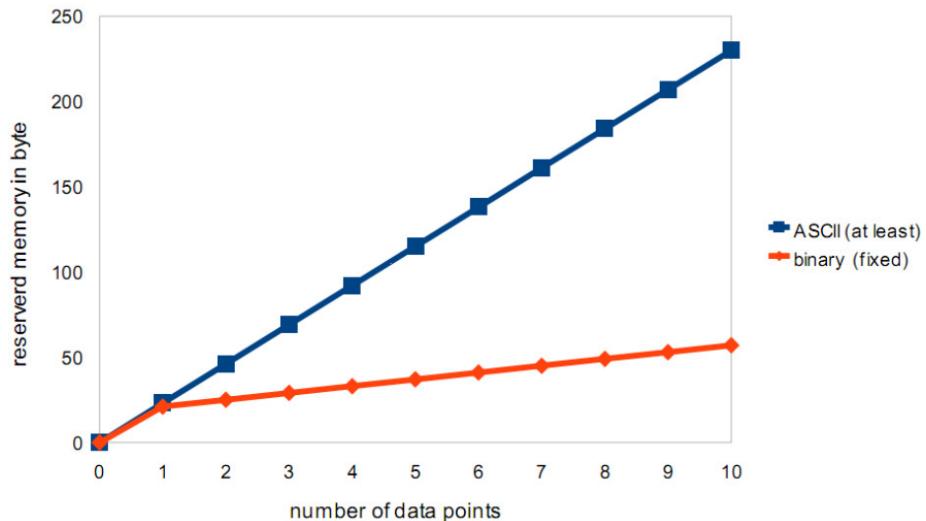


Abbildung 70: Speicherbedarfsvergleich ASCII/binär [15, S. 44]

Tabelle 17: Labordatenstrukturdarstellung, angelehnt an ISA-S95 und B2MML

100_Informations	200_ProcessEngineering	300_Trials
101_PrincipalsPersonel-Information	201_MechUnitOperations	301_TrialPersonalInformation
102_SupervisorPersonel-Information	202_ThermalUnitOperations	302_TrialDevices
103_GroupInformations	203_ChemicalUnitOperations	303_UniversalDeviceDatas
104_DeviceInformations	204_FoodProcessing-UnitOperations	304_UnitOperationTrialDatas
105_StandardOperating-Procedures	205_EnviromentalProcessing-UnitOperations	305_StandaloneDevice-ProcessDatas
106_MaterialSafetySheets		
107_DeviceManuals		
108_ComponentsGeometry-data		

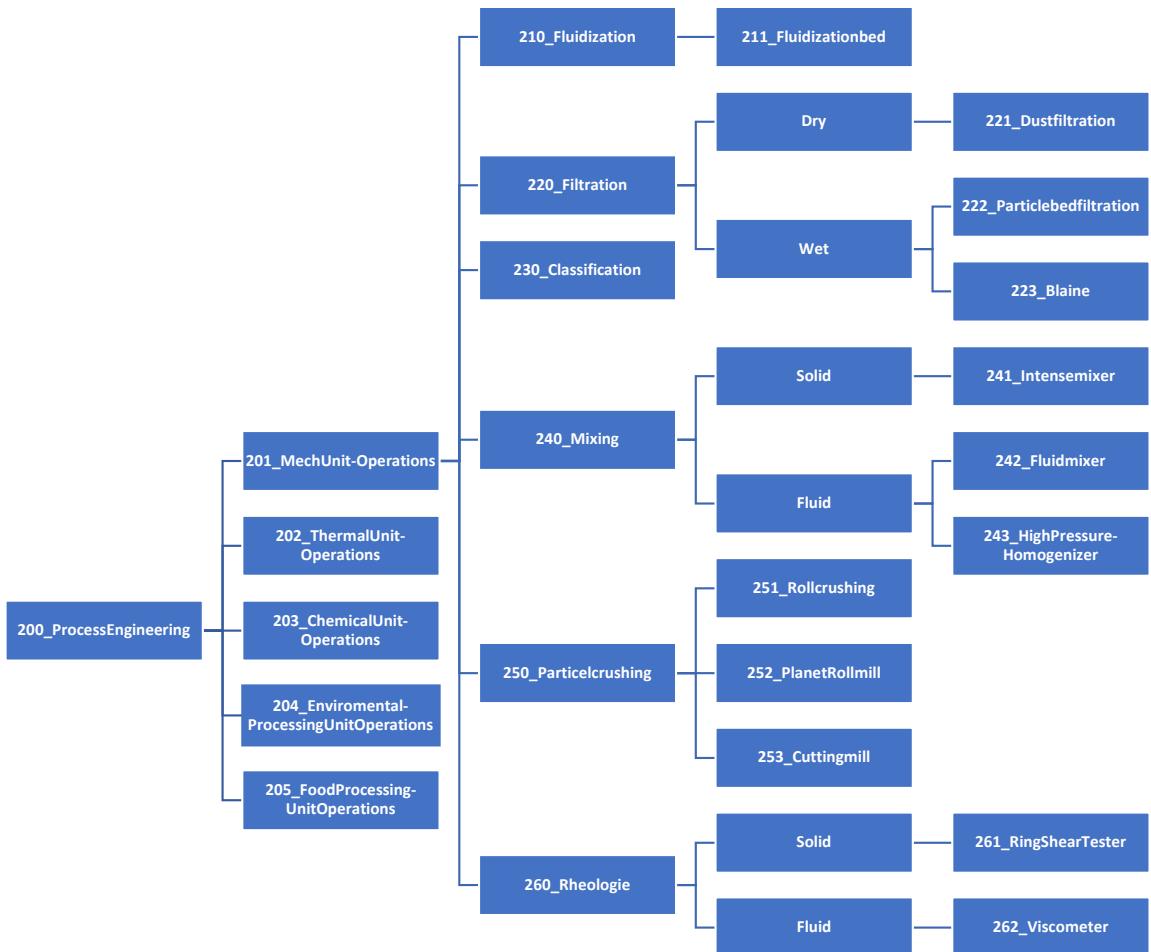


Abbildung 71: 200ProcessEngineering, angelehnt an ISA-S95



Abbildung 72: 300_Trials, angelehnt an ISA-S95

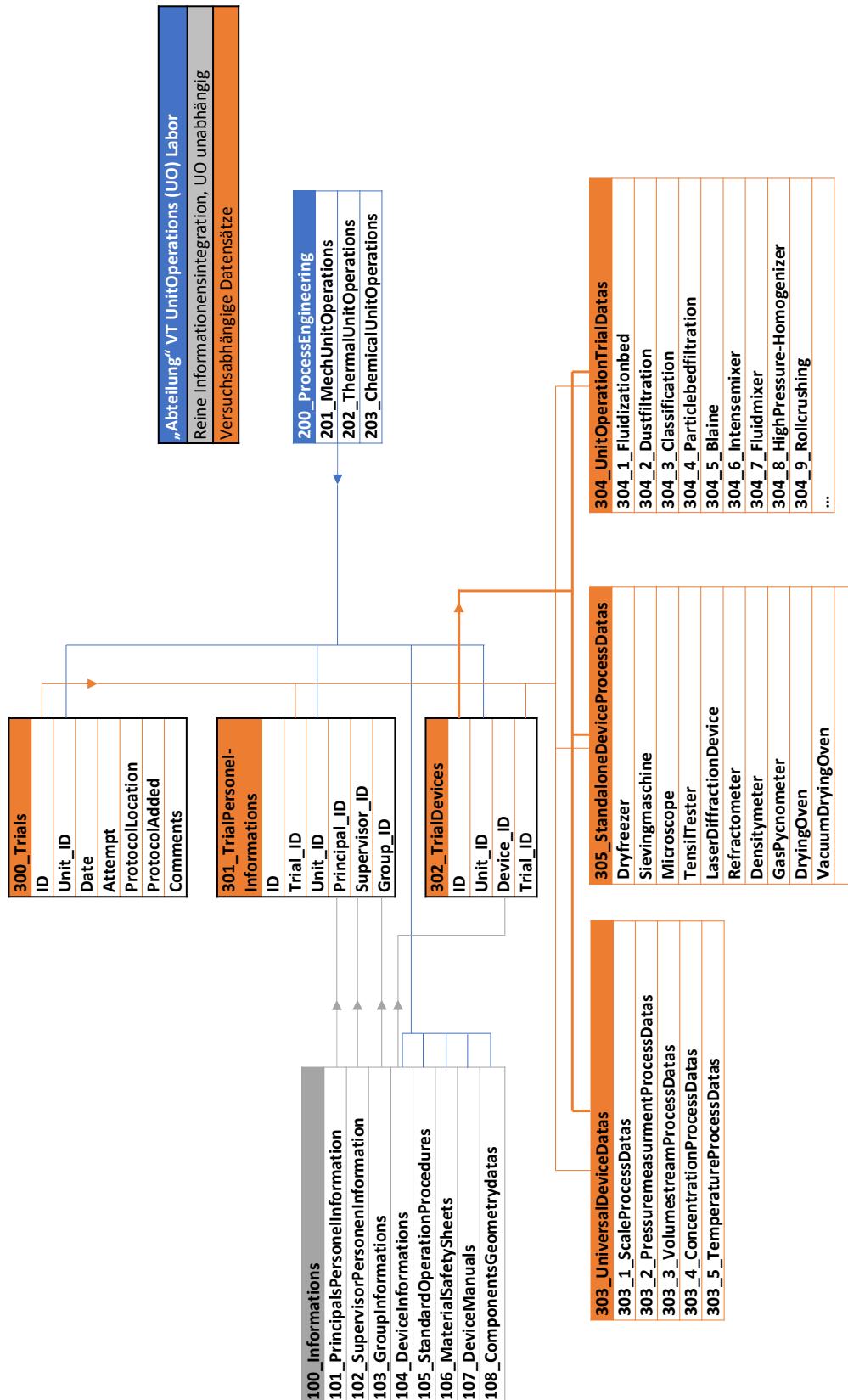


Abbildung 73: Abstrakte Dataflow Darstellung einer potentiellen SQL DB

Anhang

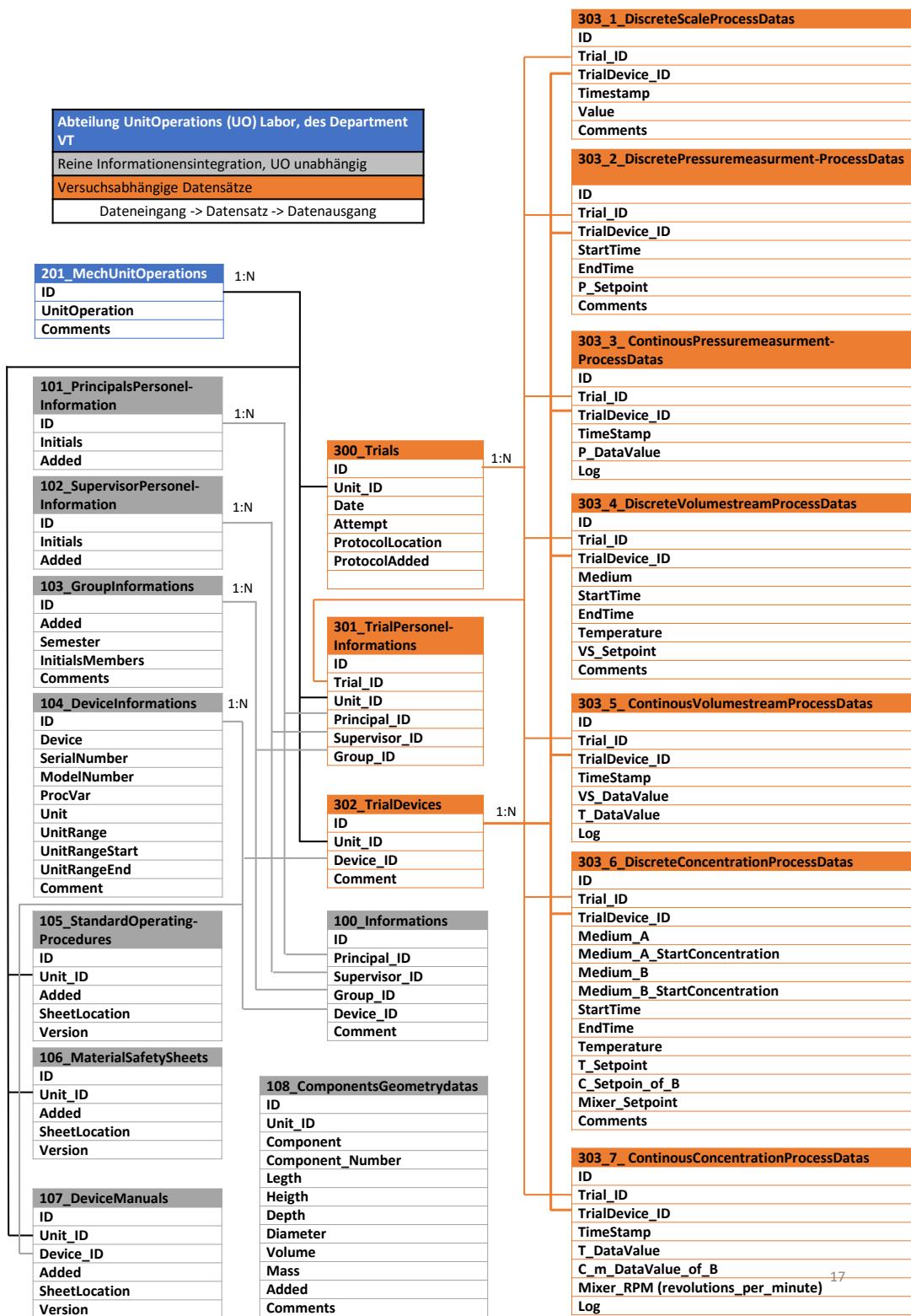


Abbildung 74: potentieller SQL Dataflow (Abstraktionsebenen gemix)

Anhang

Diagramm der Datenstruktur:

Datensatz:

Name	Attribut	Datenwert
benutzer	Name	Winkelmann
	Vorname	Paul
	Email	winkelma...@gmx.de
	GebDat	1980-10-02
		Lindemann
		Erwin
		lindemannerwin71@...de
		1971-09-18
		Heubel
		Monika
		heubel.m@lgocg.com
		1993-01-15
		Faas
		Sigrid
		faas.sig@ohay.de
		1983-05-02

Datenfeld:

Tabelle 201: MechUnitOperations

ID	MechUnitOperations	Comments
1	Fluidizationbed	
2	Particlebfiltration	
3	Classification	
4	Fluidmixing	
5	Particelcrushing	
6	Dustfiltration	
...

Tabelle 300: Trial

ID	Unit_ID	Attempt	Date	Protocol-Location	Protocol-Added	Comments
1	1		15-09-2020			...
...

Tabelle 301: TrialPersonel-Information

ID	Unit_ID	Trial_ID	Principal_ID	Supervisor_ID	Group_ID
1	1	1	1	1	20

Tabelle 302: TrialDevice

ID	Unit_ID	Device_ID	Devices	Comments
1	1		FloatTypeFlowMeter1	
2	1		AnalogPressureMeasurment	
3	1		DigitalPressureMeasurment	
4	1		Scale_FB_Classification	
5

Tabelle 102: SupervisorPersonel-Information

ID	Initials	Added
1	MH	
2	SW	

Tabelle 103: GroupInformations

ID	Added	Semester	InitialMembers	Comments
1	16-09-2020	WS2020	2	
2

Tabelle 101: PrincipalsPersonelInformation

ID	Initials	Added
1	MG	
2

Tabelle 104: DeviceInformations

ID	Device_ID	Device	Serial-Number	Model-Number	ProcVar	Unit	Unitrange	UnitRange-Start	UnitRange-End	Comments
1	1	FloatType-FlowMeter_1			Vs	l/s	5	0	5	
2	2	Pressure-Measurment_1			p	mbar	20	5	25	
3	3	Scale_FB_Classification			m	g				
4

Tabelle 303_1: ScaleProcessDatas

ID	Trial_ID	TrialDevice_ID	Timestamp	ValueName* (could be XY-Scanner)	Value
1	1	5	10:25		22.5
2

Tabelle 303_2: PressuremeasurmentProcessDatas

ID	Trial_ID	Device_ID	TimeStamp	DataValues	Log
1	1	4	00:01	1,01	
2

Tabelle 303_4: DiscreteVolumestream-ProcessDatas

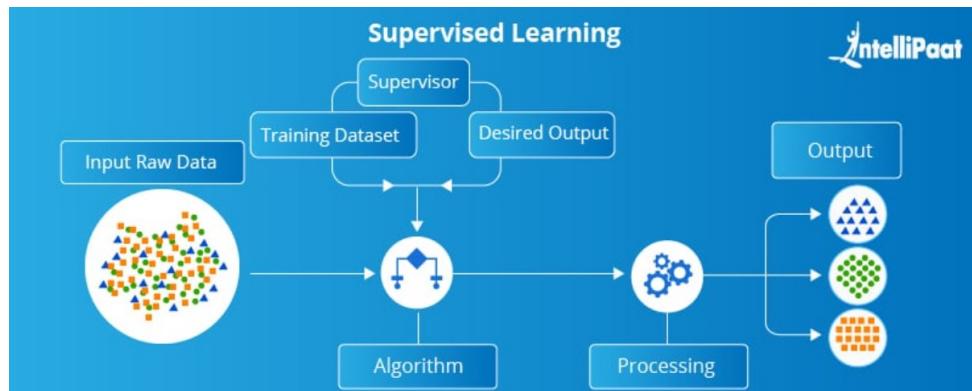
ID	Trial_ID	Device_ID	Medium	StartTime	Endtime	Temperatur	VS_Setpoint	Comments
1	1	1	Air	10:34	10:54		5	
2	1	2	air	1				
3

Abbildung 75: Unterste Abstraktionsebene der SQL Datenstruktur

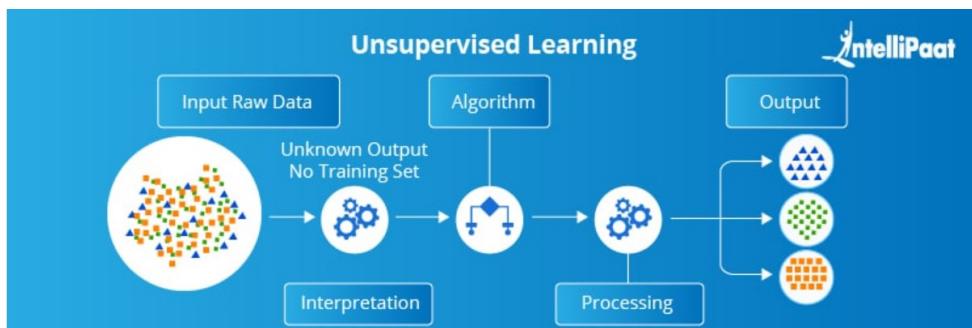
Machine Learning

Einige Algorithmen müssen angelernt werden (*engl. supervised or semi-supervised*), z.B. das vom Google-Brain-Team entwickelte open-source Modell, Tensor Flow [46]. Andere werden nicht angelernt und generieren Ihren Algorithmus autonom (*engl. unsupervised*; vgl. 76a im Anhang). **Supervised Learning** (vgl. Abbildung 76a im Anhang) bedeutet, dass Datensätze vorgegeben werden und zugleich deren Lösung bzw. Klassifizierungen. Anhand des dadurch generierten Algorithmus werden unbekannte Datensätze klassifiziert. Je nach verwendetem Algorithmus können Klassifizierungen binär wie Spam, nicht Spam; Mann, Frau oder Multi-Class-Klassifizierungen, wie z.B. Porsche, Mercedes, VW, sein. **Semi-Supervised Learning** Algorithmen verändern ihren angelernt Algorithmen, bzw. können Ihren Algorithmus, anhand der unvalidierten Daten, die sie nach dem anlernen erhalten, optimieren. KI Algorithmen benötigen große, Datenmengen. Einfache KI's benötigen zusätzlich strukturierte Daten, damit ist die Verfügbarkeit der Daten in Datenbanken gemeint. Die neusten, komplexen [39] maschine learning Algorithmen können auch mit unstrukturierten Daten verwertbare Ergebnisse generieren, doch es gilt stets GIGO, „garbage in, garbage out!“ Die Güte der Ergebnisse einer KI wird durch die Güte der Ausgangsdaten bestimmt, demnach ist die Wahrscheinlichkeit mit strukturierten Daten ein sinnvolles, aussagekräftiges Ergebnis zu erzielen wahrscheinlich höher. Es existiert noch eine vierte Kategorie, die **Reinforced Learning** Algorithmen. Diese Algorithmen unterscheiden sich von den unsupervised Algorithmen in der Art, dass sie mit einer Umgebung (*engl. environment*) interagieren und dadurch lernen [6, S. 15 f.]. Als Beispiele können Roboter oder KI in einem Spiel wie Schach, Go (die asiatische, komplexere Schachvariante), aber auch in Videospielen genannt werden [24].

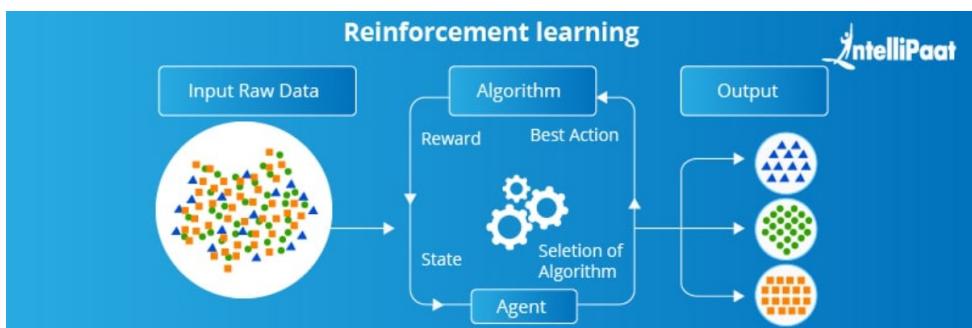
Zu den **Unsupervised Learning** Algorithmen zählen z.B. die Regressionsanalyse oder die k-mean Clusteranalyse [6, S. 114]. Um bei **supervised** oder **semi-supervised** verwertbare Ergebnisse zu erhalten, wird in vielen Anwendungsfällen eine „große“ Datenbasis benötigt. Des Weiteren ist die Güte der Daten entscheidend für die Güte der Ergebnisse. Weniger komplexe KI Algorithmen benötigen strukturierte Daten, wofür sich Datenbankmanagementsysteme (DBMS) auszeichnen.



(a) Supervised Machine Learning



(b) Unsupervised Machine Learning



(c) Reinforced Machine Learning

Abbildung 76: Machine Learning [20]

Anhang der Teilumsetzung

Industrielle Drucktransmitter	
Messbereich	-1 ... +100 bar, 12 Typen
Überlast	Siehe Tabelle
Restfehler Linearität / Hyst.	< $\pm 0,2\%$ FS
Temperaturkoeffizient	TCO < $\pm 0,015\%$ FS / K TCG < $\pm 0,010\%$ FS / K
Temperatur	
Einsatzbereich	-20...+80 °C
Werkstoff Sensor	Keramik, Al ₂ O ₃
Werkstoff Gehäuse	Aluminium AlMgPbCu, blau eloxiert
Dichtung	Viton
Druckanschluss	1/4" Innengewinde
Abmessungen	30 x 30 x 89 mm
Anschluss	4-poliger Industriestecker, DIN 43650
Schutzart	IP65
CE-Konformität	2004/108/EG
EMV-Störaussendung	EN 61000-6-3:2011
EMV-Störfestigkeit	EN 61000-6-1:2007
Ausführung 4...20 mA	
Ausgangssignal	4...20 mA, Zweileiter
Zulässige Bürde	R _a [Ω]=(U _v [V]-10 V)/0,02 A
Ausführung 0...10 V	
Ausgangssignal	0...10 V, Dreileiter
Hilfsenergie	12...30 V DC / 5 mA

Abbildung 77: Technische Daten des DRTR-AL-10V-RV1 Drucksensors, (dem technischen Datenblatt entnommen [4])

Abbildung 78: Dehnungsresistive Sensoren

Spezifikation	Einheit	Metallobasis	Halbleiterbasis		
Hersteller	[–]	HBM	ESA Messtechnik	ME-Systeme	ZSE Electronic
Modell	[–]	Y-Serie (Universal-DMS)	C-Serie (für hohe Temperaturen)	LI66-10/350 (für Faserverbund-Werkstoffe)	EA-XX-500BH-120 SS-060-033-500P KSP-2-120-E3
Ausführung	[–]	Folien-DMS	Folien-DMS	Folien-DMS	Folien-DMS
max. Dehnbarkeit [µm/m] Betrag					
pos.	50.000 (≈ 5 %)	20.000 (≈ 2 %)	±50.000 (≈ 5 %)	±50.000 (≈ 5 %)	±3.000 (≈ 0,3 %) ±3.000 (≈ 0,3 %)
neg.	100.000 (≈ 10 %)	100.000 (≈ 10 %)			
Nennwiderstand [Ω]	120, 350, 700, 1000 (je nach Typ) (je nach Typ)	120, 350	350	120	540
Toleranz [%]	±0,3–0,35	±0,3–0,35	±0,3–0,35	±0,15	±9,26
Nennwiderstand					
k-Faktor	[–]	ca. 2	ca. 2,2	ca. 2	2,05
Toleranz	[%]	±1,0–1,5	±1	±1	–
k-Faktor					
Temperaturbereich	[°C]	stat. dyn.	-70–200 -200–200	-200–200 -269–250	-40–180 -40–180
Lastwechsel	[–]	> 10 Mio. Zyklen	> 10 Mio. Zyklen	5 Mio.–10 Mio. Zyklen (je nach Typ und Amplitude)	100.000–100 Mio. Zyklen (je nach Typ und Amplitude)
Abmessungen	[mm]	je nach Typ	je nach Typ	22 × 10	18,29 × 4,45
Preis	[Euro]	67–450	91–270	290	auf Anfrage
Website	[–]	hbm.com	hbm.com	hbm.com	esa-messtechnik.de me-systeme.de zse.de

Anhang

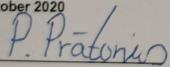
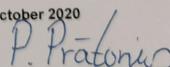
 CS INSTRUMENTS GmbH & Co. KG Werkszertifikat		 CS INSTRUMENTS GmbH & Co. KG Calibration certificate																	
Gerät:	VA525	Device:	VA525																
Serien Nr.:	41204871	Serial number:	41204871																
Test Bedingungen: Referenz Volumenstrom: Trommelgaszähler Rohrinnendurchmesser: 5,0 mm Testmedium: Air Mediumtemperatur: 18...26 °C Messergebnisse im CS-Labor:** bezogen auf 1000 hPa, 20°C, DIN 1945, ISO 1217																			
<table border="1"> <thead> <tr> <th>Sollwert [ltr/min]*</th> <th>zulässige Toleranz</th> <th>Istwert [ltr/min]*</th> <th>Bewertung</th> </tr> </thead> <tbody> <tr> <td>3,05</td> <td>±1,5% v.Mw. ±0,3% v.E.</td> <td>2,93</td> <td>o.k.</td> </tr> <tr> <td>13,25</td> <td>±1,5% v.Mw. ±0,3% v.E.</td> <td>13,04</td> <td>o.k.</td> </tr> <tr> <td>38,26</td> <td>±1,5% v.Mw. ±0,3% v.E.</td> <td>38,08</td> <td>o.k.</td> </tr> </tbody> </table> <p>***v. Mw. = vom Messwert; v. E. = vom Endwert. Endwert bei Durchmesser 5,0 mm = 55 ltr/min</p>				Sollwert [ltr/min]*	zulässige Toleranz	Istwert [ltr/min]*	Bewertung	3,05	±1,5% v.Mw. ±0,3% v.E.	2,93	o.k.	13,25	±1,5% v.Mw. ±0,3% v.E.	13,04	o.k.	38,26	±1,5% v.Mw. ±0,3% v.E.	38,08	o.k.
Sollwert [ltr/min]*	zulässige Toleranz	Istwert [ltr/min]*	Bewertung																
3,05	±1,5% v.Mw. ±0,3% v.E.	2,93	o.k.																
13,25	±1,5% v.Mw. ±0,3% v.E.	13,04	o.k.																
38,26	±1,5% v.Mw. ±0,3% v.E.	38,08	o.k.																
Kundenspezifische Einstellungen bei Auslieferung: Rohrinnendurchmesser: 5,0 mm Gas: Air Volumenstrom Einheit: ltr/min Bezugsnorm: ISO1217, DIN 1945: 20°C, 1000 hPa Analog Ausgangssignal Durchfluss Skalierung: " 4 mA 0,0 ltr/min 20 mA 50,0 ltr/min																			
<small>* Bitte entnehmen Sie der Bedienungsanleitung des Sensors die Skalierung des Analogausgangs in Volumenstrom-Einheiten wie m³/h, m³/min, kg/h, cfm etc.</small>																			
Datum: 6. Oktober 2020	Prüfer: PP	Date: 6. October 2020	Inspector: PP																
Unterschrift: 		Signature: 																	
CS Instruments GmbH & Co. KG - Gewerbehof 14, D-24955 Harrislee www.cs-instruments.com		CS Instruments GmbH & Co. KG - Gewerbehof 14, D-24955 Harrislee www.cs-instruments.com																	

Abbildung 79: VA-525 Kalibrierzertifikat

Ludwig, Daniel

Von: Ludwig, Daniel
Gesendet: Montag, 8. Februar 2021 12:34
An: Geweke, Martin; Wittkowski, Stefan; Hannappel, Marc
Betreff: Cloudnutzung.. eher nicht

Sehr geehrtes Team,

Ich hatte gerade ein langes informatives Telefonat mit einem ITSC Mitarbeiter.

Vorweg nenne ich meine Empfehlung, welche die Einrichtung eines lokalen Serverraums, für alle VT-Praktika ist. Als Grundlage dient meine folgendes, Bewertungsrankings (die durch den folgenden Volltext begründet ist):

Lokale Servereinrichtung	+++++
EMIL	+++
HAW-Cloud	+

Anmerkung: Cloudcomputing existiert seitdem es Computer gibt und ist nur jetzt auch Breit, aufgrund von etablierten Infrastrukturen, verfügbar. Die Studenten von morgen werden die Cloudcomputing Kompetenzen damit mit hoher wahrscheinlich anderorts (privat, so wie auch ich) erwerben, daher wäre ein Verzicht an dieser Stelle nicht „verheerend“. 😊

HAW-Cloud (ursprüngliche Idee, siehe PS)

Die ursprüngliche Idee funktioniert so (wie privat mit OneDrive) nicht. Die Cloud der HAW ist generell Nutzerbezogen. Nicht natürliche Personen, sprich Labor/Labor Accounts dürfen laut Richtlinien der HAW keinen HAW-Account haben. Auch ein externer, z.B. ein Professor im Falle einer Kooperation, dürfte offiziell keinen Zugang zu dem Pfad der HAW Cloud haben.

Einen Pfad in einem Programm, LabVIEW, zu etablieren würde vllt. funktionieren, wenn ein Teammitglied sich an den Versuchsrechnern anmeldet und daraufhin ein Pfadverzeichnis veruscht zu generieren, **doch das ist natürlich ein no go**, denn Studenten würden infolgedessen ja mit dem Acc. von z.B. Marc, an den Labor PC's, operieren.

Abgewandelte Idee, mit Cloud Nutzung

Wenn die HAW-Cloud genutzt werden soll, dann müssten Supervisor (Profs./Dipl.-Ing.) eine Struktur in der Cloud, auf Ihrem Account einrichten, d.h.:

„MarcsCloud“/MVT/Wirbelschicht AND ...

„MarcsCloud“/CVT/AdiabaterReaktor AND ..

„StefansCloud“/TWT/Waermuebertrager AND ...

Option #1: Daraufhin müssten diese Ordner freigegeben werden, indem jeder Studie einen individuellen Link erhält. Voraussetzung dafür ist, dass die Studies sich bereits einmal auf der HAW-Cloud angemeldet haben.

- Vorteil: Studenten, die nicht die Vorlesung besuchen ist es nicht möglich Daten up/down zu laden, durch personalisierte Links
- Nachteil: Jeder Studie muss angeschrieben werden (wie genau ist mir unbekannt)

Anhang

Option #2: Das Teammitglied generiert einen Link (mit Definition eines Ablaufdatums des Links möglich) und dieser wird an alle potenziellen Teilnehmer gesendet.

- Nachteil: Studenten, die nicht die Vorlesung besuchen, ist es möglich Daten up/down zu laden
- Vorteil: Nicht jeder Student muss angeschrieben werden, durch globale Verteilung, z.B. bei einer Einführungsveranstaltung

Ich persönlich, so wie der ITSC Mitarbeiter, würde davon abraten, denn die „**Kosten**“ übersteigen dem **Nutzen**, meiner Meinung nach.

Nutzung von EMIL (Empfehlung des ITSC)

Die Empfehlung des ITSC Mitarbeiter ist die Nutzung von Emil. Laut ITSC (entzieht sich meiner Kenntnis, da ich nur die studentische Seite der emil Welt kenne) Haben Dozenten genaue Infos über die Daten, wann geuploaded etc., und Funktionen wie Kommentar etc..

Lokale Servereinrichtung (über Frau Matych; alternative Methode)

Meine ersten Schritte auf dieser Reise waren in Richtung Frau Matych. Frau Matych hat angemerkt, dass es ihr möglich wäre einen lokalen Serverraum einzurichten. Die von Ihnen/uns erwünschten Verzeichnisstrukturen wären möglich. Alle Programme (LabVIEW heute, Datenbank und Applikationen im Rahmen von Industrie 4.0 und BigData morgen 😊) könnten darauf referenziert werden. Die administrativen Rechte können möglicherweise relativ einfach eingerichtet werden.

Ich wünsche Ihnen einen super Start in die Woche!

Mit allerbesten Grüßen,

Daniel Ludwig

PS

a. PS Cloudverzeichnis

i. Die Anforderungen sind die Folgenden:

1. Verzeichnisreferenzierung wie mit der Nutzung von OneDrive
2. Supervisor (wissenschaftliche Mitarbeiter, Professoren, ggf. Tutoren) sollen Admin-/Masterrechte haben.
3. Labor/Labor (Dialog mit Frau Matych wird unverzüglich erfolgen) sowie alle anderen Accounts sollen lediglich Up-/Downloadrechte besitzen.

ii. In Absprache mit Stefan werde ich gleich eine Mail, in Bezug auf ein gewünschtes Cloudverzeichnis, an Frau Matych senden. Die Anforderung, ist eine Handhabung wie mit OneDrive, d.h. es soll ein Verzeichnis/Ordnerpfad geben, auf den in LabVIEW referenziert werden kann, z.B. VT-Labor/MVT/Filterkuchenversuchsstand/Datalogs AND Protokolle. (Mit AND sind mehrere Ordner, im Verzeichnis Filterkuchenversuchsstand, gemeint)

iii. Meine Strukturempfehlung für die Verzeichnisse wäre die folgende:

1. VT-Labor/MVT AND TVT AND CVT (AND MSRP; Mess-/Steuer-/Regelungstechnik Praktikum),
2. UT-Labor/... ,
3. FoodScience (FS)-Labor/... ,
4. ANS/...

Anhang der Schnittstellenprogrammierung

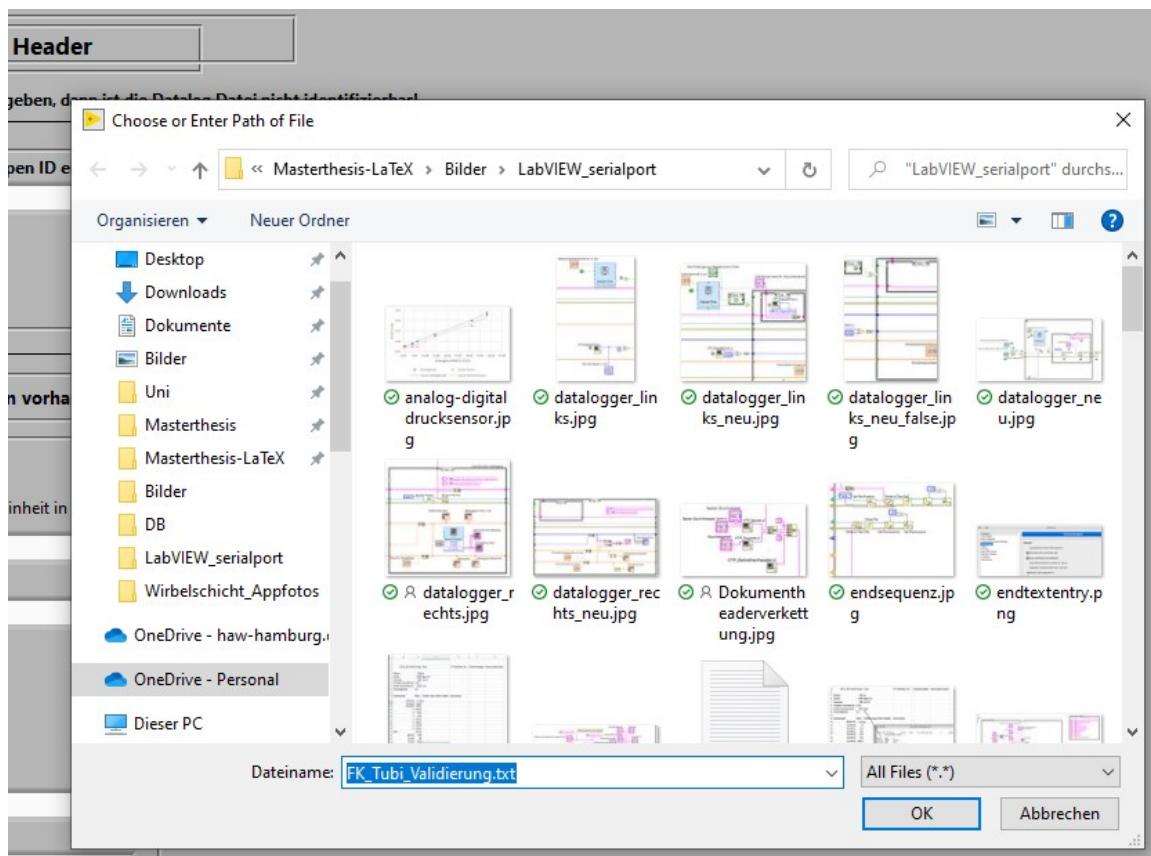


Abbildung 82: Dateidialog

Anhang

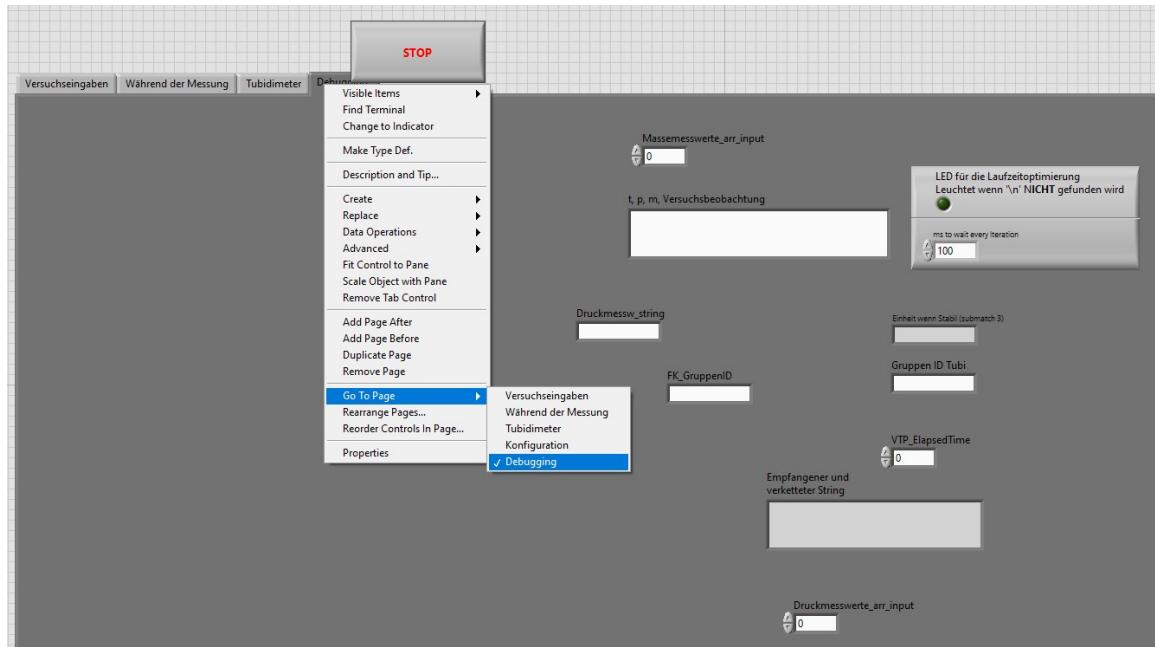


Abbildung 83: Go to hidden tabs

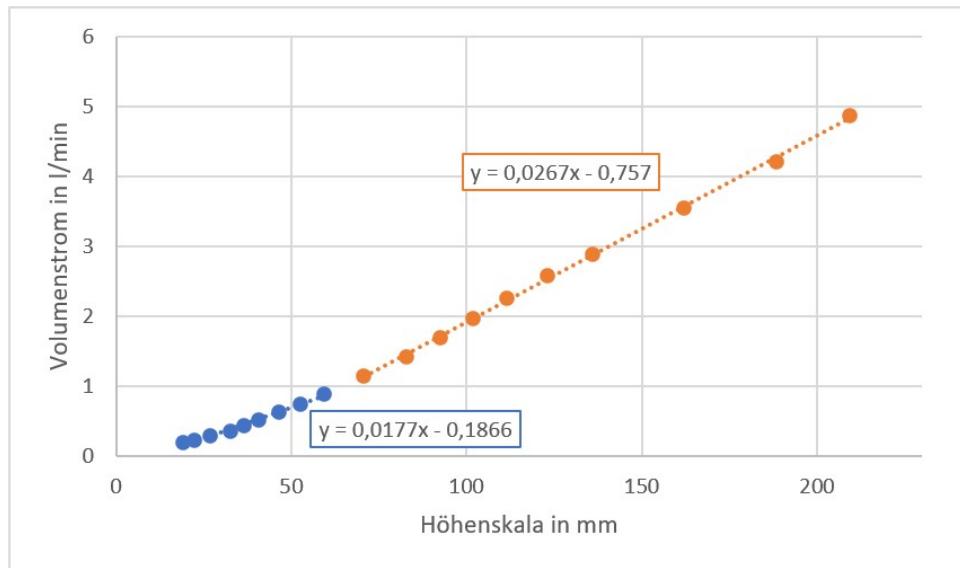


Abbildung 84: Schwebekörperdurchflussmesser ($m=0,6585$ g) mit einer Messspanne von 0,185 bis 5 l/min, zzgl. Approximationen der Kennlinie

Tabelle 18: Tabelle mit den Werten aller Sensordiagramme

Index	kleiner SKDM in mm	großer SKDM in mm	SKDM Umrechnungsfunktionen
1	19,2		$y=0,0177x-0,1866$
2	59		$y=0,0177x-0,1866$
3	189		$y=0,0177x-0,1866$
4		45	$y=0192x-2,0927$
5		150	$y=0,2577x-7,4485$
6		177	$y=0,2577x-7,4485$
7		85	$y=0,2577x-7,4485$
8		31,5	$y=0192x-2,0927$

Tabelle 19: Tabelle mit den Werten Aller Sensordiagramme

Index	Drucksensor in bar	Volumen- stromsensor in l/min	Manometer- differenz in mm	SKDM in l/min	Manometer- druck in bar	absolute Volumenstrom- messungsdifferenz in $ \frac{1}{\text{min}} $
1	0,99	0,03	2	0,99	0,15	0,12
2	0,99	1,09	7	0,99	0,86	0,23
3	0,99	4,17	32	0,99	4,29	0,12
4	0,99	6,97	52	1,00	6,55	0,42
5		34,83	253	1,01	31,21	3,62
6		44,9	327	1,02	38,16	6,74
7		15,65	118	1,00	14,46	1,19
8	0,99	3,98	32	0,99	3,96	0,02

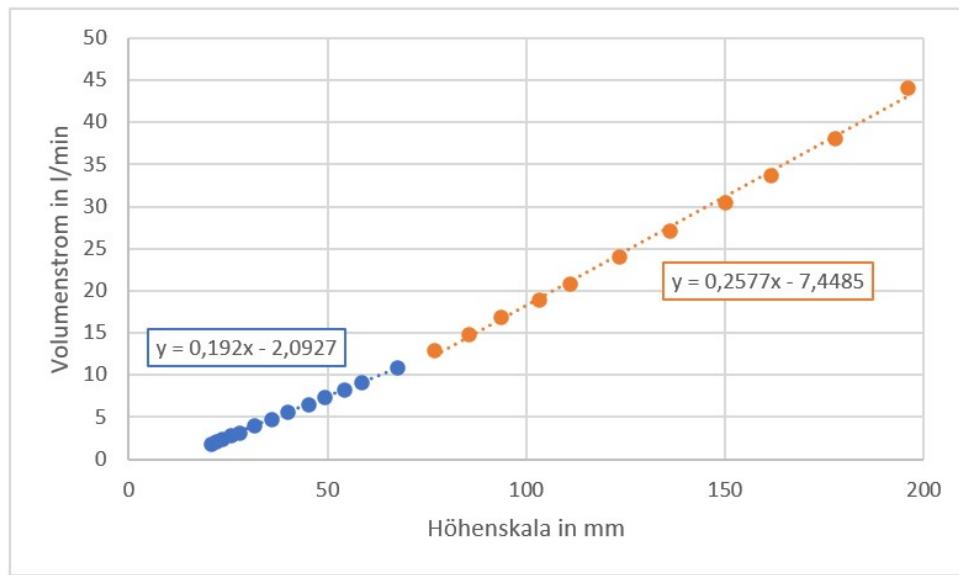


Abbildung 85: Schwebekörperdurchflussmesser ($m=2,225$ g) mit einer Messspanne von $1,86$ l/-min bis 50 l/min, zzgl. Approximationenfunktionen der Kennlinie

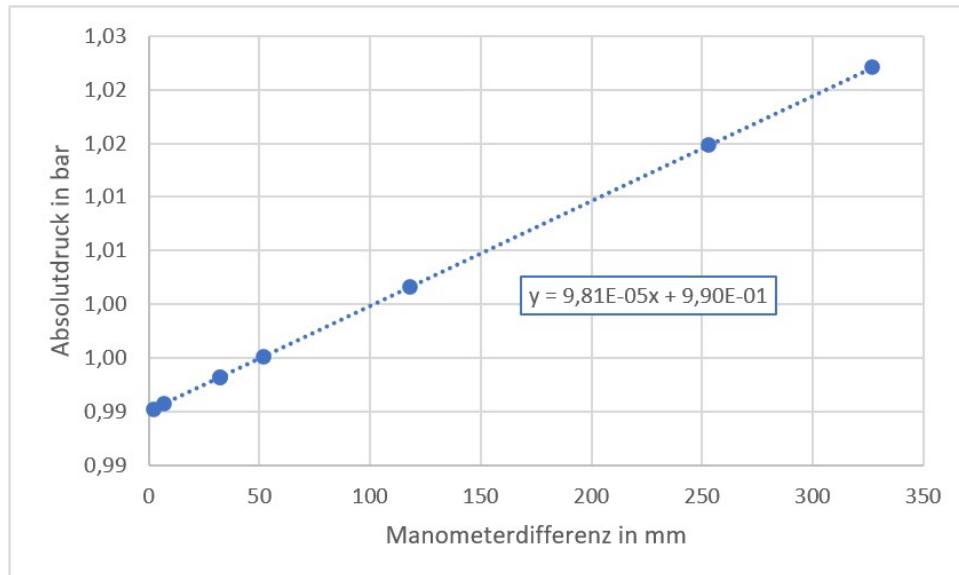


Abbildung 86: Manometerkennline; Druckdifferenz zzgl. 0,99 bar, laut digitalem Drucksensor am 17.12.20, um 13:02 Uhr