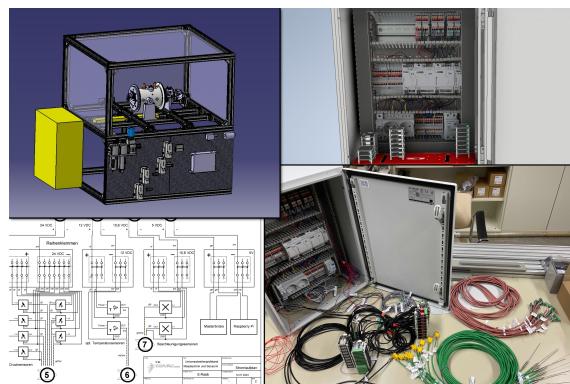


IFP: Messtechnik am Universalwellenprüfstand

Konzeption, Beschaffung, Aufbau und Datenerfassung



Bericht

im Rahmen des Studiengangs **M.Sc. Mechatronik**

vorgelegt von
SteffenRaab
746495

ausgegeben und betreut von
Prof. Dr.-Ing. Brita Pyttel

Darmstadt, den 15.01.2023

Inhaltsverzeichnis

1 Einleitung	3
2 Konzept des Messsystems	4
2.1 Vorgesehene Messtechnik	4
2.2 Positionierung der Sensorik	5
2.3 Iterativer Prozess	6
3 Auswahl und Beschaffung	7
3.1 Vorauswahl der Komponenten	7
3.2 Nutzwertanalyse	8
3.3 Ausgewählte Komponenten	10
3.4 Beschaffung	10
4 Modulares Messsystem	11
4.1 Anforderungen	11
4.2 Konzept	11
4.3 Umsetzung	13
5 Inbetriebnahme des Messsystems	15
5.1 Installation am Prüfstand	15
5.2 Einrichtung der Messsoftware	16
5.3 Durchführung und Auswertung von Messungen	18
6 Zusammenfassung und Ausblick	21

1 Einleitung

Im Rahmen des Projektes „Universalwellenprüfstand“ an der Hochschule Darmstadt soll ein Prüfstandsaufbau erfolgen, um eine luftgelagerte Welle genauer zu untersuchen. Bei der Welle handelt es sich um ein Design, welches in einer Mikrohybridturbine[1] zum Einsatz kommen wird. Die Mikrohybridturbine ist eine Kombination aus Gas- und ORC-Turbine, welche beide auf einer Welle arbeiten sollen. Ziel ist es die mechanischen und thermischen Belastungen der Welle genauer zu untersuchen und daraus notwendige Verbesserungen abzuleiten. Ebenfalls soll untersucht werden, ob eine Luftlagerung für diesen speziellen Anwendungsfall geeignet ist und auch in Serie eingesetzt werden kann.

Die Aufgabenstellung im Rahmen des „Integrierten Forschungsprojektes“ ist die Bereitstellung eines Messsystems. Um die Belastungen des Prüfstandes ermitteln zu können, muss entsprechende Sensorik und Messtechnik verbaut werden. Dies ist die primäre Aufgabenstellung des *IFPs*.

Während der Bearbeitungszeit wurde die Verwendung des Messsystems für ein weiteres Projekt beschlossen und daher musste das bereits erarbeitete Konzept erweitert werden. Somit wurde die Aufgabenstellung erweitert, um die neuen Anforderungen zu erfüllen:

- Konzept des Messsystems
- Auswahl und Beschaffung des Messsystems
- Modulares Messsystems
- Inbetriebnahme des Messsystems

Ziel dieses Berichtes ist es, den Ablauf des *IFPs* zu dokumentieren und die durchgeführten Bearbeitungsschritte aufzuzeigen.

2 Konzept des Messsystems

In diesem Abschnitt wird auf die Ermittlung der notwendigen Messgrößen eingegangen.

Die Bestimmung der Messgrößen fand im Rahmen mehrerer Arbeitsgruppentreffen innerhalb des Gesamtprojektes statt. Hierfür wurde gemeinsam die Anforderungen an die Messtechnik und Sensorik festgelegt. In einem ersten Schritt wurden Handskizzen im Blockschaltbildformat erarbeitet und ein Systemkonzept grafisch ausgearbeitet. Parallel dazu wurde bereits durch andere Projektteilnehmer ein CAD-Modell für die Konstruktion angefertigt.

2.1 Vorgesehene Messtechnik

Zunächst sollen die aufzunehmenden physikalischen Messgrößen dargestellt werden:

- Temperatur:
 - Lagerluft
 - Kühlkreislauf
 - Wellentemperatur
- Druck
 - Lagerluft
- Drehzahl
- Beschleunigung
- Gasgeschwindigkeit

2.2 Positionierung der Sensorik

Die parallel stattfindende Konstruktion machte es erforderlich ein erstes Konzept vorzulegen und darauf beruhend ebenfalls die Positionierung der Sensorik festzulegen:

Universalwellenprüfstand

Blockschaltbild mit Messstellen

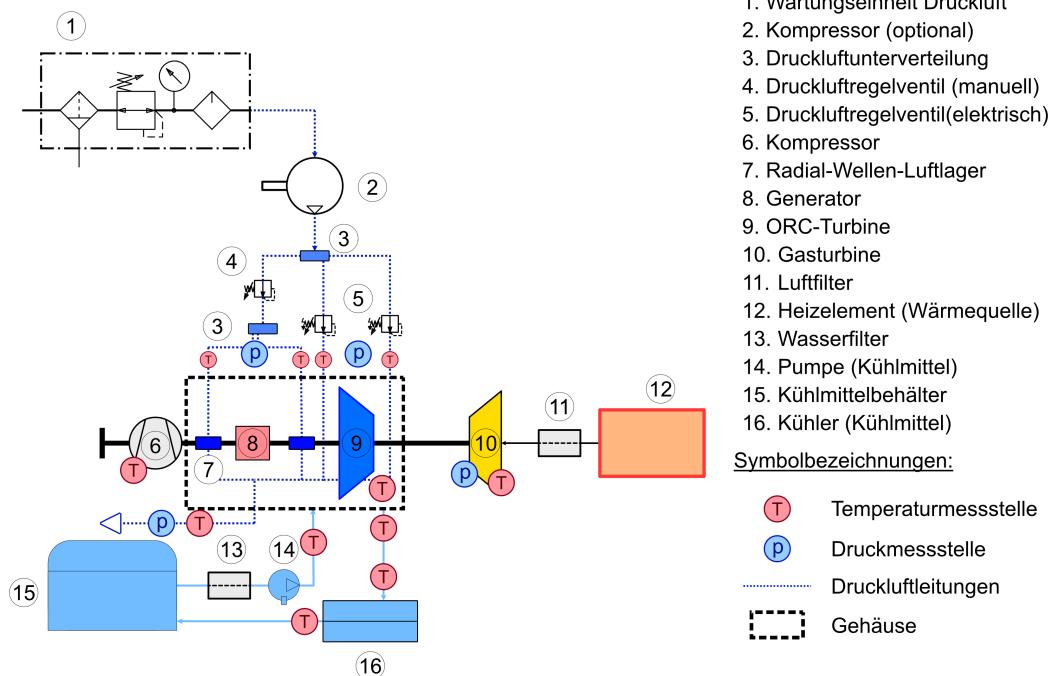


Abbildung 2.1: Blockschaltbild des Universalwellenprüfstandes

In Abbildung 2.1 ist der Aufbau der Einzelkomponenten des Universalwellenprüfstandes erkennbar. Ebenfalls wird die Aktorik und Sensorik des Prüfstandes abgebildet. Diese Darstellung wurde ebenfalls in Arbeitsgruppentreffen erstellt, um die Positionierung der Sensorik festlegen zu können. Nachdem dieser Schritt abgeschlossen war, konnten im CAD die Positionen am realen Prüfstand festgelegt werden. Hierbei war erneut eine enge Zusammenarbeit mit dem Konstrukteur von Nöten.

2.3 Iterativer Prozess

Zusätzlich gilt es zu erwähnen, dass die gesamte Festlegung der Messtechnik ein iterativer Prozess war. Während der Arbeitsgruppentreffen wurden mehrere Änderungen bezüglich der Auswahl, der Anzahl und auch der Positionierung der Sensorik festgelegt. Die dargestellten Abbildungen in diesem Bericht zeigen jeweils den aktuellsten Stand der Entwicklung der Messumgebung. Durch diese Vorgehensweise war es jedoch auch von Vorteil, dass bereits während der Planung eine Kostenaufstellung durchgeführt wurde und eine Kostenoptimierung möglich war. Ein Beispiel hierfür ist die Anpassung der Konstruktion an die ausgewählte Sensorik, um eine kostengünstige Umsetzung zu ermöglichen.

3 Auswahl und Beschaffung

In diesem Kapitel soll nun auf die Auswahl der verwendeten Messtechnik eingegangen werden. Zur Dokumentation der Auswahl wurde in einer projektweiten Stückliste gearbeitet, so dass dieser Prozess für alle Beteiligten zu jeder Zeit einsehbar war.

3.1 Vorauswahl der Komponenten

Im ersten Schritt wurde eine Online-Recherche durchgeführt, um mögliche Sensorik und Messaufnahmesysteme zu recherchieren. Für die Messaufnahme, also die Umwandlung und Aufbereitung der elektrischen Sensor-Signale in eine rechnergesteuerte Messumgebung, haben sich folgende drei umsetzbaren Möglichkeiten ergeben:

- National Instruments Hardware
- Tinkerforge-Hardware
- Mikrocontroller

Diese dargestellten Möglichkeiten sind jeweils mit unterschiedlichem Aufwand und Aufwendung verbunden, worauf noch genauer eingegangen wird.

Auch für die Messung der Temperaturen am Prüfstand haben sich verschiedene Typen von Sensoren recherchieren lassen. Folgende sinnvoll einsetzbare Typen konnten ermittelt werden:

- Thermoelemente
- PTC
- NTC

Diese Sensoren bieten jeweils Vor- und Nachteile, welche später noch genauer betrachtet werden.

Durch bereits vorhandene Sensorik an der Hochschule, konnten folgende Messgrößen in Bezug auf die Auswahl passender Sensorik für die Vorauswahl ausgeschlossen werden:

- Drucksensoren
- Drehzahlsensoren
- Beschleunigungssensoren
- Gasgeschwindigkeitssensoren

Für diese Positionen liefern die aufgeführten Messaufnehmer jedoch auch entsprechende Anschlussmöglichkeiten, um die Signale der Sensoren zu verarbeiten.

3.2 Nutzwertanalyse

Um einen übersichtlichen und möglichst objektiven Vergleich der Vorauswahl durchführen zu können wurde eine Nutzwertanalyse durchgeführt. Zunächst war es erforderlich Kriterien festzulegen, welche für die Umsetzung der Messtechnik am Prüfstand relevant sind. Vor allem die Auswahl der Sensoren zur Temperaturmessung und die Messaufnehmer mussten aufgrund der vielfältigen Angebote genauer verglichen werden.

Tabelle 3.1: Nutzwertanalyse für Auswahl der Temperatursensoren

Nutzwertanalyse		Vergleich: Thermoelement, PTC, NTC					
Kriterium	Gewichtung	Thermoelement		PTC		NTC	
		Bewertung (1-5)	Bewertung (gewichtet)	Bewertung (1-5)	Bewertung (gewichtet)	Bewertung (1-5)	Bewertung (gewichtet)
Linearität	15	3	45	5	75	1	15
Schnelligkeit	25	4	100	2	50	2	50
Genauigkeit	25	3	75	5	125	4	100
Temperaturbereich	20	5	100	3	60	1	20
Preis	15	2	30	4	60	5	75
Summe	100		350		370		260

Tabelle 3.2: Nutzwertanalyse für Auswahl der Messaufnehmer

Nutzwertanalyse		Vergleich Messhardware: National Instruments, Tinkerforge, beliebiger Microcontroller					
Kriterium	Gewichtung	NI-Hardware		Tinkerforge		μ -Controller	
		Bewertung (1-5)	Bewertung (gewichtet)	Bewertung (1-5)	Bewertung (gewichtet)	Bewertung (1-5)	Bewertung (gewichtet)
Messverfahren*	25	5	125	4	100	1	25
Erweiterbarkeit	25	3	75	5	125	2	50
Komfort & Handhabung*	20	4	80	5	100	1	20
Flexibilität*	10	3	30	5	50	5	50
Preis	20	1	20	4	80	5	100
Summe	100	●	330	●	455	●	245

* Anmerkungen:
- bewertet wurden Messgenauigkeit und Abtastrate
- einfache Installation und vielseitige Nutzungsmöglichkeiten sollten gewährleistet sein
- Wiederverwendbarkeit ist wichtig

Mit Hilfe der Nutzwertanalysen 3.1 und 3.2 wird visuell verdeutlicht, welche Vor- und Nachteile die einzelnen Optionen jeweils bieten. Die Gewichtungen wurden subjektiv festgelegt, die Bewertung der einzelnen Punkte wurde durch Recherche im Internet unterstützt. Hierzu wurden die folgenden Quellen recherchiert:

- Messaufnehmer:
 - National Instruments [2]
 - Tinkerforge [3]
- Temperaturmessungen:
 - TE-connectivity [4]
 - RCT-online [5]

Mit Hilfe dieser Analysen konnten nun die finalen Komponenten bestimmt werden. Die Temperaturmessstellen am Prüfstand werden je nach Anforderungen in Bezug auf den Messbereich, die Genauigkeit aber auch die Messgeschwindigkeit entweder als PTC oder aber als Thermoelement realisiert. Die Bezugsquelle der Messaufnehmer ist durch die Verwendung der Hardware von *Tinkerforge* bereits festgelegt, jedoch war es notwendig einen Anbieter für die ausgewählte Sensorik zu recherchieren.

3.3 Ausgewählte Komponenten

Nach abgeschlossener Nutzwertanalyse und Preisrecherche konnten folgende Hersteller und Lieferanten für die Sensorik und Messtechnik festgelegt werden:

- Sensoren: sensorshop24.de
- Messaufnehmer: tinkerforge.com

Die genannten Anbieter haben im Preis-Leistungs-Verhältnis überzeugt und auch das Angebot und die Anpassungsmöglichkeiten der notwendigen Komponenten ist vielfältig.

3.4 Beschaffung

Im nächsten Schritt wurde die ausgewählte Messtechnik bei den ausgewählten Händlern bestellt. Nach Eingang der Artikel an der Hochschule wurden diese einer Funktionsprüfung unterzogen, um die einwandfreie Funktion zu gewährleisten.

4 Modulares Messsystem

4.1 Anforderungen

Während der Bearbeitung wurde klar, dass eine modulare Lösung des Messsystems von Vorteil ist. Für ein weiteres Projekt, welches sich mit dem Aufbau eines Prüfstandes zur Untersuchung einer einfachen Dampfturbinengeometrie beschäftigt, wurde ebenfalls ein Messsystem gesucht. Aufgrund der engen Zusammenarbeit der Projektteilnehmer beider Projekte wurde festgelegt, das entwickelte Messsystem dort ebenfalls zu verwenden. Die dort vorhandene Sensorik kann ebenfalls mit den beschafften Komponenten verwendet werden.

Einige der ausgewählten und beschafften Komponenten beider Projekte, machten es erforderlich eine Spannungsversorgung bereitzustellen, da diese bereits digitalisierte Signale erzeugen. Auch eine zusätzliche und konstante Spannungsversorgung der Messaufnehmer von *Tinkerforge* ist grundsätzlich sinnvoll. Spannungsschwankungen können möglicherweise zu Abweichungen in den Messwerten führen. Zunächst wurden folgende notwendigen Anforderungen definiert:

- Anschluss an 230 V-Netz möglich
- Bereitstellung von Gleichspannungen 5 V, 12 V und 24 V
- Modularität, um Verwendung an anderen Prüfständen zu ermöglichen
- Sicherheit der Bedienenden

4.2 Konzept

Zunächst wurde ein Konzept erarbeitet, damit die gegebenen Anforderungen umgesetzt werden konnten. Ein besonderes Augenmerk lag hierbei auf der Einhaltung der Sicherheitsaspekte. Die Bedienersicherheit soll durch Unterbringung der Komponenten in einem abschließbaren Kompaktschalschrank erreicht werden.

Da der Anschluss ans 230V-Netz möglich sein soll, ist ein zusätzlicher Fehlerstromschalter geplant. Die Installation der geforderten Komponenten soll jederzeit änderbar sein, weswegen Tragschienenprofile (TH35) zum Einsatz kommen sollen. Eine saubere Führung der gesamten Verkabelung soll durch geschlitzte Kabelkanäle erfolgen. Auch Änderungen an der Verkabelung der Sensorik und Messtechnik ist gegebenenfalls beim Umrüsten nötig, daher sollen Installationsklemmen eingesetzt werden. Ebenfalls sollen zwei weitere Schutzkontaktsteckdosen im Schaltschrank verfügbar sein, um eventuelle Zusatzkomponenten mit handelsüblichen Schaltnetzteilen betreiben zu können.

Im nächsten Schritt wurde eine Vorauswahl der Komponenten getroffen, wobei bereits eine erste Kostenabschätzung erstellt werden konnte und gegebenenfalls bereits im Vorfeld preisgünstigere Alternativen ermittelt wurden. Diese Auswahl wurde wieder in der bereits angesprochenen Tabelle festgehalten. Mit Hilfe dieser Stückliste wurde ein Stromlaufplan erarbeitet, um die Bestellmengen zu ermitteln und den Aufbau des Messsystems zu unterstützen. Dieser ist in Abbildung 4.1 auszugsweise dargestellt.

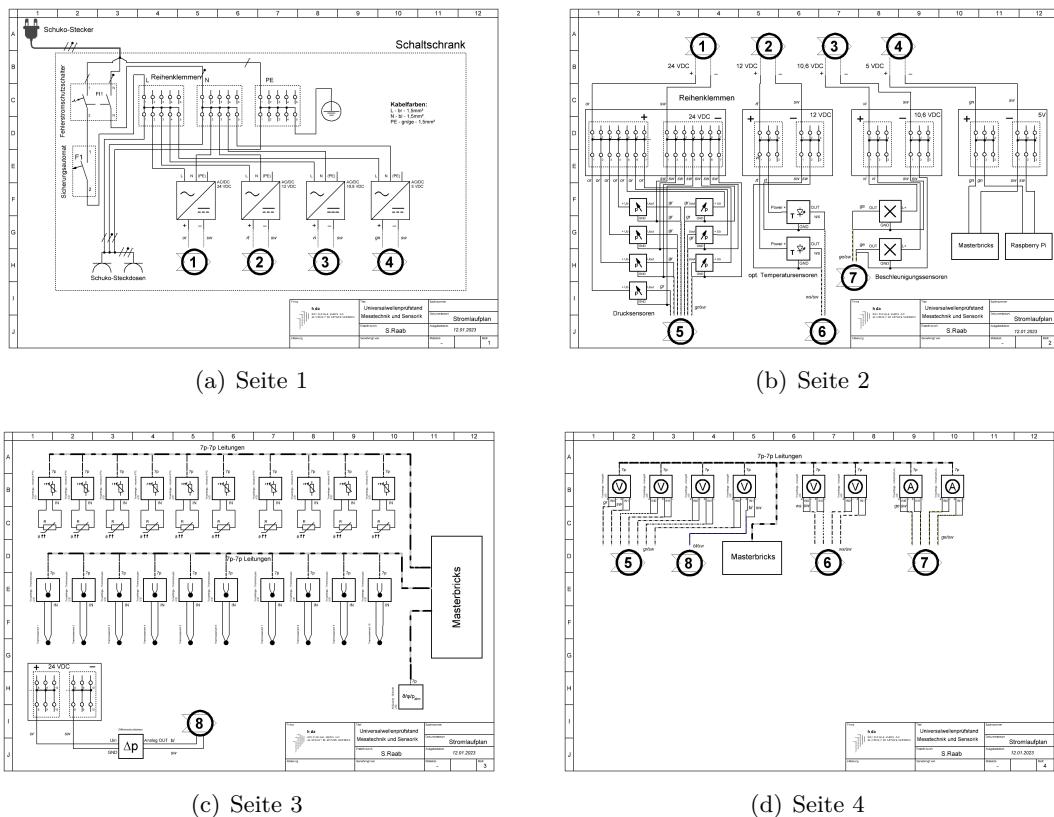


Abbildung 4.1: Stromlaufplan des Messschrankes

4.3 Umsetzung

Nach Fertigstellung des Konzeptes konnte nun die Beschaffung erfolgen und mit dem Aufbau begonnen werden. Im ersten Schritt wurde der Kompaktschalschrank zusammengebaut und mit den bestellten Komponenten bestückt.

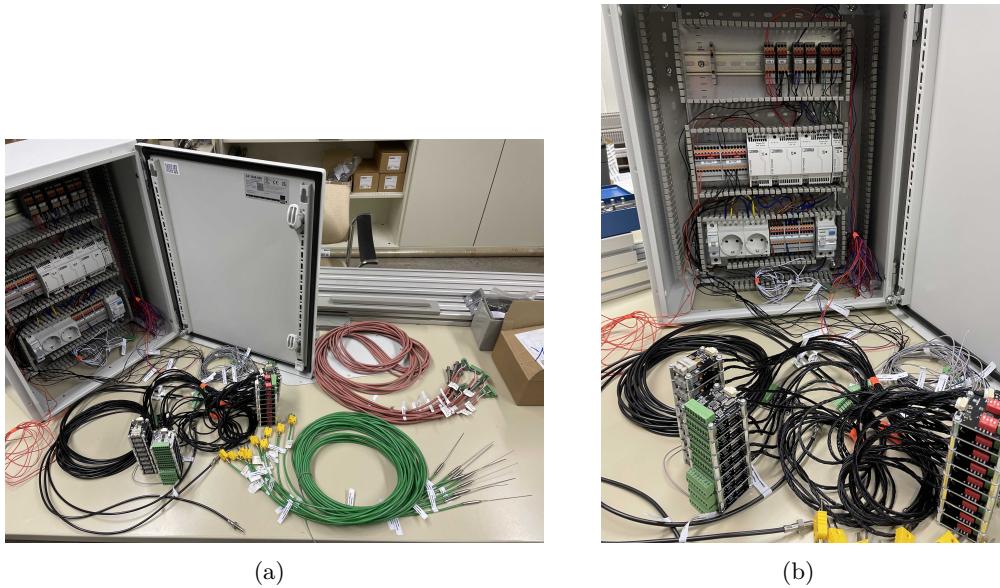


Abbildung 4.2: Teilweise aufgebautes Messsystem

In Abbildung 4.2 ist der derzeitige Aufbau des Messsystems dargestellt. Die Komponenten und die Verkabelung des Messschrankes ist gemäß dem Stromlaufplan (siehe: Abbildung: 4.1) erfolgt. Sämtliche Sensoren sind entsprechend ihrer vorgesehenen Einbauposition beschriftet und können somit einfach am Prüfstand installiert werden. Bei den gezeigten gestapelten Platinen handelt es sich um die *Tinkerforge*-Module, welche ebenfalls im Messschrank platziert werden sollen. Hierfür wurde eine Bodenplatte zur Montage der Module im Messschrank konstruiert:

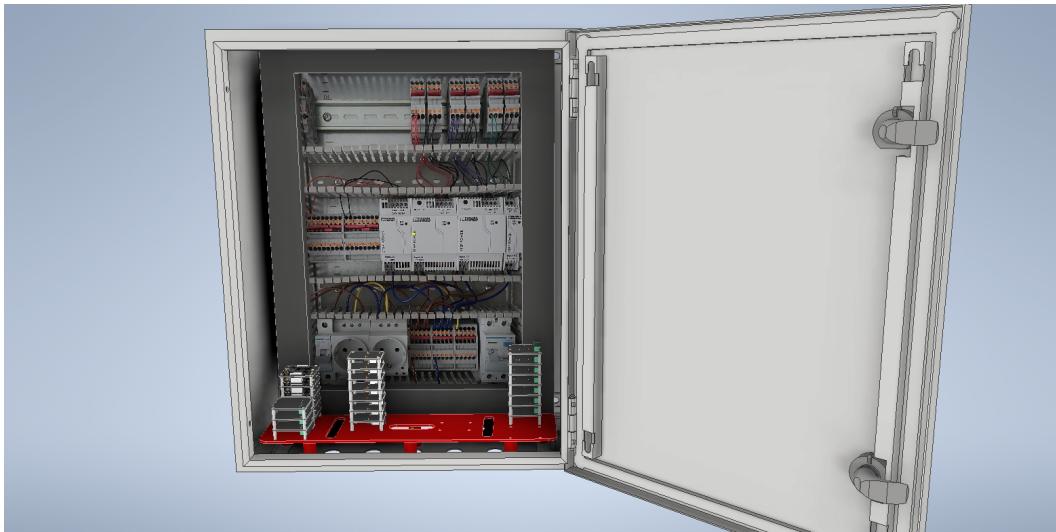


Abbildung 4.3: CAD-Modell: Platzierung der Montageplatte für *Tinkerforge*-Module

In Abbildung 4.3 ist die konstruierte Montageplatte in rot dargestellt. Diese befand sich zum Zeitpunkt der Berichterstellung noch in der Fertigung und konnte noch nicht im Messschrank installiert werden.

Während des Aufbaus ist die Idee entstanden ein zusätzliches Hutschienenprofil mit Reihenklemmen direkt an den Prüfständen zu installieren. Somit kann eine einfache Trennung des Messschrankes und der Sensorik erfolgen und die Messtechnik an einem anderen Prüfstand verwendet werden. Hierfür ist es jedoch erforderlich zusätzliche Komponenten nachzubestellen und diese Änderungen ebenfalls entsprechend zu dokumentieren (Schaltpläne, Beschriftungen, usw.) Diese Änderungen werden dann im IFWP „*Aufbau eines Datenloggers sowie Messdatenverwertung und Visualisierung*“ umgesetzt.

Im darauffolgenden Kapitel soll dennoch der Aufbau und die Inbetriebnahme des Messsystems durchgeführt und eine erste Testmessung gezeigt werden.

5 Inbetriebnahme des Messsystems

5.1 Installation am Prüfstand

Die zukünftige Positionierung der Sensorik und Messtechnik soll in diesem Abschnitt mit Hilfe von Bildausschnitten des CAD-Modells des Universalwellenprüfstandes dargestellt werden.

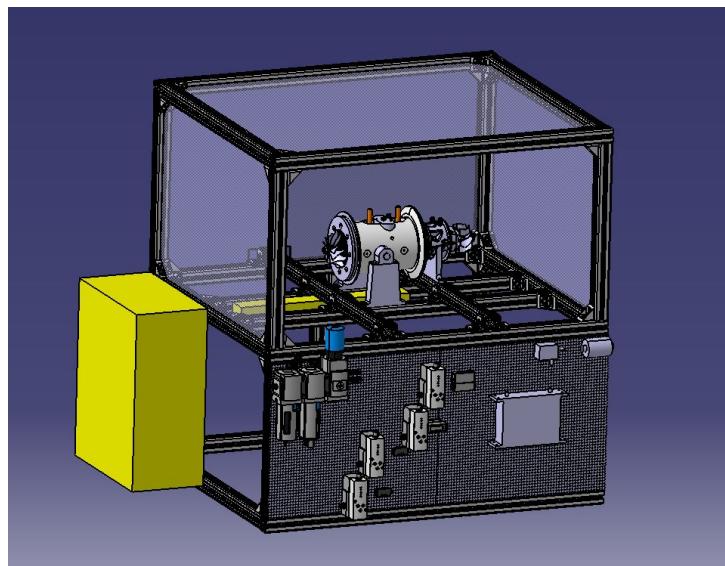


Abbildung 5.1: Universalwellenprüfstand in CATIA

In Abbildung 5.1 wird die Installation des Schaltschrankes sowie ein geplanter Kabelkanal mit den zusätzlich geplanten Reihenklemmen (in gelb dargestellt) gezeigt. Somit können die Sensoradern sauber verlegt werden und sind zusätzlich geschützt.

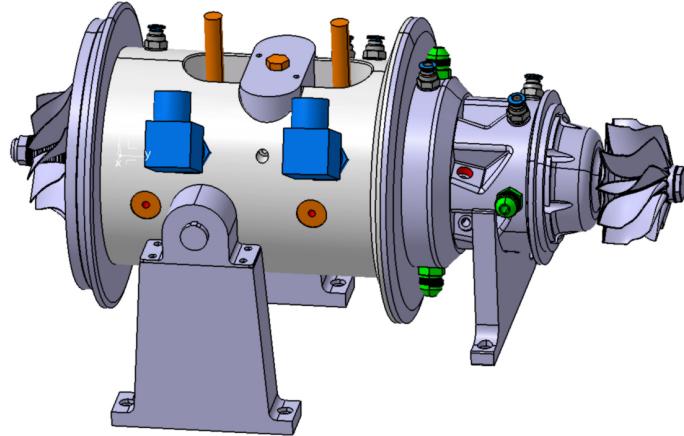


Abbildung 5.2: Platzierung der Sensoren am Turbinengehäuse

In Abbildung 5.2 sind nun die Positionen der einzelnen Sensoren im Turbinengehäuse dargestellt. In orange sind verschiedene Temperatursensoren dargestellt. Auch die Drucksensoren (hier blau) zur Überwachung des Luftdruckes der Luftlager sind erkennbar. Weitere Sensoren sind in Messblöcken geplant und nicht direkt am Turbinengehäuse installiert. Diese können beim Aufbau beliebig platziert werden. Ebenfalls sind Sensoren im Kühlkreislauf beschafft worden, wovon jedoch noch kein Aufbau bzw. ein CAD-Modell vorliegt.

5.2 Einrichtung der Messsoftware

Zur Durchführung der Messungen wurde auf die von *Tinkerforge* bereitgestellte Software *Brickv 2.4.23* zurückgegriffen. Mit dieser wurde zunächst die gesamte Messtechnik auf Funktion überprüft. Hierfür war es ebenfalls erforderlich bei einigen Komponenten ein Firmware-Update durchzuführen, um eine einwandfreie Funktionalität zu gewährleisten. Die Benutzeroberfläche von *Brickv 2.4.23* soll nun gezeigt werden.

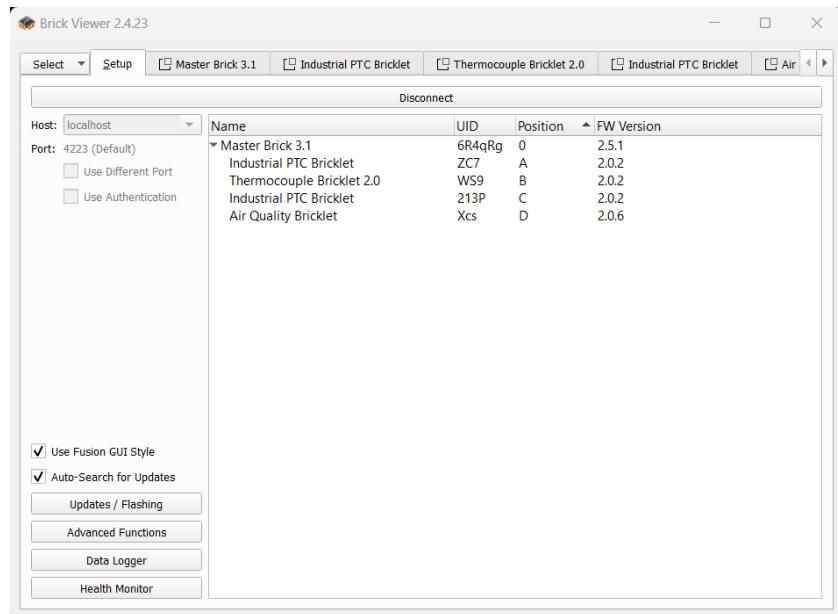


Abbildung 5.3: Benutzeroberfläche von *Brickv 2.4.23*

In Abbildung 5.3 ist die GUI von *Brickv 2.4.23* mit angeschlossenen und verbundenen *Tinkerforge*-Modulen dargestellt. Mit diesem Tool können bereits Sensorwerte ausgelesen werden, welche bei Temperatursensoren bereits Temperaturwerte liefern. Für die spätere Verwendung am Prüfstand muss gegebenenfalls eine Kalibrierung durchgeführt werden. Hier wird auch deutlich, dass sämtliche Module von *Tinkerforge* mit einer einzigartigen „UID“ versehen sind. Diese wurden entsprechend ausgelesen und tabellarisch dokumentiert.

Modulübersicht Tinkerforge												
Pos. 1 ist immer das oberste Modul!												
Master-Bricks			Industrial Dual Analog IN				Industrial Dual 0-20mA					
Pos.	UID	Sonstiges	Pos.	UID	Sensor 0	Sensor 1	Sonstiges	Pos.	UID	Sensor 0	Sensor 1	Sonstiges
1	6DbMMY		1	Ye9				1	ZLW			
2	6Lg19G		2	Ybj				2				
3	6DAfkj		3	23RH				3				
4	6LgwY		4	YaM				4				
5	6koAcl		5	Ybs				5				
6	6kpADw		6	Ycy				6				
7	6fd5hy		7	23RK				7				
8	6DnPcp		8	Ybx				8				
9			9					9				
10			10					10				
11			11					11				
Industrial PTC			Thermocouple									
Pos.	UID	Sensor	Pos.	UID	Sensor	Sonstiges	Pos.	UID	Sensor 0	Sensor 1	Sonstiges	
1	Ye9		1	Ye9								
2	Ybj		2	Ybj								
3	23RH		3	23RH								
4	YaM		4	YaM								
5	Ybs		5	Ybs								
6	Ycy		6	Ycy								
7	23RK		7	23RK								
8	Ybx		8	Ybx								
9			9	Rhf								
10			10	Wta								
11			11									

Abbildung 5.4: Auflistung und Position der verbauten *Tinkerforge*-Module

Abbildung 5.4 zeigt die Auflistung der verbauten *Tinkerforge*-Module. Diese Auflistung liegt als Excel-Dokument auf dem eingesetzten Cloud-Speicher des Projektes und kann somit von jedem eingesehen bzw. bearbeitet werden. Beim Aufbau sollen dann entsprechend die eingebauten Sensoren den Modulen zugeordnet werden. Diese Arbeit dient ebenfalls als Vorbereitung für das darauffolgende IFWP, da die UIDs für die Programmierung einer Visualisierung benötigt werden.

5.3 Durchführung und Auswertung von Messungen

Mit den angeschlossenen Modulen aus Abbildung 5.3 und den dazugehörigen Sensoren wird nun eine Testmessung durchgeführt. Zu diesem Zweck wurden folgende Sensoren an die passenden *Tinkerforge*-Module angeschlossen:

- Temperatur: PTC-Widerstand
- Temperatur: Thermoelement Typ K
- Temperatur und Luftfeuchte: Fertigmodul *Tinkerforge Air Quality Bricklet*

Mit dieser Konfiguration soll nun eine Testmessung erfolgen. Hierfür muss nun das Datatalogging entsprechend in *Brickv 2.4.23* noch eingestellt werden:

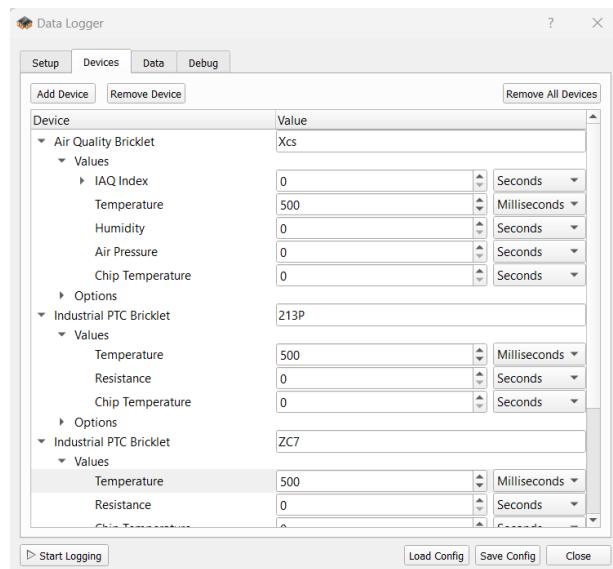


Abbildung 5.5: Auflistung und Position der verbauten *Tinkerforge*-Module

In Abbildung 5.5 ist die Benutzeroberfläche zur Einstellung des Dataloggings dargestellt. Folgende Zeitintervalle für das Abspeichern der Messwerte wurden festgelegt:

- Temperaturen: 500 ms
- Luftfeuchte: 1000 ms

Ebenfalls wurde vorher in einem anderen Menü-Tab das Abspeichern der Messwerte als CSV-Datei eingestellt, um eine spätere Auswertung zu ermöglichen. Das Datalogging wurde nun gestartet und während der Messung wurden die Sensoren mit der Hand umschlossen, um eine Änderung der Messwerte zu erreichen. Die aufgenommenen Messwerte sollen nun auszugsweise dargestellt werden:

TIME	NAME	UID	VAR	RAW	UNIT
12.01.23 12:36	Air Quality Bri Xcs		Temperature	2090	°C/100
12.01.23 12:36	Industrial PTC ZC7		Temperature	1992	°C/100
12.01.23 12:36	Industrial PTC 213P		Temperature	1996	°C/100
12.01.23 12:36	Air Quality Bri Xcs		Temperature	2091	°C/100
12.01.23 12:36	Industrial PTC 213P		Temperature	1996	°C/100
12.01.23 12:36	Industrial PTC ZC7		Temperature	1992	°C/100
12.01.23 12:36	Air Quality Bri Xcs		Humidity	4993	%RH/100
12.01.23 12:36	Air Quality Bri Xcs		Temperature	2091	°C/100
12.01.23 12:36	Industrial PTC ZC7		Temperature	1993	°C/100
12.01.23 12:36	Industrial PTC 213P		Temperature	1996	°C/100
12.01.23 12:36	Air Quality Bri Xcs		Humidity	4993	%RH/100
12.01.23 12:36	Air Quality Bri Xcs		Temperature	2091	°C/100
...

Abbildung 5.6: Auszug aus der Datalog CSV-Datei

In Abbildung 5.6 wird deutlich, dass eine Aufbereitung der Messdaten stattfinden muss, um eine sinnvolle Auswertung durchführen zu können. Am späteren Prüfstand ist es notwendig die Position der Sensoren den passenden „UIDs“ der Module zuzuordnen. Hierfür kann wieder auf die bereits gezeigte Auflistung (siehe 5.4) zurückgegriffen werden. Auch für Visualisierungen müssen die Daten getrennt und die „RAW“-Werte korrigiert werden. Im Rahmen der Testmessung und Visualisierung wurden die Daten von Hand in *Excel* gefiltert und für jedes Modul ein separates Tabellenblatt erstellt. Anschließend wurden die Messdaten in Diagrammen visualisiert:

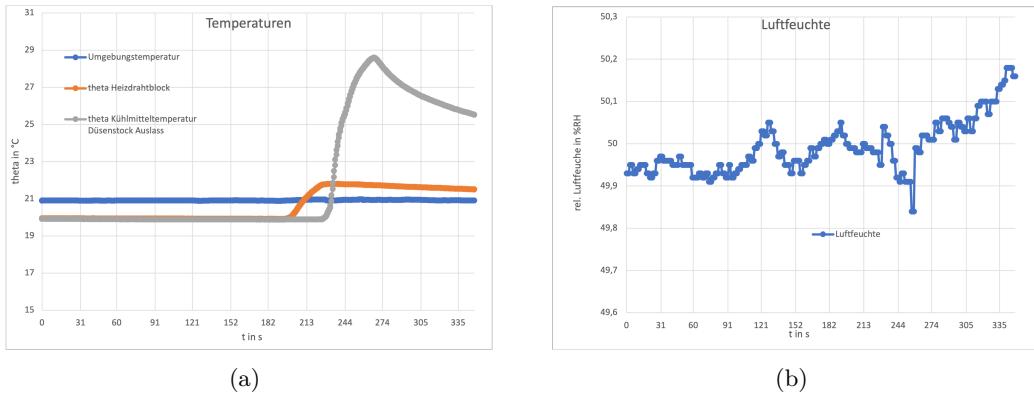


Abbildung 5.7: Visualisierung der Testmessung

In Abbildung 5.7 werden die aufgenommen und korrigierten Messwerte der Testmessung nun dargestellt. Anzumerken ist, dass die Sensoren nicht zeitgleich erwärmt wurden, sondern nacheinander. Die erwarteten Änderungen der Temperaturen sind zu erkennen. Besonders gut erkennt man hier den Unterschied der Messgeschwindigkeit zwischen einem PTC-Widerstand(orange) und einem Thermoelement(grau). Auch die Abkühlung des PTC-Widerstandes dauert wesentlich länger. Die Luftfeuchte ist über die Messzeit relativ konstant, was zu erwarten war. Möglicherweise ist der Anstieg am Ende durch die Bewegung im Raum während der Erwärmung der Temperaturmessstellen zu erklären. Der Ausschlag der Messwerte ist jedoch so gering und generell von Schwankungen betroffen, dass es sich auch lediglich um Messungenauigkeiten handeln kann.

6 Zusammenfassung und Ausblick

Die Aufgabenstellung des *IFPs* konnte fast vollständig erfüllt werden. Das Messsystem zur Verwendung am Universalwellenprüfstandes wurde entsprechend der Anforderungen ausgelegt und folgende Ergebnisse wurden erzielt:

- Konzepterarbeitung
- Beschaffung der Komponenten
- Aufbau eines modularen Messsystems
- Inbetriebnahme für Testzwecke

Besonders hervorzuheben ist die stetige Entwicklung und Änderung des Konzeptes, welches im Laufe der Aufgabenstellung mehrfach erweitert und verfeinert werden musste. Einige Änderungen wurden erst kurz vor Ende des *IFPs* beschlossen, welche jedoch essentielle Verbesserung der Modularität und Vielseitigkeit des Messsystems versprechen. Dieser Prozess ist bedingt durch die ständige Weiterentwicklung des Prüfstandes und erforderte daher eine hohe Flexibilität in der Umsetzung des Messsystemkonzeptes. Hierbei spielt vor allem eine gute Dokumentation in Form von Stücklisten, Blockschaltbildern, Konstruktionsmodellen oder aber auch dieser Bericht eine große Rolle. Somit wäre es möglich eine Übergabe der Aufgabe nach Abschluss des *IFPs* durchzuführen und den Einstieg in die Weiterentwicklung zu erleichtern.

Aufbauend auf die Umsetzung des Messsystems wären folgende Erweiterungen und Weiterentwicklungen denkbar:

- teil-automatisierte Datenauswertung mit *Microsoft Excel* oder *MATLAB*
- Entwicklung einer Software-Messumgebung mit Visualisierung der Messdaten
- Steuerung und Regelung von Aktorik mit weiteren *Tinkerforge*-Modulen
- Weiterentwicklung zur vollwertigen Software zur Überwachung und Steuerung des Prüfstandes

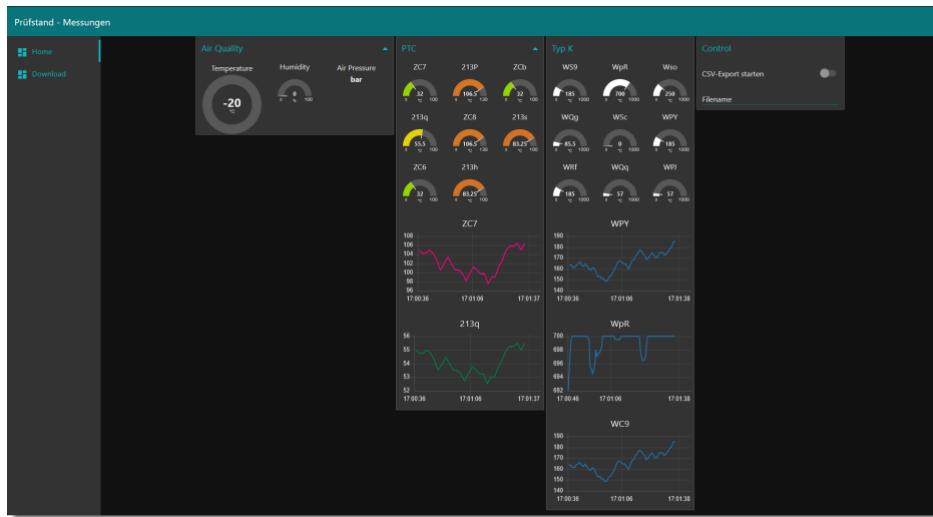
Alles in allem konnte das *IFP* erfolgreich bearbeitet werden, hat tiefgreifende Einblicke in die Entwicklung einer Messumgebung ermöglicht und hat dazu beigetragen, dass viele neue Erkenntnisse gewonnen werden konnten. Besonders interessant war das Arbeiten mit sich ständig verändernden Anforderungen und den daraus notwendigen Anpassungen. Auch die vielfältigen Möglichkeiten zur Weiterentwicklung des entstandenen Messsystems haben einen tiefen Einblick ins Arbeiten in die Forschung vermitteln können.

Literatur

- [1] Dipl.-Ing. (FH) Michael Cremer; Prof.Dr. Wolfgang Heddrich; M.Eng. Lukas S. Zajac; „Die Mikrohybridturbine - ein modulares Konzept für die Energiewende“. In: *VDI - Technik und Mensch* (2021), S. 10–12. URL: https://www.vdi.de/fileadmin/pages/vdi_de/redakteure/vor_ort/bv/bv-frankfurt-darmstadt/dateien/T_u_M_Strategie.pdf.
- [2] National Instruments. *PC-basierte Mess- und Steuersysteme*. Nov. 2022. URL: <https://www.ni.com/de-de/shop/pc-based-measurement-and-control-system.html>.
- [3] Tinkerforge. *Einstieg*. Nov. 2022. URL: <https://www.tinkerforge.com/de/doc/Primer.html#primer-bricks>.
- [4] TE connectivity. *Einen Temperatursensor auswählen*. Nov. 2022. URL: <https://www.te.com/deu-de/products/sensors/temperature-sensors/intersection/selecting-temperature-sensors.html>.
- [5] Reichelt Chemie Technik GmbH und Co. *Die elektrische Temperaturmessung - schnell und präzise*. Nov. 2022. URL: <https://www.rct-online.de/magazin/elektrische-temperaturmessung/>.

Datenlogging, Messdatenverwertung und Visualisierung

Anleitung zur Installation, Einrichtung und Verwendung am Universalwellenprüfstand



WiSe 2022/23

Erstellt von:

Steffen Raab

746495

Studiengang: M. Sc. Mechatronik

Betreut von:

Prof. Dr.-Ing. Brita Pyttel

Projekte:

Universalwellenprüfstand und 3kW-Dampfturbine

Inhalt

Kapitel 1 - Vorstellung und Datenfluss	3
Kapitel 2 – Installationsanleitung	4
Installation OS auf Raspberry PI (RPi)	4
Setup des Raspberry Pi.....	4
Einrichten der NodeRED-Flows	8
Kapitel 3 – Anwendungsbeispiel	9
Kapitel 4 – Data-Integrity	10
Backup der Messwerte.....	10
Backup und Restore	10
Kapitel 5 – Begleit- und Videodokumentation.....	11
Begleitdokumentation	11
Videodokumentation	11
Für User	11
Für Systembetreuer und Interessierte	11
Kapitel 6 – Ausblick	12
Quellen:.....	13
Installation des Betriebssystems - Raspbian OS:	13
Nützliche Links für die Verwendung der Befehlszeile via SSH:	13
Setup der TinkerForge-MQTT-Schnittstelle:	13
Setup der NodeRED-Umgebung:.....	13
Weiterführende Tools zum Testen, Konfigurieren und Simulieren der MQTT-Schnittstelle:	13
Anhang	14
Konfigurationsfile der MQTT-Schnittstelle:	14
Backup-Skript	14

Kapitel 1 - Vorstellung und Datenfluss

Mit Hilfe dieser Dokumentation soll vor allem die Nutzung des Messsystems am Universalwellenprüfstand oder 3kW-Prüfstand genauer beschrieben werden. Dieses Dokument enthält darüber hinaus auch wichtige Informationen für den zukünftigen Systembetreuer (Installationsanleitung, Begleitdokumente und Data-Integrity).

Als Einstieg soll zunächst der Datenfluss kurz dargestellt werden:

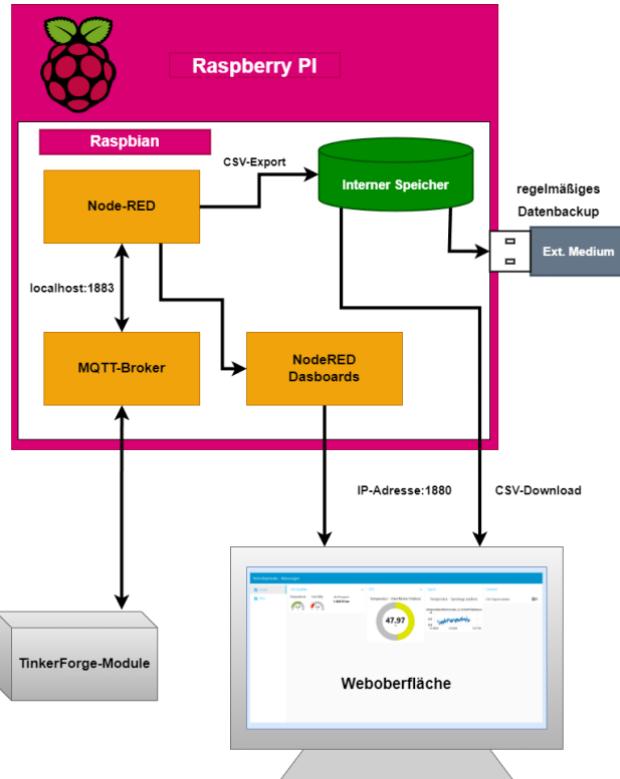


Abbildung 1: Data-Flow im Messsystem

Sämtliche Sensorik wird mit Hilfe von Messumformern von *TinkerForge* betrieben. Diese bieten die Möglichkeit die Messdaten, nach entsprechender Konfiguration, über das *Message Queuing Telemetry Transport* (MQTT) periodisch zu übermitteln. Diese Konfiguration wird mit Hilfe eines *Raspberry Pi* vorgenommen, welcher zeitgleich als MQTT-Broker fungiert. Dieser veröffentlicht die Daten über den Standard Port „1883“ via TCP-IP. Die Auswertung und Weiterverarbeitung der Daten geschieht mit Hilfe von *NodeRED*. Dies wird ebenfalls auf dem *Raspberry* installiert und stellt dafür eine Webapplikation zur Verfügung. Diese kann über einen beliebigen Internetbrowser aufgerufen werden und wird für die Weiterverarbeitung der Messdaten verwendet. Dort wird flow-basiert und unter Verwendung von JavaScript-Code die Anzeige und Speicherung der Messwerte programmiert. Ein webbasiertes-Dashboard wurde ebenfalls erstellt und bildet die Schnittstelle für den User der Messumgebung.

Die Messdaten werden, nach Starten der Messungen, als CSV-File mit Zeitstempeln auf den internen Speicher des *Raspberry Pis* abgelegt und können anschließend über das Dashboard heruntergeladen werden. Auch ein regelmäßiges Backup der Messdaten auf ein externes Speichermedium ist implementiert.

Kapitel 2 – Installationsanleitung

Dieses Kapitel richtet sich in erster Linie an den Systembetreuer. Sollten zukünftige Erweiterungen und Anpassungen notwendig sein, ist das hier dargestellte Vorgehen essenziell. Auch für den interessierten Nutzer ist es interessant das System und dessen Installation genauer zu verstehen. Alle Schritte werden unter Nutzung von Windows 11 durchgeführt und die genannte Software zur Installation ist teilweise nur für Windows verfügbar. Bei Verwendung eines anderen Betriebssystems muss auf andere Software zurückgegriffen werden.

Installation OS auf Raspberry PI (RPi)

Als Grundlage für die zu installierenden Programme ist eine *Raspbian Lite* Installation ausreichend. Dieses OS verzichtet auf eine grafische Nutzeroberfläche, welche nicht zwangsläufig benötigt wird. Alle Installationen können über die Befehlszeile durchgeführt werden und der Upload von Dateien ist via FTP-Client möglich. Die Installation kann aber auch problemlos mit einer *Raspbian* Installation mit GUI durchgeführt werden.

Anleitung – Step-by-step:

1. Download des [Raspberry Pi Imagers](#) (02.04.2023 13:46)
2. Starten, passende Micro SD-Karte einlegen und Konfigurieren des Imagers:
 - a. Betriebssystem wählen: „*Raspberry PI OS LITE (32-BIT)*“
 - b. SD-Karte wählen
 - c. Zahnrad unten rechts ermöglicht weitere Einstellungen:
 - i. SSH mit „Password zur Authentifizierung verwenden“ auswählen
 - ii. Benutzername und Passwort festlegen:
 - **User: admin**
 - **Passwort: uniwelle_3kW**
 - iii. (optional) Wifi und Spracheinstellungen festlegen
 - d. „SCHREIBEN“ drücken und Installation abwarten
3. Micro-SD-Karte in *Raspberry* einlegen und starten

Setup des Raspberry Pi

1. Verbindung zu *RPi* herstellen (beide Geräte müssen sich im selben Netzwerk befinden)
2. Windows-Power-Shell starten
3. Verbindung via SSH zum *RPi* herstellen (siehe [Abbildung 2: Verbindung zum RPi via SSH herstellen](#)):
 - a. Folgenden Befehl ausführen:

```
ssh admin@192.168.xxx.xxx
```

- i. User
- ii. IP-Adresse
- b. Beim ersten Verbinden erscheint eine Abfrage, ob diesem Host vertraut werden darf.
Diese muss mit der Eingabe von „yes“ und Enter bestätigt werden
- c. Danach wird das Passwort des **Users** abgefragt
- d. Ist die Verbindung erfolgreich, ist man auf der Befehlszeile des *Raspberry* und kann mit der Installation fortfahren

The screenshot shows a Windows PowerShell window titled "Windows PowerShell" running on a Windows host. It displays the command "ssh admin@192.168.0.10" being entered. The response indicates that the host's fingerprint is being checked, and the user is prompted to continue connecting ("yes/no/[fingerprint]?)". The user types "yes" and provides the password "raspberry". The final output shows the Linux terminal prompt "admin@raspberrypi:~ \$".

```
PS C:\User\3. A. > ssh admin@192.168.0.10
The authenticity of host '192.168.0.10 (192.168.0.10)' can't be established.
ED25519 key fingerprint is SHA256:hb
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes 3. b.
Warning: Permanently added '192.168.0.10' (ED25519) to the list of known hosts.
admin@192.168.0.10's password: 3. c.
Linux raspberrypi 5.15.84-v8+ #1613 SMP PREEMPT Thu Jan 5 12:03:08 GMT 2023 aarch64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.

Last login: 3. d.
```

Abbildung 2: Verbindung zum RPi via SSH herstellen

4. Update des RPi durchführen:

```
sudo apt update
sudo apt upgrade
```

- Bestätigen mit Eingabe von „y“

5. Installation der *TinkerForge*-Module:

- Befolgen der Anleitungen:

- https://www.tinkerforge.com/de/doc/Software/APT_Repository.html#apt-repository (02.04.2023 13:45): Einrichtung durchführen:
 - GPG-Schlüssel importieren
 - APT-Repository hinzufügen
 - APT Paketliste updaten
- https://www.tinkerforge.com/de/doc/Software/Brickd_Install_Linux.html#brickd-install-linux (02.04.2023 13:45): *brickd* und *brickv* Pakete installieren

```
sudo apt install brickd
sudo apt install brickv
sudo apt install tinkerforge-mqtt
```

6. Installation von NodeRED:

- Folgende Anleitung befolgen: [\(02.04.2023 13:55\)](https://nodered.org/docs/getting-started/raspberrypi)

```
bash <(curl -sL https://raw.githubusercontent.com/node-red/linux-installers/master/deb/update-nodejs-and-nodered)
```

- “Autostart on boot” aktivieren:

```
sudo systemctl enable nodered.service
```

7. Erstellung einer Verzeichnisstruktur:

- Konfigurationsdateien:

```
mkdir config
```

- Skripte:

```
mkdir scripts
```

- Messungen:

```
mkdir measurements
```

8. Einrichtung der Tinkerforge-Module für MQTT-Übertragung

- Installation eines FTP-Clients zur Datenübertragung (z.B. FileZilla)
- Verbindung zum RPi herstellen (siehe Abbildung 3)
 - Eintragen der Netzwerkadresse mit Präfix „sftp://...“, Benutzername, Passwort und Port „22“
 - Verbinden drücken und Verbindung herstellen
 - Die Verzeichnisstruktur des RPi wird auf der rechten Seite (roter Kasten) angezeigt, die Verzeichnisstruktur des verbundenen PCs befindet sich auf der linken Seite (gelber Kasten)
 - Dateien können via Drag & Drop runter- bzw. hochgeladen werden

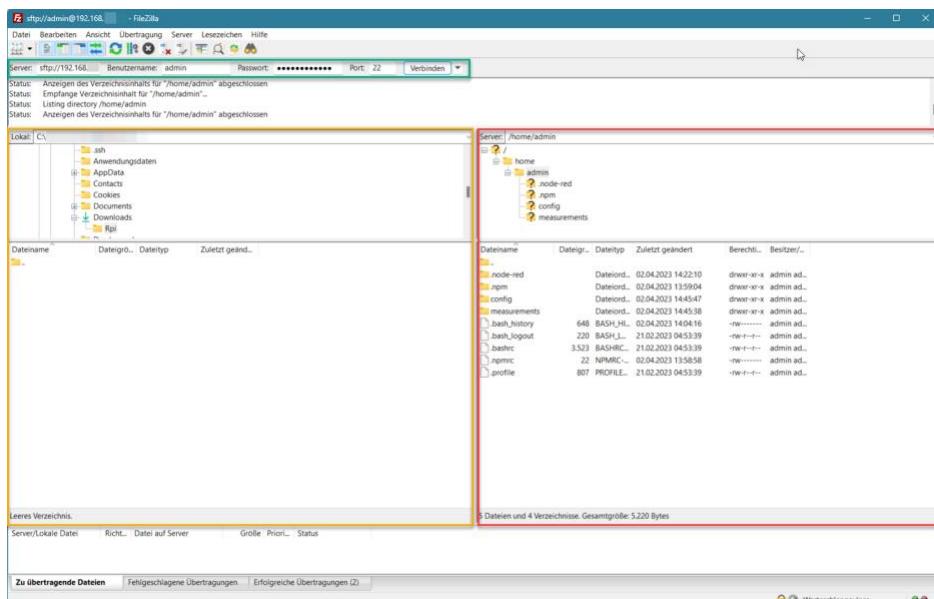


Abbildung 3: Verbindung zum Raspberry Pi via FTP für Fileupload

- c. Konfigurationsdatei „tf_mqtt_config.txt“ für die MQTT-Schnittstelle in „config“-Ordner hochladen
- d. Skript „data_backup.sh“ in „scripts“-Ordner hochladen
 - i. Skript verfügbar machen:

```
cd scripts/
sudo cp data_backup.sh /usr/local/bin/
sudo chown root /usr/local/bin/data_backup.sh
sudo chgrp root /usr/local/bin/data_backup.sh
sudo chmod 755 /usr/local/bin/data_backup.sh
```

9. Einrichtung der automatischen MQTT-Konfiguration und Daten-Backup
 - a. Bearbeitung der Crontab-Tabelle aufrufen:

```
sudo crontab -e
```

- b. Es öffnet sich ein Editor dort müssen am Ende folgende Zeilen eingefügt werden:

```
@reboot tinkerforge_mqtt --debug --broker-host localhost --broker-port 1883 --init-file /home/admin/config/tf_mqtt_config.txt
```

- i. Nach einem Reboot wird automatisch die MQTT-Schnittstelle mit der vorhandenen Konfiguration gestartet
- ii. Alle 5 Minuten wird das erstellte Backup-Skript ausgeführt und speichert alle Messdaten auf ein externes Medium
- c. Die beiden Konfigurationsdateien werden in [Kapitel 5](#) – Begleit- und Videodokumentation genauer beschrieben und befinden sich ebenfalls als Auszug im Anhang

Einrichten der NodeRED-Flows

1. Die Oberfläche zur weiteren Einrichtung der *NodeRED-Flows* kann über die IP-Adresse des RPis aufgerufen werden:
 - a. Öffnen eines beliebigen Internetbrowsers
 - b. Eingabe der IP-Adresse mit Standard-Port „1880“ im Adressfeld und mit Enter bestätigen
 - c. Es öffnet sich die Benutzeroberfläche von *NodeRED*:

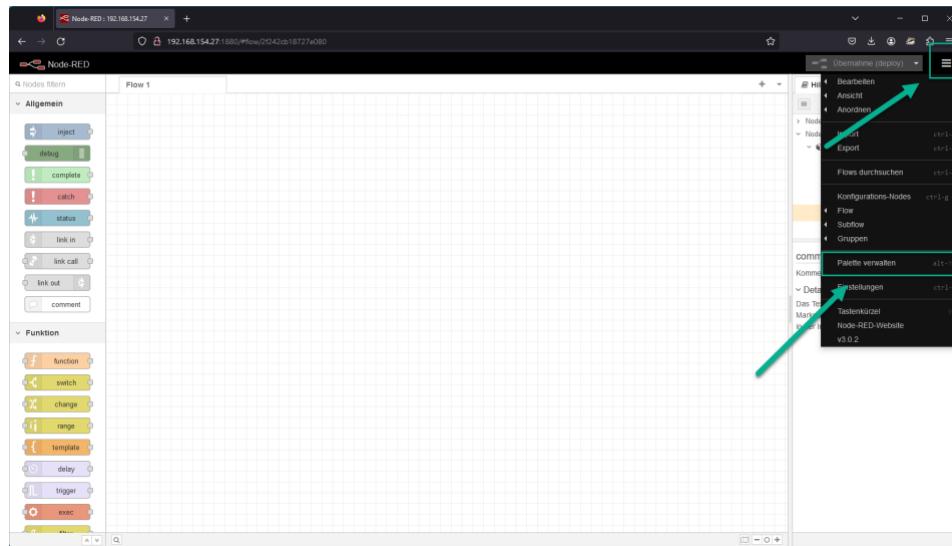


Abbildung 4: NodeRED Nutzeroberfläche

- d. Unter der Einstellung Palette verwalten müssen nun folgende Nodes installiert werden:
 - i. node-red-dashboard
 - ii. node-red-contrib-fs
 - e. Danach kann das bereitgestellte JSON-File importiert werden
 - i. Aufrufen des Menüs (siehe Abbildung 4)
 - ii. Import auswählen
 - iii. Im Import-Fenster kann nun die passende „.json“-Datei auf dem lokalen Rechner ausgesucht und importiert werden:
<https://hbx.fhrz.net/getlink/fiGJEvfLM5VM5xddNiwX5f/flows.json>
 - iv. Danach müssen die Flows noch durch Klicken auf „Übernahme (deploy)“ initialisiert werden
 - v. Anmerkung: Es kommt ein angepasster File-Browser auf Grundlage eines erstellten Flows von Csongor Varga zum Einsatz (<https://www.youtube.com/watch?v=3QgK4IAAqcQ> 02.04.2023 16:22)
2. Anpassung der Flows/Dashboards:
 - a. Im Tab „TinkerforgeModule“ befindet sich die Konfiguration der MQTT-Schnittstelle zu *NodeRED*, sowie eine erste Aufbereitung der Messdaten
 - b. Im Tab „CSV-Export“ wird den Messdaten ein Timestamp hinzugefügt und die Speicherung als CSV-Datei konfiguriert
 - c. Im Tab „Dashboard“ werden die einzelnen Anzeigen des Dashboards konfiguriert
 - d. (Optional) für Testzwecke können im Tab „TemperaturSensor“ die Anzeigen des Dashboardes mit Zufallswerten beschrieben werden
 - i. Die Nodes der Anzeigen können nach fertiger Konfiguration dann in den Tab „Dashboard“ kopiert und mit den entsprechenden Messstellen verbunden werden (ebenso umgekehrt)

Kapitel 3 – Anwendungsbeispiel

Nach Abschluss der Installationsanleitung sind Messungen am Prüfstand möglich. Das aktuelle Dashboard, welches mit simulierten Messdaten betrieben wird, kann nun unter folgendem Link über einen Browser aufgerufen werden:

192.168.x.x:1880/ui/

Hierbei sind die beiden hinteren Oktette (xxx) mit der entsprechenden IP-Adresse zu ergänzen. Wird der *Raspberry* mit grafischer Oberfläche installiert und genutzt, kann am *Raspberry* mit angeschlossenem Monitor und Eingabegeräten auch folgenden Adressen im Browser eingegeben werden:

localhost:1880/ui/

Im Browser wird dann folgendes Dashboard angezeigt:

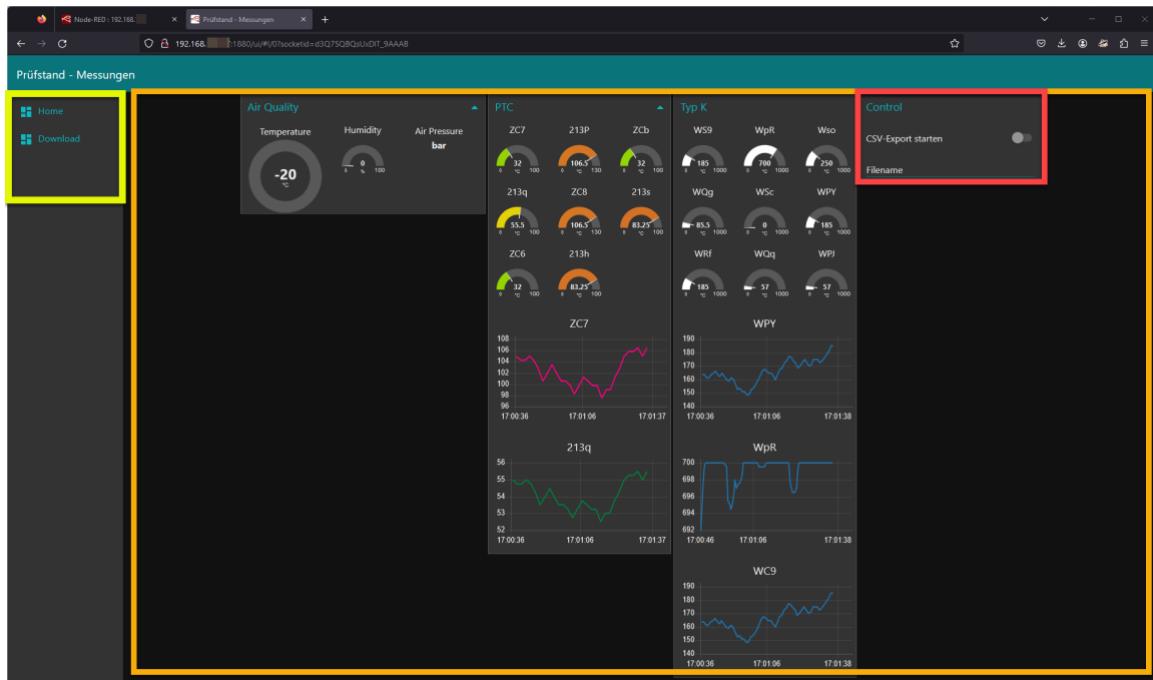


Abbildung 5: Dashboard des Messsystems

- Im orangenen Bereich befinden sich die konfigurierbaren Anzeigen, welche die Live-Daten der Messungen darstellen oder aber auch Diagramme mit zeitlichen Verläufen
- Im roten Bereich kann die Messung mit dem Schiebeschalter „CSV-Export starten“ begonnen werden, nachdem im Eingabefeld „Filename“ ein Dateiname eingegeben wurde
- Im gelben Bereich kann zwischen der Dashboard-Anzeige („Home“) und dem Download-Bereich („Download“) gewechselt werden. Im Downloadbereich wird das Verzeichnis angezeigt, in welchem die CSV-Dateien abgespeichert werden und können dort heruntergeladen werden

Das Dashboard ist von jedem Endgerät, welches sich im gleichen Netzwerk wie der RPi befindet erreichbar. Smartphones und Tablets werden ebenfalls von *NodeRED* unterstützt.

Sollte es dennoch zu Problemen kommen ist vor allem auch als Referenz [Kapitel 5](#) vorgesehen. Dort wird die Begleit- und Videodokumentation behandelt, welche weitere Einblicke in die Funktionsweise der Messumgebung bietet und bei der Fehlersuche behilflich sein kann.

Kapitel 4 – Data-Integrity

Aufgrund der höheren Fehleranfälligenkeiten von Micro-SD-Karten, bedingt durch den verbauten Flash-Speicher, ist es notwendig die Integrität der Daten abzusichern. Dafür wurden zwei Absicherungsmechanismen entwickelt, um zum einen das Betriebssystem aber auch die Messdaten abzusichern und eine Wiederherstellung zu ermöglichen.

Backup der Messwerte

Für das Backup der Messwerte wurde ein Skript entwickelt, welches das Ablageverzeichnis der Messdaten periodisch auf einen externen Datenträger absichert. Dieser kann durch einen am RPi angeschlossenen USB-Stick oder aber idealerweise eine externe HDD realisiert werden. Auf das genannte Backup-Skript wurde bereits in [Kapitel 2](#) referenziert (`data_backup.sh`) und die Installation sowie Ausführung beschrieben. Im Anhang dieser Anleitung befindet sich der Aufbau des Skriptes, welches ebenfalls ein Teil der Begleitdokumentation ist.

Backup und Restore

Auch das Wiederherstellen des Betriebssystems und der Konfiguration des Messsystems muss bei einem Ausfall möglich sein. Hierfür wurde ein Image der Micro-SD-Karte erstellt, welches bei einem Defekt mit Hilfe des Tools „*Raspberry Pi Imager*“ auf eine neue Micro-SD-Karte aufgespielt werden kann. Somit ist die Konfiguration des Messsystems wiederhergestellt und der Datenverlust ist minimal. Das Image wird ebenfalls in einem Verzeichnis auf der Hessenbox abgelegt:

<https://hbz.fhrz.net/getlink/fiWHL1f92F9UfUwgeqF3oM/Images>

Bei zukünftigen Änderungen ist es zwingend erforderlich das Image zu aktualisieren und die neue Datei im Verzeichnis auf der Hessenbox abzulegen. Hierfür kann folgendes Tool verwendet werden und Image der Micro-SD-Karte erstellt werden (nur Windows Betriebssysteme):

<https://win32diskimager.org/> (02.04.2023 21:03)

Kapitel 5 – Begleit- und Videodokumentation

Begleitdokumentation

Folgende Begleitdokumente sind Bestandteil dieser Anleitung und können bei Bedarf angepasst werden, um zukünftige Anpassungen und Änderungen zu dokumentieren:

1. Excel-Tabelle: MeasurementEquipment_TagList.xlsx
 - a. Dokumentation der Hardware-seitigen Verkabelung der *TinkerForge-Bricks*
 - b. Dokumentation der Brick-IDs am Prüfstand
 - c. Dokumentation der angeschlossenen Sensoren (Positionen) am Prüfstand
2. Konfigurationsdatei für MQTT-Schnittstelle der *TinkerForge-Module*:
 - a. Konfiguration der Nachrichtenübermittlung via MQTT:
 - i. Einstellen der Publish-Periode
 - ii. Weitere Optionen möglich siehe jeweiliges API-Binding der Bricklets (weiterführende Links in den Quellen)
3. Data_Backup-Skript:
 - a. Festlegen des Speicherortes und Benennung der Backups

Die genannten Dateien werden sowohl im Anhang als Auszug dargestellt aber auch in einem Verzeichnis auf der Hessenbox abgelegt:

<https://hbx.fhhrz.net/getlink/fiXxtwcusPmDdnoCahNRqt/Dokumentation>

Videodokumentation

Für User

Intro:

https://hbx.fhhrz.net/getlink/fiSmWJdsbjkpyPL9fifP1P/00_Messsystem_Intro.mp4

Verwendung des Dashboards:

https://hbx.fhhrz.net/getlink/fiEhiNwHgf4dQt4gCfaCSv/06_Messsystem_Dashboard-Vorstellung.mp4

Für Systembetreuer und Interessierte

Konfiguration der TinkerForge-Module:

https://hbx.fhhrz.net/getlink/fiWSbR1NCQmtSAh9hFtpBB/01_Messsystem_TF-Module.mp4

Messungen als CSV-Datei einrichten:

https://hbx.fhhrz.net/getlink/fiFA5PDzyRMBb2r1Y1XLJv/02_Messsystem_CSV.mp4

Konfiguration des Dashboard-Flows:

https://hbx.fhhrz.net/getlink/fiMyb7o3DHfV6XheK71Yki/03_Messsystem_Konfig-Dashboard.mp4

Konfiguration des Filebrowser im Dashboard:

https://hbx.fhhrz.net/getlink/fi6pbidvsB3DJbwZvd4EEb/04_Messsystem_Filebrowser.mp4

Messstellensimulation:

https://hbx.fhhrz.net/getlink/fi4TxHFL69qTji142uYdnc/05_Messsystem_SensorSim.mp4

Kapitel 6 – Ausblick

Das hier vorgestellte Messsystem soll keine finale Version sein, sondern vor allem die zukünftigen Nutzer dazu einladen, das System weiter zu verbessern und auszubauen. Beim Entwickeln und Aufbauen sind einige Ideen aufgekommen, welche Erweiterungen bzw. Verbesserung man implementieren könnte. Diese sollen nun kurz aufgelistet werden:

- Erweiterung des Dashboardes:
 - Implementierung weiterer/angepasster Visualisierungen
- Erweiterung des Systems:
 - Weitere Messstellen installieren und einbinden
 - Erweiterung zum Prüfstandskontrollsysteem durch Installation/Implementierung von Aktorik/Anzeigen/Steuerungen
- Speicherung der Messdaten auf eine ausgelagerte Datenbank (lokal/Web)
 - Interessante Lösung: InfluxDB v2.x mit integriertem Webserver für Datendownload und erste Visualisierung
 - Idee für eine Tagliste ist Excel-File der Begleitdokumentation aufgelistet
 - 64-bit Betriebssystem auf *Raspberry* zwingend erforderlich

Quellen:

In diesem Bereich sind alle Quellen aufgelistet, welche bei der Entwicklung des Systems nützlich waren. Diese müssen nicht zwangsläufig in der Anleitung erwähnt sein, bieten aber möglicherweise für die zukünftige Weiterentwicklung gute Anhaltspunkte:

Installation des Betriebssystems - Raspbian OS:

- <https://www.raspberrypi.com/software/> (02.04.2023 11:57)

Nützliche Links für die Verwendung der Befehlszeile via SSH:

- <https://wiki.ubuntuusers.de/Shell/Bash-Skripting-Guide> f%C3%BCr Anf%C3%A4nger/ 12.03.2023 11:47
- <https://wiki.ubuntuusers.de/mkdir/> 12.03.2023 13:59
- <https://wiki.ubuntuusers.de/cp/> 12.03.2023 14:18
- <https://wiki.ubuntuusers.de/Cron/> 12.03.2023 14:27

Setup der TinkerForge-MQTT-Schnittstelle:

- https://www.tinkerforge.com/de/doc/Software/APT_Repository.html#apt-repository (02.04.2023 13:45)
- https://www.tinkerforge.com/de/doc/Software/Brickd_Install_Linux.html#brickd-install-linux (02.04.2023 13:45)

Alle am 17.03.2023 21:24:

- https://www.tinkerforge.com/de/doc/Software/Bricklets_MQTT.html
- https://www.tinkerforge.com/de/doc/Software/Bricklets/IndustrialDualAnalogInV2_Bricklet_MQTT.html#industrial-dual-analog-in-v2-bricklet-mqtt-api
- https://www.tinkerforge.com/de/doc/Software/Bricklets/ThermocoupleV2_Bricklet_MQTT.html#thermocouple-v2-bricklet-mqtt-api
- https://www.tinkerforge.com/de/doc/Software/Bricklets/IndustrialPTC_Bricklet_MQTT.html#industrial-ptc-bricklet-mqtt-api
- https://www.tinkerforge.com/de/doc/Software/Bricklets/IndustrialDual020mA_V2_Bricklet_MQTT.html#register/industrial_dual_0_20ma_v2_bricklet/%3CUID%3E/current
- https://www.tinkerforge.com/de/doc/Software/Bricklets/IndustrialDual020mA_V2_Bricklet_MQTT.html#industrial-dual-0-20ma-v2-bricklet-mqtt-examples

Setup der NodeRED-Umgebung:

- https://nodered.org/docs/getting-started/raspberry_pi (02.04.2023 13:55)
- <https://www.youtube.com/watch?v=3QgK4IAAqcQ> (02.04.2023 16:22)

Weiterführende Tools zum Testen, Konfigurieren und Simulieren der MQTT-Schnittstelle:

- https://mosquitto.org/man/mosquitto_pub-1.html (17.03.23 16:29)
- http://www.steves-internet-guide.com/mosquitto_pub-sub-clients/ (17.03.23 16:29)

Anhang

Konfigurationsfile der MQTT-Schnittstelle:

```
1 | 
2 |     "+tinkerforge/request/industrial_ptc_bricklet/8C7/set_temperature_callback_configuration": {"period": 2000, "value_has_to_change": false, "option": "x", "min": 3000, "max": 0},
3 |     "+tinkerforge/register/industrial_ptc_bricklet/8C7/temperature": {"register": true},
4 | 
5 |     "+tinkerforge/request/industrial_ptc_bricklet/213P/set_temperature_callback_configuration": {"period": 2000, "value_has_to_change": false, "option": "x", "min": 3000, "max": 0},
6 |     "+tinkerforge/register/industrial_ptc_bricklet/213P/temperature": {"register": true},
7 | 
8 |     "+tinkerforge/request/industrial_ptc_bricklet/8C9/set_temperature_callback_configuration": {"period": 2000, "value_has_to_change": false, "option": "x", "min": 3000, "max": 0},
9 |     "+tinkerforge/register/industrial_ptc_bricklet/8C9/temperature": {"register": true},
10 | 
11 |     "+tinkerforge/request/industrial_ptc_bricklet/213g/set_temperature_callback_configuration": {"period": 2000, "value_has_to_change": false, "option": "x", "min": 3000, "max": 0},
12 |     "+tinkerforge/register/industrial_ptc_bricklet/213g/temperature": {"register": true},
13 | 
14 |     "+tinkerforge/request/industrial_ptc_bricklet/213h/set_temperature_callback_configuration": {"period": 2000, "value_has_to_change": false, "option": "x", "min": 3000, "max": 0},
15 |     "+tinkerforge/register/industrial_ptc_bricklet/213h/temperature": {"register": true},
16 | 
17 |     "+tinkerforge/request/industrial_ptc_bricklet/8C6/set_temperature_callback_configuration": {"period": 2000, "value_has_to_change": false, "option": "x", "min": 3000, "max": 0},
18 |     "+tinkerforge/register/industrial_ptc_bricklet/8C6/temperature": {"register": true},
19 | 
20 |     "+tinkerforge/request/industrial_ptc_bricklet/8C8/set_temperature_callback_configuration": {"period": 2000, "value_has_to_change": false, "option": "x", "min": 3000, "max": 0},
21 |     "+tinkerforge/register/industrial_ptc_bricklet/8C8/temperature": {"register": true},
22 | 
23 |     "+tinkerforge/request/industrial_ptc_bricklet/213s/set_temperature_callback_configuration": {"period": 2000, "value_has_to_change": false, "option": "x", "min": 3000, "max": 0},
24 |     "+tinkerforge/register/industrial_ptc_bricklet/213s/temperature": {"register": true},
25 | 
26 |     "+tinkerforge/request/thermocouple_v2_bricklet/W89/set_temperature_callback_configuration": {"period": 1000, "value_has_to_change": false, "option": "x", "min": 3000, "max": 0},
27 |     "+tinkerforge/register/thermocouple_v2_bricklet/W89/temperature": {"register": true},
28 | 
29 |     "+tinkerforge/request/thermocouple_v2_bricklet/W8P/set_temperature_callback_configuration": {"period": 1000, "value_has_to_change": false, "option": "x", "min": 3000, "max": 0},
30 |     "+tinkerforge/register/thermocouple_v2_bricklet/W8P/temperature": {"register": true},
31 | 
32 |     "+tinkerforge/request/thermocouple_v2_bricklet/W80/set_temperature_callback_configuration": {"period": 1000, "value_has_to_change": false, "option": "x", "min": 3000, "max": 0},
33 |     "+tinkerforge/register/thermocouple_v2_bricklet/W80/temperature": {"register": true},
34 | 
35 |     "+tinkerforge/request/thermocouple_v2_bricklet/W8Q/set_temperature_callback_configuration": {"period": 1000, "value_has_to_change": false, "option": "x", "min": 3000, "max": 0},
36 |     "+tinkerforge/register/thermocouple_v2_bricklet/W8Q/temperature": {"register": true},
37 | 
38 |     "+tinkerforge/request/thermocouple_v2_bricklet/W8G/set_temperature_callback_configuration": {"period": 1000, "value_has_to_change": false, "option": "x", "min": 3000, "max": 0},
39 |     "+tinkerforge/register/thermocouple_v2_bricklet/W8G/temperature": {"register": true},
40 | 
41 |     "+tinkerforge/request/thermocouple_v2_bricklet/W8C/set_temperature_callback_configuration": {"period": 1000, "value_has_to_change": false, "option": "x", "min": 3000, "max": 0},
42 |     "+tinkerforge/register/thermocouple_v2_bricklet/W8C/temperature": {"register": true},
43 | 
44 |     "+tinkerforge/request/thermocouple_v2_bricklet/W8Y/set_temperature_callback_configuration": {"period": 1000, "value_has_to_change": false, "option": "x", "min": 3000, "max": 0},
45 |     "+tinkerforge/register/thermocouple_v2_bricklet/W8Y/temperature": {"register": true},
46 | 
47 |     "+tinkerforge/request/thermocouple_v2_bricklet/W8R/set_temperature_callback_configuration": {"period": 1000, "value_has_to_change": false, "option": "x", "min": 3000, "max": 0},
48 |     "+tinkerforge/register/thermocouple_v2_bricklet/W8R/temperature": {"register": true},
49 | 
50 |     "+tinkerforge/request/thermocouple_v2_bricklet/W8Q/set_temperature_callback_configuration": {"period": 1000, "value_has_to_change": false, "option": "x", "min": 3000, "max": 0},
51 |     "+tinkerforge/register/thermocouple_v2_bricklet/W8Q/temperature": {"register": true},
52 | 
53 |     "+tinkerforge/request/thermocouple_v2_bricklet/W8J/set_temperature_callback_configuration": {"period": 1000, "value_has_to_change": false, "option": "x", "min": 3000, "max": 0},
54 |     "+tinkerforge/register/thermocouple_v2_bricklet/W8J/temperature": {"register": true},
55 | 
56 | 
57 |     "+tinkerforge/request/industrial_dual_0_20ma_v2_bricklet/Z1M/set_current_callback_configuration": {"period": 1000, "value_has_to_change": false, "option": "x", "min": 3000, "max": 0},
58 |     "+tinkerforge/register/industrial_dual_0_20ma_v2_bricklet/Z1M/current": {"register": true},
59 | 
60 | 
61 | 
62 |     "+tinkerforge/request/industrial_dual_analog_in_v2_bricklet/Ye9/get_all_voltages_callback_configuration": {"period": 1000, "value_has_to_change": false},
63 |     "+tinkerforge/register/industrial_dual_analog_in_v2_bricklet/Ye9/voltage": {"register": true},
64 | 
65 |     "+tinkerforge/request/industrial_dual_analog_in_v2_bricklet/Yb1/get_all_voltages_callback_configuration": {"period": 1000, "value_has_to_change": false},
66 |     "+tinkerforge/register/industrial_dual_analog_in_v2_bricklet/Yb1/voltage": {"register": true},
67 | 
68 |     "+tinkerforge/request/industrial_dual_analog_in_v2_bricklet/Ye0/get_all_voltages_callback_configuration": {"period": 1000, "value_has_to_change": false},
69 |     "+tinkerforge/register/industrial_dual_analog_in_v2_bricklet/Ye0/voltage": {"register": true},
70 | 
71 |     "+tinkerforge/request/industrial_dual_analog_in_v2_bricklet/Yb0/get_all_voltages_callback_configuration": {"period": 1000, "value_has_to_change": false},
72 |     "+tinkerforge/register/industrial_dual_analog_in_v2_bricklet/Yb0/voltage": {"register": true},
73 | 
74 |     "+tinkerforge/request/industrial_dual_analog_in_v2_bricklet/Yam/get_all_voltages_callback_configuration": {"period": 1000, "value_has_to_change": false},
75 |     "+tinkerforge/register/industrial_dual_analog_in_v2_bricklet/Yam/voltage": {"register": true},
76 | 
77 |     "+tinkerforge/request/industrial_dual_analog_in_v2_bricklet/Ybs/get_all_voltages_callback_configuration": {"period": 1000, "value_has_to_change": false},
78 |     "+tinkerforge/register/industrial_dual_analog_in_v2_bricklet/Ybs/voltage": {"register": true},
79 | 
80 |     "+tinkerforge/request/industrial_dual_analog_in_v2_bricklet/Yco/get_all_voltages_callback_configuration": {"period": 1000, "value_has_to_change": false},
81 |     "+tinkerforge/register/industrial_dual_analog_in_v2_bricklet/Yco/voltage": {"register": true},
82 | 
83 |     "+tinkerforge/request/industrial_dual_analog_in_v2_bricklet/YRk/get_all_voltages_callback_configuration": {"period": 1000, "value_has_to_change": false},
84 |     "+tinkerforge/register/industrial_dual_analog_in_v2_bricklet/YRk/voltage": {"register": true},
85 | 
86 | 
87 | 
88 |     "+tinkerforge/request/air_quality_bricklet/X8A/set_all_values_callback_configuration": {"period": 1000, "value_has_to_change": false, "option": "x", "min": 0, "max": 0},
89 |     "+tinkerforge/register/air_quality_bricklet/X8A/all_values": {"register": true}
```

Eine Textdatei befindet sich in folgendem Verzeichnis auf der Hessenbox:

<https://hbx.fhhrz.net/getlink/fi9XBFTNSsbQVjUEaRQ8dP/Konfigurationsdateien>

Backup-Skript

```
1 | #!/bin/bash
2 | 
3 | datum=$(date +'%Y-%m-%d')
4 | quelle=/home/admin/measurements
5 | ziel1=/home/admin/backups/$datum
6 | ziel2=/media/usb/backups/$datum
7 | 
8 | sudo mkdir $ziel1 $ziel2
9 | sudo cp -uvr $quelle $ziel1
10 | sudo cp -uvr $quelle $ziel2
11 | 
```

Das Skript befindet sich in folgendem Verzeichnis auf der Hessenbox:

<https://hbx.fhhrz.net/getlink/fi9XBFTNSsbQVjUEaRQ8dP/Konfigurationsdateien>