

MASTERTHESIS
Sebastian Farrenkopf

Konzept und Entwicklung einer Leitwarte für die Sektorkopplung mit Was- serstoff am Anwendungs- fall Elektrolyse mit biolo- gischer Methanisierung

FAKULTÄT TECHNIK UND INFORMATIK
Department Informations- und Elektrotechnik

Faculty of Engineering and Computer Science
Department Information and Electrical Engineering

Sebastian Farrenkopf

**Konzept und Entwicklung einer Leitwarte für
die Sektorkopplung mit Wasserstoff am An-
wendungsfall Elektrolyse mit biologischer Me-
thanisierung**

Masterarbeit eingereicht im Rahmen der Masterprüfung
im Studiengang *Master of Science Automatisierung*
am Department Informations- und Elektrotechnik
der Fakultät Technik und Informatik
der Hochschule für Angewandte Wissenschaften Hamburg

Betreuender Prüfer: Prof. Dr. Lutz Leutelt
Zweitgutachter: Prof. Dr.-Ing. Hans Schäfers

Eingereicht am: 22. April 2022

Sebastian Farrenkopf

Thema der Arbeit

Konzept und Entwicklung einer Leitwarte für die Sektorkopplung mit Wasserstoff am Anwendungsfall Elektrolyse mit biologischer Methanisierung

Stichworte

Leitwarte, Kommunikationsprotokolle, Datenbanken, Schaltentscheidungen, Routinen, Systemdienlichkeit, Power-to-Gas, Elektrolyseur, Methanisierung, Sektorenkopplung

Kurzzusammenfassung

In dieser Masterarbeit wird eine neue Leitwarte für das Technologiezentrum Energie-Campus aufgebaut. Anhand des Anwendungsfalls netzdienlicher Betrieb eines Elektrolyseurs mit einer biologischen Methanisierung wird die Entwicklung von Funktionen und Routinen sowie die Überführung der eigenentwickelten Funktionen im Einsatz mit den realen Anlagen getestet.

Dabei wird simulativ der Anwendungsfall netzdienlicher Betrieb des Power-to-Gas Konzept untersucht. Als Eingangsdaten für den netzdienlicher Betrieb wird zum einen die Stromnetzdaten vom Bezirk Bergedorf und zum anderen die Stromnetzdaten vom Technologiezentrum Energie-Campus verwendet.

Sebastian Farrenkopf

Title of Thesis

Concept and development of a control center for sector coupling with hydrogen taking electrolysis with biological methanation as a use case.

Keywords

power-to-gas, control room, communication protocols, databases, switching strategies, routines, system efficiency, electrolyser, methanation, sector coupling

Abstract

In this master thesis a new control station for the Technology Center Energy-Campus (TEC) is constructed. The development of functions and routines for an electrolyser with biological methanation will be analysed by the use case of grid-serving operation. These functions will be transfer to the new control room and tested on real plants of the TEC. As input data for the grid-serving operation, the power grid data from the district of Bergedorf and the power grid data from the TEC will be used.

Inhaltsverzeichnis

Abbildungsverzeichnis	vii
Tabellenverzeichnis	x
Listings	xi
Abkürzungsverzeichnis	xii
1 Einleitung	1
1.1 Motivation	1
1.2 Aufgabenstellung	2
1.3 Aufbau der Arbeit	2
2 Grundlagen	3
2.1 Die Leitwarte von Kieback & Peter	3
2.2 Zenon Supervisor	5
2.3 Internet of Things	6
2.4 Aufbau des Stromnetzsystems	7
3 Anforderungsanalyse	10
3.1 Konzept der Leitwarte des TECs	10
3.2 Hardware- und Softwareanforderungen an die Leitwarte am TEC	11
3.3 Anwendungsfall Systemdienlicher Betrieb	11
3.4 Eingangsdaten	13
3.5 Zusammenfassung und Fazit der Anforderungsanalyse	14
4 Technische Analyse	15
4.1 Normalbetrieb Elektrolyse und Methanisierung	15
4.1.1 Beschreibung der Verschaltung der Elektrolyse und Methanisierung	15
4.1.2 Betrieb Elektrolyse und Methanisierung	20
4.1.3 Ermittlung Regelungsgrenzen und Flexibilität	22

4.2	Analyse Leitwartenanbindung.....	24
4.2.1	Analyse der Leitwartenkonzepte	24
4.2.2	Analyse Kommunikationsprotokolle.....	25
4.3	Anbindung, Methoden zur Steuerung der Leitwarte	27
4.3.1	Analyse der Programmiersprache	27
4.3.2	Regelungs- / Steuerungskonzept für den Use Case netzdienlicher Betrieb.....	28
4.4	Analyse der Netzzustände mittels Daten von Stromnetz Hamburg	28
4.4.1	Aufbereitung der „Daten vom Energieportal Hamburg“	29
4.4.2	Auswertung Netzzustände für Hamburg und den Bezirk Bergedorf.....	30
4.5	Datenbanken.....	31
4.5.1	Daten in der Influx	31
4.5.2	Grafana	32
4.5.3	Datenbankkonzept.....	32
4.6	Design	33
5	Implementierung	34
5.1	Die Hardware	34
5.1.1	Die Firewall.....	34
5.1.2	Nginx mit Let's Encrypt.....	35
5.2	Der Leitwartenrechner.....	36
5.2.1	Das Admin-Interface	36
5.2.2	Die Services	37
5.2.3	Das User Managment.....	38
5.2.4	Workbench	39
5.2.5	SSL-Zertifikat für den Leitwartenrechner	41
5.3	Der Datenbankserver.....	41
5.3.1	Installation und Konfiguration der Datenbanken Influx und PostgreSQL	41
5.3.2	Weitere Konfigurationen.....	43
5.4	Development & Deployment Server	44
5.4.1	Docker Container	44
5.4.2	Container-Aufbau.....	45
5.4.3	Portainer	49

6	Evaluierung	52
6.1	Anwendungsfall Netzdienlichkeit im Bezirk Bergedorf.....	52
6.1.1	Beschreibung des Anwendungsfall Netzdienlichkeit im Bezirk Bergedorf....	52
6.1.2	Die Eingangsdaten für den Anwendungsfall Netzdienlichkeit im Bezirk Bergedorf	53
6.1.3	Umsetzung der Regelung der Elektrolyse mit Wasserstoffspeicher	57
6.1.4	Auswertung des Anwendungsfall Netzdienlichkeit im Bezirk Bergedorf	60
6.2	Anwendungsfall Netzdienlichkeit am TEC.....	63
6.2.1	Beschreibung des Anwendungsfall am TEC.....	63
6.2.2	Die Eingangsdaten für den Anwendungsfall Netzdienlichkeit am TEC	63
6.2.3	Auswertung des Anwendungsfall Netzdienlichkeit am TEC	64
6.3	Kommunikationstest „Netzdienlichkeit am TEC“	66
6.3.1	Beschreibung des Kommunikationstests „Netzdienlichkeit am TEC“	66
6.3.2	Die Eingangsdaten für den Kommunikationstest.....	67
6.3.3	Durchführung des Kommunikationstests „Netzdienlichkeit am TEC“	67
6.4	Zusammenfassung der Ergebnisse der Simulation.....	70
7	Fazit & Ausblick	71
	Literaturverzeichnis	73
	Digitaler Anhang	76

Abbildungsverzeichnis

Abbildung 1: Die Bedienoberfläche der Kieback & Peter Leitzentrale	4
Abbildung 2: Screenshot des OPC-Servers.....	4
Abbildung 3: Das Zenonprojekt TEC (links) mit dem Bild der Methanisierung (rechts).....	5
Abbildung 4: die Überdrucksicherung der Methanisierung mittels strukturierten Text (ST) ...	6
Abbildung 5: Das deutsche Stromnetz (Quelle: VKU e.V.)	7
Abbildung 6: Nettostromerzeugung in Deutschland, KW15, 2022 (Quelle: Fraunhofer ISE, energy-charts.info)	9
Abbildung 7: Anforderungen an das System	10
Abbildung 8: Verfügbarkeitsmatrix mit Blick auf den Power-2-Gas Prozess	12
Abbildung 9: Schaubild des Power-to-Gas-Systems am TEC. (Quelle: Schaubild Hitachi Zosen INOVA, Projekt 1911-50 für HAW-Hamburg)	15
Abbildung 10: PEM- Elektrolyseur (Quelle: [17]).....	16
Abbildung 11: Bilder der Methanisierung und des Wasserstoffspeichers am TEC (Quelle: Eigene Aufnahmen)	17
Abbildung 12: Bild des Elektrolyseurs (links) und des Gasmischers (rechts) am TEC (Quelle: Eigene Aufnahme)	18
Abbildung 13: Zeitlicher Verlauf der Leistung des Elektrolyseurs und des Speicherdrucks..	21
Abbildung 14: Der Wasserstoffspeicher mit seinen Beladegrenzen	22
Abbildung 15: Ladeverlauf des H ₂ -Speichers	23
Abbildung 16: Ablaufdiagramm der Schaltroutine für die Elektrolyse	23
Abbildung 17: Screenshot von jupyter.ec.haw-hamburg.de und Node-RED	27

Abbildung 18: Screenshot der Seite "energieportal-hamburg.de"	29
Abbildung 19: Vergleich der Daten von der Website und der Datenbank (KW 07, 2022).....	30
Abbildung 20: Verlauf der Residuallast und der Stromüberschuss-Viertelstunden (KW 10, 2022)	31
Abbildung 21: Realisierung des Systems aus den Überlegungen der Anforderungsanalyse ..	33
Abbildung 22: Das User Interface der Connectware.....	37
Abbildung 23: Rollen und Rechte des Users Netzdieners	39
Abbildung 24: Grenzdruck-Flow unter Node-RED	40
Abbildung 25: Measurement- und Datenabfrage in der Influx	42
Abbildung 26: Screenshot des Grafana-Dashboards „Wetter am Energie-Campus“	44
Abbildung 27: Docker Logo Moby Dock. (Quelle: www.docker.com).....	45
Abbildung 28: Dateien für den kontinuierlichen Container	46
Abbildung 29: Dateien für den Aufbau des routine-Containers	48
Abbildung 30: Screenshot des erstellten Containers	49
Abbildung 31: Screenshot von „docker inspect“ im Terminal und in Portainer	50
Abbildung 32: Weboberfläche Portainer.....	50
Abbildung 33: Ablauf und Analyse der Simulation für den Bezirk Bergedorf.....	53
Abbildung 34: Jahresdauerlinien der Residuallast im Bezirk Bergedorf.....	54
Abbildung 35: Vergleich der Daten aus der Datenbank mit dem Energieportal-Hamburg	56
Abbildung 36: Netzzustandsvektor und Dauer der positiven Schaltzustände.....	56
Abbildung 37: Überschussstunden im Bezirk Bergedorf von 2019-2021.....	57
Abbildung 38: Ablauf der Regelung der Elektrolyse mit dem Wasserstoffspeicher	59
Abbildung 39: Verlauf der Jahressimulation mit einem Ausschnitt vom Januar 2018.....	60
Abbildung 40: Ausschnitt des Verlaufs der Simulation vom März 2018	61
Abbildung 41: Verlauf der Speicherdrücke der Simulationszeiträume 2019 - 2021	62
Abbildung 42: Überschussstunden am TEC	64
Abbildung 43: Vergleich des Speicherdrucks mit dem Netzzustand vom TEC 2021	65

Abbildungsverzeichnis

Abbildung 44: Simulierter Speicherdruck im 2021 mit Sommerregelung.....	65
Abbildung 45: Leistungsverlauf und Netzdienlichkeitssignal der KW 13, 2022.....	67
Abbildung 46: Grafana-Screenshot Einschaltverhalten Elektrolyseur.....	68
Abbildung 47: Ausschaltvorgänge Elektrolyse.....	69

Tabellenverzeichnis

Tabelle 1: Die DDCs am TEC mit den aufgeschalteten Anlagen	3
Tabelle 2: Daten PEM-Elektrolyseur am TEC.....	18
Tabelle 3: Daten Methanisierung am TEC.....	19
Tabelle 4: Daten des Gasmischers am TEC	19
Tabelle 5: Gasflaschenbündel	19
Tabelle 6: Benötigte Datenpunkte für den Anwendungsfall	20
Tabelle 7: Permit Regeln der Firewall als Access-list	35
Tabelle 8: Auszug Messgrößentabelle	38
Tabelle 9: Analyse der Jahresviertelstunde.....	55
Tabelle 10: Simulationsergebnisse über den Netzzustand des Bezirk Bergedorfs.....	62
Tabelle 11: Simulationsergebnisse über den Hausanschluss des TEC.....	66
Tabelle 12: Der Inhalt der beiliegenden CD	76

Listings

Listing 1: Lastabfrage durch „Energieportal-Hamburg-Scraper.py“	29
Listing 2: Certbot wird mit dem nginx Plugin ausgeführt.....	36
Listing 3: Cybus-Script für die Installation ausführen	36
Listing 4: YAML-File des Siemens PAC3200 aus Raum3.....	37
Listing 5: Grenzdruck-Funktion als Node in der Workbench.....	40
Listing 6: Aktualisieren des SSL-Zertifikats des Leitwartenrechner	41
Listing 7: Installation und Einrichten der Datenbanken.....	42
Listing 8: Installation und Einrichten von Grafana	43
Listing 9: Inhalt der requirements.txt-Datei	46
Listing 10: Inhalt der config.ini-Datei.....	46
Listing 11: Beispielscript mit configparser	46
Listing 12: Inhalt der Dockerfile.....	47
Listing 13: Inhalt der batch_config.xml-Datei.....	48
Listing 14: Inhalt der docker-complse.yml-Datei	49
Listing 15: Initialisierung der Simulation des Power-to-Gas-Prozesses	58

Abkürzungsverzeichnis

HAW	Hochschule für Angewandte Wissenschaften
CC4E	Competence Center für Erneuerbare Energien und EnergieEffizienz
TEC	Technologiezentrum Energie-Campus
P2G	Power-to-Gas
BHKW	Blockheizkraftwerk
kW	Kilowatt
MW	Megawatt
ESS	Energiespeichersystem
DBMS	Datenbankmanagementsystem
FM	Facility Management
GLT	Gebäudeleittechnik
ITSC	Informationstechnik Service Center
VPN	Virtual Private Network
API	Application Programming Interface / Programmierschnittstelle
HTTP (S)	Hypertext Transfer Protocol (Secure)
DDS	Development & Deploy-Server
ODbL	Open Database License

1 Einleitung

1.1 Motivation

Die klimaschutzpolitischen Grundsätze und Ziele der Bundesregierung wurden 2016 im Klimaschutzplan 2050 verabschiedet. Dieser orientiert sich am Ziel des Pariser Klimaabkommens, indem das nationale Ziel, langfristig weitgehend Treibhausgasneutralität zu erreichen, festgehalten wird [1]. Mit dem Beschluss des Bundesverfassungsgerichtes 2021, wonach das Klimaschutzgesetz in Teilen nicht mit den Grundrechten vereinbar ist, da die Freiheitsrechte der jüngeren Generation durch drohende drastische Einschränkungen betroffen sein könnten, trat 2021 das novellierte Bundes-Klimaschutzgesetz in Kraft. Darin werden verbindliche Zwischenziele für die 2020er und -30er Jahre festgelegt. Die gesamten Treibhausgasemissionen müssen bis zum Jahr 2030 um 65% und bis 2040 um 88% gegenüber 1990 vermindert werden und die Treibhausgasneutralität muss bereits bis 2045 erreicht werden.

Die zentralen Bausteine dahin sind der Ausbau der erneuerbaren Energien und die Steigerung der Energieeffizienz, aber auch die Nutzung von Vorteilen und Chancen durch die Digitalisierung z.B. in der Modernisierung der Verteilnetze [2].

Im Grünbuch Energieeffizienz spiegelt sich dabei der hohe Stellenwert der Sektorkopplung, das Lösen der Auftrennung in einzelne Energiesektoren, wider, da durch verstärkten Einsatz von erneuerbarem Strom fossile Energieträger substituiert werden, und dadurch ein entscheidender Beitrag zur Erreichung der Schutzziele geleistet werden kann [3].

Der stetige Wandel der Energiewende und damit verbunden der Umbau zur Kopplung der Sektoren Mobilität, Wärme, Industrie und Wasserstoff, ergeben neue Anforderungen an die Digitalisierung. Insbesondere das Steuern und Regeln von Anlagenverbänden gehen über die Funktionsweise von klassischen Leitwarten hinaus. Hier bedarf es neue Lösungskonzepte, speziell durch das immer stärkere Auftreten von Internet of Things (IoT) Lösungen. Hierbei geht es darum Anlagen smart zu machen um die Betriebsweise der Anlagen zu verändern und um diese neuen, übergeordneten Ziele erfüllen zu können [4].

1.2 Aufgabenstellung

Ziel dieser Arbeit ist die Analyse und Realisierung eines Leitwartenkonzepts, das den netzdienlichen Betrieb für die Anlagen am Technologiezentrum Energie-Campus (TEC) des Competence Center für Erneuerbare Energien und EnergieEffizienz (CC4E) der HAW Hamburg unterstützt. Für den Kommunikationstest der Leitwarte soll das Power-to-Gas Konzept (P2G) dienen. Die neue Leitwarte soll den künftigen Anforderungen an dem Forschungsbetrieb sicherstellen. Hierbei betrifft es insbesondere die simulative Entwicklung von Funktionen/ Routinen in den Forschungsprojekten, sowie die Überführung der eigenentwickelten Funktionen im Einsatz mit den realen Anlagen.

Vor diesem Hintergrund soll simulativ der Anwendungsfall netzdienlicher Betrieb des Power-to-Gas Konzept untersucht werden. Als Eingangsdaten für den netzdienlicher Betrieb sollen zum einen die Stromnetzdaten vom Bezirk Bergedorf und zum anderen die Stromnetzdaten vom TEC verwendet werden. Neben dem netzdienlichen Betrieb soll vor allem auch das zu entwickelnde Regelungskonzept aufgebaut und untersucht werden.

Das P2G Konzept am TEC besteht aus einer Elektrolyse, mit einem Wasserstoffspeicher und einer biologischen Methanisierung. Für die biologischen Methanisierung wird noch CO₂ benötigt. Dieses wird in Flaschen vorgehalten.

1.3 Aufbau der Arbeit

In Kapitel 2 werden die Grundlagen und der Stand der Technik vorgestellt. Darin gibt es einen Überblick über die aktuell verbauten Leitwarten und die damit verbunden Anlagen am TEC. Zudem werden Begrifflichkeit, die im Zusammenhang mit dem IoT und dem Energiesystem verbunden sind, eingeführt. Im Kapitel 3 Anforderungsanalyse werden die Anforderungen an die neue Leitwarte vorgestellt und die Anwendungsfälle näher dargestellt. Darauf anschließend ergeben sich Fragen, die notwendig sind, um die Fragestellung dieser Arbeit umzusetzen. Diese Fragen werden in dem Kapitel 4 Technische Analyse näher analysiert, damit am Ende des Kapitels das finale Design der neuen Leitwarte erarbeitet werden kann. In Kapitel 5 Implementierung wird der Aufbau und die Einrichtung der neue Leitwarte näher beschrieben. Im Kapitel 6 Evaluierung werden zunächst die beiden Anwendungsfälle untersucht und der finale Kommunikationstest der neuen Leitwarte durchgeführt. Abschließend erfolgen in Kapitel 7 das Fazit und der Ausblick.

2 Grundlagen

In diesem Kapitel wird zunächst die vorhandene Leitwarte am TEC beschrieben, sowie der Rahmen, in dem die neue Leitwarte arbeiten soll.

Dazu gehört zum einen die Gebäudeleittechnik (GLT) von Kieback & Peter, die vom Facility Management (FM) der HAW Hamburg genutzt wird, und zum anderen die Leitwartensoftware der Forschenden, der Zenon Supervisor von Copa Data.

2.1 Die Leitwarte von Kieback & Peter

Das FM nutzt zur Überwachung der verschiedenen Anlagen die Gebäudeautomatisierung von Kieback & Peter. Die Anlagen werden in der Werkshalle des TECs in fünf Schaltschränken zusammengeführt, wo jeweils eine Direct-Digital-Controller-Zentrale (DDC 4200) die Daten aufnimmt. Die unterschiedlichen Anlagen sind mit dem standardisierten Kommunikationsprotokoll BACnet (Building Automation and Control Network) verbunden.

Tabelle 1: Die DDCs am TEC mit den aufgeschalteten Anlagen

	<i>Geräte</i>
DDC 1 - Heizung	Hydraulik, Heizflächen
DDC 2 - Lüftung	Lüftungsanlage, Sicherheitssysteme
DDC 3 - Kälte	Kältemaschine
DDC 4 - Speicher	Heizspeicher 4-6, Kältespeicher, Wärmepumpe, Heizstäbe
DDC 5 - Raumluft	BHKW

Die Datentypen der einzelnen Werte können Analog, Binär oder Multistate sein, und unterscheiden sich in Soll- und Istwerte. Für das Schalten, Stellen, Melden, Messen und Zählen der Sensoren und Aktoren sind die DDCs an eine GLT-Leitzentrale von Kieback und Peter angeschlossen, einem Computer auf dem das echtzeitfähigen QNX-Betriebssystem läuft.

Die GLT-Leitzentrale besteht dabei gleichzeitig aus einem Server und einem Client. Der Server hat dabei den direkten kommunikativen Zugriff auf alle Informationen der DDCs, der Client stellt die Oberflächensoftware Neutrino-GLT zur Verfügung, für deren Aufbau und Darstellung er verantwortlich ist. Die Anfragen, die über die Bedienoberfläche an den Client gestellt werden, werden vom Client an den Server weitergeleitet und die Antworten wieder vom Client graphisch dargestellt.

In Abbildung 1 ist die Anlage Kältemaschine mit den Plattenwärmetauscher und dem Kältespeicher abgebildet. In dem Anlagenbaum links können die verschiedenen Anlagenbilder mit ihren Parametern ausgewählt werden.

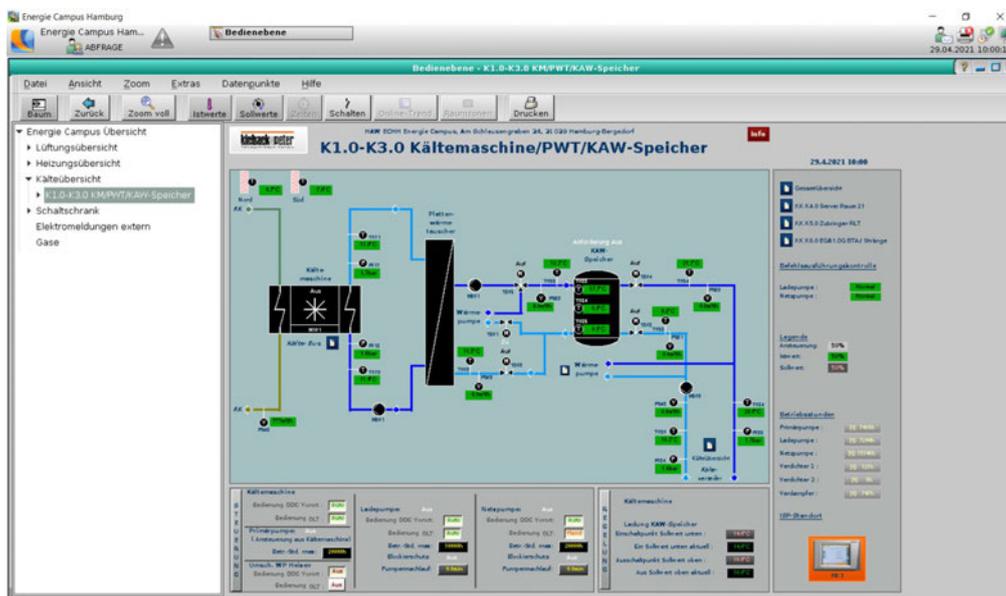


Abbildung 1: Die Bedienoberfläche der Kieback & Peter Leitzentrale

Für das TEC wurde in der GLT-Leitzentrale von Kieback & Peter zusätzlich die Möglichkeit implementiert, die Datenpunktmenge über OPC (OLE for Process Control) im Gebäudenetz zu übertragen. Da OPC auf Mechanismen von Microsoft Windows Betriebssystemen basiert, wird der OPC-Server auf einem PC im Microsoft Windows Netzwerk installiert. Die Leitzentrale, ebenfalls Teilnehmer in diesem Netzwerk, versorgt den OPC-Server mit den aktuellen Wertinformationen.

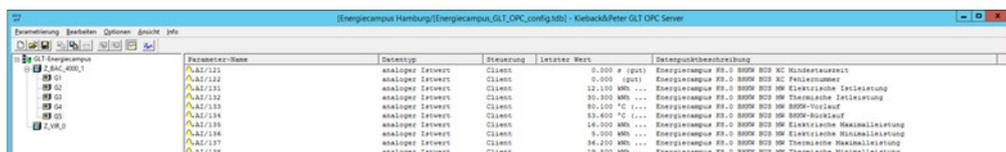


Abbildung 2: Screenshot des OPC-Servers

Abbildung 2 zeigt die analogen Ist-Werte der DDC5 – Raumlufte an, unter der das BHKW aufgeschaltet ist. Der OPC-Server läuft auf dem Leitwartenrechner des Smart-Grid Labors.

2.2 Zenon Supervisor

Beim Einzug des Smart-Grids zum TEC wurde ein Windows-Server Leitstand konfiguriert, der die Leitwartensoftware Zenon Supervisor von Copa-Data installiert hat. Das Programm besteht aus dem Zenon Editor und aus der Zenon Runtime. Der Editor ist ähnlich dem Client von Kieback & Peter für die Werteanzeige und das visuelle Erscheinen zuständig, während die Runtime für die Werteabfrage zuständig ist. Für die Anzeige und die Schaltungen der einzelnen Anlagen müssen von der Leitwarte sowohl Kommunikationsschnittstellen der Anlagen des Gebäudes unterstützt werden, als auch die Forschungsanlagen. Um die Gebäudeanlage zu lesen, wurde auf dem Leitstand des Smart-Grid Labors der OPC-Server von Kieback & Peter installiert (vgl. Abbildung 2). Auf diese Werte kann der Editor zugreifen, daneben hat der Editor zur SPS-Anbindung den dazugekauften S7 TCP-IP Treiber, der durch die Standardnetzwerkarte über das Gebäudenetz mit einer Siemens Steuerung kommunizieren kann. Mit diesem Treiber können Variablen angelegt werden, die auf Datenbausteine der angeschlossenen SPS zugreifen. Diese Variablen bekommen je nach Datentyp vom Treiber zugeordneten Objektamen.

Für die Modbus-Kommunikation gibt es einen Zenon-Treiber, der das Open Modbus TCP für TCP/IP Verbindungen unterstützt. Auch hier werden wieder Variablen im Zenon-Editor erstellt, die auf die Adressen des jeweiligen Modbusregisters verweisen.

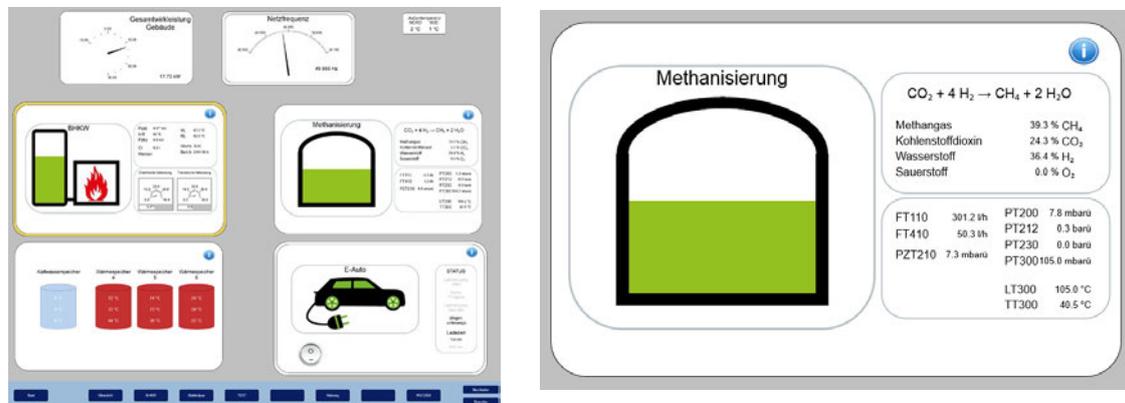


Abbildung 3: Das Zenonprojekt TEC (links) mit dem Bild der Methanisierung (rechts)

Die im Editor erstellten und adressierten Variablen können dann mit Hilfe von sich selbst aktualisierenden Bildern unter der Runtime angezeigt werden.

Für die Speicherung der Daten aus der Runtime wurde das Zenon-Paket Archiv-Server hinzugebucht. Dieses ermöglicht die aufgenommenen Werte in einer SQL-Datenbank zu speichern.

Die Speicherung kann zyklisch erfolgen, zum Beispiel jede Stunde, oder ereignisgesteuert, wenn bestimmte Schwellwerte erreicht werden.

Der Zenon Editor hat ein integriertes SPS-System, die Zenon Logic. Damit ist es möglich, Variablen und Datentypen in einem Programmiersystem, zum Beispiel als Simulation zu erstellen oder zu verändern. Das Programm kann in den fünf typischen SPS-Sprachen (AWL, ST, KOP, FUP und AS) geschrieben werden.



```
Druck2
IF Methanisierung_Ueberdruck THEN
  // Ueberdruck Sicherung EIN
  IF Methanisierung_FT200 > 12.0 THEN
    Methanisierung_V220_Handbetrieb := TRUE;
    Methanisierung_V220 := TRUE;
  END_IF;
  // Ueberdruck Sicherung AUS
  IF Methanisierung_FT200 < 8 AND Methanisierung_V220 = TRUE THEN
    Methanisierung_V220_Handbetrieb := TRUE;
    Methanisierung_V220 := FALSE;
    Methanisierung_V220_Handbetrieb := FALSE;
  END_IF;
END_IF;
```

Abbildung 4: die Überdrucksicherung der Methanisierung mittels strukturierten Text (ST)

Die Leitwarte von Kieback&Peter, die das Facility Management der HAW am TEC nutzt, braucht für jedes Protokoll ein Hardware-Gateway.

2.3 Internet of Things

Das Internet der Dinge (Internet of Things - IoT) soll die Vernetzung von Dingen definieren, die für die Erfassung, Verarbeitung und Meldung von Informationen genutzt werden. Die Anwendung und die Nutzung dieser Informationen nehmen ständig zu, da immer mehr Sensoren mit dem Internet verbunden werden. Wichtige IoT Elemente sind die Sensoren, die Kommunikation und die Identifikation, die Big Data-Erfassung sowie die Endanwendung [5].

IoT ist dabei keine einzelne Technologie, sondern wird als Überbegriff für verschiedene Techniken genutzt. Bei den IoT-Produkten im Wohnbereich spricht man häufig von SmartHome, deren Weiterentwicklungen zu Smart Buildings bis hin zu Smart Citys werden können.

Bei der Flexibilisierung von Produktion und Lieferketten kommt häufig der Begriff der Industrie 4.0 und Industrial Internet of Things (IIoT) mit dazu, bei der Flexibilisierung des Stromnetzes ist häufig die Rede vom Smart-Grid [6].

Auch die Nutzung von Smartphones und Wearables hat das IoT von der Maschinen-zu-Maschinenkommunikation erweitert und den Anwender zum Mitglied der Kommunikation gemacht.

Dieser Fortschritt in der Erfassung, Verarbeitung und Auswertung von Informationen soll auch in die Leitwarte am TEC mit einfließen, da der Energiesektor durch die Sektorenkopplung zwischen Strom zu Wärme, zur Mobilität oder zur H₂ – Wirtschaft an der Digitalisierung deutlich

zunimmt und dadurch sich auch neue Geschäftsfelder ergeben. Diese gilt es auch im Forschungsbetrieb des TECs näher zu untersuchen.

2.4 Aufbau des Stromnetzsystems

Das deutsche Stromnetz ist Bestandteil des europäischen Verbundnetzes. Dieses Verbundnetz ist auf der Höchstspannungsebene eng vermascht, wodurch es sehr robust ist und großflächige Netzausfälle selten auftreten [7].

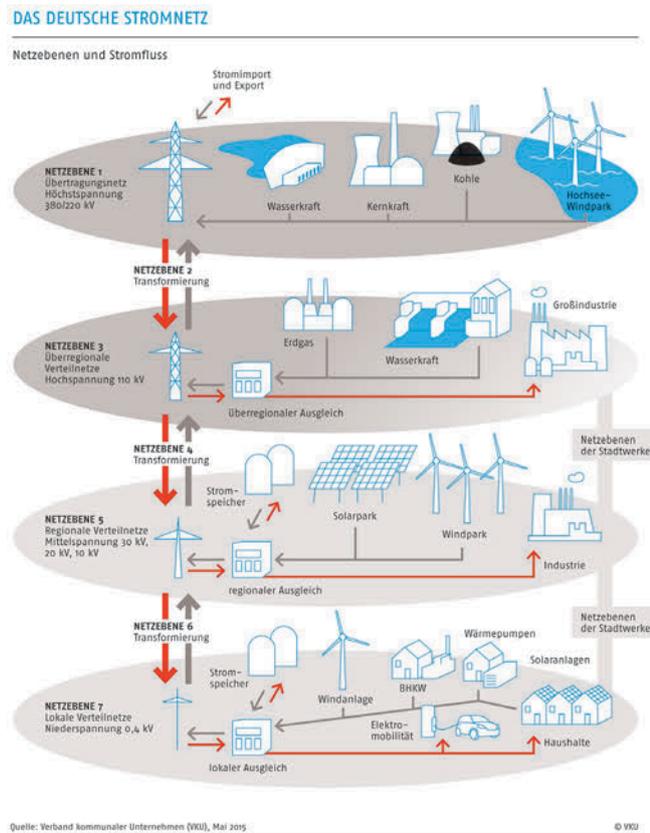


Abbildung 5: Das deutsche Stromnetz (Quelle: VKU e.V.)

Die vier Übertragungsnetzbetreiber (ÜNB) in Deutschland (TenneT, 50Hertz Transmission, Amprion und TransnetBW) bauen, betreiben und erhalten die Infrastruktur der Übertragungsnetzes. Kontrolliert werden die ÜNBs durch die Bundesnetzagentur.

Das Übertragungsnetz ermöglicht die überregionale Verteilung des Stroms. Da sich der Energieverlust bei der Verteilung mit der Höhe der Spannung verringert, arbeitet das Übertragungsnetz mit Spannungen bis zu 380.000 Volt. Über Kuppelleitungen ist das Übertragungsnetz an das europäische Verbundnetz angeschlossen. Die Stromkreislänge der Übertragungsnetze

beträgt rund 37.000 Kilometer. In den Netzebenen unter dem Übertragungsnetz wird der Strom in Hoch-, Mittel- und Niederspannung übertragen, die meist von einem lokalen bzw. kommunalen Verteilnetzbetreiber betrieben werden. Das Hochspannungsnetz ist mit Umspannwerken an das Übertragungsnetz angeschlossen. Das Hochspannungsnetz verteilt den Strom mit 110 kV zu den Ballungszentren oder auch direkt an große Industriebetriebe. Das Mittelspannungsnetz verteilt den Strom innerhalb der einzelnen Stadt- oder Landbezirke an regionalen Transformatorstationen. Einen direkten Anschluss an das Mittelspannungsnetz haben nur größere Einrichtungen wie beispielsweise Krankenhäuser oder Fabriken. Das Niederspannungsnetz transportiert den Strom zu den Endverbrauchern. An das Niederspannungsnetz sind die meisten Verbraucher angeschlossen, es hat eine Stromkreislänge von rund 1.190.000 km [7].

Für den verstärkten Einsatz von erneuerbarem Strom kommen mehrere Herausforderungen zusammen. Es werden jetzt und künftig noch mehr neue Stromerzeugungsanlagen verbaut werden. Diese werden vermehrt dezentral auf Nieder- und Mittelspannungseben verbaut [2]. Die Anlagen müssen effizient und optimal eingesetzt werden und benötigt dafür entsprechende Speicherkapazitäten, damit eine Flexibilität für den Ausgleich zwischen der Stromerzeugung und dem -verbrauch geschaffen wird. Im Gegensatz zur Stromerzeugung von Atom-, Kohle- und Gaskraftwerken, die je nach Bedarf gut steuerbar waren, ist die Stromgewinnung aus regenerativen Quellen wie Wind und Sonne un stetig, da diese Wetterabhängig sind. Durch das Verdrängen bzw. Abschalten konventioneller Kraftwerke und den vermehrten Ausbau der Erneuerbaren Energien, ergibt sich die Herausforderung auf die schwankende EE-Erzeugung zu reagieren. Dabei entstehen Zeiten, in der die Stromproduktion die Stromnachfrage übersteigt (Stromüberschuss) oder unterschreitet (Strommangel), doch muss zu jeder Zeit sichergestellt sein, dass Erzeugung und Verbrauch im Gleichgewicht sind. Durch die Flexibilisierung der Stromnachfrage können diese Schwankungen zum Teil ausgeglichen werden, insbesondere unter Einbezug der Energiesektoren von Wärme und Mobilität. Daher spielt bei der Transformation zu einem klimaneutralen Energiesystem gerade die Aufnahme- und Speicherfähigkeit von Energie, insbesondere im Bereich der Wasserstoffwirtschaft, eine wichtige Rolle [8].

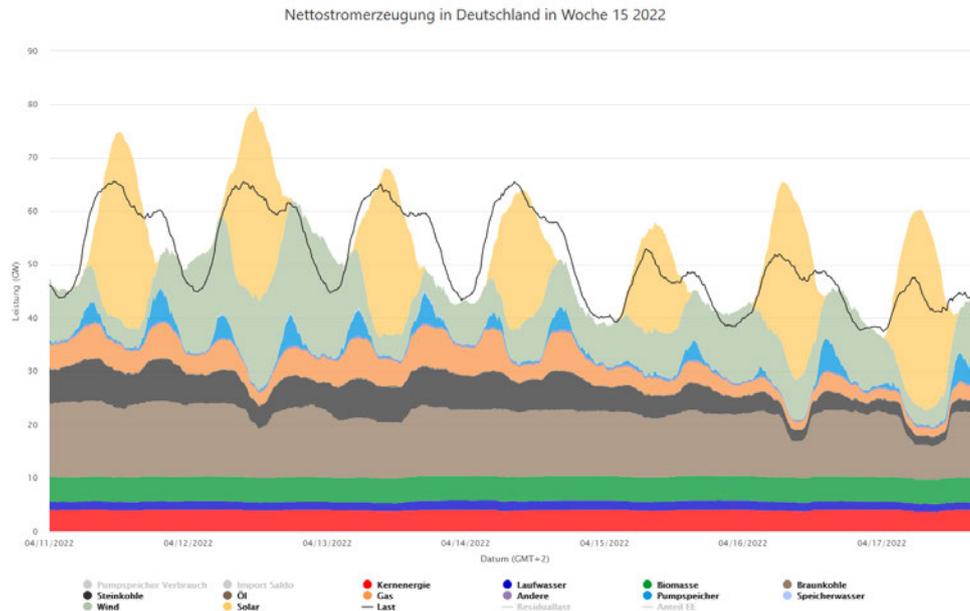


Abbildung 6: Nettostromerzeugung in Deutschland, KW15, 2022 (Quelle: Fraunhofer ISE, energy-charts.info)

Die Wichtigkeit von Flexibilität wird bei der Betrachtung der Abbildung 6 deutlich. Diese zeigt die Stromerzeugung und den Stromverbrauch in der Karwoche 2022. Es ist zu erkennen, dass die Summe der Stromerzeugung der sonnenreichen Tage in den Mittagsstunden die Last überschreitet. Auch die Ungleichgewichte bei der Stromerzeugung durch Flauten oder Tage ohne Sonnenschein oder die Zeiten der heruntergeregelten oder gar abgeschalteten Erzeugungsanlagen können mit Speichern reduziert und bestenfalls vermieden werden. Energiespeichersysteme (ESS) können die Nutzung der klimaneutral gewonnenen Energie zeitlich von der Erzeugung trennen und den verschiedenen Sektoren zur Verfügung stellen. Besonders die Umwandlung von Strom in synthetische Kraftstoffe oder in Wärme/Kälte durch „Power-to-X“-Prozesse, bietet zu der Speicherfähigkeit durch die Sektorenkopplung weitere Flexibilitätspotentiale [9] [10].

3 Anforderungsanalyse

In dieser Arbeit wird die Systemdienlichkeit einer Elektrolyse in Kombination mit einer Methanisierung untersucht, die über eine Leitwarte geregelt wird. Unter der Berücksichtigung der Feldbus-Kommunikation des Gebäudemanagements, und mit Einbezug von verschiedenen Anlagen des TECs, soll die Leitwarte Zustände erkennen, aufnehmen und ändern können. Der Anwendungsfall zur Netzdienlichkeit der Fahrweise der Elektrolyse mit einer biologischer Methanisierung soll dabei helfen, die Leitwarte hinsichtlich der Systemdienlichkeit zu bewerten.

3.1 Konzept der Leitwarte des TECs

Das FM der HAW Hamburg nutzt für die Gebäudeautomation am TEC ein System von Kieback&Peter, darauf aufbauend wird von der Nutzerschaft und den Forschenden das Betriebssystem Zenon Supervisor von Copa Data eingesetzt (siehe Kapitel 2).

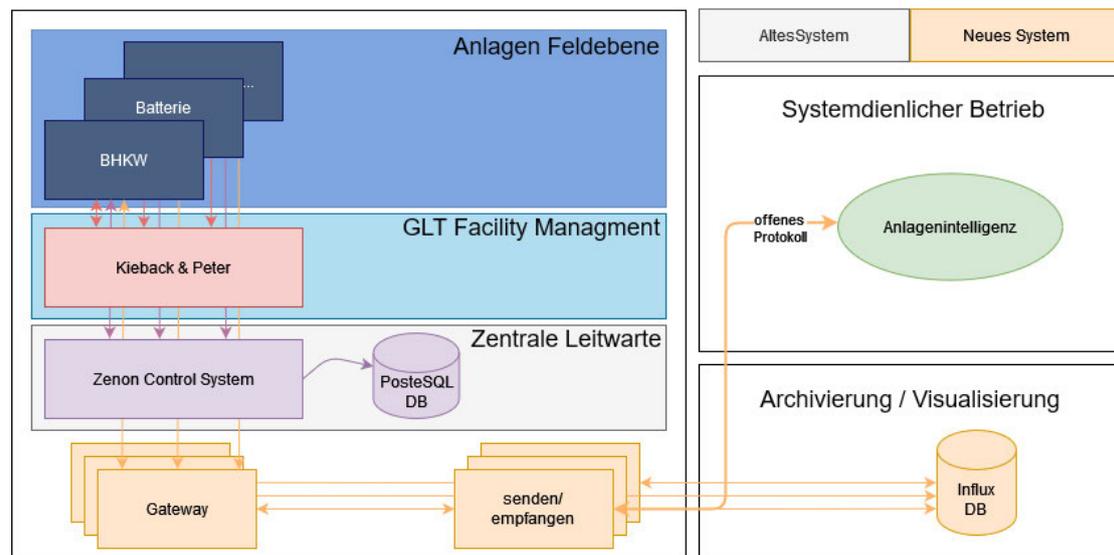


Abbildung 7: Anforderungen an das System

Durch die Fülle an Protokollen, die am TEC und den angeschlossenen Anlagen zum Einsatz kommen, ist der modulare Aufbau und der Bezug der einzelnen Lizenzen bei Zenon mit hohen Kosten verbunden. Zudem ist das System sehr unflexibel für den Betrieb in der Forschung,

obwohl dieses ein zentraler Bestandteil am TEC ist. Aus diesem Grund wird ein neues paralleles System aufgebaut, um einheitliche Schnittstellen für den Forschungsbetrieb zu ermöglichen.

Hierzu ist es notwendig, dass die Anlagen mit ihren unterschiedlichen Protokollen mit Hilfe einer Middleware vereinheitlicht werden. Zum einen wird eine Lesefunktion benötigt, um die Daten der Anlagen zu erfassen und zum anderen wird eine Schreibfunktion benötigt, um Sollwerte setzen zu können. Des Weiteren haben Vorarbeiten gezeigt, dass das Speichern von großen Datenmengen (insbesondere der Statuswerte der Anlagen) in relationalen Datenbanken als sehr unpraktisch ist und hierfür Zeitreihendatenbanken besser geeignet sind. Das beste Ergebnis hat dabei die Influx DB erzielt, die ebenfalls im Rahmen der Arbeit implementiert wird.

Ein weiterer Aspekt, der für den Forschungsbetrieb relevant ist, ist die Möglichkeit unterschiedliche Betriebszenarien mit den Anlagen darstellen zu können. Hierbei muss es eine übergeordnete Ebene geben, die es ermöglicht eine Funktion zu hinterlegen und eine Umgebung zu schaffen, mit der es möglich ist, Funktionen simulativ zu entwickeln und zu testen, bevor diese in Produktion mit den Anlagen in Interaktion treten.

3.2 Hardware- und Softwareanforderungen an die Leitwarte am TEC

Die Leitwarte sollte offen für neue Anlagen und Protokolle sein, dieses beinhaltet auch eine schnelle Aufnahme der neuen Anlagen und eine Testumgebung, die auf die Anlagen zugreifen kann um im Livebetrieb Wertänderungen oder Wertmanipulationen ausführen zu können.

Gesucht wird eine Leitwarte, die den Forschenden und Studierenden am TEC eine Schnittstelle zur Verfügung stellt, um mit unterschiedlichen Programmiersprachen und unter verschiedenen Programmierumgebungen Anlagendaten zu verfolgen und Anlagen zu steuern. Die Erweiterbarkeit des Anlagenparks soll ohne Lizenzweiterung möglich sein und die Leitwarte sollte sich nicht wie eine Blackbox verschließen, sondern offen und zugänglich Daten bereitstellen, und auch eine einfache Fehleranalyse in den Kommunikationsprotokollen erlauben. Hierzu zählt ebenfalls die Möglichkeit einer dezentralen Visualisierung, die möglichst browserbasiert die Visualisierung aller Verläufe anbietet, sodass von überall ein Überblick der aktuellen Situation der Zustände möglich ist. Abschließend ist ein Nutzermanagement notwendig, um Zugriffe zu beschränken und Rechte zu erteilen

3.3 Anwendungsfall Systemdienlicher Betrieb

Im Stromnetz kommt es durch die erneuerbaren Energien zur fluktuierenden Stromerzeugung. Je nachdem wie zu einem Zeitpunkt X die Stromabnahme ist, herrscht entweder ein Strommangel oder Stromüberangebot. Im ersten Fall kann die Stromabnahme nicht gedeckt werden, und es müssen spontan zusätzliche Erzeuger, meist Gaskraftwerke zugeschaltet werden. Im

zweiten Fall, der Überproduktion, entsteht eine Überdeckung der Stromabnahme und es kommt zur Abschaltung von Erzeugern und zu Negativpreisen an der Börse, was zum erhöhten Stromexport an Nachbarländern führt. Damit aber ein stabiler Netzbetrieb möglich ist, ist es zwingend notwendig, dass die Stromerzeugung im Einklang mit der Stromabnahme steht. Der systemdienliche Betrieb achtet darauf, das Ungleichgewicht im Netz auszugleichen. Je nach Stromsituation können die Anlagen am TEC diesen systemdienlichen Betrieb unterstützen. In der folgenden Abbildung lässt sich über eine Verfügbarkeitsmatrix die Nutzungsmöglichkeit der einzelnen Anlagen für die Unterstützung des systemdienlichen Betriebs ablesen.

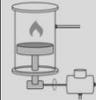
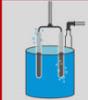
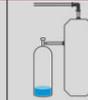
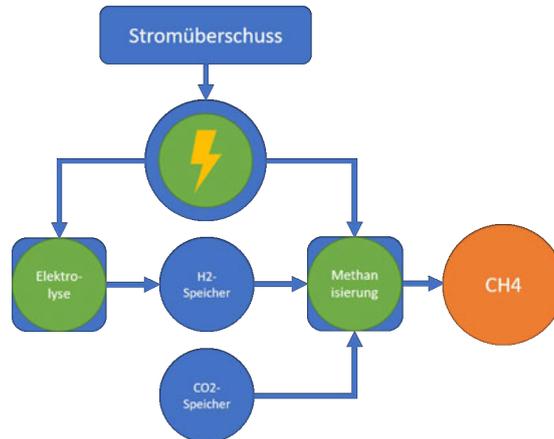
			DAC						
	Batterie	BHKW		E-Auto	Heizstab	Elektrolyse	Methanisierung	PV	Wärmepumpe
Stromüberschuss									
Strommangel									

Abbildung 8: Verfügbarkeitsmatrix mit Blick auf den Power-2-Gas Prozess

Der Anwendungsfall, der in dieser Arbeit betrachtet wird, nutzt das Power-to-Gas Konzept (P2G) am TEC. Dieses Konzept eignet sich Überschussstrom zu nutzen um Bio-Methan zu erzeugen und zu speichern. Um den Energiegehalt des Gases zu erhöhen, wird in einem Bioreaktor mit Hilfe von Archaeen Wasserstoff mit Kohlenstoffdioxid kombiniert und Methan generiert.



Für den systemdienlichen Betrieb müssen die Anlagen näher analysiert werden:

- der PEM-Elektrolyse, die Wasser in seine zwei atomaren Bestandteile Wasserstoff und Sauerstoff spaltet. Der Elektrolyseur beendet die Wasserstoffgeneration selbstständig,

sobald der Druck des Flaschenbündels einen gewissen Grenzdruck erreicht hat. Im Standardfall sind das 15 bar.

- die biologische Methanisierung, nutzt das Verfahren des Sabatier-Prozesses. Die Reaktionsgleichung dazu ist $\text{CO}_2 + 4 \text{H}_2 \rightarrow \text{CH}_4 + 2 \text{H}_2\text{O}$. Dieses Methan kann zum einen dem Erdgasnetz beigemischt werden oder zum anderen im BHKW am TEC rückverstromt werden.

Für den systemdienlichen Betrieb stellen sich daher Fragen, wie flexibel können die Anlagen betrieben werden und welche Restriktionen ergeben sich dabei im Betrieb?

3.4 Eingangsdaten

Um das TEC mit seinen Anlagen systemdienlich zu betreiben, müssen die aktuellen Netzzustände ermittelt werden. Auch prognostizierte Zustände sind hilfreich, sodass eine Anlagenplanung stattfinden kann und eine Bewertung des aktuellen Zustands vorgenommen werden kann.

Im Rahmen von zwei Anwendungsfällen werden unterschiedliche Eingangsdaten verwendet, um das Verhalten des P2G-Konzeptes simulativ zu analysieren. Bei der Bewertung der Anwendungsfälle werden historische Daten verwendet:

1. Um den Netzzustand zu ermitteln, werden die öffentlich zugänglichen Daten des energieportal-hamburg.de genutzt. Dort bietet die Stromnetz Hamburg GmbH historische und aktuelle Daten für jeden Bezirk in Hamburg an. Hierbei werden die Daten vom Bezirk Bergedorf genutzt
2. Anwendungsfall: Hier werden die Erzeugungs- und Verbrauchsdaten vom TEC verwendet, um den systemdienlichen Betrieb zu bestimmen.

Die Daten müssen für den systemdienlichen Betrieb aufbereitet werden, zudem müssen die Netzzustände ausgewertet werden (vgl. Kapitel 6).

Der Idee des geschlossenen Kohlenstoffkreislaufs, der unter dem Anwendungsfall betrachtet werden soll, beruht auf der Annahme, dass nur Strom aus regenerativen Energien für den Prozess eingesetzt wird. Auch die zeitliche Flexibilität, die der Prozess durch den Wasserstoffspeicher und der Fahrweise der Methanisierung hat, gibt dem Verfahren die besten Bedingungen, als Anwendungsfall betrachtet zu werden.

Wie in Abbildung 19 zu entnehmen, könnten Führungsgrößen wie regionaler oder lokaler Stromüberschuss als Steuerungssignal genutzt werden, um die Anwendungsfälle zu bedienen.

3.5 Zusammenfassung und Fazit der Anforderungsanalyse

Auf Grundlage der Anforderungen lassen sich drei wesentliche Fragen festhalten, die nachfolgend näher analysiert werden müssen:

1. Welches Kommunikationsprotokoll ist für eine Vereinheitlichung der Kommunikation am TEC geeignet?
2. Welche Dauer und welche Form haben die Netzzustände und wie werden diese für den systemdienlichen Betrieb für das TEC genutzt?
3. Welche Regelungsgrenzen bzw. flexiblen Betrieb bieten die Elektrolyse und die Methanisierung?

4 Technische Analyse

4.1 Normalbetrieb Elektrolyse und Methanisierung

4.1.1 Beschreibung der Verschaltung der Elektrolyse und Methanisierung

Das Power-to-Gas Anlagenkonzept befindet sich im Raum 3 des TECs. Dieses besteht aus dem PEM-Elektrolyseur mit einem Gastrockner und der biologischen Methanisierung.

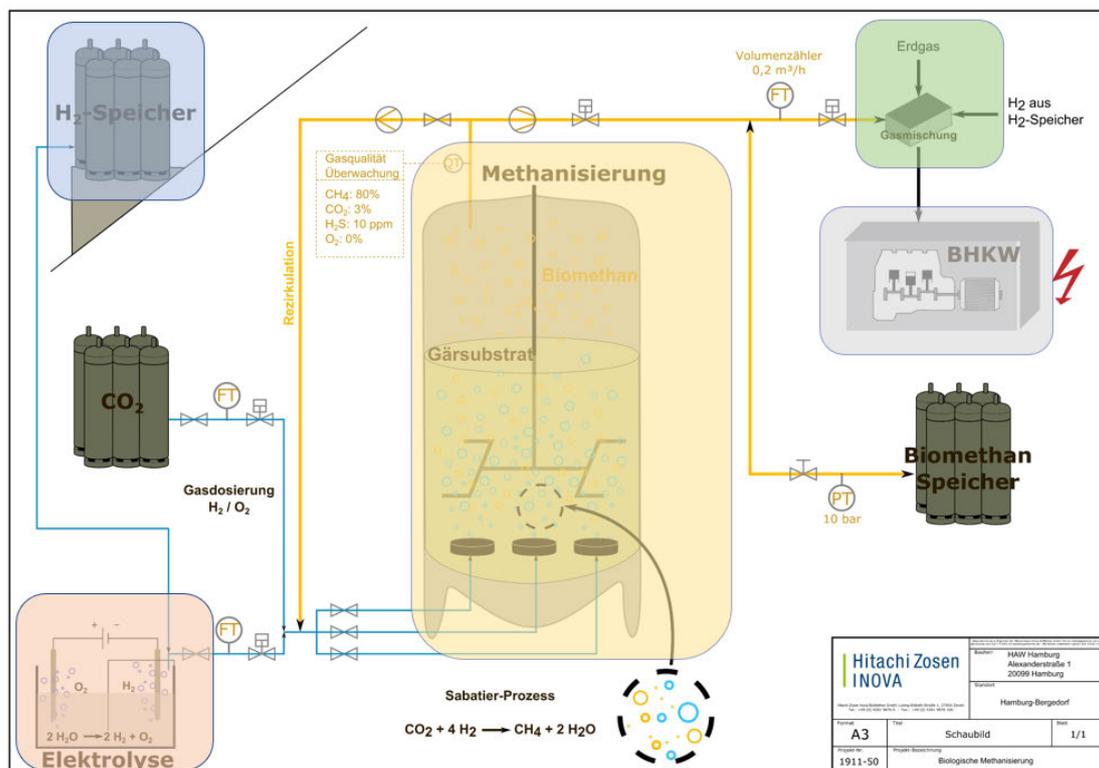


Abbildung 9: Schaubild des Power-to-Gas-Systems am TEC. (Quelle: Schaubild Hitachi Zosen INOVA, Projekt 1911-50 für HAW-Hamburg)

Allgemeine Beschreibung der Anlagenverschaltung am TEC

In der Abbildung 9 ist die Anlagenverschaltung des Power-to-Gas Verfahrens zu sehen. Die Verschaltung besteht aus der PEM-Elektrolyse (roter Bereich) der gekoppelt ist mit dem

Wasserstoffspeicher (blauer Bereich). Die Koppelstelle ist auch mit der Methanisierung verbunden (gelber Bereich). Da zur Herstellung von Bio-Methan CO_2 benötigt wird, wird CO_2 über Flaschen bereitgestellt, Das erzeugte Bio-Methan aus der Methanisierung kann zum einen in den Bio-Methanspeicher gespeichert werden und zum anderen kann das Bio-Methan über den Gasmischer (grüner Bereich) mit bis zu 20 % Wasserstoff angereichert und dem BHKW zugeführt werden, um die Wärmeversorgung des TECs sicherzustellen.

Beschreibung der Elektrolyse am TEC

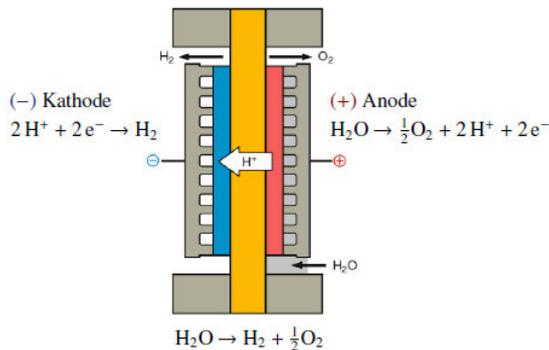


Abbildung 10: PEM- Elektrolyseur (Quelle: [17])

Die Elektrolyse am TEC ist eine PEM Elektrolyse. Das PEM steht für Protonenaustauschmembrane oder im englischen „Proton Exchange Membrane“. Bei dem Elektrolyseprozess wird das Wasser aus dem Trinkwassernetz bezogen, dieses wird gefiltert und deionisiert. Das gereinigte Wasser wird in der Elektrolyse in Stacks geleitet, wo es in seine atomaren Bestandteile in zwei Wasserstoffatome und ein Sauerstoffatom zerlegt wird (vgl. Abbildung 10).

Der Wasserstoffstrom wird über einen Tropfenabscheider entfeuchtet und als reiner Wasserstoff durch eine Druckreglung auf bis zu 15 bar in dem 600 Liter Flaschenbündel – Wasserstoffspeicher - auf dem Dach gespeichert.

Mit der Elektrolyse ist eine Wasserstoffproduktion bis zu 2,9 Nm^3/h möglich.



Abbildung 11: Bilder der Methanisierung und des Wasserstoffspeichers am TEC
(Quelle: Eigene Aufnahmen)

Beschreibung der Methanisierung

Die biologische Methanisierung ist mit Archebakterien gefüllt. Diese benötigen für die Herstellung von Bio-Methan zwei Wasserstoffatome und vier Kohlenstoffdioxid-Moleküle. Der Wasserstoff wird durch die Wasserstoffspeicher zur biologischen Methanisierung geführt. Das Kohlenstoffdioxid wird aktuell durch eine 50 Liter Tauschstahlflasche zugeführt. Seit dem Herbst 2021 befindet sich eine Direct-Air Capture (DAC) Anlage auf dem Dach des TECs der HAW Hamburg, mit der künftig der benötigte Kohlenstoff aus der Atmosphäre gewonnen wird und so der Kohlenstoffkreislauf geschlossen werden soll. Der Wasserstoff wird zusammen mit Kohlenstoffdioxid mittels eines Massendurchflussreglers über die Tellerbelüfter in den unteren Reaktorbereich eingeführt. Im Reaktor befindet sich ein Rührwerk, diese vermischt die Flüssigkeit mit dem Gas.

Die Methanisierung kann bis zu 200 l/h Produktgasgemisch, mit einem Methananteil von mindestens 80 Vol.-%, erzeugen.

Beschreibung Gasmischer am TEC

Der Gasmischer dient zur variablen Beimischung der drei am TEC vorhandenen Gase: Wasserstoff aus der Elektrolyse, methanhaltiges Gas aus der Methanisierung und Erdgas aus dem

öffentlichen Netz. Das Gasgemisch wird anschließend zum BHKW geleitet und dort in Strom und Wärme umgewandelt.



Abbildung 12: Bild des Elektrolyseurs (links) und des Gasmischers (rechts) am TEC
(Quelle: Eigene Aufnahme)

Überblick der Anlagenkenndaten

In den folgenden Tabellen sind die wichtigsten Informationen von der Elektrolyse und der Methanisierung, sowie des Gasmischers und des Wasserstoffspeichers zusammengefasst.

Tabelle 2: Daten PEM-Elektrolyseur am TEC

<i>Hersteller</i>	<i>ITM POWER</i>
Typ	HPac 40
Funktionsweise	PEM
Wasserstoffproduktion	2,92 Nm ³ /h
Wasserstoff-Ausgangsdruck	15 bar
Wasserstoffreinheit	99,99 %
Nennleistung	13,5 kW

Tabelle 3: Daten Methanisierung am TEC

Hersteller	MT BioMethan GmbH
Typ	Biologische Methanisierung – Prototyp
Produktionsmenge	0,2 m ³ /h
Methangehalt	Mind. 80 Vol.-%
Nennleistung / Anschlussleistung	13 kW / 17,3 kVA

Tabelle 4: Daten des Gasmischers am TEC

Hersteller	LT GASETECHNIK
Mischbarkeit	0 - 100% für jedes Gas
Flussregelung	MFC - Mass Flow Controller

Tabelle 5: Gasflaschenbündel

Hersteller	Herose
Inhalt	12 x 50 Liter Druckgasflaschen
max. Betriebsdruck Leitungssystem	20 bar

Kommunikation der Anlagen

Für die Überwachung und Erfassung der Energiewerte des kompletten P2G Raums ist im Verteilerkasten das Netzanalysegerät Sentron PAC3200 der Firma Siemens installiert. Mit diesem Gerät können elektrische Kenngrößen wie Spannungen, Ströme, Leistungen und Frequenz gemessen und überwacht werden.

Sowohl die Elektrolyse als auch die Methanisierung werden durch speicherprogrammierbare Steuerungen (SPS) betrieben.

Die Elektrolyse nutzt die SIMATIC S7-1200, diese wiederum nutzt die CPU 1215C. Die Kommunikationsschnittstelle erfolgt über PROFINET.

Im Schaltschrank der Methanisierung ist eine SIMATIC IPC427C mit einem Intel Celeron verbaut. Das Gerät ist an der Hutschiene des Schaltschranks verbaut und wird von dort über 24V DC versorgt. Auf der SIMATIC läuft eine Windows Embedded 2009, in der über eine WinCC-

Visualisierung der Prozess der Methanisierung angezeigt wird. Für den Datenaustausch und die Kommunikation gibt es ebenfalls eine PROFINET Schnittstelle.

PROFINET (Process field Ethernet) ist einer der führenden Kommunikationsstandards in der Fertigungs- und Prozessautomatisierung für schnelle und komplexe Prozessdatenübertragungen. Durch die Verwendung der Ethernet-Netzwerktechnologien wird eine Verlängerung der Feldebene bis in den Bürobereich geschaffen [11].

Die Erfassung der kompletten Energiewerte des Raumes erfolgt über den PAC 3200. Die Kommunikationsschnittstelle dieses Messgeräts erfolgt über Modbus TCP.

Modbus-TCP ist wie das PROFINET ein Industrial-Ethernet-Bus und in der Automatisierung auch sehr weit verbreitet. Modbus-TCP nutzt das Client-/Server-Prinzip zur Vernetzung von Automatisierungsgeräten und hat eine sehr hohe Datenübertragungsleistung [12].

Für die Steuerung der Anlagen im Power-to-Gas Raum werden die folgenden Datenpunkte gebraucht, um den Anwendungsfall betrachten zu können:

Tabelle 6: Benötigte Datenpunkte für den Anwendungsfall

<i>Datenpunkt</i>	<i>Einheit</i>	<i>Protokoll</i>	<i>Adresse</i>	<i>Beschreibung</i>
Elektrolyse			192.168.241.100	
Status	Zahl	S7	DB210, INT2	Anlagenstatus (100-999)
Start / Stop	Boolean	S7	DB209, X0.1	Elektrolyse starten/beenden
Methanisierung			192.168.241.103	
Status	Zahl	S7	MI104	Anlagenstatus (0-9)
Durchfluss H ₂	l/h	S7	DB1, REAL752	Wasserstoffdurchfluss
Speicherdruck H ₂	bar	S7	DB1, REAL2402	Druck im Wasserstoffspeicher
Sentron PAC 3200			192.168.241.038	
Wirkleistung	Watt	modbus	FC3, 65	

Bei den dargestellten Datenpunkten ist für die Implementierung des Anwendungsfall eine einheitliche Kommunikation nötig.

4.1.2 Betrieb Elektrolyse und Methanisierung

Der Wasserstoffspeicher hat ein Fassungsvermögen von 600 Litern, dieser kann mit bis zu 15 bar befüllt werden. Das Gesetz von Boyle-Mariotte (1) gibt den Zusammenhang zwischen Druck und Volumen eines idealen Gases bei konstanter Temperatur an:

$$p \sim \frac{1}{V} = \text{const.} \quad \text{bzw.} \quad \frac{p_1}{p_2} = \frac{V_2}{V_1} \quad (1)$$

Somit ist unter Berücksichtigung gleichbleibender Temperatur und gleichbleibender Stoffmenge der Druck im Wasserstoffspeicher umgekehrt proportional zum Volumen. Bei einem Speicherdruck von 1 bar ist das Volumen 600 Liter, ist der Speicherdruck 15 bar, beträgt das Volumen 9.000 Liter.

Rechnerisch würde sich daher bei einer Wasserstoffproduktion der Elektrolyse von 2,92 Nm³/h, (umgerechnet ca. 3000 l/h) für die Befüllung des Wasserstoffspeichers mit 15 bar eine Laufzeit von 180 Minuten ergeben.

$$600 \text{ Liter} \times \frac{15}{3000 \frac{\text{Liter}}{\text{h}}} = 3 \text{ h} \quad (2)$$

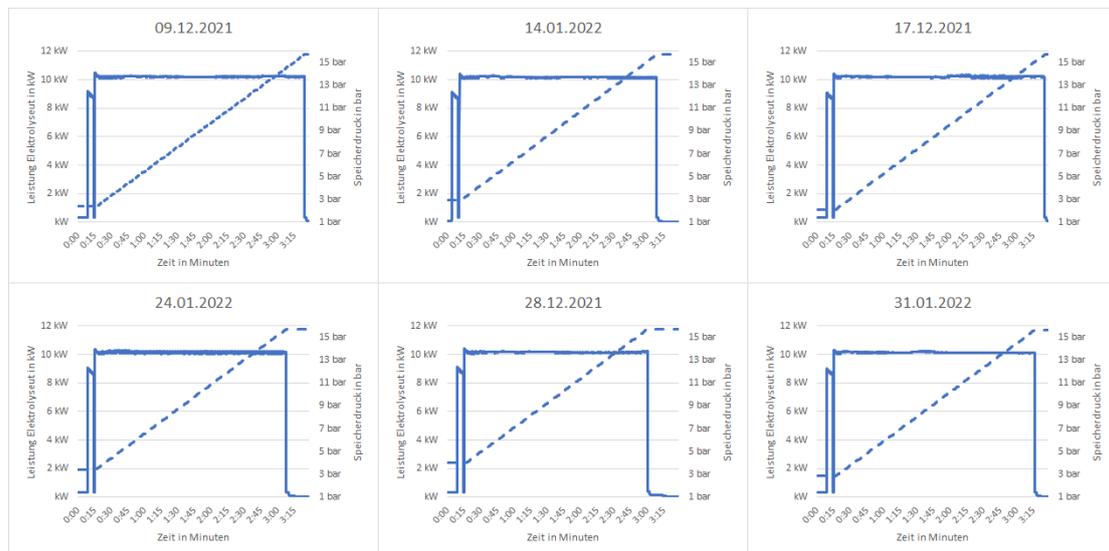


Abbildung 13: Zeitlicher Verlauf der Leistung des Elektrolyseurs und des Speicherdrucks

In Abbildung 13 ist das Ladeverhalten des Wasserstoffspeichers abgebildet. Am Verlauf der Grafen ist zu erkennen, dass nach dem Drucktest (der erste Peak mit 9 kW) die konstante elektrische Leistung des Elektrolyseurs während der Wasserstoffgeneration bei 10kW liegt. Auch der Anstieg des Speicherdrucks ist dabei konstant. Der Wasserstoffspeicher wird von der Elektrolyse in ca. 3 Stunden aufgefüllt. Die Dauer der Laufzeit, bis der Speicher seine 15 bar erreicht hat, hängt unter anderem vom Anfangsdruck und mit der Höhe der gleichzeitigen Entnahme ab.

4.1.3 Ermittlung Regelungsgrenzen und Flexibilität

Zur Ermittlung der Regelungsgrenzen gibt es zum einen die harten Druckgrenzen von 1 bis 15 bar, aber auch die Laufzeitbegrenzung der Elektrolyse. So sollten schnell aufeinanderfolgende Start- und Stoppbefehle an die Elektrolyse vermieden werden, da vor jedem Start die Elektrolyse selbst eine Reihe an Sensormesswerten prüft und einen Drucktest durchläuft. Dabei wird der Wassertank mit Wasser befüllt und auf den maximalen Arbeitsdruck gebracht, um sicherstellen, dass kein Leck vorhanden ist. Als Laufzeitbegrenzung wird deshalb eine minimale Ein- und Ausschaltzeit von 15-Minuten festgelegt.

Um die Flexibilisierung der Anlage größtmöglich nutzen zu können, wird ein Reserveband um den Speicherdruck gelegt, sodass ein möglichst breiter Schaltungskorridor verwendet werden kann. Innerhalb des Reservebands kann je nach Netzstatus die Elektrolyse ein- oder ausgeschaltet werden.



Abbildung 14: Der Wasserstoffspeicher mit seinen Beladegrenzen

Der Wasserstoffspeicher hat seine maximale Ladekapazität bei 15 bar. Fällt der Druck auf 10 bar ab, kann die Elektrolyse den Wasserstoffspeicher erneut beladen. Spätestens bei einem Speicherdruck von 3 bar muss die Elektrolyse wieder Wasserstoff produzieren und wird automatisch eingeschaltet bis mindestens 5 bar erreicht sind. Bei der Zwangseinschaltung ist es egal, wie der Netzstatus ist. Die Einschaltung entspricht einer Mindestlaufzeit von ca. 30 Minuten. Aufgrund der Analysen des Wasserstoffspeichers, wird eine Hystereseregulierung verwendet, um die Regelbänder sicherzustellen. Solange die Hystereseregulierung nicht aktiv ist, können beliebige Schaltbefehle für die Elektrolyse gesetzt werden.

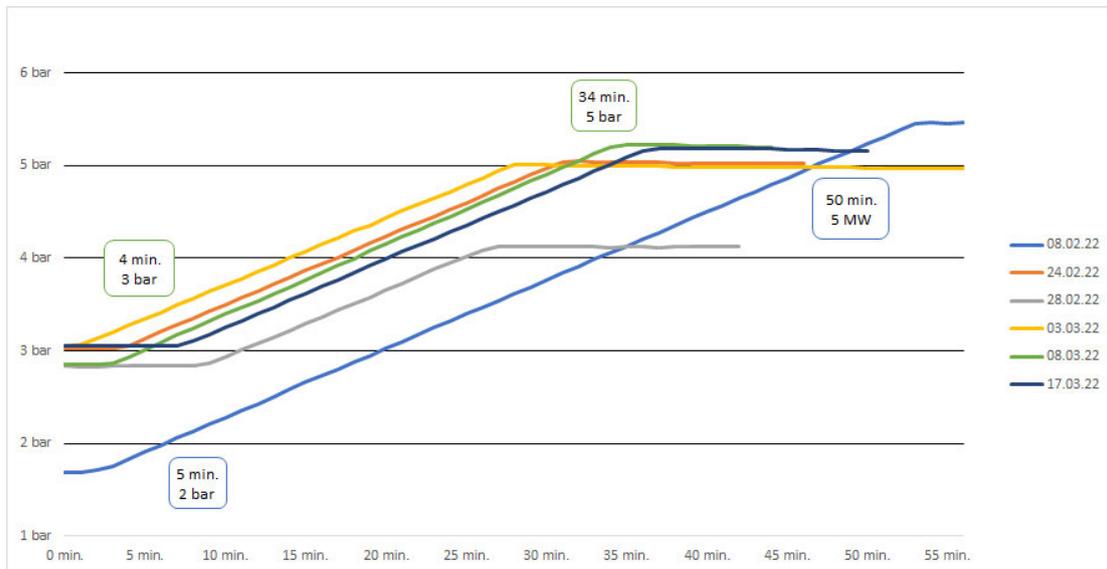


Abbildung 15: Ladeverlauf des H₂-Speichers

Das Diagramm aus Abbildung 15 zeigt den unteren Beladeverlauf an verschiedenen Tagen. Deutlich zu erkennen ist der lineare Verlauf der Druckerhöhung, aus den markierten Datenpunkten kann die Steigung von 1 bar pro 15 Minuten abgelesen werden. Demnach wird bei einer Laufzeit von 30 Minuten der Druck von 3 auf 5 bar angehoben.

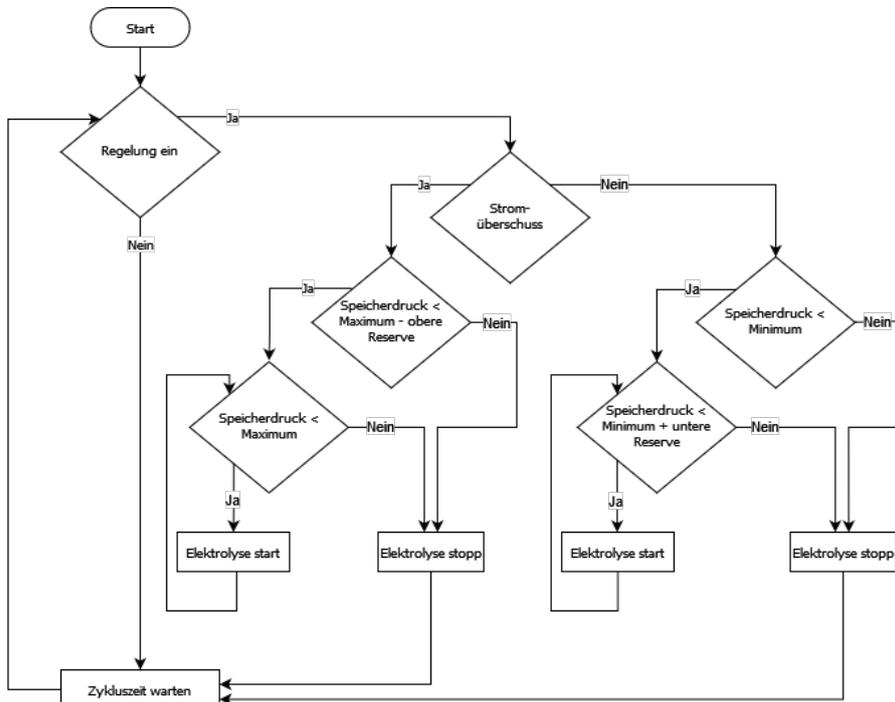


Abbildung 16: Ablaufdiagramm der Schaltungsroutine für die Elektrolyse

Anhand der Routineabfrage aus Abbildung 16 können Einschalt- und Ausschaltbefehle des Elektrolyseurs entschieden werden. Zudem ist auch sichtbar wie die Hystereseregulierung abgebildet ist. Das Ändern des Zustands des Stromüberschusses und damit das Ausführen der Routine bestimmt die Schalthäufigkeit der Anlage.

4.2 Analyse Leitwartenanbindung

Unter Kapitel 2.1 wurde beschrieben, welche Leitwarte vom FM der HAW im Einsatz ist und für die Grundsicherung verantwortlich ist, und welche Software beim Einzug des Smart-Grid Labors in das TEC angeschafft wurde.

Die Bestandsanalyse in 4.1 hat gezeigt, dass eine gesicherte Erfassung von Messwerten der bestehenden Anlagen zur Archivierung, Auswertung und Visualisierung mit der Software Zenon der Leitwarte möglich ist.

Dennoch ist die Erweiterbarkeit des Anlagenparks nur begrenzt möglich, da die installierten Treiber der Zenon Software, auf die bereits bestehenden Kommunikationsprotokolle begrenzt möglich sind. Hinzu kommt das die Implementierung einer übergeordneten Steuerungs- und Regelungslogik mit den realen Anlagen in einem Testbetrieb aktuell nicht möglich ist.

Auch das aktive Managen und Regeln der einzelnen Anlagen kann nur über die zentrale Instanz des Zenon Logic erfolgen, da dies über das integrierte SPS-System auf der Runtime des Leitwartenrechner geschieht. Damit ist keine Anbindung an Programme möglich, die auf anderen Systemen als dem Zenon-Rechner laufen, auch die Anbindung von Regelungen kann nur über SPS-Programme erfolgen, die auf das Soft-SPS-System der Runtime integriert werden.

Zenon speichert alle Variablen in einer Datenpunktliste, dort ist jedes Datenelement hinterlegt. Diese können Sensoren oder Aktoren im Anlagenpark sein, sowie Hilfs- oder Laufvariablen aus der Programmierung. Durch das Nutzen dieser Datenpunktliste ohne irgendeine Rechteabfrage, ist es jedoch jedem Programm möglich, auf alle Stellsignale zu senden und jegliche Zustände zu ändern.

4.2.1 Analyse der Leitwartenkonzepte

Die intelligente Einbindung aller Anlagen aus den Sektoren Mobilität, Wärme, Strom und Gas braucht hingegen eine Leitwarte, die eher als eine Art Middleware zwischen der Feldebene und dem Anwender fungiert. Die Leitwarte sollte als Vermittler dienen, welche die benötigten Rechte zum Stellen von Signalen prüft, aber auch als Übersetzer zwischen den verschiedenen Protokollen arbeitet. Die Leitwarte sollte eine Verwaltungsebene zwischen kontrollierten Datenzugriff und lokaler Infrastruktur darstellen [13].

Durch das Entkoppeln der Datenverarbeitungsanlage, also dem Rechner, auf dem die einzelnen Regelungen der Anlagen laufen, sowie die Kommunikation zu den Anlagen, kommt es zu einer besseren Trennung zwischen Steuern und Ausführen.

Eine Multi-User Fähigkeit sorgt für modulare, leicht ersetzbare und leicht implementierbare Prozesse, die dezentral steuern können, welche Signale geschaltet werden sollen, und die Prozess- und Ablaufsteuerung wird diese Schaltsignale prüfen und ausführen [14].

Es muss eine Schnittstelle geschaffen werden, bei dem die Möglichkeit besteht, übergeordnete Routinen/Funktionen zunächst simulativ zu entwickeln und diese anschließend zu überführen, um diese mit den realen Anlagen zu koppeln.

Ein weiterer Aspekt der erfüllt werden muss, ist die Datensicherung, die zum einen durch die Anlagen erfasst werden und zum anderen durch die eigenen entwickelten Routinen/Funktionen berechnet wurden.

Berücksichtigt man diese Anforderungen ergeben sich klare Trennungen der Ebenen im Gesamtkonzept. Diese sind:

1. Leitwarterechner: Dieser sorgt zum einen dafür, dass die Kommunikation zu den Anlagen sichergestellt wird, trotz der unterschiedlichen Kommunikationsprotokolle in der Feldebene und zum anderen für eine einheitliche Kommunikation zu den anderen Ebenen.
2. Datenbankserver: Dieser soll genutzt werden, um die Daten der Anlagen und den eigenen entwickelten Routinen/Funktionen zu sichern. Als Datenbank eignen sich besonders Zeitreihendatenbanken. Im Rahmen der Arbeit „Evaluation of Tools for visually analysing Time Series Data“ [15] fanden Voranalysen zu Zeitreihendatenbanken statt, sodass eine InfluxDB implementiert wird.
3. Development- & Deployment Server: Dieser dient zum einen zur Entwicklung von eigenen Routinen/Funktionen, die simulativ getestet werden und zum anderen erlaubt er die eigenen Routinen/Funktionen mit den realen Anlagen zu interagieren.

4.2.2 Analyse Kommunikationsprotokolle

Die stärkere Vernetzung und dezentralisierte Organisation zukünftiger Anlagenstrukturen sowie die weltweit möglich gewordene Kommunikation von technischen Systemen über das Internet fordern den Einsatz von Middleware-Systemen für die Umsetzung der Industrie 4.0-Kommunikation [16].

Die Middleware soll dabei als Plattform zwischen den Anwendungsprogrammen und der Infrastruktur fungieren. Dieses kann mit Kommunikationsprotokollen realisiert werden. Die Kommunikation, die am TEC zukünftig für die Vermaschung der Anlagenstrukturen sorgt, braucht eine klare und einfache Struktur, die trotz einer großen Anzahl an Maschinen und

Maschinendaten Informationen bündelt und gruppieren kann. Nutzer können benötigte Informationen somit leicht eingrenzen und die Leitwarte kann über die gruppierten Informationen eine übersichtliche Rechtevergabe erteilen.

Die Kommunikationsprotokolle der Anlagen am TEC sind:

- PROFINET
- Modbus-TCP
- OPC-DA

Diese Protokolle müssen mittels der Middleware vereinheitlicht werden. Durch den wachsenden Trend des Internet of Things (IoT) haben sich neue Wege der Interaktion ergeben, die auch für das TEC eine wichtige Rolle in der künftigen Entwicklung spielen. Es werden nicht nur immer mehr Geräte „smart“ und können von überall gesteuert werden, sondern auch das Netzwerk ändert sich. Es sind eine Reihe von IoT-Protokollen entstanden, die zur Kommunikation zwischen den verbundenen Geräten verwendet werden.

Ein Protokoll, welches schon 1999 speziell für die Machine-to-Machine Kommunikation entwickelt wurde, ist das Message Queuing Telemetry Transport (MQTT) Protokoll. Es wurde damals entwickelt, um durch verhältnismäßig kurze Nachrichten Ressourcen wie geringem Speicher oder geringer Bandbreite sparsam zu nutzen. Durch diesen platzsparenden Datenaustausch sind viele gleichzeitige Verbindungen mit kurzen Latenzzeiten möglich. [17]

Basierend auf der Publish-Subscriber-Architektur werden Nachrichten über eine zentrale Einheit, dem Broker, vom Sender zum Empfänger geleitet. Auch die Authentifizierung übernimmt der MQTT-Broker, gegenüber den sich die Teilnehmer mittels Benutzername und Passwort verifizieren müssen. Die Verschlüsselung der Nachrichten und die Authentifizierung der Teilnehmer geschieht bereits in der Transportebene und findet mittels TLS (Transport Layer Security) statt. So kann selbst über unsichere Kanäle wie dem Internet eine sichere Kommunikation stattfinden.

Aufgrund dieser Anforderungen wurde die Lösung von CYBUS am TEC implementiert. Durch Cybus werden Anlagen ganz einfach mit IT-Systemen verbunden. Es können gleichermaßen interne und externe digitale Services integriert werden. Dieses wird mittels der Connectware Dateninfrastruktur sichergestellt. Cybus bietet unterschiedliche Kommunikationsprotokolle zu den Anlagen und vereinheitlicht diese über MQTT. Dies stellt die zentrale Komponente für den Leitwartenrechner dar.

4.3 Anbindung, Methoden zur Steuerung der Leitwarte

4.3.1 Analyse der Programmiersprache

Am TEC werden in unterschiedlichen Projekten diverse Programmiersprachen genutzt, um Simulationen zu erstellen, Algorithmen zu formulieren oder problemorientierte Anweisungen zu schreiben.

So hat die HAW zum einen Verträge mit Herstellern von proprietärer Software, wie zum Beispiel mit Mathworks Matlab, die eine akademische Nutzung von MATLAB und Simulink erlaubt. Zum anderen werden aber aus den Drittmittelprojekten am TEC der Einsatz von freier Software, wie z.B. Python oder Julia vermehrt genutzt.

Die Grenzen einer kommerziellen Software sind schnell erreicht. Außerhalb eines Campusvertrages ist die Grundlizenz für die Nutzung teuer, aber auch für zusätzliche Toolboxen, die die Funktionalität von Software erweitern sollen, fallen Gebühren an. Bei einer Auftragsforschung oder Kooperationsforschung, spätestens aber bei der Ausgründung einer Forschergruppe kommt die Frage zum Nutzungsrecht von nicht freier Software auf.

Als vielseitigste, und vor allem schnell und einfach zu erlernende Programmiersprache hat sich die offene Software Python durchgesetzt. Besonders die klare und prägnante Syntax hat in diversen und interdisziplinären Arbeitsgruppen große Beliebtheit erlangt. Die Standardbibliothek ist sehr umfangreich, und es gibt eine breite Sammlung an Python Bibliotheken, die von der Datenanalyse oder der Automatisierung bis hin zum wissenschaftlichen Rechnen Anwendungen findet [18].

Am TEC gibt es ein Jupyter-Hub Server, auf dem Python Code über einen Webbrowser geschrieben und ausgeführt werden kann. Der Server stellt dabei mehreren Benutzern sogenannte Jupyter-Notebooks bereit, um darauf Code zu generieren und Daten zu errechnen bzw. zu simulieren.

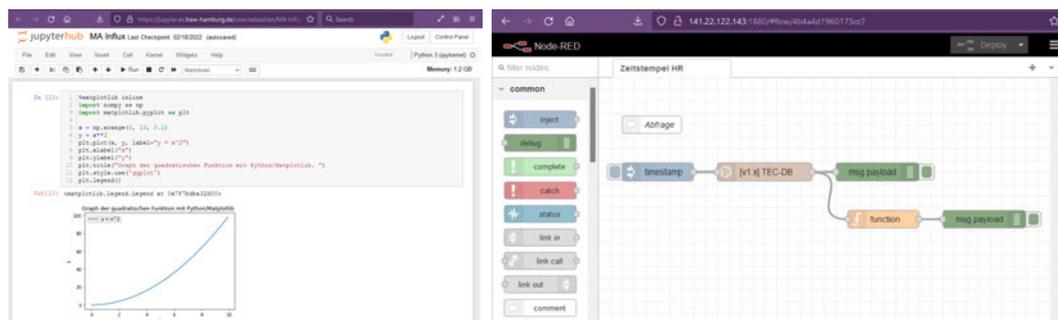


Abbildung 17: Screenshot von jupyter.ec.haw-hamburg.de und Node-RED

Zudem genießt die Software Node-RED eine große Beliebtheit am TEC. Diese wurde ursprünglich von IBM entwickelt und 2016 unter Apache-Lizenz als Open-Source-Projekt veröffentlicht [19]. Mit der Node-RED Workbench können im Baukastenprinzip Anwendungsfälle umgesetzt werden, dadurch eignet sie sich für den schnellen und einfachen Prototyping-Test. Die einzelnen Funktionsbausteine aus der Node-RED Workbench werden durch Ziehen von Verbindungen (Drag and Drop) verknüpft, so lassen sich leicht Sensoren mit Aktoren verbinden und die verschiedenen Signale kontrollieren, manipulieren und steuern. Der browserbasierte Editor, der auch auf einfachen Einplatinenrechnern wie dem Raspberry Pi läuft, macht es einfach, Simulationen zu erstellen.

Mit Node-RED als Programmierwerkzeug lassen sich Geräte, APIs und Online-Dienste auf einfache Weise miteinander verbinden.

4.3.2 Regelungs- / Steuerungskonzept für den Use Case netzdienlicher Betrieb

Für den Anwendungsfall der Elektrolyse mit der biologischen Methanisierung für den netzdienlichen Betrieb soll Python als Programmiersprache genutzt werden. Um Regelungen nicht nur permanent als Prozess laufen zu lassen, soll das Regelungskonzept mithilfe einer betriebssystem-internen Zeittafel, dem sogenannten Crontab, ausgeführt werden. Der Crontab steuert in vielen unixbasierten Systemen das periodische und automatische Ausführen von Anweisungen, wie das Archivieren oder Löschen von Logdateien, oder das Aktualisieren von Systemprogrammen. Durch das mit einem Cronjob automatische Starten von Routinen können so auch Programme gesteuert werden, die zu einer bestimmten Zeit ausgeführt werden.

Für die Schaltberechtigung wird der User erstellt, der die Rechte zur Veränderung von Werten für Anlagen hat. Für den Anwendungsfall erhält dieser Nutzer daher Schreibrechte auf alle Datenpunkte, die für die Regelung nötig sind. Diese betrifft für die Arbeit das Ein- und Ausschalten der Elektrolyse. Für die allgemeine Werteabfrage wird ein weiterer User erstellt. Dieser Nutzer hat Leserechte auf alle Datenpunkte. Durch die Trennung zwischen lesendem Nutzer und schreibberechtigten Nutzer soll die Rechtevergabe einfach gehalten werden. So können beim Testen weiterer Regelungen alle Zustände ohne zusätzliche Rechte abgefragt werden, jedoch nicht direkt in die Prozesse eingegriffen werden.

4.4 Analyse der Netzzustände mittels Daten von Stromnetz Hamburg

Für die Anwendungsfälle sollen verschiedene Eingangsgrößen als Schaltempfehlungen analysiert werden. Ein Anwendungsfall betrachtet dabei den netzdienlichen Betrieb im Bezirk Bergedorf. Die Daten zu der EE-Stromerzeugung und dem Stromverbrauch sind detailliert nach den sieben Bezirken der Stadt Hamburg, über die Online Plattform *energieportal-hamburg.de* abrufbar. Die Daten werden vom Verteilnetzbetreiber Stromnetz Hamburg veröffentlicht. Als Eigentümer des Hamburger Stromverteilungsnetzes und der dazugehörigen Netzanlagen, ist

Stromnetz Hamburg verantwortlich für den sicheren Netzbetrieb in Hamburg. Auf dem Energieportal kann anhand von Diagrammen der Strom, der in Hamburg benötigt wird und den Strom, der im Hamburger Netz erzeugt wird, abgelesen werden. Die Daten können auch als Textdatei vom Portal heruntergeladen werden und sollen für den Anwendungsfall auf dem Datenbankserver in der InfluxDB hinterlegt werden.

Energieportal Hamburg



Abbildung 18: Screenshot der Seite "energieportal-hamburg.de"

4.4.1 Aufbereitung der „Daten vom Energieportal Hamburg“

Mit Hilfe des Python Pakets „Requests“ können die Daten über das Portal direkt und kontinuierlich abgerufen werden. Für die Automatisierung der Abfrage wurde ein Skript geschrieben, dass eine Anfrage stündlich ausführt und die Daten zur Weiterverarbeitung in die Datenbank des TECs schreibt.

Listing 1: Lastabfrage durch „Energieportal-Hamburg-Scrapper.py“

```
def run():
    lst = []
    # alle 7 Bezirke werden abgefragt
    for region in ['eimsbuettel', 'nord', 'wandsbek', 'mitte'...]:
        url = f'https://www.energieportal-hamburg.de/?region-key={region}'...
        try:
            # Werte werden als JSON gespeichert
            s = requests.session()
            s.get(url, timeout=(5,20))
            value = json.loads(s.get(..., timeout=(5,20)).text)
        except:
            logging.info('Scrapper Failed. Timed Out')
            return None
        try:
            ...
            # der JSON Objekt bekommt die Struktur der Datenbank
```

```
lst.append(  
    {  
        "measurement": "Last",  
        "tags": {  
            "Region": f"(region)",  
        },  
        "time": f"(end_time)",  
        "fields": {  
            "Last [MW]": value['graphset']["0"]...['values'][-1:]["0"]  
        }  
    }  
)
```

In der Datenbank sind die Erzeugungs- und Verbrauchswerte der 7 Bezirke von Hamburg bis zurück zum Januar 2012 in 15-minütiger Auflösung abrufbar.

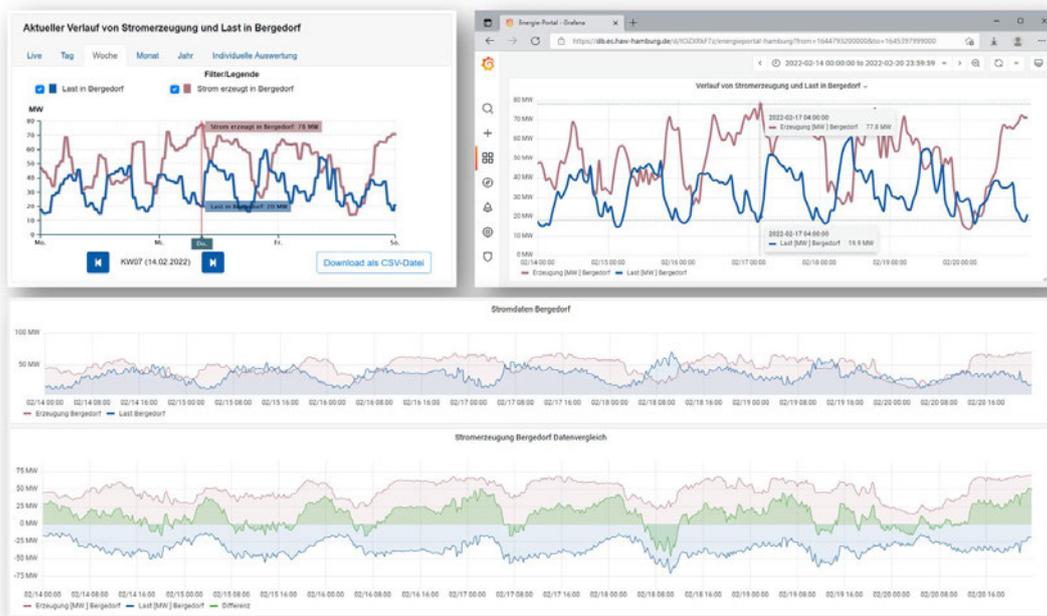


Abbildung 19: Vergleich der Daten von der Website und der Datenbank (KW 07, 2022)

4.4.2 Auswertung Netzzustände für Hamburg und den Bezirk Bergedorf

Mit den Werten des Stromportals können die Zeiten gefunden und genutzt werden, in denen es in den Bezirken einen Stromüberschuss gibt. Diese sollen genutzt werden, um den Betrieb der Elektrolyse netzdienlich zu betreiben.

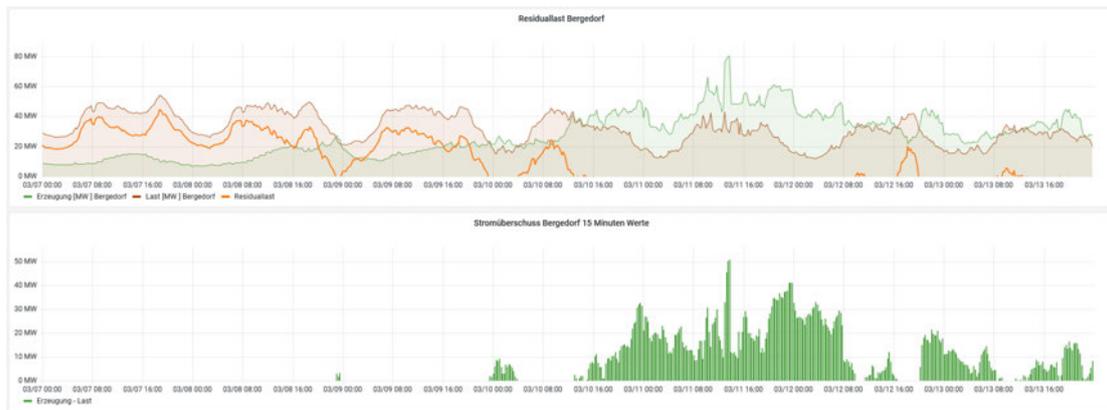


Abbildung 20: Verlauf der Residuallast und der Stromüberschuss-Viertelstunden (KW 10, 2022)

4.5 Datenbanken

Durch das enorme Wachsen der digitalen Technologien und dem Verarbeiten und Auswerten der anfallenden Datenmengen, sind im Bereich der Datenverarbeitung eine große Bandbreite an Konzepten und Softwarelösungen entstanden. Die Tabellenkalkulationen, wie sie am Heimrechner oder bei Präsentationen genutzt werden, berechnen Daten aus fest vorgegebenen Bereichen. Aufwändig wird es jedoch, wenn sich der Bereich vergrößert, weil neue Daten hinzukommen, oder Bereiche auf weitere Systeme erweitert werden.

Um Daten weiterhin in Tabellenform zu verarbeiten, ist das Konzept der Datenbank bedeutend. In den Datenbanken wird das Schema, wie das Format der Daten vorzuliegen hat, vorgegeben. Es wird auf reiner Datenebene gearbeitet, eine Formatierung der Daten findet dabei nicht statt.

Für die Speicherung der anfallenden Daten wird unter zenon eine PostgreSQL Datenbank benutzt. Dieses Datenbankmanagementsystem (DBMS) hält sich stark an den ISO und IEC SQL-Standard, indem die Sprache auf der relationalen Algebra, also Operationen wie Vereinigung, Verbund oder Selektion und Projektion basiert.

Die anfallenden Daten der Sensoren, die am TEC aufgenommen werden, liefern hingegen einen konstanten Datenstrom. Das Aufnehmen der Daten geschieht als Tupel, es gibt einen Wert mit einem bestimmten Zeitstempel. Bei einer Abfrage dieser Werte geht es dann um Zeitintervalle, um Trends zu erkennen oder Peaks und Unregelmäßigkeiten zu entdecken.

4.5.1 Daten in der Influx

Am TEC gibt es ein Open Source DBMS speziell für Zeitreihen, die InfluxDB. Es befinden sich schon mehrere Datensätze in der InfluxDB. So gibt es die schon erwähnten Daten vom Stromportal, es werden Wärme- und Windprognosen gespeichert, auch Raumtemperaturen

werden auf der InfluxDB gespeichert. Die Erweiterung für weitere Datenpunkte sind sehr simple umzusetzen, so sind die Anlagendaten des TECs wie folgt abgelegt:

*„io/energie-campus/ *Projekt* oder *Anlage* / Datengruppe / Datenpunkt“*

4.5.2 Grafana

Zur Auswertung und graphischer Darstellung ergänzt sich zur Influx Datenbank das Software-tool Grafana. Die plattformübergreifende Open-Source Anwendung kann Daten aus einer Influx-Datenbank abfragen und in verschiedenste Anzeigenformen mittels eines Webbrowsers ausgeben. Eine Bachelorarbeit beim CC4E zur Evaluation eines Administrationswerkzeugs für Datenbankmanagementsystem kam zu der Empfehlung, Grafana für Zeitreihendatenbanken einzusetzen [15].

4.5.3 Datenbankkonzept

Das Konzept der Speicherung der Daten soll der Architektur der Datenpunkte, wie sie über die Kommunikation über MQTT stattfinden, angelehnt sein. So sollen die Daten direkt nach dem messen im JSON-Format an eine Datenbank des TECs übergeben werden, mit mindestens 3 Informationen: Name des Messwerts, gemessener Wert und Zeitstempel.

4.6 Design

Auf Basis der Anforderungsanalyse, die in der technischen Analyse näher betrachtet wurde, konnte ein finales Design für diese Arbeit bestimmt werden und ist in der folgenden Abbildung 21 dargestellt.

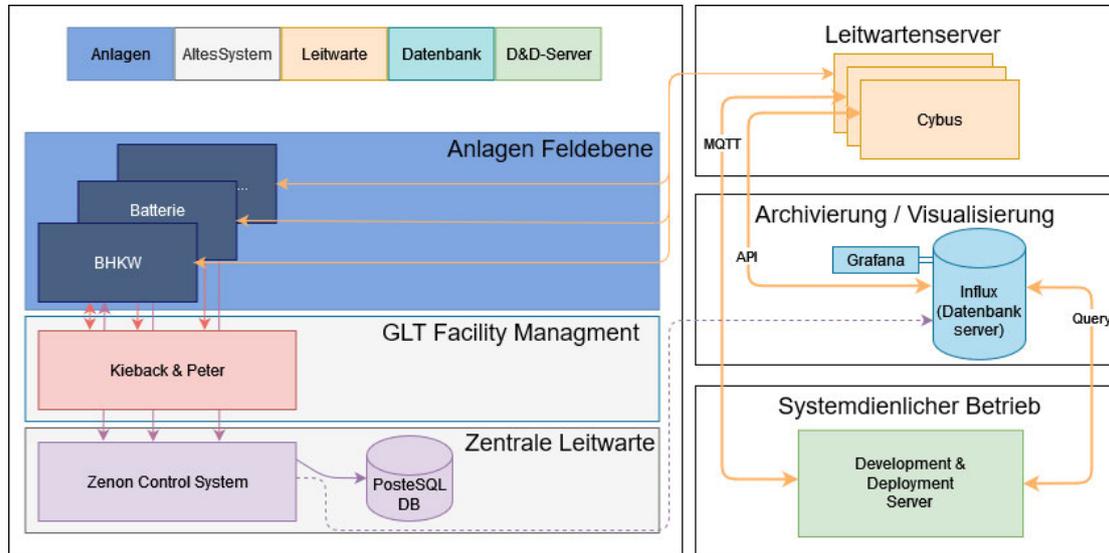


Abbildung 21: Realisierung des Systems aus den Überlegungen der Anforderungsanalyse

Die neue Leitwarte wird modular aufgesetzt und besteht aus drei Servern – Leitwarte, Development- & Deploymentserver und Datenbankserver (vgl. Abbildung 21).

- Leitwarte: Der Cybus mit der Connectware wird auf dem Leitwartenrechner laufen (orange), auf dem das Mapping der einzelnen Protokolle auf MQTT stattfindet und die Rechtevergabe auf die einzelnen Datenpunkte.
- Datenbankserver: Alle Datenpunkte sowie die Änderungen derer wird direkt auf den Datenbankserver (blau) geschrieben, der auch die Visualisierung der Daten mit Grafana übernimmt.
- Development- & Deploymentserver: Den kontinuierlichen Ablauf oder auch den zeitlich gesteuerten Ablauf der Algorithmen übernimmt der DD-Server (grün), der die Zustände der Anlagen und Sensoren über MQTT überwacht und auch Abfrage auf die Leitwarte stellen kann. Von dort werden durch die Routinen die Anlagen und die Aktoren über MQTT geschaltet.

5 Implementierung

In diesem Kapitel wird auf die Integration der neuen Leitwarte und die Ansteuerung der Anlagen am TEC eingegangen.

5.1 Die Hardware

Für die Umsetzung des Designs in Kapitel 4.6 stehen am TEC drei Server zur Verfügung. Die Spezifikationen der Server sind:

- einen Datenbankserver, auf dem auch die Visualisierung laufen wird
 - Intel Xeon 4110 8-Core, 2.40GHz, 32GB RAM
- den Leitwartenrechner, auf dem die Connectware von Cybus läuft
 - E5-2620V3 6-Core, 3.20GHz, 16GB RAM
- und einen Development Server für den Einsatz der Dockercontainer
 - AMD EPYC 7302P 16-Core, 3.00GHz, 64GB RAM

Auf den drei Servern ist das Betriebssystem Ubuntu Linux-Distribution als Server Edition implementiert. Dieses bietet neben diversen zusätzlichen Serverdiensten und einer begrenzten grafische Nutzeroberfläche, auch einen verlängerten Long Time Support gegenüber den Desktop Editionen an.

5.1.1 Die Firewall

Die HAW Hamburg wird bei den zentralen IT-Diensten durch das Informationstechnik Service Center (ITSC) betreut. Dazu gehört auch der Betrieb der Kommunikationsnetzwerke und die dazugehörigen Basisdienste, wie die Überwachung des laufenden Datenverkehrs. Um zu entscheiden, welche Netzwerkpakete durchgelassen oder welche Netzwerkzugriffe abgewehrt werden, sind festgelegte Regeln definiert. Im Rahmen der Arbeit mussten für die Implementierung der neuen Leitwarte die Netzwerkadressen angepasst werden, damit die Server nicht nur intern im HAW Hamburg Netz zur Verfügung stehen, sondern auch von außen über VPN erreichbar sind (vgl. Tabelle 7).

Tabelle 7: Permit Regeln der Firewall als Access-list

<i>Netzwerkadresse</i>	<i>Quelle</i>	<i>Ziel</i>	<i>Traffic / Port</i>
Datenbankserver	any	Datenbankserver	eq www
	any	Datenbankserver	eq 443
	141.22.0.0 0.0.255.255	Datenbankserver	eq 5432
	141.22.0.0 0.0.255.255	Datenbankserver	eq 8086
Leitwartenrechner	any	Leitwartenrechner	eq www
	any	Leitwartenrechner	eq 443
	141.22.0.0 0.0.255.255	Leitwartenrechner	eq 1883
	141.22.0.0 0.0.255.255	Leitwartenrechner	eq 8883
Development und Deployment Server	141.22.0.0 0.0.255.255	DDS	eq www
	141.22.0.0 0.0.255.255	DDS	eq 443

In Tabelle 7 sind die drei Server mit den Regeln für die TCP-Verbindungen abgebildet. Der Leitwartenrechner und der Datenbankserver sind über http (www) und über http over SSL/TLS (443, https) aus dem gesamten Web (wildcard „any“) erreichbar. Bei dem Datenbankserver sind die Ports für die PostgreSQL (notwendig für die ZENON Leitwarte) und die InfluxDB geöffnet. Beim Leitwartenrechner sind die Ports für MQTT und Secure MQTT ebenfalls geöffnet. Der Development & Deployment Server (DDS), der mit Docker versehen laufen wird, wird nur im VPN der HAW Hamburg erreichbar sein.

Bei dem Subnetz des TECs handelt es sich um ein Netzwerk der Klasse C und wird technisch mit 141.22.122.0/24 klassifiziert. Das letzte Oktett der IP kann an die Teilnehmer (Hosts) im Netzwerk selbst vergeben werden. Da dies eine Länge von 8 Bit hat, die 0 allerdings für das Netz und die 255 für die Broadcast-Adresse reserviert ist, ergibt sich eine maximale Teilnehmeranzahl von 254 Hosts, die in dem Subnetz adressiert werden können. In dem Subnetz des TECs gibt es keine dazwischengeschaltete Firewall, innerhalb des Netzes gibt es weder technische Regeln noch Beschränkungen für die Netzwerkzugriffe unter den einzelnen Teilnehmer.

5.1.2 Nginx mit Let's Encrypt

Das ITSC hat unter der Subdomäne „*ec.haw-hamburg.de*“ weitere Domains eingerichtet, die von den Servern verwendet werden. Um mit diesen registrierten Domains eine verschlüsselte SSL-Verbindung aufbauen zu können, gibt es freie und offene Zertifizierungsstellen. Diese

stellen dem Server ein SSL-Zertifikat aus. Für die Server am TEC wird der Service von Let's Encrypt verwendet. Mit Hilfe des Zertifikats kann sichergestellt werden, dass die Verbindung zu oder von zum Beispiel Grafana über HTTPS verschlüsselt wird.

Um vom Internet auf die Dienste der Server weitergeleitet zu werden, wird die freie Webserver-Software Nginx verwendet. Dazu wird auf dem Datenbankserver eine Webserver-Software installiert, um für Anfragen als Reverseproxy das Softwaretool Grafana von außen erreichbar zu machen. Es wird zuerst in der Konfigurationsdatei unter `/etc/nginx/sites-available/` für die Seite unter location „`proxy_pass http://localhost:3000;`“ eingetragen. Dadurch werden alle Anfragen, die an den Server gerichtet sind, auf den Standardport von Grafana weitergeleitet.

Um nun ein Zertifikat bei Let's Encrypt zu erstellen und regelmäßig zu erneuern, wird der Certbot aus der Paketquelle von Ubuntu als Nginx-Plugin installiert und ausgeführt.

Listing 2: Certbot wird mit dem nginx Plugin ausgeführt

```
$ sudo certbot --nginx -d db.ec.haw-hamburg.de
```

Nach der Eingabe startet der Certbot eine Abfrage an den Let's Encrypt-Server und validiert die Domäne, für die das Zertifikat erstellt wird. Dieses ist nach der Prüfung 90 Tage gültig und wird durch den `certbot.timer` automatisch erneuert. Wird die Grafana-Seite nun im Browser aufgerufen, wird die verschlüsselte HTTPS-Verbindung anhand des Zertifikats angezeigt.

5.2 Der Leitwartenrechner

Auf dem Leitwartenrechner wird nach der Installation der Server-Edition von Ubuntu und das Einrichten der beiden Netzwerkkarten das aktuelle Installationsskript der Firma Cybus über deren Internetseite heruntergeladen und implementiert. Durch das Skript wird die Connectware installiert, die dann über den Browser erreichbar ist.

Listing 3: Cybus-Script für die Installation ausführen

```
$ wget -O ./connectware-online-installer.sh https://download.cybus.io/latest/connectware-online-installer.sh
$ chmod +x ./connectware-online-installer.sh
$ sudo ./connectware-online-installer.sh
```

5.2.1 Das Admin-Interface

Zum Einrichten der Connectware wird die Server IP in den Browser eingegeben, worauf hin das Admin-Interface mit dem Home Screen angezeigt wird. Es gibt in einer seitlichen Navigationsleiste weitere Menüpunkte wie Services, User Management und Workbench.

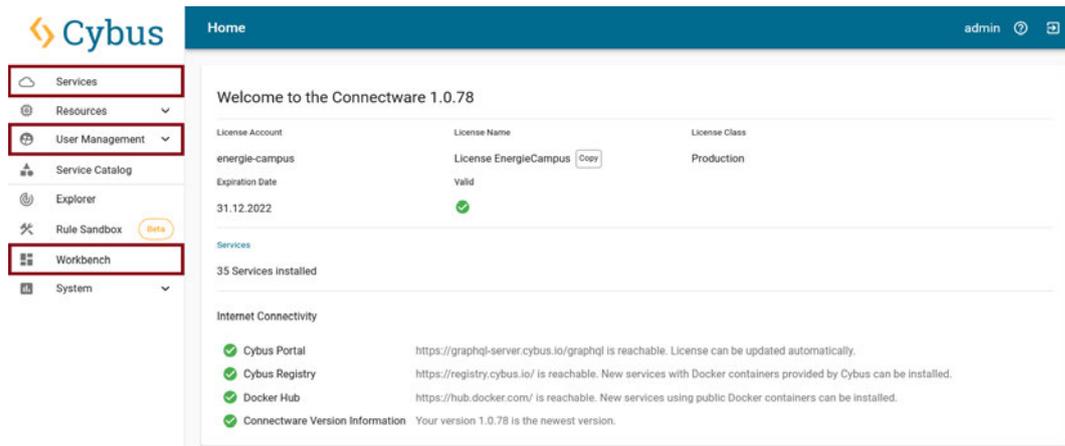


Abbildung 22: Das User Interface der Connectware

Auf der Startseite werden die ersten, wichtigen Informationen wie die ausgeführte Version und die Anzahl der installierten Services angezeigt. Zudem wird der Verbindungsstatus zu Internet-Serveradressen, die für weitere Dienste erforderlich sind, angezeigt. Dies ist zum Beispiel der Docker Hub, der Dienste von Drittanbietern installiert und aktiviert. Für den Betrieb der Connectware am TECs ist keine der Verbindungen erforderlich. Für die grundlegende Konfiguration werden in den folgenden Unterkapiteln auf die rot markierten Menüpunkte in Abbildung 22 eingegangen.

5.2.2 Die Services

Unter den Services sind Anlagen mit all Ihren Ressourcen und Einstellungen aufgeführt. Ein Service wird über eine commissioning-Datei eingepflegt. In dieser Inbetriebnahmedatei, eine textbasierte Datei im YAML-Format, sind alle Parametrierungen wie das Herstellen von Verbindungen oder das Festlegen von Berechtigungen, aufgelistet.

Listing 4: YAML-File des Siemens PAC3200 aus Raum3

```

description: PAC3200 - Elektrolyse
metadata:
  name: PAC3200 - Elektrolyse
  version: '1.0'
definitions:
  CYBUS MQTT ROOT: io/cybus/energie-campus/raw/energie/elektrolyse
  IP Address: 192.168.241.38
  Port_Number: 502
resources:
  modbusConnection:
    type: Cybus::Connection
    properties:
      protocol: Modbus
      targetState: connected
    connection:
      host: !ref IP_Address
      port: !ref Port_Number

```

```
wirkleistungsgesamt:
  type: Cybus::Endpoint
  properties:
    protocol: Modbus
    connection: !ref modbusConnection
    topic: wirkleistung/gesamt
  subscribe:
    fc: 3
    length: 2
    address: 65
    interval: 5000
    dataType: floatBE
```

In Listing 4 ist der Aufbau der YAML des Netzanalysegeräts PAC3200 für den Power-to-Gas Raum am TEC aufgeführt. Mit Hilfe des YAML -Files können Konfigurationen durchgeführt werden. Beginnend werden Metadaten wie die Beschreibung des Geräts und der Name festgehalten, sowie die Versionsnummer zur Aufnahme der Änderungen. Danach kommen die Definitionen der Anlagen, das sind die Parameter der Anlage, die parametrisiert werden. Die Parameter sind IP-Adresse und der Port, über den die Kommunikation stattfindet. Zudem wird die MQTT Hauptadresse angegeben. Da alle Netzanalysegeräte unter dem Topic „energie“ laufen, wurde für dieses Gerät die Adresse „./energie/elektrolyse“ angegeben. Der Abschnitt Ressourcen besteht zum einen aus der Verbindung, in diesem Fall ist es die Modbus-TCP Verbindung, und den Endpunkten, die in der Datei beispielhaft auf den Wirkleistungswert verweist.

Tabelle 8: Auszug Messgrößentabelle

<i>Offset</i>	<i>Anzahl Register</i>	<i>Name</i>	<i>Format</i>	<i>Einheit</i>	<i>Zugriff</i>
65	2	Summe der Wirkleistung	Float	W	R

Die Messgröße *Summe der Wirkleistung* wird, wie in Tabelle 8 aufgelistet, im Modbus-TCP-Register der Adresse 65 zur Verfügung gestellt. Da der Zugriff lesend ist, wird der Modbus Function Code „read Analog Output Holding Registers“, FC=03, in das YAML-File geschrieben. Das in dem YAML-File angegebene Intervall von 5000 gibt die Abfragegeschwindigkeit in Millisekunden an, in diesem Fall alle 5 Sekunden.

5.2.3 Das User Managment

Das User Management dient der Rechteverwaltung. Hier können Identitäten als Nutzer erstellt, die dem Nutzer zugehörigen Datenberechtigungen bearbeitet oder Passwörter geändert werden. Auch die Rollen, die Nutzer zu einer Gruppe vereinen, z.B. die administrative Zugriffsberechtigungen, werden im User Management verwaltet.

Die einzelnen Berechtigungen, die einem Nutzer zugeordnet werden sollen, können entweder den Zugriff auf Topics über den Broker mittels MQTT oder den Zugriff über die API mittels HTTPS sein. Die Zugriffsarten werden über den Ressourcenpfad entlang der MQTT-Topic

Strukturen und mit den Berechtigungen, *read*, *write* oder *readWrite* für beides vergeben werden.

In der technischen Analyse 4.3.2 wurde die Erstellung zweier Nutzer gezeigt, um die Rechtevergabe zu steuern. Die beiden Nutzer wurden dafür in dem User Management erstellt, ein User, der auf alle Topics Leserechte hat (*Netzdienlichkeit*), und ein Nutzer, der auf die Schaltpunkte Schreibzugriff hat (*Netzdiener*).

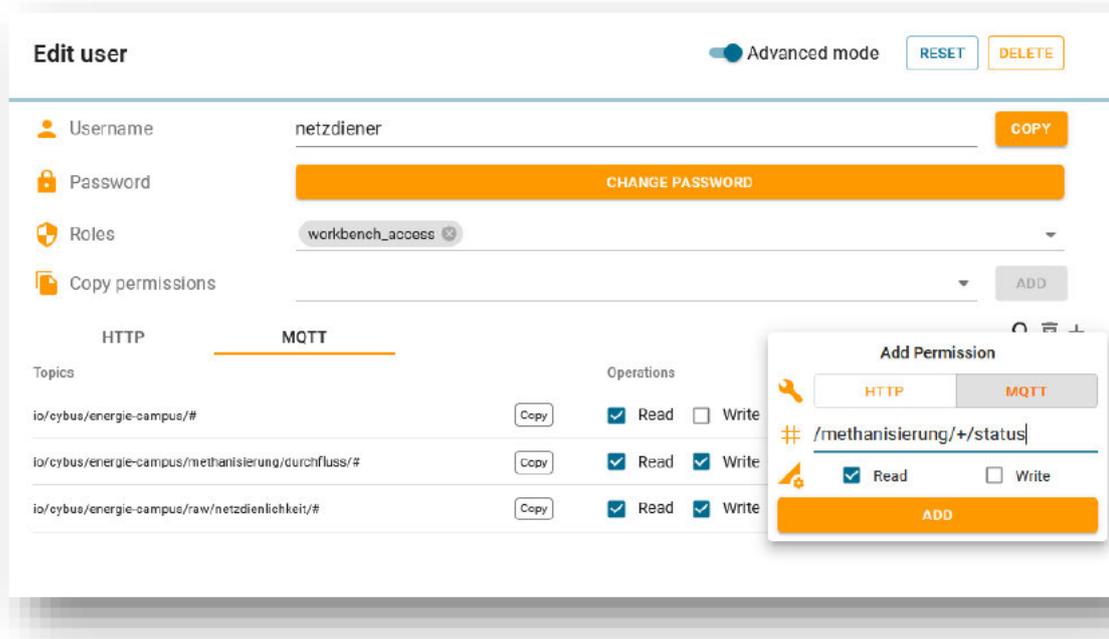


Abbildung 23: Rollen und Rechte des Users Netzdiener

In Abbildung 23 sind die Rechte des schreibberechtigten Nutzers „netzdiener“ zu sehen, der Nutzer hat Zugang zu der workbench und leserechte auf die Topics, die mit *io/cybus/energie-campus/* starten, da das '#'-Zeichen für den vollständigen Teilbaum der Hierarchie steht, es muss daher auch das letzte Zeichen in der Zeichenkette sein. Die neu erstellte Berechtigung für das Topic */methanisierung/+/status* steht für eine einzelne Ebene der Hierarchie und wird als Begrenzungszeichen verwendet. Die */methanisierung/+/status* Resource trifft auf die Status der Ventile *methanisierung/V210/status* wie auch auf *methanisierung/V220/status* zu.

5.2.4 Workbench

Die Cybus Connectware wird auch mit einer laufenden Node-RED Instance ausgeliefert. Sobald im Navigationsbereich auf den Menüpunkt Node-RED geklickt wird, öffnet sich der Node-Red Editor in einem neuen Browserfenster.

Über den Browser können über die Palette auf der linken Seite Nodes ausgewählt werden und von dort auf die Arbeitsfläche gezogen und verbunden werden. Die verknüpften Nodes geben in dem entstandenen Flussdiagramm ihre Nachrichten als Objekte (*msg*) weiter und es kann mittels Java-Script-Funktionen direkt in den Nachrichtenverlauf eingegriffen werden.

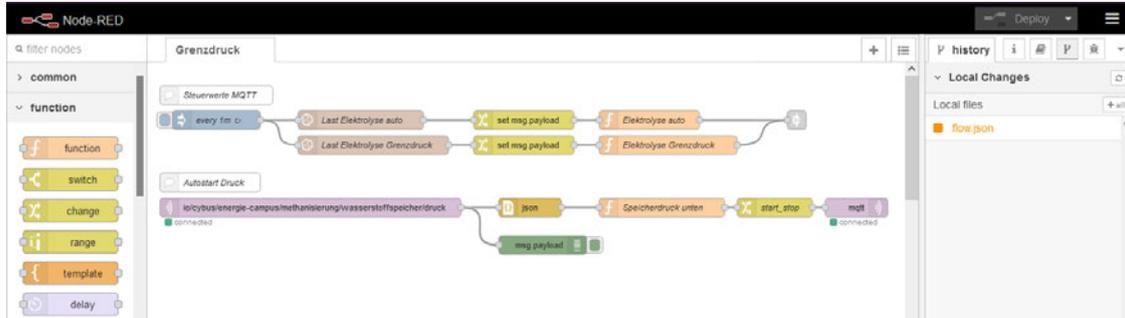


Abbildung 24: Grenzdruck-Flow unter Node-RED

In Abbildung 24 ist die Druckhaltung des Wasserstoffspeichers unter Node-RED abgebildet. Die Druckhaltung ist ein Prozess, der aktiviert werden kann, wenn keine Regelung der Wasserstoffgeneration stattfindet. So soll sichergestellt werden, dass ab einem bestimmten Grenzwert (z. Zt. 3bar) die Elektrolyse startet und den Speicher wieder füllt. Der Grenzwert und auch ob die Druckhaltung aktiviert ist, wird im oberen Bereich der Workbench durch eine Datenbankabfrage (braune Nodes) geprüft. Die Datenbankabfrage wird minütlich der blauen Inject-Node getätigt, sodass selbst bei einem Neustart der Connectware die Werte aktuell sind.

Darunter wird das MQTT-Signal unter dem Topic *io/cybus/energie-campus/methanisierung/wasserstoffspeicher/druck* abonniert (lila Node). Das Signal geht von dort zum einen in eine Debug-Node (grün), das dem Nutzer direkt ankommende Werte anzeigt und gleichzeitig geht das Signal als json-Objekt in einen Funktionsblock, wo das ankommende MQTT-Signal verarbeitet wird:

Listing 5: Grenzdruck-Funktion als Node in der Workbench

```
if (msg.payload.value < context.global.elek_grenz && context.global.elek_auto &&
!context.global.elek_running)
{
  msg.payload = {value:true};
  context.global.elek_triggered = true
  return msg;
}
```

Durch die geschriebene Funktion wird bei der eingehenden Nachricht der Wert mit dem Grenzwert verglichen. Ist dieser unterschritten und die Steuerung aktiviert, und ist die Elektrolyse noch nicht gestartet, wird ein „True“ Signal ausgegeben, das dann über MQTT an die Connectware gesendet wird und zum Einschalten der Elektrolyse führt.

In der Workbench wurde auch die Git- Integrationsfunktionen aktiviert, um die Quellcodekontrolle eines Versionsverwaltungssystem zu aktivieren. Es können somit Änderungen auf den Git-Server der HAW mit *push* übertragen werden, und im Notfall auch das Repository geklont werden. Das Repository besteht aus den Flow-Dateien, einer Credential-Datei (.json-Datei) und einer Runtime (einige .js-Dateien) sowie einer package-json-Datei, die alle benötigten Anwendungscoddateien enthält.

5.2.5 SSL-Zertifikat für den Leitwartenrechner

Um das SSL-Zertifikat zu ändern oder zu aktualisieren, darf die Connectware nicht laufen. Um sie zu beenden, muss ein Shell-Interface zum Leitwartenrechner, z.B. über das Terminal, aufgerufen werden, wo der Service gestoppt und dann der certbot aufgerufen und auch das Zertifikat der Connectware hinzugefügt werden kann.

Listing 6: Aktualisieren des SSL-Zertifikats des Leitwartenrechner

```
$ sudo bash
$ service connectware stop
$ certbot renew
$ service connectware start
$ docker cp -L /etc/letsencrypt/live/energie-campus.cybus.io/privkey.pem
connectware connectware 1:/connectware certs/cybus server.key
$ docker cp -L /etc/letsencrypt/live/energie-campus.cybus.io/fullchain.pem
connectware connectware 1:/connectware certs/cybus server.crt
$ docker restart connectware_connectware_1
```

Danach kann der Connectware-Proxy neu gestartet werden, um die geänderten Dateien zu übernehmen.

5.3 Der Datenbankserver

Auf dem Datenbankserver sollen die entstandenen und aufgenommenen Werte des TECs abgelegt werden. Darunter fallen die Werte, die der Leitwartenrechner erfasst, und die Werte die Zenon Runtime aufnimmt und in eine PostgreSQL-Datenbank speichert. Für die neue Leitwarte sollen auf dem Datenbankserver Simulationswerte und Datenreihen in eine InfluxDB abgelegt werden können, um diese effizient und dauerhaft zu speichern. Im Folgenden wird die Einrichtung näher vorgestellt.

5.3.1 Installation und Konfiguration der Datenbanken Influx und PostgreSQL

Alle Server sind im Serverraum des TECs. In dem Serverraum besteht die Möglichkeit die Installation vor Ort durchzuführen. Hierfür steht ein Standtisch mit Monitor und Tastatur bereit. Da es keine Restriktionen innerhalb des Campusnetzes gibt, können nach dem Einstellen des BIOS, des Hardwareraids und der Vergabe der statischen IP-Adresse alle Befehle mit einem anderen Rechner über eine SSH-Verbindung an den Server gesendet werden.

Listing 7: Installation und Einrichten der Datenbanken

```
# Paketlisten eingelesen und aktualisieren,
# und falls vorhanden, auf neue Paketversion upgraden
$ sudo apt-get update && sudo apt-get upgrade
# Runterladen und installieren der InfluxDB.deb
$ wget https://dl.influxdata.com/influxdb/releases/influxdb2-2.x.x-xxx.deb
$ sudo dpkg -i influxdb2 2.x.x xxx.deb
# InfluxDB starten
$ sudo service influxdb start
# PostgreSQL aus der offiziellen Quelle installieren
$ sudo apt-get install postgresql
# Nach einem Neustart testen, ob der Service automatisch läuft
$ systemctl status postgresql
● postgresql.service - PostgreSQL RDBMS
  Loaded: loaded
  (/lib/systemd/system/postgresql.service; enabled; vendor preset: enabled)
  Active: active (running)
$ systemctl status influxdb
● influxdb.service - InfluxDB is an open-source, distributed, time series database
  Loaded: loaded
  (/lib/systemd/system/influxdb.service; enabled; vendor preset: enable>
  Active: active (running)
  └─ /usr/bin/influxd -config /etc/influxdb/influxdb.conf
```

Für die Werte aus dem Leitwartenrechner wird die Datenbank TEC erstellt und der Nutzer Leitwarte mit Schreibrechten erzeugt. Das Benutzermanagement soll auf der Leitwarte mit unpersönlichen Accounts erfolgen, es werden die Lese- und Schreibrechte über Nutzer vergeben, die Projekt- oder Arbeitsgruppen (AG)-bezogen benannt werden. Die Struktur der Ablage von Datenpunkten auf die Datenbank TEC soll der Topologie des Anlagenparks am TEC widerspiegeln und entspricht daher auch der Hierarchie, wie Nachrichten von der Leitwarte über die jeweiligen Topics gesendet werden.

```
> SHOW MEASUREMENTS ON TEC WITH MEASUREMENT =~ /elektrol.*/ LIMIT 2
name: measurements
name
----
io/cybus/energie-campus/elektrolyse/start stop
io/cybus/energie-campus/elektrolyse/status
>
>
>
> select last("value") from "io/cybus/energie-campus/elektrolyse/status"
name: io/cybus/energie-campus/elektrolyse/status
time                last
----                ----
1649325118111       103
> █
```

Abbildung 25: Measurement- und Datenabfrage in der Influx

In Abbildung 25 ist eine Abfrage von Influx-Datenpunkte aufgenommen. Es wird zuerst nach den ersten beiden Datenpunkten der „elektrolyse“ abgefragt. Die zweite Abfrage soll den letzten Wert des Datenpunkts „./elektrolyse/status“ ausgeben. Es ist zu erkennen, dass die Ausgabe des Datenpunkts dem Inhalt des MQTT-Protokolls entspricht. Das MQTT-Topic ist in der Datenbank mit dem Namen des Measurements „io/cybus/energie-

campus/elektrolyse/status“ belegt. Die gesendete MQTT-Message {"value":103,"timestamp":1649325118111}" stellt dabei ein Datentuple aus Zeitstempel und Wert da.

Auch die Visualisierungsanwendung Grafana wird direkt auf dem Datenbankserver laufen.

Listing 8: Installation und Einrichten von Grafana

```
# Dependencies
$ sudo apt-get install -y adduser libfontconfigl
# Runterladen und installieren der .deb
$ wget https://dl.grafana.com/oss/release/grafana 8.x.x amd64.deb
$ sudo dpkg -i grafana 8.x.x amd64.deb
# Grafana starten
$ sudo systemctl enable grafana-server
# Nach Neustart testen, ob der Service automatisch läuft
$ systemctl status grafana-server
● grafana-server.service - Grafana instance
   Loaded: loaded
          (/lib/systemd/system/grafana-server.service; enabled; vendor preset: enabled)
   Active: active (running)
           └─ /usr/sbin/grafana-server --config=/etc/grafana/grafana.ini
```

Die Datenbanken sind installiert und Grafana ist nun einsatzbereit.

5.3.2 Weitere Konfigurationen

Die Konfigurationsdatei für Grafana ist unter `./etc/grafana/grafana.ini` gespeichert, für die Influx unter `./etc/influxdb/influxdb.conf`. Hier können weitere Einstellungen und Direktive erstellt und aktiviert werden. Bei den Datenbanken werden die Speicherorte auf die größere Datenfestplatte festgelegt und „Storage“ genannt.

Bei Grafana wird das SMTP-Verfahren für die E-Mail-Benachrichtigung konfiguriert. Hier werden Warnungen oder Alarme eingestellt und dann unter dieser eingerichteten Adresse automatisiert verschickt. Ebenfalls wurde in der Grafanakonfiguration eine Organisation festgelegt. Diese kann nichts editieren, aber sich anonym (ohne sich einzuloggen) mit Grafana verbinden. Damit können, anstelle einer Login-Seite als Startseite, ein festgelegtes Dashboard ohne Anmeldedaten angezeigt werden. Mit der Adresse „`https://db.ec.haw-hamburg.de/d/CC4E/tec`“ kann der aktuelle Zustand der Anlagen am TEC dargestellt werden (vgl. Abbildung 26).

gekapselten Umgebung frei vom Haupt-Betriebssystem laufen. Diese Unabhängigkeit soll dazu dienen, bei den einzelnen Algorithmen und Anwendungen nicht auf bestimmte Versionen des Betriebssystems oder auf bestimmte Software-Bibliotheken des Host-Betriebssystems Rücksicht nehmen zu müssen.

Anders als bei einer virtuellen Maschine wird die Software, die mittel Docker in einem Container gekapselt gestartet und ausgeführt wird, als einfacher Prozess laufen. Wohingegen bei Virtualisierungsumgebungen gleich ganze virtuelle Computer mit virtueller Hardware wie RAM, Prozessor und Festplatte zur Verfügung gestellt werden [20].

Ein Container braucht daher keinen Kernel-Module- oder Hardwaretreiber, und es muss auch kein Betriebssystem gebootet werden.



Abbildung 27: Docker Logo Moby Dock. (Quelle: www.docker.com)

Ein Docker-Container wird aus einem Abbild erstellt, dem Container-Image. Dieses Image wird auf dem Host-Betriebssystem des DDS als laufender Docker Container gestartet. Das Container-Image enthält das auszuführende Programm mit den notwendigen Abhängigkeiten [20]. Da das Container-Image kein Gast-Betriebssystem benötigt und nur das auszuführende Programm mit den benötigten Komponenten beinhaltet, eignet es sich sehr gut als Template. So kommt es selbst bei häufig geänderten Entwicklungsversionen von einzelnen Containern zu keiner Vermüllung des Systems, da die Prozesse in sich geschlossen sind und Container schnell entfernt oder erneuert werden können.

5.4.2 Container-Aufbau

Für das Erstellen eines Containers, in den dann das Programm kontinuierlich oder zeitgesteuert ausgeführt werden soll, müssen verschiedene Schritte ausgeführt werden.

Es wird als erstes ein Image benötigt, das als Vorlage zum Erzeugen eines oder auch mehrerer identischer Container dient. Durch die Dockersoftware wird dieses Image dann als Docker-Container ausgeführt. Das kann auf den verschiedensten Systemen passieren, auf denen Docker installiert ist.

Für den kontinuierlichen Betrieb ist zu dem Python Programm mit zugehöriger requirement- und config-File, ein Dockerfile und die docker-compose-YAML Datei notwendig.



Abbildung 28: Dateien für den kontinuierlichen Container

Die Datei *requirements.txt* ist eine Anforderungsdatei, die alle Python-Module, die für das auszuführende Skript notwendig sind, beinhaltet. Es handelt sich um eine reine Textdatei, in der die benötigte Bibliothek und, wenn nötig, auch die zu benutzende Paketversion steht. Durch den Befehl `pip3 install -r requirements.txt` werden alle in der requirements-Datei ausgeführten Module in die Projektumgebung installiert. Viele Programmierumgebungen erstellen auch mit wenigen Klicks eine Liste der requirements, oder installieren aus dieser direkt, sodass beim Teilen oder beim zur Verfügung stellen von Code auch die richtigen Bibliotheken installiert werden können.

Listing 9: Inhalt der requirements.txt-Datei

```
pandas
matplotlib == 3.5.1
numpy >= 1.22.0
```

Oben abgebildete requirements.txt-Datei erfordert für das Python Programm die Programm-bibliotheken pandas, Matplotlib Version 3.5.1 und numpy mit der Version 1.22.0 oder höher.

In der *config.ini*-Datei werden alle Zugangsdaten und Passwörter hinterlegt, sowie weitere kritische Informationen, die nicht weiterverbreitet werden sollen. Das Speichern dieser Informationen außerhalb der Programmdatei soll das versehentliche veröffentlichen von Zugangsdaten verhindern.

Listing 10: Inhalt der config.ini-Datei

```
[USERINFO]
benutzername = user
passwort = 123456
[SERVERCONFIG]
host = 192.168.1.100
port = 80
```

Oben abgebildet ist das Schema der config-Datei, in dem Programmcode können dann die einzelnen Informationen mit der Python-Bibliothek *configparser* als Variablen eingebunden werden.

Listing 11: Beispielscript mit configparser

```
from configparser import ConfigParser

# Einlesen der config.ini Datei
config_object = ConfigParser()
config_object.read("config.ini")

# Ausgabe der Informationen
```

```
userinfo = config_object["USERINFO"]
print("Das Passwort ist {}".format(userinfo["password"]))
```

Wenn der Programmcode in Repositories bereitgestellt wird, z.B. auf einem GIT, kann in der Reposit-Datei *.gitignore* die config-Datei hinterlegt werden. Dadurch ist die Config-datei vom pushen, also vom Hochladen der lokalen Version auf einen Git-Server, ausgenommen. Es sollte dann aber ein ‚leeres‘ Template der config-Datei mit angehängt werden, dass allen Mitarbeitern bekannt ist, wie der Aufbau der Datei aussieht und welche benötigten Zugänge in der Datei hinterlegt werden sollen. Aus diesem Grund wird eine weitere Datei *config-example.ini* erstellt, in der alle kritischen Werte gelöscht, aber die benötigten Informationen als Template erkennbar sind.

Zum Erstellen des Images benötigt Docker abschließend das sogenannte *Dockerfile*, eine Textdatei, die alle Befehle in der Reihenfolge enthält, die für die automatische Herstellung nötig sind.

Listing 12: Inhalt der Dockerfile

```
FROM python:3.10.2-slim
RUN apt-get update && apt-get install -y gcc
COPY ./requirements.txt requirements.txt
# Install any needed packages specified in requirements.txt
RUN pip3 install -r requirements.txt
RUN pip3 install pytz
# Change TZ locale
RUN echo "Europe/Berlin" > /etc/timezone
RUN dpkg-reconfigure -f noninteractive tzdata
ENV TZ = "Europe/Berlin"
# Copy the batch_routines directory contents into the container at /batch_routines
COPY . /batch_routines
# Set the working directory to /batch_routines
WORKDIR /batch_routines
ENTRYPOINT ["python3", "run_batch_routines.py"]
```

In dem oben dargestellten Dockerfile, dass auch für die Images, die eine Python Applikation beinhalten, angewendet wird, werden folgenden Anweisungen ausgeführt:

- FROM erstellt das Docker-Image anhand des „Basis-Image“ `python:3.10.2-slim`, das aktuellste bereitgestellte und gepflegte Python-Image vom offiziellen Docker-Hub.
- RUN führt Befehle innerhalb des Containers aus
- COPY fügt Dateien aus dem aktuellen Verzeichnis dem Image hinzu
- WORKDIR legt das Arbeitsverzeichnis für z.B. RUN und COPY fest
- ENTRYPOINT (im JSON-Array-Format) gibt die Anweisung vor, die beim Start in den ausführenden Container gemacht werden soll. Hier soll die Python-Datei mit `Python3` ausgeführt werden.

Mit diesem Dockerfile wird ein Image auf Grundlage des Python-Slim-Layer erstellt, nochmals geupdated und die benötigte Requirements installiert und die Zeitzone auf die Mitteleuropäische gestellt.

Da es häufiger Aufgaben oder Funktionen von Programmen gibt, die nicht nur gestartet werden und dann kontinuierlich laufen oder nach dem Start einmal ihren Algorithmus ausführen und sich dann beenden, soll es die Möglichkeit geben, Programme in Containern zu festgelegten Zeiten wiederholt zu starten. Umgesetzt wurde das mit der Datei `batch_config.xml`, die Programme wiederkehrend als Cronjob ausführt.

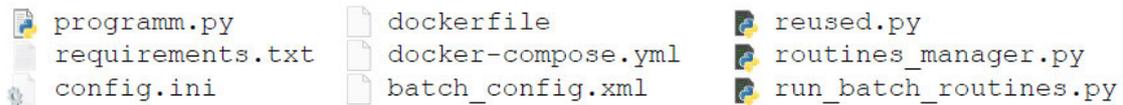


Abbildung 29: Dateien für den Aufbau des routine-Container

Die `batch_config.xml`-Datei ist eine xml-typische, hierarchisch strukturierte Textdatei, in der Anweisungen für den Anwendungscode im Container stehen, wenn dieser zeitgesteuert aufgerufen und abgelaufen werden soll.

Listing 13: Inhalt der `batch_config.xml`-Datei

```
<?xml version="1.0" encoding="UTF-8"?>
<batch>
  <routines>
    <job name="Test1"> <!-- Insert Name for the job. Will be used for logging -->
      <script>Script1.py</script> <!--Python Script-->
      <cron_timing> 30 10 * * 3 </cron_timing> <!--Cron Timing.-->
      <environment>python3</environment> <!--Python exec -->
      <active>True</active> <!--True / False-->
    </job>
    <job name="Test2"> <!-- Insert Name for the job. Will be used for logging -->
      <script>Script2.py</script> <!--Python Script-->
      <cron_timing> */30 * * * * </cron_timing> <!--Cron Timing.-->
      <environment>python3</environment> <!--Python exec -->
      <active>True</active> <!--True / False-->
    </job>
  </routines>
</batch>
```

In der oben abgebildeten `batch_config.xml`-Datei wird unter „Test1“ die Programmdatei `Script1.py` mittels Python jeden Mittwoch um 10:30Uhr ausgeführt (30 = Minute, 10 = Stunde, 3 = Wochentag). Der zweite Cron-Job „Test2“ sorgt dafür, dass die Python-Datei `Script2.py` alle 30 Minuten ausgeführt wird.

Für den Container, der Programme als Cronjob startet, werden die zusätzlichen Python Dateien `reused`, `routines_manager` und `run_batch_routines` in dem Image gebraucht. In den Dateien wird beschrieben, wie der Cronjob aus der xml-Datei gelesen werden kann und wie mit den Protokolldateien umzugehen ist.

Die erstellten und erforderlichen Dateien werden in einen Ordner auf dem DDS gespeichert, aus dem dann der Container gebaut werden kann.

Dazu erleichtert die `docker-compose.yml`-Datei das starten von Containern, da zusätzliche Konfiguration und Parametrierung, die dem Container beim Erstellen aus dem Image

mitgegeben werden, darin definiert werden können. So muss nicht an den „*docker run <Name des Images>*“-Befehl, der das Image als Container startet, der Name des Containers oder der Port, über den der Container verfügbar sein soll, angehängt werden. Allerdings muss vorher das Docker-Compose Programm installiert werden, da es nicht direkt mit Docker mitgeliefert wird.

Listing 14: Inhalt der *docker-compose.yml*-Datei

```
version: '3.8'
services:
  service_name:
    build:
      context: .
    restart: always
    container_name: YYYYMMDD_docker_container_name
    network_mode: "host"
    volumes:
      - ./:/batch_routines
```

In der oben abgebildeten *docker-compose.yml*-Datei wird nach der Schemaversion der Abschnitt, in dem der Container gestartet werden soll, als Service benannt. Darin kann der Containername angegeben und die restart-policy, also die Neustart-Optionen bei Fehlern oder Ereignissen, die den Container stoppen, vorgegeben werden. Es kann auch ein Pfad angegeben werden, auf welchen der Container speichern kann. Durch das Verknüpfen von Pfaden vom Container auf das Host-System bleiben Dateien beim Update oder Neustart des Containers erhalten. Ohne solch symbolischen Link gehen alle generierten Daten des Containers verloren, da die Container als geschlossene Systeme arbeiten.

Mit dem Anfügen eines Projekt-Namens wird die Containerzusammenstellung mit dem Befehl "*docker-compose -p project_name up --build -d*" im Projektordner gestartet. Der Projektname erlaubt das Bündeln der einzelnen Services für ein gemeinsames Projekt und zeigt diese dann gruppiert mit dem Namen des Projekts als Stack an.

```
cc4e@containerport:~/TEC_Excess_slim$ sudo docker ps -l
CONTAINER ID   IMAGE                                COMMAND                                CREATED        STATUS        PORTS        NAMES
5755fa94ff5e   netzdienlichkeit_signal_excess      "python3 TEC_Excess..."            8 minutes ago  Up 8 minutes  20220127_
netzdienlichkeit_signal_excess
```

Abbildung 30: Screenshot des erstellten Containers

Die Abbildung 30: Screenshot des erstellten Containers zeigt den erstellten Container „20220127_netzdienlichkeit_signal_excess“, der den Zustand des Hausanschlusspunktes vom TEC berechnet und sendet.

5.4.3 Portainer

Die ganzen Befehle der letzten Kapitel lässt erkennen, dass Docker ein klassisches Kommandozeilentool ist. Bei der Auflistung aller laufenden Container wird der Befehl „*docker ps*“ verwendet, auch nicht laufende Container können mit dem Befehl „*docker ps -a*“ angezeigt

Implementierung

werden. Für die detaillierten Informationen über einen Container gibt es den Befehl „*docker inspect*“

Um einen leichteren und besseren Überblick der Container zu bekommen, wird am TEC das Softwaretool Portainer verwendet. Dieses nutzt eine graphische Oberfläche, um die Container zu verwalten. Portainer ist als Community Edition eine Open-Source-Software, die über eine Weboberfläche das Verwalten von Containern und Images ermöglicht. Dabei braucht die Software nicht installiert werden, sie kann direkt als eigener Docker-Container auf dem DDS gestartet werden [21].

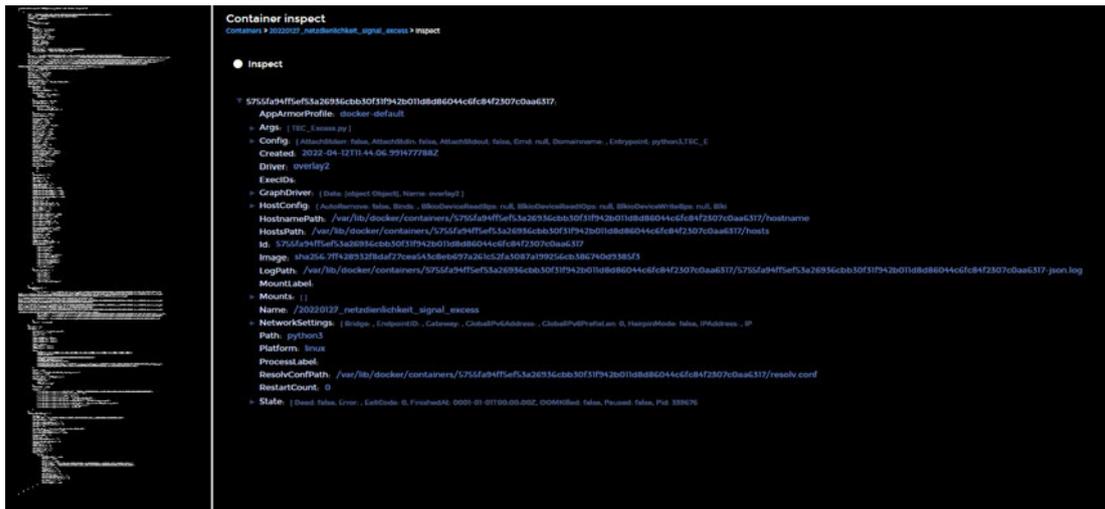


Abbildung 31: Screenshot von „docker inspect“ im Terminal und in Portainer

In dem oben abgebildeten Screenshot ist die Ausgabe des Befehls „*docker inspect*“ für den erstellten Container „TEC_excess“ aufgenommen. Auf der linken Seite ist die Ausgabe, so wie sie aus dem Terminal ausgegeben wird, zu sehen. Rechts die Ausgabe im Portainer, gruppiert und mit der Maus auswählbar und ausklappbar.

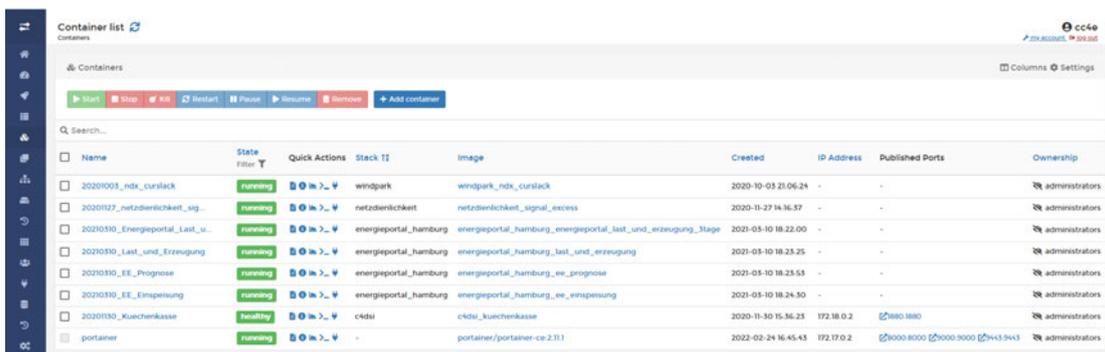


Abbildung 32: Weboberfläche Portainer

Implementierung

Durch die grafische Übersicht lassen sich die laufenden Container auch durch Mausklicks administrieren. Es können gezielt Container ausgewählt werden, um Statistiken wie CPU- und Arbeitsspeicherverbrauch zu erhalten, die Netzwerkauslastung zu betrachten oder sie neu zu starten.

6 Evaluierung

In Kapitel 4 wurden die Voraussetzungen geklärt, die eine Leitwarte benötigt, um das Ziel des Anwendungsfalls einer Elektrolyse mit biologischer Methanisierung für die Sektorkopplung am TEC umzusetzen. Im vorherigen Kapitel 5 wurden die nötigen Schritte beschrieben, um die neue Leitwarte am TEC aufzubauen. Im folgenden Kapitel werden zwei Anwendungsfälle zum netzdienlichen Betrieb simulativ analysiert. Hierzu werden im ersten Anwendungsfall die Stromdaten des Bezirks Bergedorf und im zweiten Anwendungsfall die Stromdaten vom TEC verwendet. Unter Berücksichtigung der gewonnenen Erkenntnisse aus der Simulation erfolgt der Kommunikationstest, in dem die Schaltung des Power-to-Gas Prozesses mittels der neuen Leitwarte durchgeführt wird. Abschließend werden die wichtigsten Erkenntnisse zusammengefasst.

6.1 Anwendungsfall Netzdienlichkeit im Bezirk Bergedorf

6.1.1 Beschreibung des Anwendungsfall Netzdienlichkeit im Bezirk Bergedorf

Der Anwendungsfall sieht vor, dass als Eingangsdaten die Daten des Energieportal Hamburgs „Strom erzeugt in Bergedorf“ und der „Last in Bergedorf“ verwendet werden. In der Simulation werden die Eingangsdaten zu einer Residuallast verrechnet. Mit Hilfe der Residuallast lassen sich die Intervalle bestimmen, wann in Bergedorf Stromüberschüsse vorhanden sind, um in diesen Intervallen den netzdienlichen Betrieb mit Hilfe der Elektrolyse zu aktivieren. Für die Elektrolyse gelten dabei folgenden Regeln, die in der Regelung berücksichtigt werden müssen:

1. Die Elektrolyse darf so lange angeschaltet bleiben bis keine Stromüberschüsse mehr vorliegen.
2. Der Elektrolyseur schaltet sich aus, wenn der Wasserstoffspeicher mit 15 bar voll beladen ist.
3. Wenn der Wasserstoffspeicher mit 15 bar beladen ist, darf die Elektrolyse erst wieder angeschaltet werden, wenn der Druck auf 10 bar abgefallen ist (eine Hystereseregulierung für oberes Band) und ein Stromüberschuss vorliegt.
4. Die Elektrolyse muss eine Mindestbeladung sicherstellen, diese wird aktiviert sobald der Speicherdruck bei 3 bar liegt. In diesem Fall wird die Elektrolyse angeschaltet und erst ausgeschaltet, wenn der Speicherdruck wieder bei 5 bar angelangt ist

(Hystereseregulung für unteres Band). Diese Speicherbeladung erfolgt auch in nicht netzdienlichen Betriebszeiten und soll sicherstellen, dass die Methanisierung im optimalen Zustand arbeiten kann.

Es werden drei exemplarische Jahre simuliert, die weitestgehend vollständig mit den Eingangsdaten vorliegen. In der Auswertung sollen betrachtet werden:

- Laufzeiten der Elektrolyse
- Produktionsmengen der Elektrolyse und Methanisierung
- Anteil der Laufzeiten der Elektrolyse am netzdienlichen Betrieb



Abbildung 33: Ablauf und Analyse der Simulation für den Bezirk Bergedorf

Für die Simulation wird Python als Programmiersprache verwendet, mit den Bibliotheken „NumPy“ für die Wertespeicherung und Auswertung, und „Matplotlib“ für die graphische Aufbereitung. Für die Auswertung des Netzzustands des Verteilungsnetzes vom Stadtbezirk Bergedorf wird zusätzlich die Programmbibliothek pandas (abgeleitet von panel data) verwendet.

6.1.2 Die Eingangsdaten für den Anwendungsfall Netzdienlichkeit im Bezirk Bergedorf

Um eine Analyse der Potentiale über einen langen Zeitraum zu betrachten, wird der Anwendungsfall zuerst anhand der historischen Netzzustände des Bezirks Bergedorf näher analysiert. Dabei werden verschiedene Zeiträume der Daten des Energieportals genutzt, um zunächst die Residuallast zu ermitteln und dann die Fahrweise der Elektrolyse zu simulieren und analysieren zu können.

Die Stromdaten des Energieportals sind in auf dem Datenbankserver in der Influx DB abgelegt. Die historischen Stromdaten, die auf der Influx-Datenbank gespeichert sind, werden abgerufen und lassen sich als Datenframe weiterverarbeiten. Aus den Rückgabewerten der Abfrage von Energieerzeugung und Energieverbrauch wird die Differenz gebildet und als Schaltvektor Residuallast gespeichert. Ist der Wert der Residuallast negativ, sind Stromüberschüsse durch die EE-Erzeugung vorhanden und in diesen Zeiten sollen die Stromverbrauchsanlagen angeschaltet werden, damit diese einen netzdienlichen Betrieb sicherstellen.

Für die allgemeine Datensichtung werden im nachfolgenden Diagramm die Jahres-Viertelstundenwerte der Residuallast der Jahre 2018-2021 mit dem Jahr 2012 verglichen, indem die einzelnen Jahresviertelstunden der Größe nach sortiert werde (vgl. Abbildung 34)

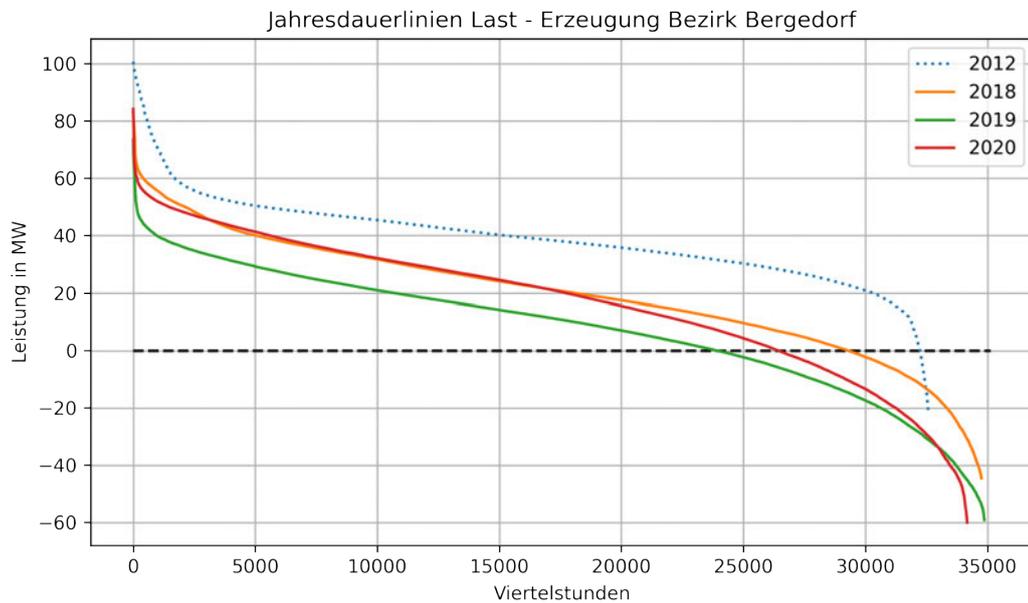


Abbildung 34: Jahresdauerlinien der Residuallast im Bezirk Bergedorf

Auffällig sind zwei markante Eigenschaften der Verläufe, die unterschiedlichen Längen und die verschiedenen Schnittpunkte mit der Nulllinie.

Die Lage von den Schnittpunkten der Graphen mit der schwarz gestrichelten Null-Achse ist für den betrachteten Anwendungsfall ein wichtiges Merkmal. Dabei stellen die positiven Werte den Netzbezug und die negativen Werte die Stromüberschüsse dar. In 2012 war die EE-Erzeugung im Bezirk noch sehr gering, weshalb die Stromüberschüsse sehr gering waren, und der Bezirk noch einen hohen Stromverbrauch aufweist. Die Jahre 2019 und 2020 weisen einen deutlich höheren Anteil an Stromüberschüssen auf. Im Jahr 2021 gehen die Stromüberschüsse zu den vorherigen Jahren zurück.

Die unterschiedlichen Längen kommen durch die nicht kontinuierlichen Werteaufzeichnungen auf dem Energieportal. Dieses lässt sich mit der teilweisen Nicht-Verfügbarkeit des Portals erklären, und mit dem Umstand dass keine Werte in das Portal nachgetragen werden. Dies führt teilweise zu längeren Datenlücken. Auch ein gewisser Anteil an Ausreißern wurde nachträglich gelöscht, welches auch zu einer Verkürzung der Jahresdauerlinien beiträgt. Hierbei wurde die Drei-Sigma Regel angewendet um die Ausreißer zu löschen.

Tabelle 9: Analyse der Jahresviertelstunde

	2012	2018	2019	2020	2021
Jahresviertelstunden	35.136	35.040	35.040	35.136	35.040
Werte vom Energieportal	33.625	35.035	35.035	34.188	34.370
Mittelwert in MW	39,605	19,357	7,350	17,110	22,795
Sigma in MW	20,379	21,288	22,197	25,967	21,514
Ausreißer nach oben	100,742	83,221	73,941	94,697	87,338
Ausreißer nach unten	-21,534	-44,507	-59,240	-59,980	-41,747
Gelöschte Ausreißer	1046	277	158	17	151
Positive Werte	32.218	28.952	23.450	26.154	28.522
Negative Werte	361	5.806	11.427	8.017	5.697
Summe	32.579	34.758	34.877	34.171	34.219

In der Tabelle 9 ist der Vergleich der negativen Jahreswerte ein Hinweis auf die Erträge aus den Erneuerbaren Energiequellen, da der Bezirk Bergedorf nur Energieerzeuger aus Wind- und Sonnenenergie hat. Die Jahre 2012 und 2020 sind Schaltjahre, weshalb hier ein Tag mehr vorhanden ist als zu den anderen Jahren und deswegen mehr Jahresviertelstunden im Jahr vorkommen müssten. Die meisten Viertelstundenwerte für ein Jahr weisen die Jahre 2018 und 2019 auf (fünf Viertelstunden, die fehlen). In dem Jahr 2020 fehlen 948 und 2021 fehlen 670 Viertelstunden. Die meisten Werte fehlen für das Jahr 2012 mit 1.511. Nach der Anwendung der Sigma-3-Regel zeigt sich, dass vor allem das Jahr 2012 die größte Anzahl an Ausreißer aufweist, hingegen hat das Jahr 2020 die Geringsten. Die restlichen Jahre liegen ungefähr auf demselben Niveau in Bezug zur Anzahl der Ausreißer. Insgesamt zeigt sich, dass die Anzahl der Werte für die Jahre 2018 – 2021 dicht beieinander liegen und deswegen auch für die Simulation genutzt werden können.

Für die Validierung des Netzzustandsvektor werden die ersten zehn Tage des Jahres 2021 mit den Daten des Energie-Portals verglichen. Der Netzzustandsvektor ist dabei die negierte Residuallast (positive Werte = Stromüberschüsse, negative Werte = Stromverbrauch), da positive Werte so auch den ‚True‘-Werten für die Freigabe zur Netzdienlichkeit erfüllen und negative Werte dem Wert ‚False‘.

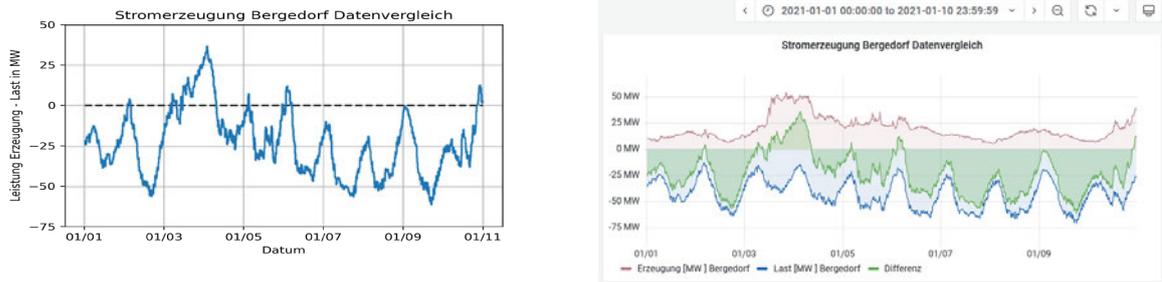


Abbildung 35: Vergleich der Daten aus der Datenbank mit dem Energieportal-Hamburg

Auf der y-Achse wird die Leistung in Megawatt dargestellt. Der Power-to-Gas Prozess ist bei der Betrachtung der Größen als Kleinlast anzusehen, da der Verbrauch des Prozesses im niedrigen Kilowattbereich stattfindet (vgl. Kapitel 4.1). Daher ist die Größe des Stromüberschusses nicht relevant, und es kann mit booleschen Werten der Netzzustand simuliert werden.

Nachfolgende Grafiken zeigen den Netzzustandsvektor für die ersten zehn Tage des Jahres 2018, eine ‚1‘ entspricht einer Viertelstunde mit Stromüberschuss, eine ‚0‘ gibt an, dass der Einsatz der Elektrolyse nicht netzdienlich ist, da mehr Last im Netz ist als lokal erzeugt werden kann.

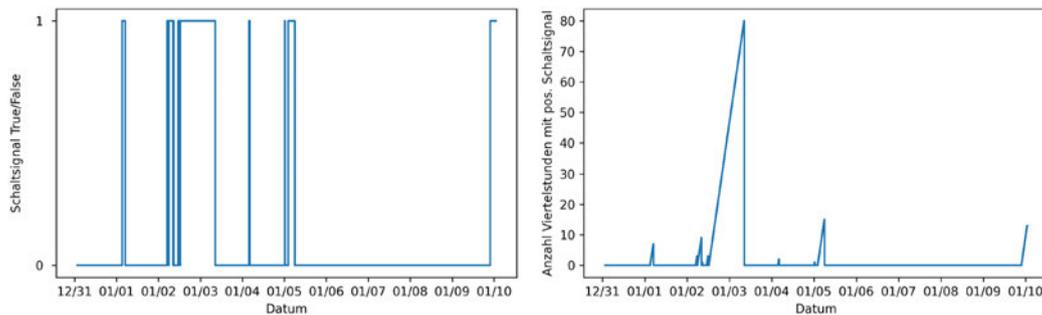


Abbildung 36: Netzzustandsvektor und Dauer der positiven Schaltzustände

Das linke Schaubild in Abbildung 36 zeigt den Schaltvektor „Stromüberschuss“, woraus sich der Zeitraum der Stromüberschüsse erkennen lässt. Das rechte Schaubild wertet die Dauer als Anzahl von Viertelstunden der Stromüberschüsse aus. Hierbei werden die Viertelstunden in den Zeiten der Stromüberschüsse aufsummiert. Aus der Grafik geht hervor, dass ab der Nacht zum 3. Januar 2021 ganze 80 Viertelstunden bzw. 20 Stunden lang ein Stromüberschuss im Netz des Bezirks Bergedorf auftrat.

Um einen Überblick über die Dauer des Stromüberschusses zu erhalten, wird das Netzdienlichkeitssignal über die Auswertungsjahre dargestellt.

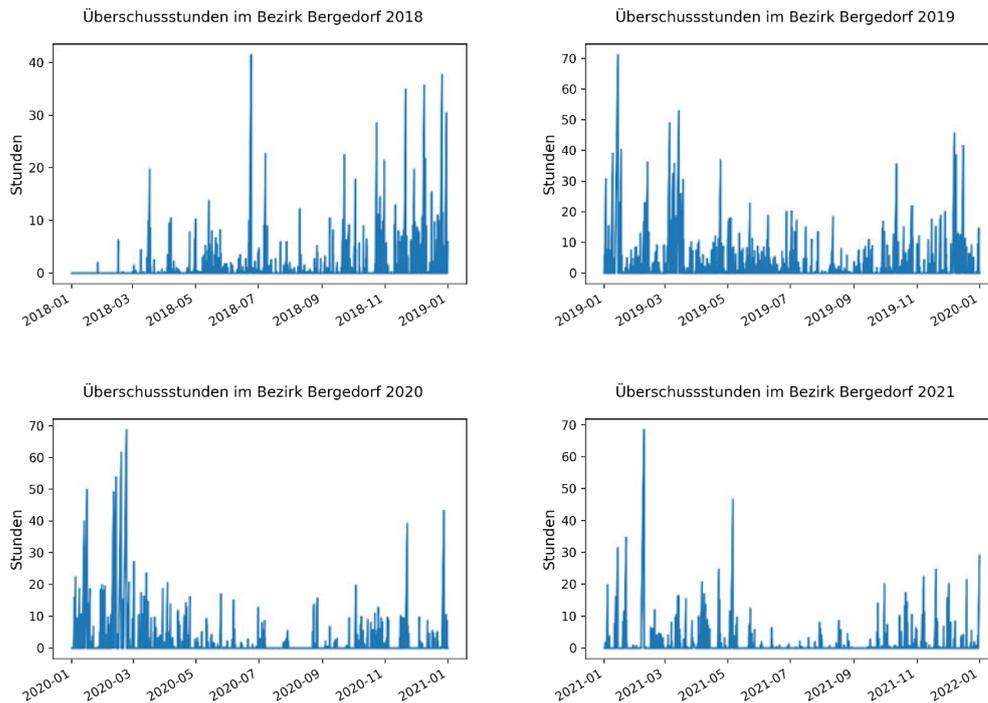


Abbildung 37: Überschussstunden im Bezirk Bergedorf von 2019-2021

Den Diagrammen in Abbildung 37 ist zu entnehmen, dass es besonders in den windreichen Monaten von November bis April viele Intervalle gibt, in denen über Tage hinweg Stromüberschuss vorhanden sind, und eine Verlagerung bzw. Flexibilisierung von leistungsintensiven Prozessen in diese Bereiche für das Verteilnetz in Bergedorf netzdienlich ist. Insbesondere die Jahre zwischen 2019 bis 2021 zeigen Intervallspitzen von bis zu 70 Stunden auf, dagegen lag in 2018 nur eine Intervallspitze bei ca. 40 Stunden.

6.1.3 Umsetzung der Regelung der Elektrolyse mit Wasserstoffspeicher

Neben den aufbereiteten Eingangsdaten für den Anwendungsfall stellt die Regelung der Elektrolyse mit dem Wasserstoffspeicher den nächsten wichtigen Baustein da. Die einzelnen Simulationsschritte werden durch die Variablen Speicherbeladung und Speicherentnahme bestimmt. Da die Residuallast für den Bezirk Bergedorf alle 15 Minuten einen neuen Wert hat, wurde dieses Intervall auch als Simulationszeitschritt gewählt. Die Speicherbeladung erfolgt konstant mit 1 m^3 , was auch der realen Produktion der Elektrolyse am TEC entspricht (vgl. Abschnitt 4.1). Die Speicherentnahme entspricht dem Mischverhältnis aus Wasserstoff und Kohlendioxid, die zur Fütterung der Archaeen benötigt werden und ist abhängig von der der Fahrweise bzw. Fütterung der Urbakterien. Der normale Betrieb der Methanisierung benötigt einen Wasserstoffbedarf von $0,01875 \text{ m}^3$ pro Viertelstunde.

Weitere Eingangsgrößen sind die Reservebänder, die eine vollständige Entnahme des Speichers oder auch das ständige Ein- und Ausschalten der Elektrolyse verhindern sollen.

Listing 15: Initialisierung der Simulation des Power-to-Gas-Prozesses

```
# obere und untere Grenzen sowie Reservebänder
SPEICHERDRUCK_MAX = 15.0      # Bis wann maximal geladen wird
SPEICHERDRUCK_MIN = 3.0      # Ab wann auf jeden Fall geladen wird

SPEICHERDRUCK_MIN_RS = 5.0    # Bis wann auf jeden Fall geladen wird
SPEICHERDRUCK_MAX_RS = 10.0  # Ab wann wieder geladen werden kann

# Initialisierung des Wasserstoffspeichers
speicherdruck = 15.0         # Speicher ist maximal aufgefüllt

# Konstanten der Methanisierung und der Elektrolyse
SPEICHERBELADUNG = 1.0      # pro viertel Stunde wird der Speicher geladen

SPEICHERENTNAHME = 0.01875  # pro Viertelstunde 0.075m³ pro Stunde, 1,8m³ pro Tag
#SPEICHERENTNAHME = 0.075

# Initialisierung der variablen Zustände
speicherdruck_aktuell = speicherdruck
druckbetankung = 0
netzzustand = 0
status_elektrolyse = 0
status_elektrolyse_change = 0
```

Die Werte *speicherdruck_aktuell*, *netzzustand*, *status_elektrolyse*, *status_elektrolyse_change*, *druckbetankung* und *druckbetankung_change* werden bei jedem Simulationsschritt als Rückgabewerte gespeichert und sollen dazu beitragen, die Simulationen zu bewerten.

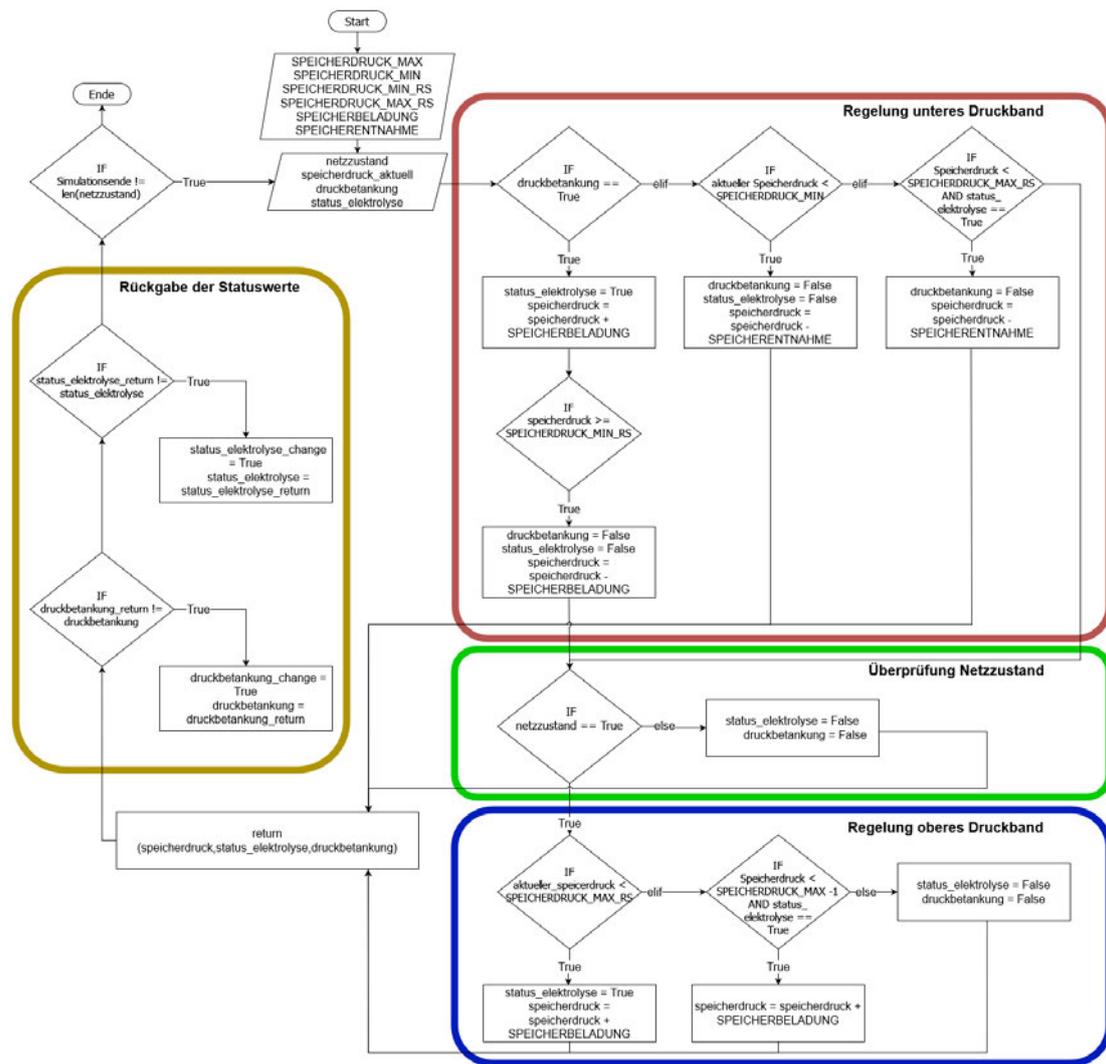


Abbildung 38: Ablauf der Regelung der Elektrolyse mit dem Wasserstoffspeicher

Mit dem Programmstart werden zunächst alle wichtigen Parameter initialisiert. Nach der Initialisierung werden die Statuswerte abgefragt und der Netzzustand überprüft. Wenn ein Stromüberschuss vorliegt, führt dieses zum Anschalten der Elektrolyse. Im nächsten Schritt erfolgt die Überprüfung der Hystereseregulierung.

In dem Programm werden in jedem Simulationsschritt zwei Bedingungen abgefragt, die Regelung für das untere Druckband (roter Bereich) und dem Netzzustand (blauer Bereich).

Für die Regelung des unteren Druckbandes gibt es drei Anweisungen:

- Der Speicher ist in der Druckbetankung und wird bis zum unteren Reserveband von 5 bar geladen

- Die Elektrolyse bleibt angeschaltet
- Der Speicher ist in der Druckbetankung und hat das Reserveband mit 5 bar erreicht
 - Die Elektrolyse wird ausgeschaltet
- Der Speicher ist gerade in den minimalen Druckbereich auf 3 bar gefalle, und die Druckbetankung wird eingeschaltet.
 - Die Elektrolyse schaltet sich an

Für die Regelung des oberen Druckbandes gibt es folgende Anweisungen:

- Es gibt es positives Netzsignal und die Elektrolyse ist unter dem oberen Druckband von 10 bar
 - Die Elektrolyse schaltet sich an
- Es gibt es positives Netzsignal und der maximale Speicherdruck von 15 bar ist erreicht
 - Die Elektrolyse schaltet sich aus.

Nach dem Durchlaufen dieser Bedingungen wird verglichen, ob sich der Zustand der Elektrolyse geändert hat und ob es zur Druckerhaltung im unteren Reservebereich kam (gelber Bereich). Diese Wertänderungen tragen zur Auswertung bei. Danach wird der nächste Simulationsschritt ausgeführt.

6.1.4 Auswertung des Anwendungsfall Netzdienlichkeit im Bezirk Bergedorf

Die erste Simulation wurde mit dem Netzzustand des Bezirks Bergedorf von 2018 gestartet. Auffällig an dem Netzzustand von 2018 ist die niedrige Anzahl der Stromüberschussstunden am Anfang des Jahres, sodass gerade am Anfang des Simulationszeitraums davon ausgegangen werden kann, dass es zu einer Minimalbeladung des Wasserstoffspeichers kommt. Zudem wurde der Wasserstoffspeicher als voll initialisiert.

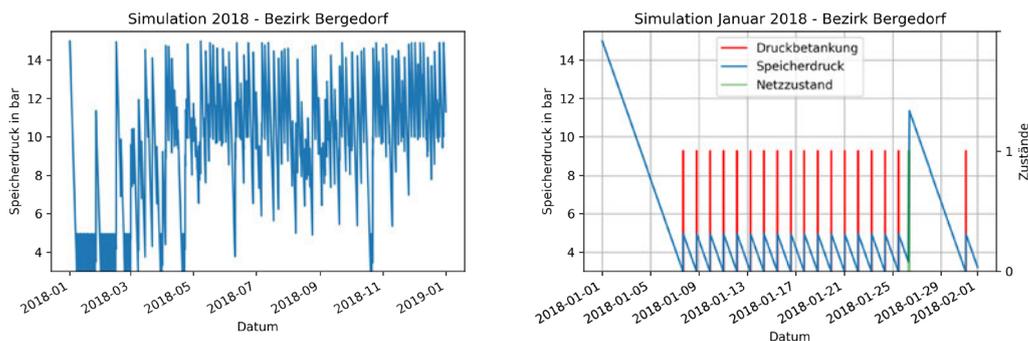


Abbildung 39: Verlauf der Jahressimulation mit einem Ausschnitt vom Januar 2018

In Abbildung 39 wird links der simulierte Speicherdruckverlauf über das Jahr 2018 und rechts der detaillierte Ausschnitt für den Januar 2018 abgebildet. Rechts zeigt sich die gehäufte Regelung zur Minimaldruckerhaltung, die am Anfang des Jahres auftritt. Bei Unterschreiten der

Speichergrenze von 3 bar (*Speicherdruck_min*) startet die Elektrolyse und belädt den Wasserstoffspeicher bis zum unteren Reserveband von 5 bar (*Speicherdruck_min_rs*).

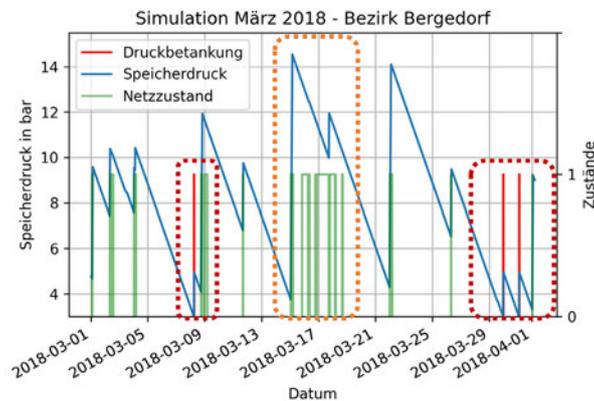


Abbildung 40: Ausschnitt des Verlaufs der Simulation vom März 2018

In der Abbildung 40 ist der Zeitraum vom 01.03.2018 bis zum 01.04.2018 abgebildet. In dem Zeitraum lässt sich zeigen, dass die implementierte Hystereseregulierung funktioniert. Die Sicherstellung der Minimalbeladung zeigen die roten gestrichelten Rechtecke in der Abbildung. Hier schwankt die Beladung von 3 – 5 bar. Die Hystereseregulierung für das obere Band ist im orangenen gezeichneten Rechteck abgebildet. Hier ist zu sehen das die netzdienliche Beladung bis 15 bar geladen wird. Dann bleibt die Elektrolyse so lang ausgeschaltet bis die 10 bar unterschritten werden und erst als das Signal für Netzdienlichkeit (grün) aktiv gestellt wird (Wert = 1), erfolgt eine Beladung der Speicher indem die Elektrolyse angeschaltet wird. Die Elektrolyse bleibt so lange an, wie auch das Signal der Netzdienlichkeit aktiv ist.

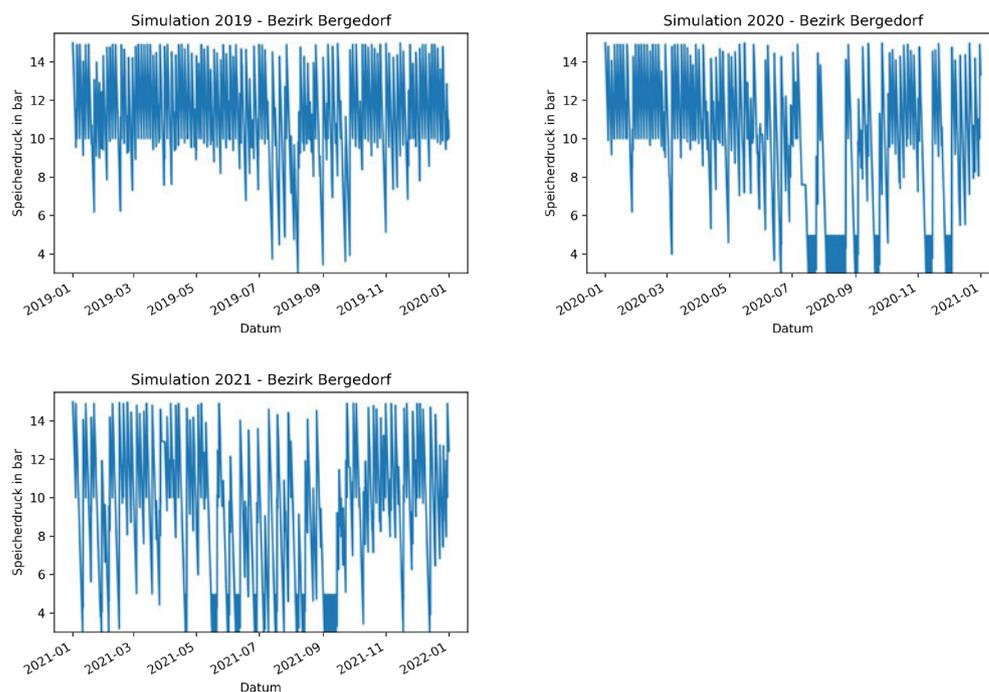


Abbildung 41: Verlauf der Speicherdrücke der Simulationszeiträume 2019 - 2021

Die Diagramme in Abbildung 41 zeigen die Simulationsergebnisse über die Überschusstunden im Bezirk Bergedorf (vgl. Abbildung 37: Überschusstunden im Bezirk Bergedorf von 2019-2021). Auffällig ist, dass im Jahr 2019 (links oben) keine Minimaldruckbetankung aktiviert wurde. Über das Jahr gesehen schwankt der Druck meist zwischen 10 und 15 bar, was der Regelung für das obere Hystereseband entspricht. In allen drei Jahren ist zu erkennen, dass der Druckzustand in den Sommermonaten abfällt. Speziell in den Jahren 2020 und 2021 wird die Hystereseregelung des unteren Bandes aktiviert.

In der folgenden Tabelle 10 werden für die Jahresauswertung die einzelnen Verläufe wie folgt ausgewertet: die Laufzeit der Elektrolyse in Stunden, die Anzahl der Starts der Elektrolyse und Starts für die Mindestdruckerhaltung.

Tabelle 10: Simulationsergebnisse über den Netzzustand des Bezirk Bergedorfs

	2018	2019	2020	2021
Jahresviertelstunden:	35.040	35.040	35.136	35.040
Simulationswerte:	34.757	34.876	34.170	34.218
Simulationswerte in Stunden:	8.689,25	8.719,0	8542,5	8.554, 5
Laufzeit Elektrolyse in Stunden:	162,0	162,25	159,75	159,75

Starts der Elektrolyse:	189	154	166	160
Starts Elektrolyse zur Druckbetankung:	45	1	44	41
Differenz Speicherdruck vorher/nachher:	3,7	4,9	1,7	2,6
Produzierter Wasserstoff in m ³ :	648,0	649,0	639,0	639,0
Verbrauchter Wasserstoff in m ³ :	648,0	649,0	639,0	639,0

In Tabelle 10 sind die wesentlichen Werte der Simulation ausgewertet. Die Laufzeit der Elektrolyse beträgt durchschnittlich 161 Stunden. Die meisten Starts der Elektrolyse liegen im Jahr 2018 vor, dieses lässt sich vor allem durch den Simulationsbeginn ableiten, da hier die Mindestdruckbetankung aktiviert ist. 2019 sind die geringsten Starts, da hier nur eine einzige Mindestdruckbetankung vorliegt und somit die Aktivierung fast nur durch den netzdienlichen Betrieb aktiviert wurde. Die letzten beiden Jahre weisen ein ähnliches Niveau auf. Der produzierte Wasserstoff wurde mit dem verbrauchten Wasserstoff verglichen. Hierbei zeigt sich in allen vier Jahren eine Übereinstimmung.

6.2 Anwendungsfall Netzdienlichkeit am TEC

6.2.1 Beschreibung des Anwendungsfall am TEC

Der Anwendungsfall wird unter dem Aspekt der Netzdienlichkeit am TEC betrachtet. Als Auslöser wird dabei der Zustand des Hausanschlusspunktes genommen, um die Elektrolyse im Niederspannungsnetz des TECs systemdienlich zu fahren. Die Stromerzeugung am TEC erfolgt durch die Aggregate BHKW und PV. Alle anderen Aggregate sind Stromverbrauchsanlagen. Auf dieser Basis wird die Residuallast gebildet und das Schaltsignal für den Netzdienlichen Betrieb ermittelt. Für die Simulation wird der Zeitraum vom 01.01.2021 – 31.12.2021 verwendet.

Die Regelung für die Elektrolyse und Methanisierung wird aus Abschnitt 6.1.3 übernommen.

6.2.2 Die Eingangsdaten für den Anwendungsfall Netzdienlichkeit am TEC

Die Jahresbetrachtung des Hausanschlusses vom TEC zeigt, dass die Energieerzeugung stark variiert, besonders im Sommer ist ein starker Rückgang der Überschussstunden zu erkennen. Das kommt daher, dass die meiste Energie durch das Blockheizkraftwerk gewonnen, und der PV-Anteil meist direkt mit der Grundlast verbraucht wird. Das BHKW hat eine elektrische Leistung von 16kW. Wenn das BHKW in Betrieb ist, läuft dieses meist über mehrere Stunden am Stück (vgl. Abbildung 42) und versorgt das TEC mit Wärme. Die Erwärmung des

Trinkwassers ist davon ausgenommen. Aus diesem Grund läuft das BHKW auch nur im Winter und in der Übergangszeit.

Da am TEC, anders als im Anwendungsfall mit den Daten vom Bezirk Bergedorf, die Leistung berücksichtigt werden muss, wurden nur die Stromüberschüsse in den Viertelstunden des Netzdienstlichkeitsvektors bestimmt. Daraufhin wurde die Stromüberschüsse aussortiert die geringer sind als 10kW, was der Höhe der Leistung entspricht, den der Elektrolyseur bei der Wasserstoffgeneration benötigt. Mit diesem Netzdienstlichkeitsvektor wird die Simulation fortgeführt.

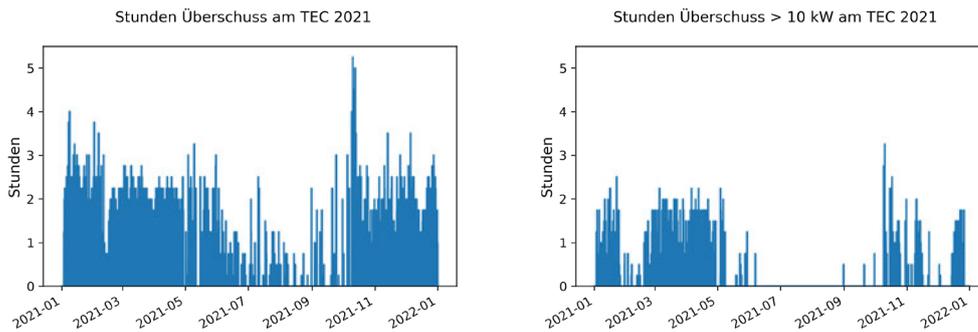


Abbildung 42: Überschussstunden am TEC

In Abbildung 42 sind die Überschussstunden über das Jahr 2021 abgebildet. Während auf der linken Seite alle Stunden mit positiver Leistung abgebildet sind, werden im rechten Diagramm ausschließlich die positiven Leistungsstunden größer als 10 kW abgebildet. Bei den Intervallen fällt auf, dass die Spitze bei 3 Stunden und in den anderen Zeiten bei ca. 2 Stunden liegt. Zudem ist auch das Sommerloch gut zu erkennen. Ein wichtiger Aspekt für die Stromverbrauchsseite ist, dass das TEC mit einer Direct Air Capture-Anlage, die eine Spitzenleistung von 15 kW besitzt, erweitert wurde. Im Herbst 2021 ist diese in Betrieb genommen worden und ist im Lastverlauf zu erkennen.

6.2.3 Auswertung des Anwendungsfall Netzdienstlichkeit am TEC

Die zuvor beschriebenen Eingangsdaten werden für die Simulation verwendet und dienen als Grundlage für die Analyse des Betriebes.

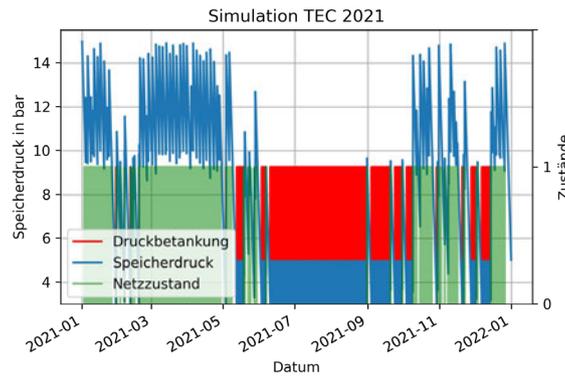


Abbildung 43: Vergleich des Speicherdrucks mit dem Netzzustand vom TEC 2021

Die simulierte Fahrweise des Elektrolyseurs über das Jahr 2021 wird in Abbildung 43 wiedergegeben. Wie zu erwarten, kommt es in den Sommermonaten, wenn das BHKW nicht läuft, zu keinem netzdienlichen Betrieb, und der Wasserstoffvorrat im Speicher kann nur durch die Minimaldruckerhaltung sichergestellt werden. Zudem wird auch im Herbst wie erwartet die Minimaldruckerhaltung aktiviert, da in diesem Zeitraum die neue Direct Air Capture-Anlage in Betrieb gegangen ist. In den Wintermonaten, wo das BHKW oft im Einsatz ist, ist die Hystereseregulierung für das obere Band zu erkennen.

Da die häufigen Starts in den Sommermonaten nicht effizient für den Betrieb der Elektrolyse sind, wurde eine weitere Simulation durchgeführt in dem die Regelung der Hystereseregulierung für das untere Band ausgeschaltet ist. Das bedeutet, dass die Elektrolyse immer eine Vollbeladung des Speichers vorsieht (vgl. Abbildung 44).

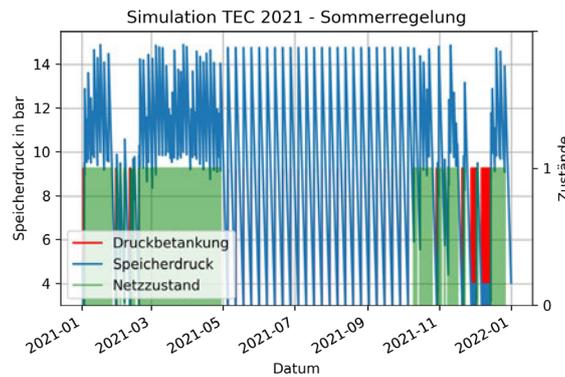


Abbildung 44: Simulierter Speicherdruck im 2021 mit Sommerregelung

Das verlängerte Laden bis zum Maximaldruck des Speichers hat in den Sommermonaten, in der das BHKW nicht lief, die Starts der Elektrolyse deutlich gesenkt.

Tabelle 11: Simulationsergebnisse über den Hausanschluss des TEC

	<i>2021</i>	<i>2021 mit Sommerregelung</i>
Simulationswerte in Stunden:	8.760	8.760
Laufzeit Elektrolyse in Stunden:	161,75	164,0
Starts der Elektrolyse:	225	143
Starts Elektrolyse zur Druckbetankung:	129	57
Produzierter Wasserstoff m ³ :	647,0	656,0
Verbrauchter Wasserstoff m ³ :	647,0	656,0

Aus Tabelle 11 kann entnommen werden, dass bei fast gleicher Laufzeit der Elektrolyse die Anzahl der Starts der Elektrolyse nur noch 63,5% und die Starts zur Druckbetankung nur noch 44% gegenüber der Fahrweise ohne Sommerregelung betragen.

6.3 Kommunikationstest „Netzdienlichkeit am TEC“

6.3.1 Beschreibung des Kommunikationstests „Netzdienlichkeit am TEC“

Damit das TEC den Power-to-Gas Prozess netzdienlich fahren kann, wurde das Simulationsprogramm aus dem Abschnitt 6.2 verwendet. Das Skript wird wie im Ablaufdiagramm in Abbildung 16 ausgeführt.

Bei dem Eingangssignal handelt es sich um die Differenz der Erzeugung und dem Verbrauch des TEC. Dieses Netzdienlichkeitssignal wird durch ein Berechnungsskript erstellt, das dauerhaft auf dem Development & Deployment-Server (DDS) implementiert und als Service minutenweise abgerufen wird. Das Signal wird direkt aus dem Berechnungsskript per MQTT an den Leitwartenrechner verschickt. Das Programm, welches die Elektrolyse steuert, nutzt dieselben Regelbänder des Wasserstoffspeichers wie sie in den Simulationen der Anwendungsfälle definiert wurden. Hierzu verschickt der Leitwartenrechner die notwendigen Anlagendaten, wie aktueller Speicherdruck und aktuellen Status an den DDS über MQTT

Auf dieser Weise kann der Leitwartenrechner je nach vorhandenem Speicherdruck und Netzzustand in die Fahrweise der Elektrolyse eingreifen und per MQTT Start- oder Stopp-Signale an die Elektrolyse senden. Hierbei kann die Elektrolyse nur zwei Leistungswerte im Betrieb annehmen. Diese sind entweder 10 kW bei eingeschaltetem oder 0 kW bei ausgeschaltetem Zustand. Zusätzlich werden alle Statusdaten vom Leitwartenrechner an den Datenbankserver

via MQTT verschickt, um dort in der InfluxDB abgespeichert zu werden. Diese können über das Tool Grafana abgerufen und visualisiert werden.

6.3.2 Die Eingangsdaten für den Kommunikationstest

Das Programm soll zu der netzdienlichen Fahrweise des Elektrolyseurs dienen, aber auch die Zuverlässigkeit der Datenkommunikation und das Absetzen der Schaltbefehle aufzeigen. Dieses Programm wird im DDS hinterlegt und ausgeführt.

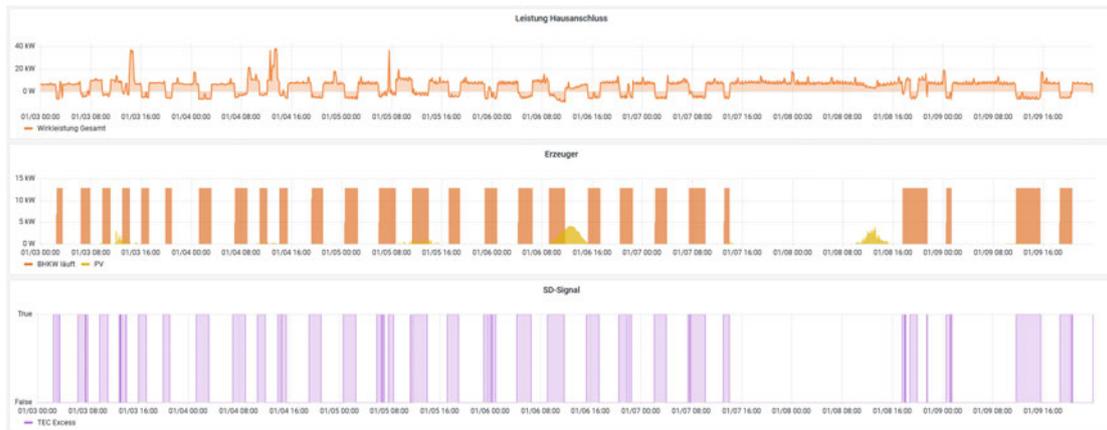


Abbildung 45: Leistungsverlauf und Netzdienstlichkeitssignal der KW 13, 2022

In dem oben abgebildeten Dashboard zeigt der obere Graph den Verlauf des Netzanschlusses an. Im mittleren Chart sind die Erzeugungsverläufe des BHKW und der PV am TEC dargestellt. Der dritte Abschnitt bildet die Zeiträume der Netzeinspeisung als Boolean am TEC ab. Nach diesem Signal soll die Schaltentscheidung des Programms erfolgen.

Geführt von diesem Signal und in Abhängigkeit vom aktuellen Druck des Wasserstoffspeichers, werden über den MQTT-Nutzer „Netzdiener“ Start- und Stopfbefehle über die Topics der Elektrolyse verschickt. Um häufiges Schalten der Elektrolyse zu verhindern, wird das Programm nach jeder Schaltentscheidung 15 Minuten warten, bevor es wieder in die Fahrweise des Elektrolyseurs eingreift.

6.3.3 Durchführung des Kommunikationstests „Netzdienstlichkeit am TEC“

Im realen Betrieb der Elektrolyse am TEC, wird beim Starten ein Drucktest automatisch durchgeführt (rotes Rechteck in der Abbildung 46). Dieser Drucktest hat eine Leistung von ca. 9 kW und dauert ca. 10 Minuten. Dieser Test dient zur Sicherstellung, dass das Wasserstoffsystem mit maximalen Betriebsdruck arbeiten kann.

Hinweis zur allgemeinen Problematik beim Drucktest der Elektrolyse am TEC:

Im Rahmen dieser Arbeit wurde die Elektrolyse der jährlichen Wartung unterzogen. Seit dieser Wartung ist ein Problem beim Einschalten des Elektrolyseurs aufgetreten. Dieses betrifft das standardmäßige Durchlaufen des Drucktestes der Elektrolyse. Es wird immer beim ersten Start des Elektrolyseurs, wenn diese zuvor mehrere Stunden ausgeschaltet war, der Drucktest abgebrochen, sodass sich der Elektrolyseur automatisch abschaltet. Hierbei tritt die Fehlermeldung „*pressure test – fill fail*“ auf. Damit der Elektrolyseur wieder genutzt werden kann, muss dieser zunächst erst manuell am TEC entriegelt werden. Startet man den Elektrolyseur ein zweites Mal erscheint die Fehlermeldung nicht mehr und der Drucktest wird erfolgreich durchgeführt. Hierzu müssen aber beide Starts zeitnah erfolgen. Erst dann kann die neue Leitwarte den Kommunikationstest durchführen.

Da die Fehleranalyse recht lange gedauert hat und der Fehler nicht richtig behoben werden konnte wird der Kommunikationstest für die Netzdienlichkeit am TEC nur exemplarisch für einen bestimmten historischen Zeitraum umgesetzt. Hierbei soll auch nur deutlich gemacht werden, dass die grundsätzliche Kommunikation mit seiner eingeschränkten Regelung funktioniert.



Abbildung 46: Grafana-Screenshot Einschaltverhalten Elektrolyseur

Realer Betrieb mit den Statuswerten der Elektrolyse und der Problematik des Drucktests:

In Abbildung 46 ist der Verlauf eines Einschaltvorgang des Elektrolyseurs zu sehen.

Der Elektrolyseur bekommt den Startbefehl über das MQTT-Topic „io/cybus/energie-campus/iso/elektrolyse/start-stop“ mit dem Wert „true“ (Abschnitt 1 in Abbildung 46). Daraufhin ändert sich der Status der Elektrolyse von 103 („Idle“) auf 303 („Start Up“). Danach geht die Anlage in Störung und auf dem Display des Elektrolyseurs erscheint die Fehlermeldung „Alarm – pressure test – fill fail“ (rote Markierung in Abschnitt 2). Diese Fehlermeldung muss an der Anlage quittiert werden.

Sobald die Anlage dann wieder einen Startbefehl bekommt, geht sie wieder in den Status 303 und von dort in den Status 501 („Pressure Test“). Nach dem durchlaufenen Drucktest wechselt die Anlage in den Status 601 („Generation“), ab da wird Wasserstoff produziert und der Druck im Wasserstoffspeicher steigt an (Abschnitt 3 in der Abbildung). Beim Senden des Wertes „false“ über „io/cybus/energie-campus/iso/elektrolyse/start-stop“ wechselt der Elektrolyseur in den Status 701 („Shut down“) und erreicht danach wieder den Idle-Status 103.

Durchführung des Kommunikationstests für die Netzdienlichkeit am TEC

Für die Netzdienlichkeit am TEC wurden zwei exemplarische Tage ausgewählt, die einen netzdienlichen Betrieb vorsehen. Diese sind

- am 12. April 2022 von 10:55 – 11:30 Uhr
- und am 14. April von 15:35 – 16:15 Uhr.

Aufgrund der Problematik des Drucktests wurde die Hystereseregulung für die Reservebänder ausgeschaltet.



Abbildung 47: Ausschaltvorgänge Elektrolyse

In Abbildung 47 ist für beide Zeiträume zu sehen, dass zunächst der Drucktest ohne Probleme durchgeführt werden konnte und danach die Elektrolyse eingeschaltet wird. Dieses lässt sich an den Statuswerten der Elektrolyse ablesen. Mit der Einschaltung der Elektrolyse steigt auch der Druck des Wasserstoffspeichers von 3 auf 5 bar an. Der Ausschaltvorgang der Elektrolyse erfolgt nach dem Ablauf des Zeitraums. Hierzu schickt der Leitwartenrechner ein „False“ über das Start-/Stop Signal. Dabei ändert sich der Status der Elektrolyse auf 701 („Shut down“) und beendet die Wasserstoffproduktion.

Zudem lässt sich auch zeigen, dass durch die erstellte Abbildung mit Grafana die Daten aus der Influx Datenbank abgerufen worden sind.

Trotz des eingeschränkten Kommunikationstests konnte aufgezeigt werden, dass das Zusammenspiel der drei Server einwandfrei funktioniert. Somit kann auch zukünftig die Vorgehensweise genutzt werden, um auch andere Anlagen am Campus zu steuern.

6.4 Zusammenfassung der Ergebnisse der Simulation

Mit Hilfe der Anwendungsfälle konnte die implementierte Regelung getestet werden. Die Anwendungsfälle zeigten, wie sich der Betrieb der Elektrolyse und damit verbunden die Beladung des Wasserstoffspeichers verhält.

Durch die Anwendungsfälle wurden die Zeiten, in denen es aus betriebstechnischen Gründen keinen Sinn macht, in die Fahrweise der Elektrolyse einzugreifen, aufgezeigt. Es wurde deutlich, dass in den Sommermonaten, in dem zum einen die Winderträge gering sind, und zum anderen das BHKW am TEC ausgeschaltet bleibt, der Speicher nicht sehr häufig vollgeladen ist, und es zu gehäuften Noteinschaltungen für die Druckerhaltung im Speicher kommt.

Bezogen auf das TEC ist es sogar ratsam, die Flexibilisierung der Fahrweise komplett umzustellen, da es durch die Zeiten, in denen das BHKW nicht läuft, es zu keinem größeren Stromüberangebot kommt und es ausschließlich zur Noteinschaltung kommt. Bei der Noteinschaltung sollte dann der Prozess der Wasserstoffgeneration solange laufen, bis die Speicher komplett voll sind. Das bündelt die Laufzeiten und die Schaltvorgänge des Elektrolyseurs werden minimiert.

Zudem konnte aufgezeigt werden, dass Programmteile aus den Anwendungsfällen erfolgreich für den Kommunikationstest und den damit verbundenen realen Betrieb erfolgreich übertragen werden kann. Künftig ist es nun möglich Simulation mit den realen Anlagen des TEC zu koppeln.

7 Fazit & Ausblick

Mithilfe der in dieser Arbeit modular aufgebauten Leitwarte lassen sich die Anlagen am TEC mit ihren unterschiedlichen Anlagenprotokollen regeln. Für den modularen Aufbau wurden drei Server - Leitwartenrechner, Datenbankserver und Development & Deployment-Server – eingerichtet und getestet.

Die Anlagen werden übergeordnet über das MQTT Protokoll angesprochen, um die Bedienung der Anlagen zu vereinheitlichen. Über den Leitwartenrechner können ausgewählte Werte an verschiedene Nutzer bereitgestellt werden und Nutzer können spezielle Wert über die Leitwarte an die Anlagen verschicken, um diese zu steuern.

Die aktuellen Zustandswerte der Anlagen werden intervall- und auch eventbasiert in die zeitreihenbasierte Influx-Datenbank gespeichert und archiviert. Über das Visualisierungstool Grafana können die Werte von der Datenbank abgerufen und angezeigt werden.

Der Development & Deployment-Server wurde zur Entwicklung von übergeordneten Regelungsprogrammen eingerichtet. Hierbei wurde ein Docker-Server aufgesetzt, um Regelungsprogramme zu entwickeln und zu testen. In der geschlossenen Umgebung können dort die Programme kontinuierlich oder in vordefinierten Routinen gestartet werden. Dieses wurde zunächst mit Hilfe der Anwendungsfälle getestet. Der erste Anwendungsfall simulierte den netzdienlichen Betrieb mit den Stromdaten vom Bezirk Bergedorf. Hierbei konnte der Zeitraum von 2018 bis 2021 näher analysiert und die Regelung getestet werden. Hierbei lässt sich feststellen, dass die EE-Erzeugung des Bezirks Bergedorf größtenteils aus Windenergieanlagen stammt. Insbesondere in den simulierten Jahren sind hohe Laufzeiten von Stromüberschüssen vorhanden, die eine netzdienlichen P2G Betrieb erlauben.

Der zweite Anwendungsfall mit den Stromdaten vom TEC zeigten, dass die Einsatzzeiten für einen netzdienlichen Betrieb über das BHKW bestimmt werden. Das BHKW wird nur für die Wärmeversorgung genutzt und nicht für Trinkwassererwärmung. Aus diesem Grund steht der netzdienliche Betrieb meist nur in den Winter- und Übergangsmonaten zur Verfügung. Eingeschränkt wurde der netzdienliche Betrieb zum Ende des Jahres, da hier eine Anlagenerweiterung durch eine Direct-Air-Capture-Anlage stattfand und dadurch der Stromverbrauch deutlich erhöht wurde.

In beiden Anwendungsfällen konnte aufgezeigt werden, dass die geplante Regelung für die Elektrolyse funktioniert. Da in den Sommermonaten das BHKW ausgeschaltet ist und es somit nicht zu einem netzdienlichen Betrieb am TEC kam, greift die implementierte Hystereseregung ein und sorgte dafür, dass eine minimale Beladung der Wasserstoffspeicher sichergestellt wurde. Dabei fiel es negativ auf, dass sehr viele Starts stattgefunden haben, weshalb in einem zweiten Simulationsdurchlauf die Hystereseregung für die Sommermonate ausgeschaltet wurde. Bei der zweiten Analyse konnte festgestellt werden, dass sich Starts deutlich reduziert haben.

Die Anwendungsfälle wurden betrachtet, die mit Eingangsgrößen aus Werten der Datenbank gespeist wurde. Durch den anschließenden Austausch der historischen Eingangswerte durch reale Zustandswerte und mit Schaltberechtigung an der Anlage des Power-to-Gas Prozesses, konnte aus dem Simulationsprogramm ein Steuerungsprogramm entwickelt werden. Auf dieser Basis wurde der Kommunikationstest zwischen den drei Servern und der Elektrolyse erfolgreich durchgeführt.

Es können aktuelle Anlagenwerte, historischer Zustandswerte oder Schaltvektoren aus anderen Quellen als Schaltempfehlung oder direkt zum Schalten der Anlagen genutzt werden. Somit ist es künftig möglich, Fahrpläne der Anlagen am TEC individuell oder als Verbund zu erstellen und abzufahren, ausgewählte Anlagen zu priorisieren oder wenn nötig auch Anlagenlaufzeiten gegeneinander zu blockieren.

Die Anforderungen an diese Arbeit konnten somit erfolgreich umgesetzt werden, als Ausblick lässt sich festhalten, dass um die Anlagen netzdienlich fahren zu können, Prognosealgorithmen entwickelt werden müssten. Zudem könnte das aufgebaute System auch um weitere Strommarktaspekte erweitert werden, wie zum Beispiel die Erstellung von Anlagenfahrpläne nach Börsenpreisen der EPEX-Spot [22].

Literaturverzeichnis

- [1] BMU, „Klimaschutzplan 2050 – Klimaschutzpolitische Grundsätze und Ziele der Bundesregierung,“ BMU, Berlin, 2016.
- [2] BMWK, „Eröffnungsbilanz Klimaschutz,“ BMWK, Berlin, 2022.
- [3] BMWi, „Grünbuch Energieeffizienz - Diskussionspapier des Bundesministeriums für Wirtschaft und Energie,“ BMWi, Berlin, 2016.
- [4] dena, „DENA-Leitstudie „Aufbruch Klimaneutralität“,“ Deutsche Energie-Agentur GmbH, Berlin, 2021.
- [5] S. W. F. Meinhardt, IoT – Best Practices, Wiesbaden: Springer Vieweg, 2021.
- [6] D. O. Aichele C., Smart Market - Vom Smart Grid zum intelligenten Energiemarkt, Wiesbaden: Springer Vieweg, 2014.
- [7] V. E. Übertragungsnetzbetreiber, „ENTSO-E website data map,“ 2022. [Online]. Available: <https://www.entsoe.eu/data/map/>. [Zugriff am 15 04 2022].
- [8] BMWK, „Ein Stromnetz für die Energiewende,“ 2022. [Online]. Available: <https://www.bmwk.de/Redaktion/DE/Dossier/netze-und-netzausbau.html>. [Zugriff am 20 04 2022].
- [9] Ö.-I. & W. I. Prognos, „„Klimaneutrales Deutschland: In drei Schritten zu null Treibhausgasen,“ 2021. [Online]. Available: <https://www.agora-energiewende.de/veroeffentlichungen/klimaneutrales-deutschland-2045/>. [Zugriff am 15 04 2022].

- [10] M. Zapf, Stromspeicher und Power-to-Gas im deutschen Energiesystem, Wiesbaden: Springer Vieweg, 2017.
- [11] F. ISE, „Wege zu einem klimaneutralen Energiesystem,“ Fraunhofer-Institut für Solare Energiesysteme, Freiburg, 2020.
- [12] G. Z. D. Wellenreuther, Automatisieren mit SPS, Wiesbaden: Vieweg+Teubner Verlag, 2005.
- [13] W. B. Schnell G., Bussysteme in der, Wiesbaden: Springer Vieweg, 2019.
- [14] B. T. t. H. M. Vogel-Heuser B., Handbuch Industrie 4.0 Bd.1, Berlin: Springer Vieweg, 2017.
- [15] H. W. W. & H. J. Kagermann, „Umsetzungsempfehlungen für das Zukunftsprojekt Industrie 4.0. Abschlussbericht des Arbeitskreises Industrie 4.0,“ Forschungsunion Wirtschaft - Wissenschaft, Deutsche Akademie der Technikwissenschaften e.V. – acatech, 2013.
- [16] N. Kaur, „Evaluation of Tools for visually analysing Time Series Data,“ CC4E - HAW Hamburg, Hamburg, 2016.
- [17] M. Friesen, „Entwicklung einer IT -Sicherheitsinfrastruktur für verteilte Automatisierungssysteme,“ Hochschule Hannover, 2020.
- [18] W. Trojan, Das MQTT-Praxisbuch, Aachen: Elektor-Verlag GmbH, 2017.
- [19] A. Sweigart, Routineaufgaben mit Python automatisieren, Heidelberg: dpunkt.verlag, 2016.
- [20] J. L. B. F. J. R. A. F. H. Dias, Visual Self-healing Modelling for Reliable Internet-of-Things Systems, Basel, Schweiz: Springer, Cham., 2020.
- [21] Docker Inc., „docker docs,“ 2022. [Online]. Available: <https://docs.docker.com/reference/>. [Zugriff am 20 04 2022].
- [22] P. C. Edition, „Portainer Dokumentation,“ Portainer Community Edition, 03 2022. [Online]. Available: <https://docs.portainer.io/v/ce-2.11/>. [Zugriff am 20 04 2022].

- [23] epexspot, „epexspot.com,“ 2022. [Online]. Available: <https://www.epexspot.com/de/>.
[Zugriff am 16 04 2022].
- [24] P. Kurzweil, Angewandte Elektrochemie, Wiesbaden: Springer Vieweg, 2020.
- [25] B. K. M. Öggl, Docker : das Praxisbuch für Entwickler und DevOps-Teams, Bonn:
Rheinwerk Verlag, 2020.

Digitaler Anhang

Teil dieser Arbeit ist ein Anhang in elektronischer Form, der auf der beiliegenden CD enthalten ist. Die Tabelle 12 listet die enthaltenen Dateien mit einer kurzen Beschreibung auf.

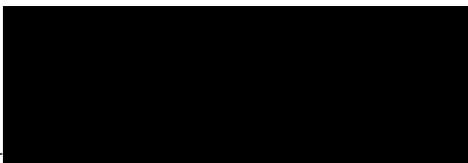
Tabelle 12: Der Inhalt der beiliegenden CD

<i>Datei</i>	<i>Beschreibung</i>
MA SFarrenkopf.pdf	Masterthesis Sebastian Farrenkopf
Datenabfrage Energieportal.html	Skript für die Daten des Energieportals
csv-Daten Bergedorf.zip	Daten des Energieportals
Datenabfrage Netzanschluss TEC.html	Skript für die Daten des TECs
csv-Daten TEC.zip	Daten des TECs
Jahresdauerlinie.html	Skript für die Erstellung der Jahresdauerlinie
Simulation.html	Skript für die Simulation des Anwendungsfalls
Elektrolyse regeln.html	Skript für die Regelung der Elektrolyse
Docker-Container	
- Blank Image	Beispiel-Image
- 20220105 Regelung Elektrolyse	Image für die Regelung der Elektrolyse
- 20220105 TEC_Excess_slim	Image für das Netzsignal am TEC
- 20220310 Energieportal	Image für das Energieportal
- 20220310 EE-Einspeisung	Image für das Energieportal EE-Einspeisung
- 20220310 EE-Prognose	Image für das Energieportal EE-Prognose

Erklärung zur selbstständigen Bearbeitung einer Abschlussarbeit

Ich versichere, dass ich meine Masterarbeit selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel verwendet habe.

Hamburg, 22.04.2022

_____ 
Sebastian Farrenkopf