



HIL NORMBOX ABWASSERSTEUERGERÄT

SCHULE

TEKO Schweizerische Fachschule Bern

DIPLOMAND

Fabian Langenegger

AUSBILDUNG

Elektrotechniker HF

JAHR

2020

Management Summary

Nach dem Abschluss des letzten Semesters der Weiterbildung zum Elektrotechniker HF soll eine Diplomarbeit geschrieben werden, um das erlernte Wissen praktisch anzuwenden.

Im Rahmen dieser Diplomarbeit wurde ein produktiv einsetzbarer HiL-Simulator, zu der sich aktuell in der Entwicklung befindlichen Abwassersteuerung, entwickelt. Mit diesem HiL-Simulator kann ein komplettes System in einer definierten Umgebung automatisch getestet werden. Der HiL-Simulator entlastet und unterstützt somit die Entwickler beim Testing der neuen Abwassersteuerung. Dadurch können viel Zeit und hohe Kosten eingespart werden und gleichzeitig gibt es dem Entwickler eine erweiterte Sicherstellung, dass sein Code mit der Hardware funktioniert. Dazu kommt, dass er die gewonnene Zeit verwenden kann, um weitere Entwicklungen zu erarbeiten. Somit werden die Entwicklungszeiten zusätzlich verkürzt.

In der Diplomarbeit wurden die verschiedenen Projektphasen des Basis-Phasenkonzepts erarbeitet. Im Zuge der Diplomarbeit konnten alle Mussziele erreicht werden. Die Hardware des HiL-Simulators ist auf einen Dauerbetrieb (24/7) ausgelegt. Die Systemschnittstellen der Abwassersteuerung werden abgedeckt und es können alle Ein- und Ausgänge simuliert und ausgewertet werden. Zusätzlich kann auch die Stromversorgung der Abwassersteuerung durch den HiL-Simulator geschaltet werden.

Die Test-Infrastruktur kann via TCP/IP den HiL-Simulator ansteuern. Während des Projekts wurde ein Kommunikationsprotokoll zusammen mit den Stakeholdern (Kunden) erarbeitet. Da der HiL-Simulator jeden Ein- und Ausgang einzeln ansteuern bzw. auslesen kann, ist eine Fehlerinjektion durch die Testprozedur möglich. Der HiL-Simulator führt nur die Befehle aus, welche ihm über TCP/IP übermittelt werden (mit entsprechendem Feedback). Somit werden im Simulator keine logischen Testabläufe abgebildet und auch keine Zustände des Systems interpretiert.

Die Kannziele wurden aus zeitlichen Gründen nicht mehr realisiert. Die Vorstudie konnte dennoch abgeschlossen werden. Es wäre daher denkbar, den HiL-Simulator in Zukunft auch noch mit diesen Features weiter zu entwickeln.

Inhaltsverzeichnis

1	VERWEISE	4
1.1	ABBILDUNGSVERZEICHNIS	4
1.2	GLOSSAR	6
1.3	ABKÜRZUNGSVERZEICHNIS	6
2	EINLEITUNG	7
2.1	AUSGANGSLAGE	8
2.2	ANFORDERUNGEN	8
2.3	AUFBAU DER DIPLOMARBEIT	9
3	INITIALISIERUNG	10
3.1	AUFGABE	10
3.1.1	<i>Form/Aufwand</i>	10
3.2	PROJEKTORGANISATION	10
3.3	WORKFLOW	11
3.4	ZIELSETZUNG	12
3.4.1	<i>Grundfunktionen (Mussziele)</i>	12
3.4.2	<i>Optionale Erweiterungen (Kannziele)</i>	13
3.5	RISIKEN	14
3.5.1	<i>Risikoanalyse</i>	14
3.5.2	<i>Massnahmen</i>	15
3.6	STAKEHOLDERMANAGEMENT	16
3.6.1	<i>Stakeholder Analyse</i>	16
4	ZEITPLAN	17
4.1	MEILENSTEINE	18
5	VORSTUDIE	19
5.1	HARDWARE AUFBAU	21
5.2	SCHNITTSTELLEN ZU ABWASSERSTEUERGERÄT	21
5.2.1	<i>Digitale Eingänge</i>	21
5.2.2	<i>Analoge Eingänge</i>	23
5.2.3	<i>Digitale Ausgänge</i>	24
5.2.4	<i>Niederspannung</i>	24
5.3	SCHNITTSTELLEN ZU TEST-INFRASTRUKTUR	24
5.4	FEHLERINJEKTION	25
5.4.1	<i>Schaltungsvorgangsfehler</i>	25
5.4.2	<i>Drahtbruch Schwimmerschalter</i>	26
5.5	BUSINESS-LOGIC	27
5.6	PROGRAMMIERSCHNITTSTELLE	27
5.7	PARAMETRIERSCHNITTSTELLE	28
5.8	MOTOREN SIMULIEREN	29
5.9	STROMMESSUNG	30
6	KONZEPT	31
6.1	VERGLEICH MANUELL ZU AUTOMATISIERT	32
6.2	LÖSUNGSVARIANTEN	34
6.3	LÖSUNGSENTSCHEID	34
7	REALISIERUNG	35
7.1	HARDWARE	35
7.1.1	<i>Mechanischer Aufbau</i>	35

7.1.2	Spannungsversorgung	36
7.1.3	Schnittstellen zu Abwassersteuergerät	37
7.1.4	Schnittstellen zu Test-Infrastruktur	38
7.1.5	National Instruments DAQ System	38
7.1.6	Übersicht der Hardware	40
7.1.7	Anschluss an die NormBox	41
7.2	ELEKTROSCHEMA	41
7.3	SOFTWARE.....	42
7.3.1	Initialisierung.....	42
7.3.2	Schnittstellen zu Abwassersteuergerät	42
7.3.3	Schnittstellen zu Test-Infrastruktur	45
7.3.4	Fehlerinjektion.....	50
8	TESTING	51
8.1	HARDWARE.....	51
8.2	SOFTWARE.....	51
8.3	HIL-SIMULATOR	51
8.3.1	Funktionstest	52
9	KONTROLLE ZIELERREICHUNG	53
9.1	GRUNDFUNKTIONEN (MUSSZIELE)	53
9.2	OPTIONALE ERWEITERUNGEN (KANNZIELE)	54
10	ABSCHLUSS	55
10.1	BERECHNUNGEN/WIRTSCHAFTLICHKEIT.....	55
10.2	REFLEXION	56
10.4	FAZIT	57
11	LITERATURVERZEICHNIS.....	58
12	DANKSAGUNG	59
13	EIGENSTÄNDIGKEITS-ERKLÄRUNG	60
14	ANHANG	61

1 Verweise

1.1 Abbildungsverzeichnis

Abbildung 1 Titelbild.....	0
Abbildung 3 Basis-PhasenkonzeptAbbildung 1 Titelbild.....	0
Abbildung 2 Kontext-Aufbau	8
Abbildung 3 Basis-Phasenkonzept	9
Abbildung 4 Kanban-Board.....	11
Abbildung 5 Mussziele	12
Abbildung 6 Kannziele	13
Abbildung 7 Risikoanalyse Tabelle.....	14
Abbildung 8 Risikoanalyse Auswertung.....	14
Abbildung 9 Massnahmen Tabelle.....	15
Abbildung 10 Massnahmen Auswertung	15
Abbildung 11 Einfluss-Interessen-Matrix	16
Abbildung 12 Kommunikationsmassnahmen	16
Abbildung 13 Zeitplan	17
Abbildung 14 Meilensteine.....	18
Abbildung 15 Wasserfallmodell.....	19
Abbildung 16 V-Modell.....	19
Abbildung 17 Scrum.....	20
Abbildung 18 Schwimmerschalter.....	21
Abbildung 19 Schaltungsvorgang Schwimmerschalter.....	22
Abbildung 20 Drucksonde.....	23
Abbildung 21 Signalberechnung Drucksonde.....	23
Abbildung 22 Testkreis.....	24
Abbildung 23 Schaltungsvorgangsfehler	25
Abbildung 24 Drahtbruch Fehler.....	26
Abbildung 25 Drahtbruch Wahrheitstabelle.....	26
Abbildung 26 SEGGER J-Link Ultra+	27
Abbildung 27 SEGGER J-Link Pinout.....	28
Abbildung 28 Parameter Abwassersteuerung.....	28
Abbildung 29 Beispielschaltung Lastwiderstand	29
Abbildung 30 Beispielschaltung Zangenstrommesser	30
Abbildung 31 Software Testpyramide.....	31
Abbildung 32 Konzept Skizze.....	31
Abbildung 33 Vorgehen manuell.....	32
Abbildung 34 Vorgehen automatisiert.....	32
Abbildung 35 Zeiten Test manuell.....	32
Abbildung 36 Zeiten Test automatisiert	32
Abbildung 37 Kosteneinsparnisse.....	33
Abbildung 38 Nutzwertanalyse.....	34
Abbildung 39 Swibox Gehäuse	35
Abbildung 40 Spannungsversorgung	36
Abbildung 41 Schnittstellen Abwassersteuerung	37
Abbildung 42 Kopplungsrelais Siemens	37
Abbildung 43 NI cDAQ-9185 CompactDAQ Chassis	38
Abbildung 44 NI 9375 Modul für digitale Signale	39
Abbildung 45 NI 9265 Stromausgabemodul	39
Abbildung 46 Übersicht der Hardware.....	40
Abbildung 47 Anschluss an die NormBox	41
Abbildung 48 Aufbau Elektroschema.....	41
Abbildung 49 Blockdiagramm Schnittstellen zu Abwassersteuergerät.....	42
Abbildung 50 LabVIEW Frontpanel Schnittstellen zu Abwassersteuergerät.....	43
Abbildung 51 LabVIEW Blockdiagramm Schnittstellen zu Abwassersteuergerät.....	44
Abbildung 52 Blockdiagramm Schnittstellen zu Test-Infrastruktur.....	45
Abbildung 53 LabVIEW-Blockdiagramm Verbindungsaufbau des Servers.....	46
Abbildung 54 PuTTY Konfigurationsseite	46
Abbildung 55 PuTTY Terminalfenster.....	47

Abbildung 56 LabVIEW Lesen der empfangenen Zeichen	47
Abbildung 57 ASCII Steuerzeichen	47
Abbildung 58 Beispiel Befehlsaufbau	48
Abbildung 59 Kommunikationsprotokoll	48
Abbildung 60 LabVIEW Set Befehl Beispiel	49
Abbildung 61 LabVIEW Get Befehl Beispiel	49
Abbildung 62 Testlauf Kommunikationsprotokoll	50
Abbildung 63 DUT-Dummy	51
Abbildung 64 Zielerreichung Mussziele	53
Abbildung 65 Zielerreichung Kannziele	54
Abbildung 66 Hardwarekosten	55
Abbildung 67 Entwicklungskosten	55

1.2 Glossar

Dismessa	Projektname Biral Abwassersteuergerät
NormBox	Produktname Biral Abwassersteuergerät
NI MAX	Konfigurationssoftware für Hard- und Software von National Instruments ¹
LabVIEW	Systementwicklungssoftware der Firma National Instruments ²
cDAQ	Hardware zur Durchführung elektrischer und physikalischer Messungen ³
EICAD	Software zum Erstellen elektrotechnischer Dokumentationen (Schemata) der Firma Aucotec AG ⁴
TCP/IP	Transmission Control Protocol/Internet Protocol ist eine Gruppe von Netzwerkprotokollen die dem Austausch von Daten zwischen Computern dient

1.3 Abkürzungsverzeichnis

NI	National Instruments
VI	Virtuelles Instrument
OSK	Ölsperrkammer
WSK	Wicklungsschutzkontakt
HiL	Hardware in the Loop
DUT	Device under Test
NO	Normally Open (Schliesser-Kontakt)
NC	Normally Closed (Öffner-Kontakt)
LS	Leitungsschutzschalter

¹ (National Instruments, 2019)

² (National Instruments, 2020)

³ (National Instruments, 2020)

⁴ (Aucotec AG, 2020)

2 Einleitung

Der Wasserverbrauch für Haushalte und Kleingewerbe beläuft sich in der Schweiz auf ca. 9'000'000 m³. Dies ergibt einen Pro-Kopf-Verbrauch von ca. 175 l/Person und Tag.⁵

Da die Abwasserentsorgung automatisch vonstatten geht, macht sich die Gesellschaft meist keine grossen Gedanken darüber, wie dies funktioniert. Es ist für sie selbstverständlich, dass zum Beispiel mit einem Knopfdruck das Abwasser wegbefördert wird. Doch was genau steckt hinter der Abwasserentsorgung?

Um diese Frage zu klären, muss zuerst die Definition von Abwasser angeschaut werden. Abwasser ist ein Oberbegriff für aus verschiedenen Quellen stammendes Wasser, das über bauliche Anlagen fortgeleitet wird. Dazu gehört Regenwasser, Schmutzwasser und Fremdwasser. Das Abwasser wird in der Kanalisation gesammelt und abtransportiert.⁶ Wenn das Abwasser nicht per Schwerkraftentwässerung entsorgt werden kann, kommt eine Abwasserhebeanlage zum Einsatz.

Eine Abwasserhebeanlage besteht im Minimum aus einem Sammelbehälter, einer Pumpe, drei Schwimmerschaltern und einer Pumpensteuerung. Das Abwasser sammelt sich im Sammelbehälter und die Pumpe wird durch die Pumpensteuerung, je nach Wasserniveau, ein- bzw. ausgeschaltet. Um die Kunden stets mit neuer Technik wie zum Beispiel Bluetooth-Konnektivität zu versorgen, versucht die Firma Biral mit Neuentwicklungen den Markt zu sättigen.

Im dritten Quartal 2020 hat die Entwicklung einer neuen Abwassersteuerungsplattform begonnen. Die Anforderungen an das Testing eines Produkts in der Entwicklungsphase nehmen stetig zu. Weiter wird die Intervall-Zeit zwischen den einzelnen Revisionen immer kürzer. Diese beiden Tendenzen führen dazu, dass das Testing zusehends automatisiert werden muss. Dies geschieht unter anderem durch das Einsetzen von HiL-Simulatoren, die es ermöglichen, ein komplettes System in einer definierten Umgebung automatisch zu testen. Im Zuge dieser Diplomarbeit soll ein HiL-Simulator, zu der sich aktuell in der Entwicklung befindlichen Abwassersteuerungsplattform, realisiert werden.

⁵ (Energie Wasser Bern, 2020)

⁶ (Wikipedia, 2020)

2.1 Ausgangslage

Für eine Neuentwicklung im Produktportfolio der Abwasserhebeanlagen «NormBox» wird ein automatisiertes Testsystem in Form eines HiL's benötigt. Im Rahmen der Diplomarbeit soll ein Simulator entwickelt werden, der alle Eingänge gemäss Vorgaben simuliert und die Ausgänge entsprechend auswertet. Die Integration des Simulators in das automatisierte Testsystem ist der Abbildung 2 zu entnehmen.

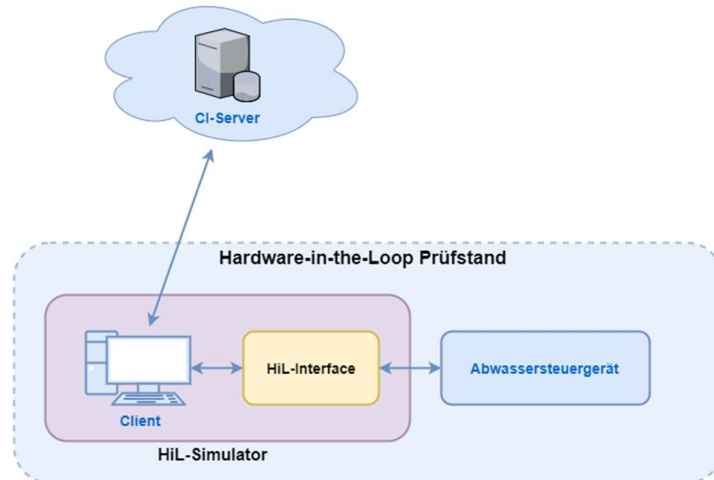


Abbildung 4 Kontext-Aufbau

2.2 Anforderungen

Die Aufgabe der Diplomarbeit umfasst die Entwicklung eines produktiv einsetzbaren HiL-Simulators, der alle Ein- und Ausgänge des Abwassersteuergerätes abdecken kann, eine Schnittstelle zur bestehenden Test-Infrastruktur hat und für einen 24/7-Betrieb ausgelegt ist. Die Anforderungen wurden im Pflichtenheft (A.1 Pflichtenheft) definiert und sollen im Zuge dieser Diplomarbeit erarbeitet und realisiert werden.

- Hardware Aufbau
- Schnittstellen zu Abwassersteuergerät
- Schnittstellen zu Test-Infrastruktur
- Fehlerinjektion
- Business-Logic

2.3 Aufbau der Diplomarbeit

Die Diplomarbeit ist auf dem Basis-Phasenkonzept⁷ aufgebaut. Das Phasenkonzept wird in Abbildung 3 dargestellt. Die Diplomarbeit umfasst die Planungs- sowie die Realisierungs- und Einführungsphase. Die Erhaltungsphase kann jedoch erst nach der Diplomarbeit abgearbeitet werden, da das Projekt nach der Präsentation noch nicht abgeschlossen ist.

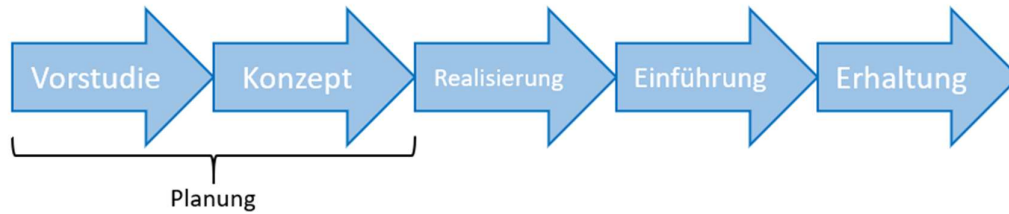


Abbildung 5 Basis-Phasenkonzept

⁷ (Projektmanagement für Führungsfachleute, 2017, S. 19)

3 Initialisierung

3.1 Aufgabe

Nach dem Abschluss des letzten Semesters der Weiterbildung zum Elektrotechniker HF soll eine Diplomarbeit geschrieben werden, um das erlernte Wissen praktisch anzuwenden. Ziel der Diplomarbeit ist dabei, eine Aufgabenstellung innerhalb einer vorgeschriebenen Zeitspanne selbstständig, umfassend und zweckmässig zu lösen sowie sauber zu dokumentieren und zu präsentieren. Zur Diplomarbeit gehört auch eine Onlinepublikation der Arbeit, auf einer durch die TEKO zur Verfügung gestellten Plattform.

3.1.1 Form/Aufwand

Die TEKO hat folgende Richtlinien zur Form und zum Aufwand bekanntgegeben:

- Schriftliche Dokumentation
- Format A4
- Einseitig beschrieben
- Fortlaufend nummeriert
- Gebunden oder in einem Ordner (wenn verlangt)
- Ca 150-250 Stunden Arbeitsaufwand
- Zeichnerisch-konstruktiver und/oder experimenteller Teil (z.B. Modell, Prototyp, Labor-Aufbau etc.)

3.2 Projektorganisation

Die Projektorganisation sieht folgendermassen aus:

Auftraggeber: Herr Bruno Christen (BCH Biral AG)
Projektleiter: Herr Fabian Langenegger (FLA Biral AG)
Projektbetreuer: Herr Jan Segessenmann (JSE Biral AG)
Diplombetreuer: Herr Xavier Pfaff (TEKO)

3.3 Workflow

Um den Workflow (Arbeitsfluss) abzubilden und zu visualisieren, wird ein Kanban-Board eingesetzt. Mithilfe dieses Tools kann man zu jedem Zeitpunkt die offenen Tasks, die Tasks in Bearbeitung und die abgeschlossenen Tasks einsehen und weiss, was noch zu erledigen ist. Gleichzeitig ist es motivierend, die abgeschlossenen Aufgaben zu sehen und es wird versucht, die sich in Bearbeitung befindlichen Tasks möglichst schnell abzuschliessen. In der Abbildung 4 ist das Kanban-Board zu sehen.



Abbildung 6 Kanban-Board

3.4 Zielsetzung

Um die Entwickler beim Testing der neuen Abwassersteuerung zu entlasten und zu unterstützen, soll ein HiL-Simulator realisiert werden. Dieser soll eine Schnittstelle zur bestehenden Test-Infrastruktur haben und das Testing automatisieren. Die Diplomarbeit wird in Grundfunktionen (Mussziele) und Optionale Erweiterungen (Kannziele) unterteilt.

3.4.1 Grundfunktionen (Mussziele)

Die Grundfunktionen haben zum Ziel, eine funktionsfähige Baugruppe zu entwickeln, die sich als HiL-Simulator in die bestehende Testumgebung integrieren lässt. Die Mussziele sind in der folgenden Abbildung ersichtlich:

Ziel	Erläuterung	Bestandteil
Hardware Aufbau	Die Hardware des HiL-Simulators muss so aufgebaut werden, dass er alle Ein- und Ausgänge des Abwassersteuergerätes abdeckt. Zudem muss er eine Schnittstelle zur bestehenden Test-Infrastruktur haben. Die Speisung des Interfaces (24V) muss den Dauerbetrieb (24/7) gewährleisten können.	Hardware
Schnittstellen zu Abwassersteuergerät	Der HiL-Simulator muss alle «Systemschnittstellen» der Abwassersteuerung abdecken. Alle Eingänge müssen gemäss Vorgaben simuliert werden können und die Ausgänge entsprechend ausgewertet werden. Die Stromversorgung der Abwassersteuerung muss über den HiL-Simulator geschaltet werden können.	Hard- und Software
Schnittstellen zu Test-Infrastruktur	Die Test-Infrastruktur muss über TCP/IP oder Serial-Interface (USB) den HiL-Simulator ansteuern können (die Kommunikation wird im Zuge des Projekts bestimmt).	Hard- und Software
Fehlerinjektion	Der HiL-Simulator muss in der Lage sein, bei allen überwachten Systemkomponenten resp. Schnittstellen gezielt Fehler zu injizieren.	Hard- und Software
Business-Logic	Der HiL-Simulator bildet nur eine Schnittstelle zwischen DUT und der Test-Infrastruktur. Die Testprozedur muss über die Test-Infrastruktur definiert werden können. Es sollen keine logischen «Testabläufe» im Interface abgebildet werden oder irgendwelche Zustände des Systems interpretiert werden.	Hard- und Software

Abbildung 7 Mussziele

3.4.2 Optionale Erweiterungen (Kannziele)

Die Kannziele sind in der folgenden Abbildung ersichtlich:

Ziel	Erläuterung	Bestandteil
Programmierschnittstelle	Die Abwassersteuerung wird über ein JTAG-Programmier-Interface verfügen. Dieses soll für das Programmieren der Firmware verwendet werden. Der HiL-Simulator übernimmt dabei eine Controller-Aufgabe – das eigentliche Programmieren soll über ein SEGGER J-Link erfolgen.	Hard- und Software
Parametrierschnittstelle	Es wird möglich sein, die Abwassersteuerung mithilfe einer Parametrierschnittstelle zu parametrieren. Dazu wird eine serielle Verbindung (UART) eingesetzt, über die durch das ProtoBuf-Protokoll kommuniziert wird. Der HiL-Simulator soll, anhand von Angaben der Test-Infrastruktur, das Abwassersteuergerät parametrieren können.	Hard- und Software
Motoren simulieren	Mit einem externen Lastwiderstand soll ein externer Motor simuliert werden, was den Betrieb einer Pumpe abbildet. Da die Abwassersteuerung über ein breites Leistungsspektrum verfügen wird, soll der gewählte Lastwiderstand einstellbar sein.	Hard- und Software
Strommessung	Auf der Abwassersteuerung wird die Stromaufnahme der Motoren/Pumpen überwacht. Um den gemessenen Strom der Steuerung prüfen zu können, muss der HiL-Simulator auch über diese Möglichkeit verfügen.	Hard- und Software

Abbildung 8 Kannziele

3.5 Risiken

Um die Risiken des Projekts abschätzen zu können ist es wichtig, diese zu kennen, um Massnahmen ergreifen zu können. Eine Risikoanalyse ist ein gutes Mittel dazu, denn darin wird auch optisch verdeutlicht, wo gewisse Risiken auftreten und welchen Einfluss sie haben könnten.

3.5.1 Risikoanalyse

In den nachfolgenden Abbildungen werden die Risiken analysiert und in tabellarischer Form wie auch in einem Diagramm veranschaulicht.

Nr	Risiko	Beschreibung
1	Krankheit	Der Diplomand wird krank (Grippe, Erkältung, Coronavirus)
2	Auftragslage	Die Auftragslage erfordert die zwingende Arbeit des Diplomanden im Tagesgeschäft
3	Beschaffung	Die gewünschte Hardware ist nicht verfügbar, durch Lieferengpässe etc.
4	Software	Die entwickelte Software funktioniert nicht ordnungsgemäss oder nur bedingt, da der Diplomand noch sehr wenig Erfahrung mit LabView hat
5	Hardware	Die Hardware geht bei der Montage, Programmierung etc. kaputt
6	Schnittstellen	Die Schnittstelle zur bestehenden Test-Infrastruktur funktioniert nicht

Abbildung 9 Risikoanalyse Tabelle

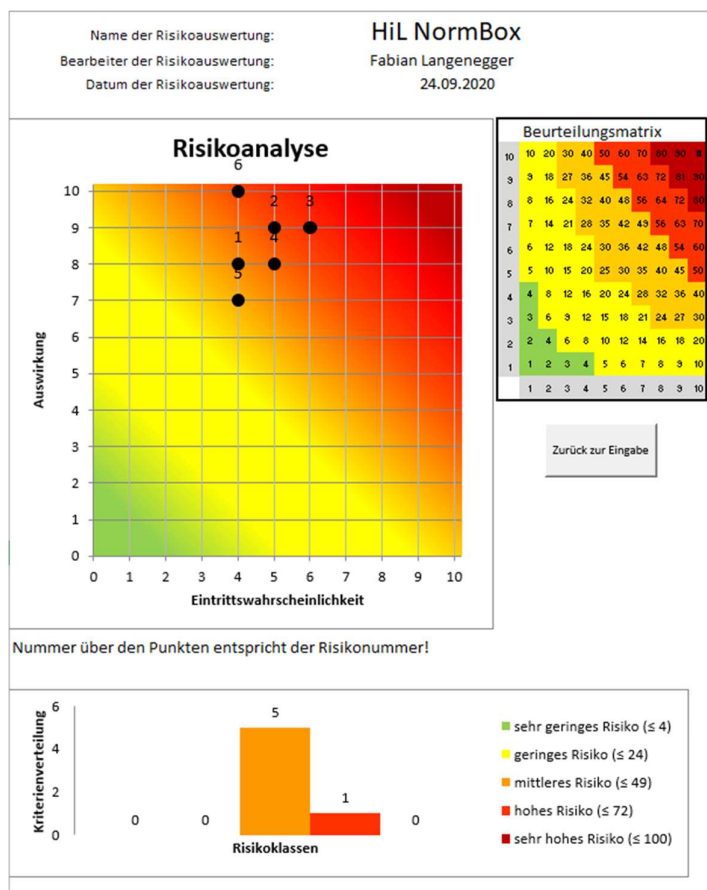


Abbildung 10 Risikoanalyse Auswertung

3.5.2 Massnahmen

Durch die entsprechenden Massnahmen konnten die Risiken minimiert werden. In den folgenden Abbildungen ist dies deutlich erkennbar.

Nr	Risiko	Massnahme
1	Krankheit	Der Diplomand kennt die Richtlinien des BAG für die Minimierung einer Ansteckung durch das Coronavirus.
2	Auftragslage	Der Diplomand wird nur einen Tag pro Woche (Mittwoch) im Tagesgeschäft tätig sein. Da im Spezialsteuerungsbau noch zwei weitere Mitarbeiter sind, welche das Tagesgeschäft übernehmen, stellt dies ein kleines Risiko dar.
3	Beschaffung	Ein Grossteil der Hardware wurde bereits im Vorfeld beschafft, um das Risiko zu minimieren. Weiter sind im internen Lager viele Komponenten, welche verwendet werden können.
4	Software	Der Diplomand hat im Vorfeld einen Onlinegrundkurs mit LabView besucht, um die Programmierung zu erlernen.
5	Hardware	Der Diplomand kennt den vorsichtigen Umgang mit der Hardware und wird diese auch gründlich vor Inbetriebnahme auf Fehler, Falschverdrahtung etc. überprüfen.
6	Schnittstellen	Da es zwei Varianten als Schnittstelle zur bestehenden Test-Infrastruktur gibt, minimiert sich so das Risiko, dass die Kommunikation nicht funktioniert.

Abbildung 11 Massnahmen Tabelle

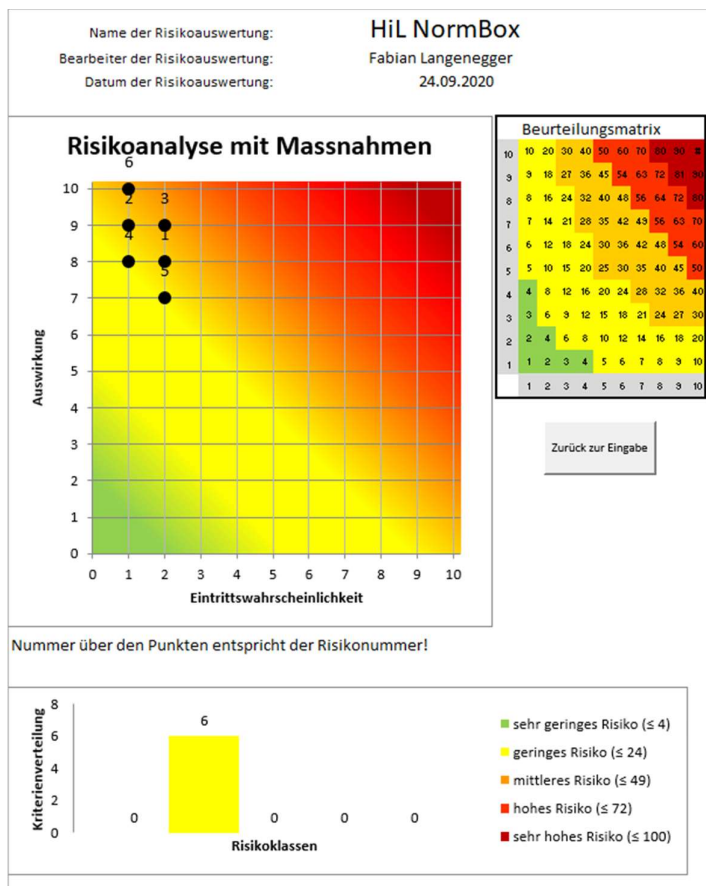


Abbildung 12 Massnahmen Auswertung

3.6 Stakeholdermanagement

Das Stakeholdermanagement bezweckt, die Akzeptanz des Projekts bei den Stakeholdern (Anspruchsgruppen) zu fördern und sie dafür zu gewinnen, die Projektziele aktiv zu unterstützen. Es ist eine wichtige Aufgabe des Projektleiters in sämtlichen Projektphasen.⁸

3.6.1 Stakeholder Analyse

In der Stakeholder Analyse geht es darum, die verschiedenen Anspruchsgruppen zu identifizieren und ihren Einfluss und ihre Interessen zu analysieren. Aus dieser Analyse können Kommunikationsmassnahmen abgeleitet werden, damit das Projekt von den Stakeholdern akzeptiert und aktiv unterstützt wird.

Die Stakeholder wurden in der Abbildung 10 mit der Einfluss-Interessen-Matrix⁹ analysiert, um einen guten Überblick zu erhalten.

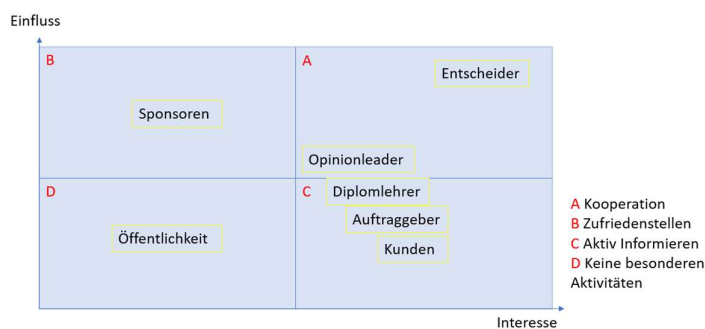


Abbildung 13 Einfluss-Interessen-Matrix

Die Kommunikationsmassnahmen konnten anhand der Einfluss-Interessen-Matrix erstellt werden und sind der Abbildung 12 zu entnehmen.

Stakeholder	Wer	Aktion	Quadrant	Massnahme
Entscheider	Lenkungsgremium des Projekts (Projektteam)	Kooperation	A	Das Projektteam besteht nur aus dem Projektleiter, der den Fortschritt des Projektes bestens kennt
Opinionleader	Abteilungsleiter Product Development & - Management	Kooperation	A	Als Partner in die Planungs- und Entscheidungsprozesse einbinden. Regelmässig, offen und umfassend informieren, persönliche Beziehung aktiv pflegen
Sponsoren	Projektförderer (Geschäftsleitung)	Zufriedenstellen	B	Bei wichtigen Entscheiden beiziehen, persönliche Beziehung pflegen. Wenn möglich von Quadrant B in Quadrant A holen, da sie grossen Einfluss haben können
Diplomlehrer	Xavier Pfaff	Aktiv Informieren	C	Regelmässig und adressatengerecht informieren
Auftraggeber	Teamleiter Product Management & PM Digitale Produkte	Aktiv Informieren	C	Regelmässig und adressatengerecht informieren
Kunden	Elektroentwicklung / Software Entwicklung	Aktiv Informieren	C	Regelmässig und adressatengerecht informieren
Öffentlichkeit	Behörde, Gesellschaft, Wirtschaft, Medien	Keine besonderen Aktivitäten	D	Da das Projekt nur Biral Intern eine Bedeutung hat sind keine besonderen Massnahmen erforderlich

10

Abbildung 14 Kommunikationsmassnahmen

⁸ (Projektmanagement für Führungsfachleute, 2017, S. 57)

⁹ (Projektmanagement für Führungsfachleute, 2017, S. 53)

¹⁰ (Projektmanagement für Führungsfachleute, 2017, S. 51-56)



4.1 Meilensteine

Die Meilensteine sind in der Abbildung 14 ersichtlich.



Abbildung 16 Meilensteine

5 Vorstudie

Die Entwicklung einer neuen Abwassersteuerungsplattform ist mit viel Aufwand verbunden. Die Hard- und Software muss dazu hohen Anforderungen entsprechen, um die Bedürfnisse der Kunden zu befriedigen. Dazu stellt sich auch immer die Frage, was die Konkurrenz zu bieten hat. Die Hardware zusammenzustellen ist meistens nicht ein grosses Problem, da man anhand eines Pflichtenhefts ziemlich gut eingrenzen kann, was benötigt wird. Hingegen die Software zu schreiben und zu testen erfordert im Verhältnis viel mehr Zeit.

Es gibt verschiedene Methoden bzw. Modelle, nach denen die Softwareentwicklung erfolgen kann. Die Bekanntesten darunter sind sicherlich das Wasserfallmodell und das V-Modell.

Das Wasserfallmodell wurde zum Beispiel auch für diese Diplomarbeit verwendet. Ist ein Punkt abgeschlossen, geht es den Wasserfall hinunter zum nächsten Schritt. In der Abbildung 15 ist ein Beispiel zu sehen.

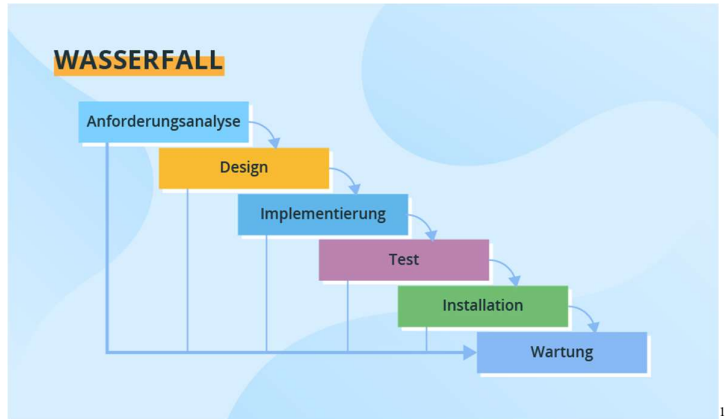


Abbildung 17 Wasserfallmodell

Man sieht im Beispiel, dass das Testing erst nach erfolgreicher Implementierung geschieht. Das Problem an diesem Modell ist, dass es keine Möglichkeit gibt, beispielsweise Softwareanforderungen während der Entwicklungsphase neu zu bewerten.

Das V-Modell ist etwas anders aufgebaut. Zuerst werden die Punkte auf der linken Seite abgearbeitet. Ist die Codierung fertiggestellt werden die einzelnen Punkte mit entsprechenden Tests geprüft. In der Abbildung 16 ist dazu ein Beispiel zu sehen.

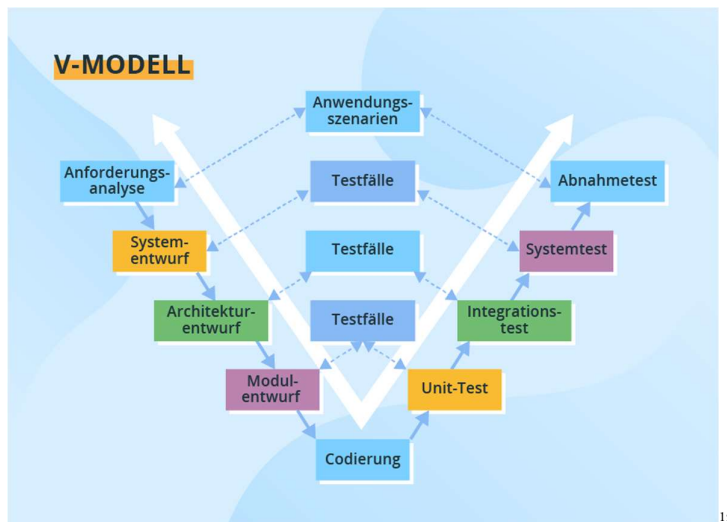


Abbildung 18 V-Modell

¹¹ (Shiklo, 2019)

¹² (Shiklo, 2019)

Auch wenn Fehler in Anforderungsspezifikation, Code und Architektur frühzeitig erkannt werden, ist es immer noch teuer und schwierig, Änderungen während der Entwicklung zu implementieren. Wie beim Wasserfallmodell werden alle Anforderungen zu Beginn erfasst und können nicht geändert werden.

Aus diesem Grund geschieht die Softwareentwicklung bei der Biral AG mit einer agilen Entwicklungsmethode, dem Scrum. Wie das Scrum aufgebaut ist, wird der Abbildung 17 entnommen.

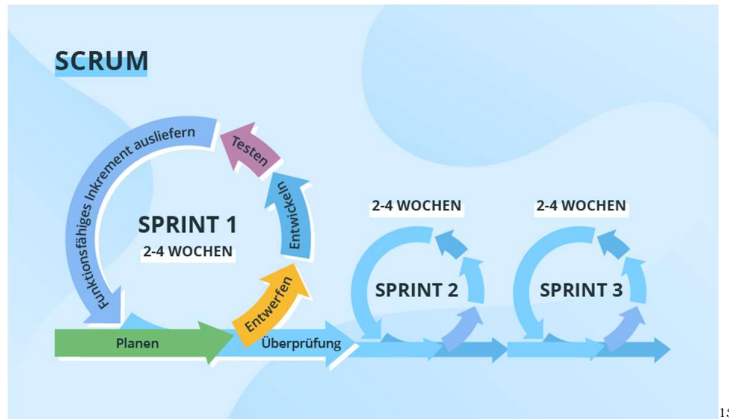


Abbildung 19 Scrum

Scrum ist eine der beliebtesten agilen Entwicklungsmethoden, welche die Aufteilung der gesamten Projektlaufzeit in kurze Iterationen (sog. Sprints) ermöglicht. Jede Iteration beginnt mit der sorgfältigen Planung und endet mit der Sprint Retrospektive, in welcher der vorherige Sprint analysiert und bewertet wird. Danach folgt der nächste Sprint, der auf den Ergebnissen des vorherigen Sprints gebaut ist (z. B. Kundenfeedback, neuen Anforderungen an das Produkt und mehr). Eine Iteration dauert üblicherweise 2 bis 4 Wochen. Nachdem die Aktivitäten für den nächsten Sprint festgelegt worden sind, können keine Änderungen mehr vorgenommen werden.

Der Vorteil hierbei ist, dass die geschriebene Software auch gleich getestet wird und man auch neue Anforderungen besser implementieren kann. Um die Entwicklung bei diesen Softwaretests zu entlasten, soll der HiL-Simulator eingesetzt werden. Der HiL-Simulator soll dafür alle Ein- und Ausgänge des Abwassersteuergerätes abdecken und eine Schnittstelle zur bestehenden Test-Infrastruktur haben. Die Abwassersteuerung kann fehlerhafte Zustände der externen Systemkomponenten erkennen. Dies soll durch beabsichtigte Fehlerinjektion getestet werden. Da der HiL-Simulator nur eine Schnittstelle zwischen DUT und der Test-Infrastruktur bildet, soll die Testprozedur über die Test-Infrastruktur definiert werden können.

¹³ (Shiklo, 2019)

5.1 Hardware Aufbau

Die Anforderungen an den Hardware-Aufbau sind im Pflichtenheft (A.1 Pflichtenheft) bereits definiert worden. Die wichtigsten Punkte werden aber in der Vorstudie nochmals angeschaut, um die Lösungsvarianten zu erarbeiten. Das Gehäuse muss so gewählt werden, dass es der gleichen Schutzart wie die Abwassersteuerung (IP54) entspricht. Stromführende Komponenten müssen den Berührungsschutz gewährleisten oder es soll mit Sicherheitskleinspannung gearbeitet werden. Die Stromversorgung des Simulators wurde auf 24V festgelegt und soll für einen Dauerbetrieb (24/7) ausgelegt werden. Der HiL-Simulator muss alle Ein- und Ausgänge des Abwassersteuergerätes abdecken. Zudem muss er eine Schnittstelle zur bestehenden Test-Infrastruktur haben. Diese Schnittstellen werden in den nächsten Abschnitten genauer angeschaut.

5.2 Schnittstellen zu Abwassersteuergerät

Die Ein- und Ausgänge der Abwassersteuerung wurden für eine bessere Übersichtlichkeit aufgeteilt in:

- Digitale Eingänge
- Analoge Eingänge
- Digitale Ausgänge
- Niederspannung

5.2.1 Digitale Eingänge

Die Digitalen Eingänge der Steuerung werden nachfolgend genauer erläutert.

- **Schwimmerschalter**

Der Schimmerschalter, auch Schwimmer oder Niveauregler genannt, ist eine Vorrichtung, welche anhand des Wasserpegels geschaltet wird. Im Inneren ist ein Wechselkontakt vorhanden. In Abbildung 18 wird dies übersichtlich dargestellt.

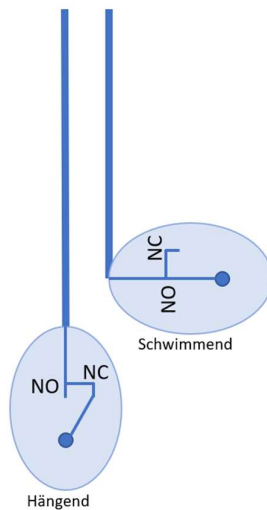


Abbildung 20 Schimmerschalter

Es gibt im Ganzen vier Schwimmerschalter bei einer Doppelpumpenanlage. Der Schaltungsvorgang wird in der nächsten Abbildung aufgezeigt.

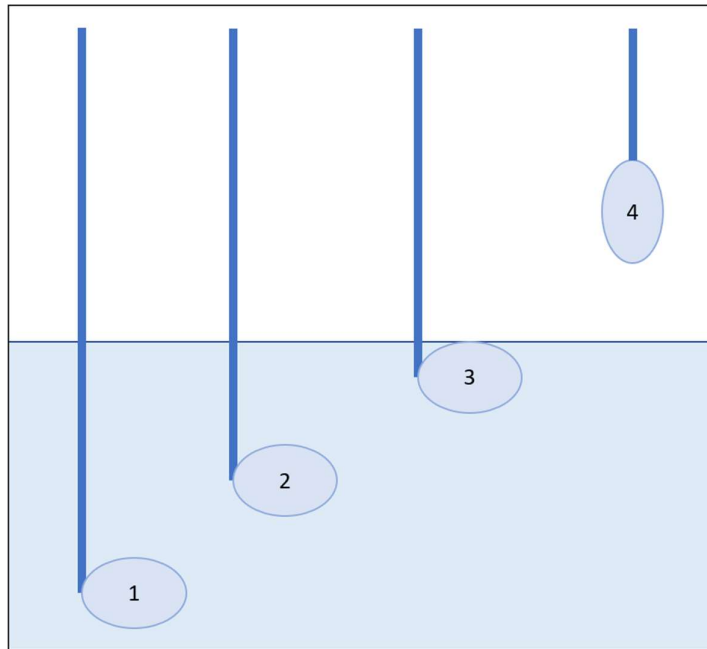


Abbildung 21 Schaltungsvorgang Schwimmerschalter

In diesem Beispiel sieht man sehr gut den Schaltungsvorgang. Steigt das Wasserniveau, werden die Schwimmer entsprechend geschaltet. Bei Schwimmer 2(EIN1) wird die erste Pumpe eingeschaltet. Steigt das Niveau weiter und erreicht den Schwimmerschalter3(EIN2) wird auch die zweite Pumpe angesteuert. Die Pumpen laufen so lange, bis der Schwimmer1(AUS) hängend wird. Erreicht das Wasserniveau jedoch den Niveauregler4(HOCH) besteht die Gefahr des Überlaufens des Abwasserschachts und es wird eine Warnmeldung gemacht.

- **Externe Sperrung**

Dieser Eingang wird benötigt, um die Steuerung zu sperren. Zum Beispiel wenn Benzin oder Öl in den Schacht läuft, muss die Steuerung gesperrt werden können.

- **Wicklungsschutzkontakt (WSK)**

Der Wicklungsschutzkontakt ist ein Bimetall Schalter, der die Motorwicklungen vor Überhitzung schützt. Er ist im Normalfall geschlossen und öffnet, wenn die Temperatur der Wicklung zu hoch wird. Die Auslösetemperatur beträgt 132 °C.

- **Ölsperrkammer (OSK)**

Die Ölsperrkammer überprüft die Dichtheit der Pumpe. Ein Dichtungssensor ist in der Ölsperrkammer eingebaut und erfasst allfälliges Eindringen von Wasser durch die untere Gleitringdichtung. Tritt Wasser durch die Gleitringdichtung ein, wird das Öl leitfähig und der Sensor schliesst einen Kontakt. Auf der Steuerung wird ein Alarm ausgegeben und der Anwender weiss, dass er die Pumpe revidieren muss. Der OSK dient somit dem Schutz, dass kein Wasser in den Motor eindringen und einen Kurzschluss verursachen kann.

5.2.2 Analoge Eingänge

Der Analoge Eingang wird verwendet, wenn eine Drucksonde anstelle der Schwimmerschalter verwendet wird. Die Drucksonde ist ein Hydrostatischer Druckaufnehmer, welcher ein Analogsignal (4 – 20 mA) je nach Wasserniveau an die Steuerung leitet. Der Standardmessbereich der Abwassersteuerung entspricht dabei 0-2 Meter. In der nachfolgenden Abbildung wird dies anschaulich dargestellt.

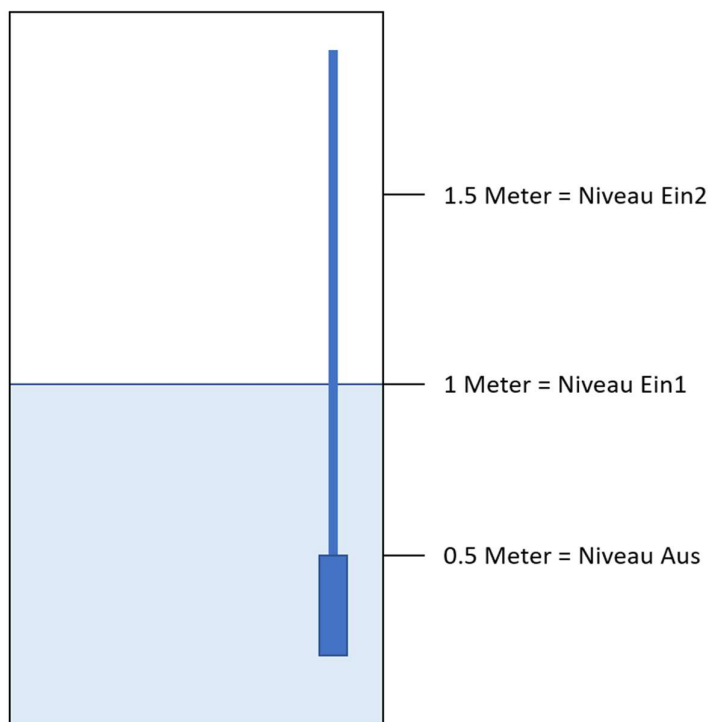


Abbildung 22 Drucksonde

Die Niveausonde verhält sich analog zu den Schwimmerschaltern. Es gibt wiederum die verschiedenen Niveaus, bei welchen die Pumpen ein- und ausgeschaltet werden.

Der Messbereich der Drucksonde ist 0-2 Meter Wassersäule (mWS). Das Ausgangssignal der Sonde ist 4-20 mA. Um den Zusammenhang zu verdeutlichen, wurden in der nachfolgenden Tabelle die Signale mit dem entsprechenden Wasserstand berechnet und tabellarisch dargestellt.

$$4-20\text{mA} == 0-2\text{mWS}$$

Signal (mA)	Wasserstand (mWS)
4	0.000
5	0.125
6	0.250
7	0.375
8	0.500
9	0.625
10	0.750
11	0.875
12	1.000
13	1.125
14	1.250
15	1.375
16	1.500
17	1.625
18	1.750
19	1.875
20	2.000

Abbildung 23 Signalberechnung Drucksonde

5.2.3 Digitale Ausgänge

Die Digitalen Ausgänge sind potentialfreie Wechselkontakte. Sie werden oft benutzt, wenn die Meldungen in einer zentralen Überwachung ausgewertet werden müssen.

5.2.4 Niederspannung

Umfasst die Schaltkontakte der Motoren (3x400VAC) und des Alarmhorns (230VAC). Diese müssen zwingend galvanisch getrennt ausgewertet werden. Die Stromversorgung der Abwassersteuerung soll ebenfalls durch den HiL-Simulator geschaltet werden können.

5.3 Schnittstellen zu Test-Infrastruktur

Die vorhandene Test-Infrastruktur besteht aus einem zentralen CI-Server (Bamboo von Atlassian) und einem dezentralen Build-Agenten. CI steht für Continuous Integration, auf Deutsch kontinuierliche Integration. Es ist eine Technik der agilen Softwareentwicklung. Die in kleineren Schritten vorgenommenen Änderungen des Codes werden in der Praxis regelmässig zusammengeführt. Die jeweils aktuellste Code-Version wird durch einen automatisierten Build-Prozess übernommen, was zu regelmässigen Versionskontrollen führt.¹⁴ Der CI-Server wird somit benutzt, um mit Git eine Versionsverwaltung des Codes zu haben. Wenn ein Entwickler den Code pusht, wird er im CI-Server aktualisiert und zusammengeführt.

Der Build Agent wird verwendet, um zum Beispiel den Code zusammen zu führen, zu kompilieren und Tasks auszuführen. Er ist also so gesehen ein Hilfsmittel des CI-Servers.

Das Ziel der Aufgabe soll bewirken, dass jede Codeänderung auf dem CI-Server aktualisiert wird und dann z.B. einen Task im Build-Agenten triggert (auslöst). Somit wird der Test mit dem HiL-Simulator durchgeführt. Das Konzept ist der Abbildung 22 zu entnehmen.

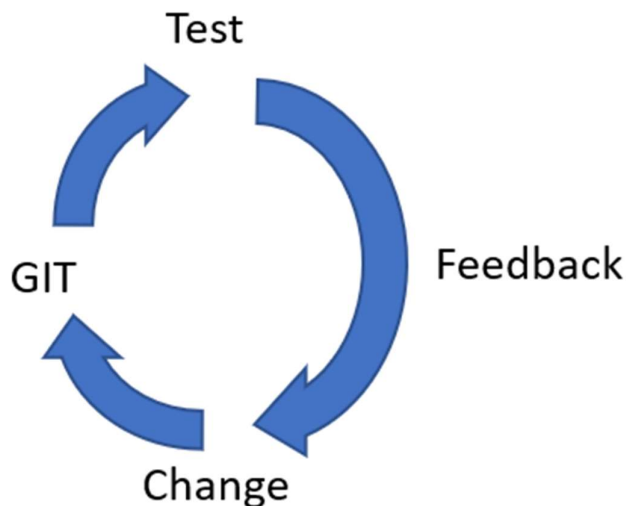


Abbildung 24 Testkreis

Der HiL-Simulator soll dabei die Rolle des Servers einnehmen. Die Verbindung wird vom Client (Build-Agent) via TCP/IP oder Serial-Interface (USB) hergestellt. Die Kommunikation soll im Zuge des Projekts unter Berücksichtigung der Stakeholder (Kunden) bestimmt werden.

¹⁴ (Dirk-Jan, 2018)

5.4 Fehlerinjektion

Die Abwassersteuerung ist in der Lage fehlerhafte Zustände der Systemschnittstellen zu erkennen. Darunter gehören:

- Schaltungsvorgangsfehler
- Drahtbruch Schwimmerschalter

5.4.1 Schaltungsvorgangsfehler

Ein Schaltungsvorgangsfehler tritt auf, wenn die Reihenfolge der Schwimmerschalter nicht korrekt ist. Dies wird in der nächsten Abbildung dargestellt.

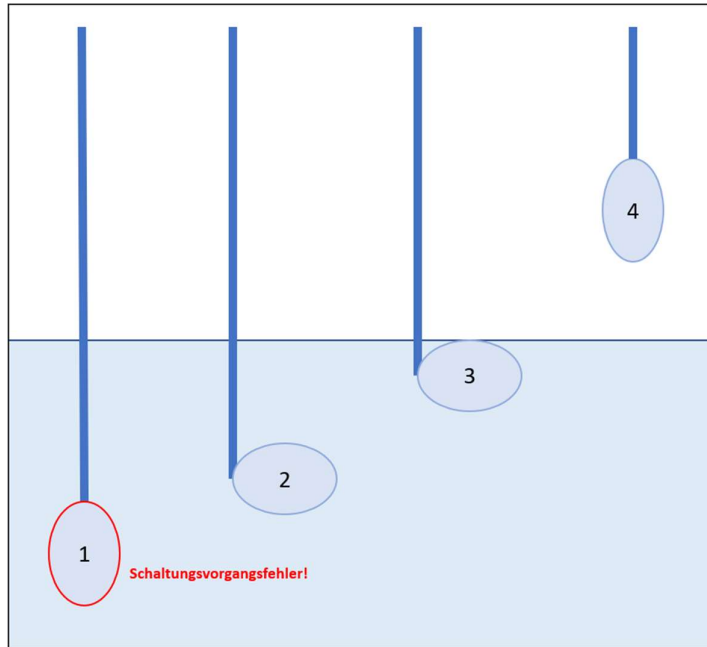


Abbildung 25 Schaltungsvorgangsfehler

Der Schwimmerschalter 1 müsste auch schwimmend sein und nicht hängend.

5.4.2 Drahtbruch Schwimmerschalter

Wird der Wechselkontakt im Schwimmerschalter bei einer Schaltung nicht sauber umgeschaltet oder schaltet gar nicht, wird dies in der Abwassersteuerung als Drahtbruch erkannt. Der Drahtbruchfehler wird in der Abbildung 24 aufgezeigt.

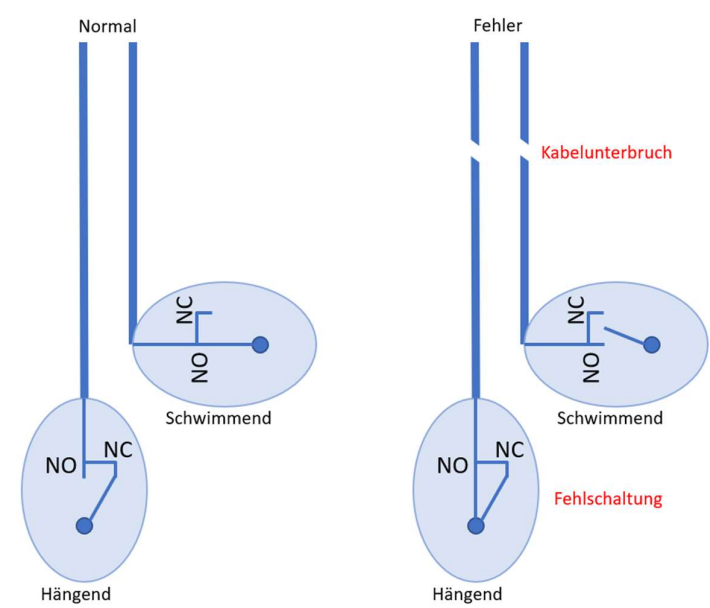


Abbildung 26 Drahtbruch Fehler

Um die Fehlermöglichkeiten aufzulisten, wurde in der nachfolgenden Abbildung die Wahrheitstabelle erstellt.

Position	Öffner Kontakt NC	Schliesser Kontakt NO
Korrekte Schaltung		
hängend	1	0
schwimmend	0	1
Fehlerhafte Schaltung		
hängend	0	0
schwimmend	0	1
hängend	0	0
schwimmend	1	1
hängend	1	0
schwimmend	1	1
hängend	1	0
schwimmend	0	0
hängend	1	1
schwimmend	0	0
hängend	1	1
schwimmend	0	1

Abbildung 27 Drahtbruch Wahrheitstabelle

5.5 Business-Logic

Business-Logic, zu Deutsch Geschäftslogik, Anwendungslogik. Bezeichnet die Abgrenzung der durch die Aufgabenstellung selbst motivierten Logik eines Softwaresystems.¹⁵ Der HiL-Simulator soll nur eine Schnittstelle zwischen DUT und der Test-Infrastruktur bilden und dabei keine logischen Testabläufe abbilden oder irgendwelche Zustände des Systems interpretieren.

5.6 Programmierschnittstelle

Die Abwassersteuerung verfügt über ein JTAG-Programmier-Interface. Joint Test Action Group (kurz JTAG), entwickelte den Standard IEEE 1149.1. Das JTAG-Protokoll ermöglicht das Programmieren, Debuggen und Testen von ICs und Prozessoren direkt in der Schaltung.¹⁶ Dieses wird für die Programmierung der Firmware der Abwassersteuerung benutzt. Die Programmierung erfolgt dabei über eine spezifische Sonde (SEGGER J-Link). In der Abbildung 26 ist ein SEGGER J-Link Ultra+ von der Entwicklungsabteilung abgebildet.



Abbildung 28 SEGGER J-Link Ultra+

¹⁵ (Wikipedia, 2019)

¹⁶ (mikrocontroller.net wiki, 2020)

Er wird via USB an einen PC/Laptop etc. angeschlossen. Auf der anderen Seite wird der 20 polige Stecker direkt auf der Platine mit der entsprechenden Buchse verbunden. Das Pinout ist der Abbildung 27 zu entnehmen.

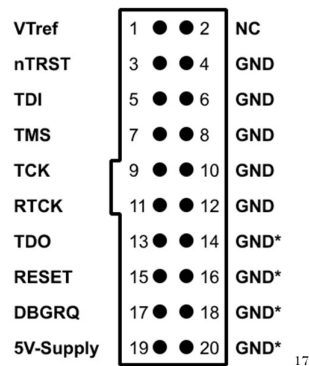


Abbildung 29 SEGGER J-Link Pinout

Mit dem beschriebenen Setup kann der Microcontroller der Abwassersteuerung programmiert werden. Angesteuert wird der SEGGER J-Link dabei via Kommandozeilenbefehle. Das Hex File des kompilierten Codes wird hierbei mit dem SEGGER J-Link in den Microcontroller programmiert. Der HiL-Simulator soll dabei eine Controller Aufgabe übernehmen.

5.7 Parametrierschnittstelle

Es wird möglich sein, die Abwassersteuerung mithilfe einer Parametrierschnittstelle zu parametrieren. Dazu wird eine serielle Verbindung (UART) eingesetzt, über die durch das «ProtoBuf»-Protokoll kommuniziert wird. Da die Firmware immer die gleiche ist, geht es darum, der Steuerung die vorliegenden Parameter mitzuteilen. In der Abbildung 28 sind die wichtigsten Parameter tabellarisch dargestellt. Die Liste ist nicht abschliessend und dient nur zur Veranschaulichung.

Parameter Abwassersteuerung	
Anzahl Pumpen	1 oder 2
Motorstrom	z.B. 5A
Schwimmerschalter	Ja / Nein
Anzahl Schwimmer	z.B 3
Niveausonde	Ja / Nein
System	Schacht / Reservoir
OSK	Ja / Nein
Externe Sperrung	Ja / Nein

Abbildung 30 Parameter Abwassersteuerung

Der HiL-Simulator soll anhand von Angaben der Test-Infrastruktur das Abwassersteuergerät parametrieren können.

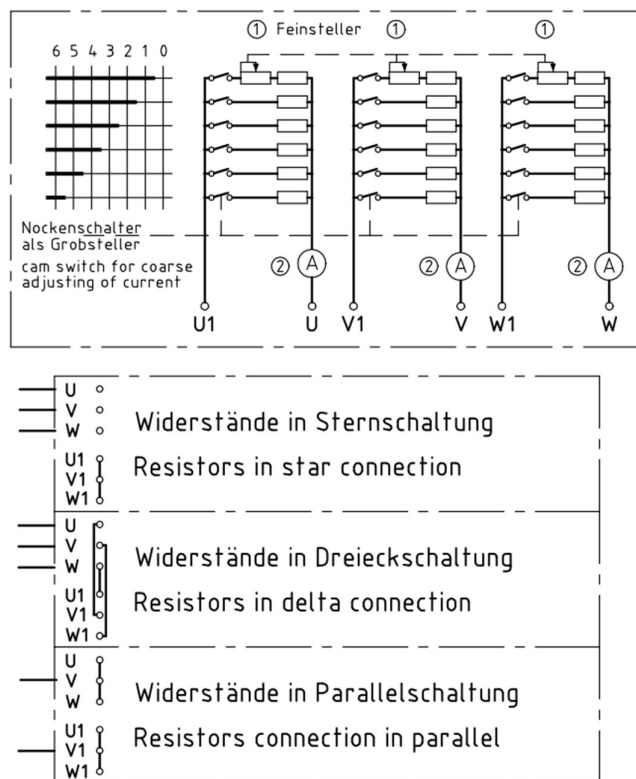
¹⁷ (SEGGER, 2020)

5.8 Motoren simulieren

Mit einem externen Lastwiderstand kann ein Motor simuliert werden. Dies bildet den Betrieb einer Pumpe ab. Da die Abwassersteuerung über ein breites Leistungsspektrum verfügen wird, muss also ein Lastwiderstand eingesetzt werden, der einstellbar ist. In der Abbildung 29 ist eine Beispielschaltung eines Lastwiderstandes zu finden.

Schaltung : B3 - 6 - 6 Dreiphasig, 6 Stufen, 6 Klemmen

Wiring : B3 - 6 - 6 Threephase, 6 steps, 6 terminals



18

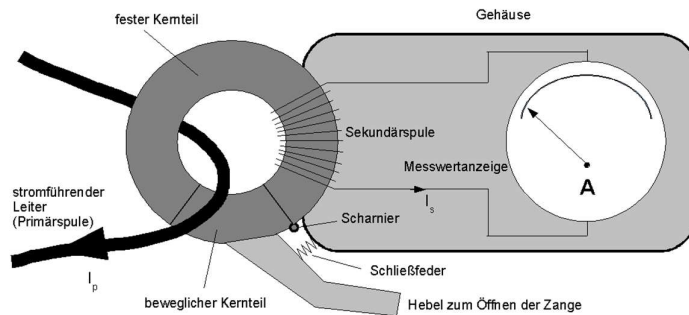
Abbildung 31 Beispielschaltung Lastwiderstand

¹⁸ (FRIZLEN, 2001)

5.9 Strommessung

Auf der Abwassersteuerung wird die Stromaufnahme der Motoren/Pumpen überwacht. Um den gemessenen Strom der Steuerung prüfen zu können, soll der HiL-Simulator auch über diese Möglichkeit verfügen. Die Strommessung kann auf verschiedene Arten durchgeführt werden. Die wohl bekannteste Art, welche auch in Multimetern anzutreffen ist, ist die Messung des Spannungsabfalls über einem Messwiderstand (Shunt). Mit Hilfe des Ohm'schen Gesetzes wird dann der Strom errechnet.

Eine alternative Methode zur Strommessung wird mit Hilfe von Stromwandlern durchgeführt. Ein Stromwandler ist im Grunde genommen ein Transformator, welcher den Primärstrom in ein kleineres elektrisches Signal umwandelt. Dies wird zum Beispiel auch bei Zangenstrommessern eingesetzt. In der nachfolgenden Abbildung wird dies veranschaulicht.



$$\text{Strom-/ Windungszahlenverhältnis: } I_p : I_s = N_s : N_p$$

Abbildung 32 Beispielschaltung Zangenstrommesser

6 Konzept

Um die Aufgaben des HiL-Simulators abzugrenzen, wurde zuerst eine Software Testpyramide erstellt. In der Abbildung 31 ist diese ersichtlich.

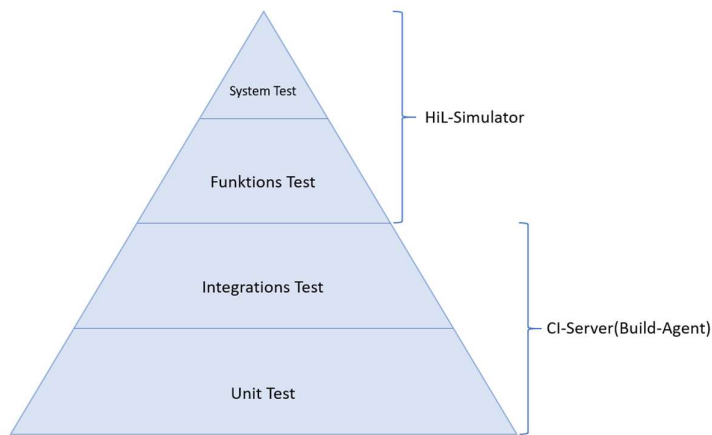


Abbildung 33 Software Testpyramide

Der CI-Server übernimmt den Unit Test und den Integrations Test. Dies wurde bereits durch das Entwicklerteam automatisiert. Der HiL-Simulator übernimmt den Funktionstest und den System Test.

Aufbauend auf der Vorstudie wurde ein Konzept gezeichnet, welches in der nachfolgenden Abbildung dargestellt wird.

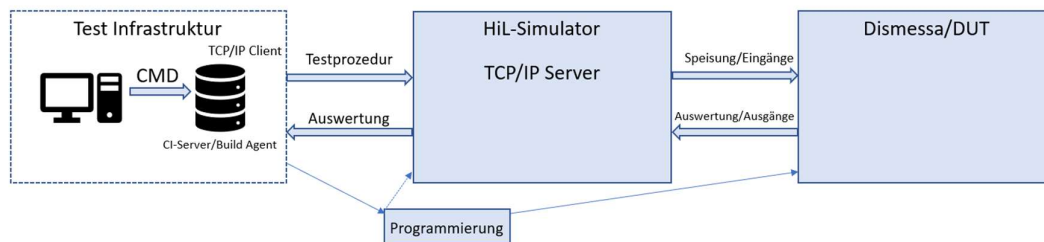


Abbildung 34 Konzept Skizze

Der Entwickler kann über die Windows Shell mit Kommandozeilenbefehlen den Build-Agenten ansprechen. Über TCP/IP wird die Testprozedur dem HiL- Simulator übergeben. Dieser kann die Eingänge gemäss Vorgaben simulieren (z.B. Niveau Ein) und wertet die Ausgänge entsprechend aus (z.B. Hat die Pumpe eingeschaltet?). Die Auswertung wird zurück an den Build-Agenten gesendet.

In der Skizze wird auch die Programmierung der Abwassersteuerung gezeigt, welche zuerst noch direkt erfolgen muss. Es ist ein optionales Ziel, dies auch über den HiL-Simulator zu machen.

6.1 Vergleich manuell zu automatisiert

Bis jetzt wurde in der Entwicklung nur der Unit Test und der Integrations Test automatisiert. Funktions- und Systemtests wurden manuell und nur sporadisch durchgeführt, da sie viel Zeit und Aufwand benötigen. Dies soll sich mit Hilfe des HiL-Simulators ändern, um den Entwicklern eine erweiterte Sicherstellung zu geben, dass der Code den sie geschrieben haben mit der Hardware funktioniert. Gleichzeitig können so die Entwicklungszeiten verkürzt und Kosten gespart werden. In der nachfolgenden Abbildung sind die einzelnen Schritte bei den manuellen Tests ersichtlich.

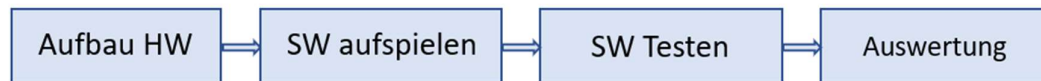


Abbildung 35 Vorgehen manuell

In der nächsten Abbildung sind die Schritte des automatisierten Tests dargestellt.



Abbildung 36 Vorgehen automatisiert

Da die Hardware nicht jedes Mal aufgebaut werden muss, entfällt dieser Schritt. Gleichzeitig wird auch das Testen mit dem HiL-Simulator automatisiert und entlastet somit die Entwickler. Damit die zwei Vorgehen noch besser miteinander verglichen werden können, werden die Zeiten für die Arbeitsschritte tabellarisch in der Abbildung 35 & 36 aufgezeigt.

Arbeitsschritt	Aufwand min.	Aufwand max.
Aufbau Hardware	4.00 h	8.00 h
Software aufspielen	0.25 h	0.50 h
Software testen	2.00 h	4.00 h
Auswertung	0.25 h	0.50 h
Total	6.50 h	13.00 h

Abbildung 37 Zeiten Test manuell

Arbeitsschritt	Aufwand min.	Aufwand max.
Software aufspielen	0.25 h	0.50 h
HiL-Simulator	0.25 h	1.00 h
Total	0.50 h	1.50 h

Abbildung 38 Zeiten Test automatisiert

Beim Vergleich der beiden Vorgehen ist zu erkennen, dass mit dem HiL-Simulator bei jedem Test sechs bis elfeinhalb Stunden eingespart werden können.

Nach Angabe des Vorgesetzten werden die Kosten für einen Mitarbeiter der Entwicklung mit CHF 100.- pro Stunde berechnet.

Mit diesen Angaben wurden die Kosteneinsparnisse berechnet und tabellarisch in Abbildung 30 dargestellt.

Die Formel dazu ist wie folgt: Zeitersparnis Max/Min x Kosten Mitarbeiter = Ersparnis pro Test

	Einsparnisse min.		Einsparnisse max.	
Zeitersparnis	6.00	h	11.50	h
Kosten MA pro Stunde	100	Chf	100	Chf
Total	600	Chf	1150	Chf

Abbildung 39 Kosteneinsparnisse

Es ist klar ersichtlich, dass hier viel Zeit und hohe Kosten eingespart werden können. Gleichzeitig gibt es dem Entwickler eine erweiterte Sicherstellung, dass sein Code mit der Hardware funktioniert. Dazu kommt, dass er die gewonnene Zeit verwenden kann, um weitere Entwicklungen zu erarbeiten. Somit werden die Entwicklungszeiten verkürzt.

6.2 Lösungsvarianten

Es gibt verschiedene Lösungsansätze, um die Anforderungen an den HiL-Simulator abzudecken. Um die bestmögliche Lösungsvariante zu finden wurde eine Nutzwertanalyse durchgeführt. In dieser wurden folgende Kriterien aufgelistet, gewichtet und eine Note vergeben:

- Flexibilität/Erweiterung
- Zeiteinsparung/Effizienz
- Schnittstellen/Kommunikation
- Datenhandling
- Investition
- Fehleranfälligkeit

In der Abbildung 38 wurde die Nutzwertanalyse tabellarisch dargestellt.

Kriterium	Gewicht	Manuell		LabView		Arduino		SPS	
		Note	N*Gew.	Note	N*Gew.	Note	N*Gew.	Note	N*Gew.
Flexibilität/Erweiterung	7	9	63	8	56	6	42	9	63
Zeiteinsparung/Effizienz	9	0	0	9	81	7	63	7	63
Schnittstellen/Kommunikation	9	0	0	9	81	5	45	4	36
Datenhandling	7	3	21	8	56	4	28	4	28
Investition	6	10	60	4	24	9	54	5	30
Fehleranfälligkeit	6	6	36	9	54	6	36	7	42
Total	44		180		352		268		262

Abbildung 40 Nutzwertanalyse

6.3 Lösungsentscheid

Durch die Nutzwertanalyse wurde klar, dass für die Diplomarbeit LabVIEW mit 352 erreichten Punkten am besten geeignet ist. LabVIEW ist eine Systementwicklungssoftware von National Instruments für Anwendungen, die Mess-, Prüf-, Steuer- und Regelungsschritte mit schnellem Zugriff auf Hardware und Erkenntnisse aus Daten erfordern.¹⁹ Aus diesem Grund wurde es für diese Diplomarbeit ausgewählt. Da der Diplomand noch keine Vorkenntnisse in Bezug auf LabVIEW-Programmierung hat, besuchte er im Vorfeld der Diplomarbeit zwei Online-Grundkurse (LabVIEW Core 1&2), um sich die Grundkenntnisse anzueignen.

¹⁹ (National Instruments, 2020)

7 Realisierung

Nachdem die Vorstudie abgeschlossen und das Konzept erarbeitet wurde, folgt die eigentliche Realisierung des Projekts. Diese wird in zwei Bereiche unterteilt: Die Hardware und die Software.

7.1 Hardware

Dieser Abschnitt befasst sich mit der Analyse und der Evaluation der einzelnen Schnittstellen und Schaltungsteilen. Es muss dabei beachtet werden, die Komponenten möglichst bei bestehenden Biral-Lieferanten zu beziehen und nicht bei jeder Komponente eine neue Evaluation durchzuführen.

7.1.1 Mechanischer Aufbau

Als Gehäuse für den HiL-Simulator, dient ein von der Biral AG nicht mehr verwendetes Gehäuse der Firma Swibox. Das Gehäuse entspricht der Schutzart IP54 und somit den im Pflichtenheft definierten Anforderungen. Dies bedeutet, dass es gegen Staub, Berührung und Spritzwasser geschützt ist. Die Abmessungen des Gehäuses sind 50x40x21cm (BxHxT). In der nachfolgenden Abbildung ist das Swibox Gehäuse dargestellt.

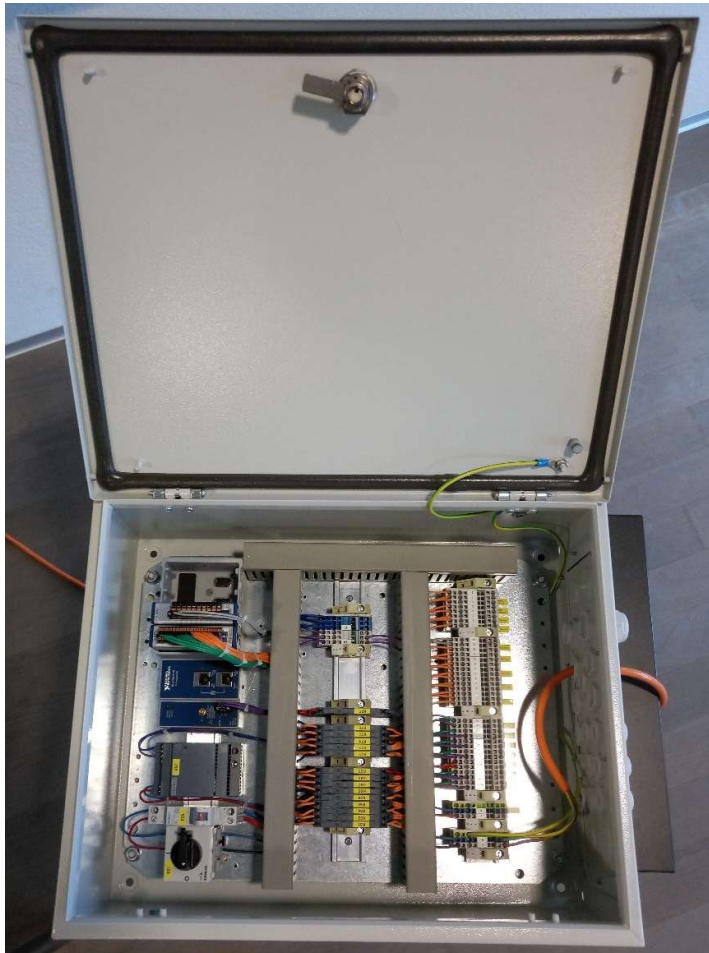


Abbildung 41 Swibox Gehäuse

7.1.2 Spannungsversorgung

Die Spannungsversorgung der Abwassersteuerung ist auf 230VAC festgelegt worden und muss über den HiL-Simulator geschaltet werden können. Die übrigen Komponenten wurden auf eine Spannung von 24VDC ausgelegt. Die Spannungsversorgung kann der Abbildung 40 entnommen werden.

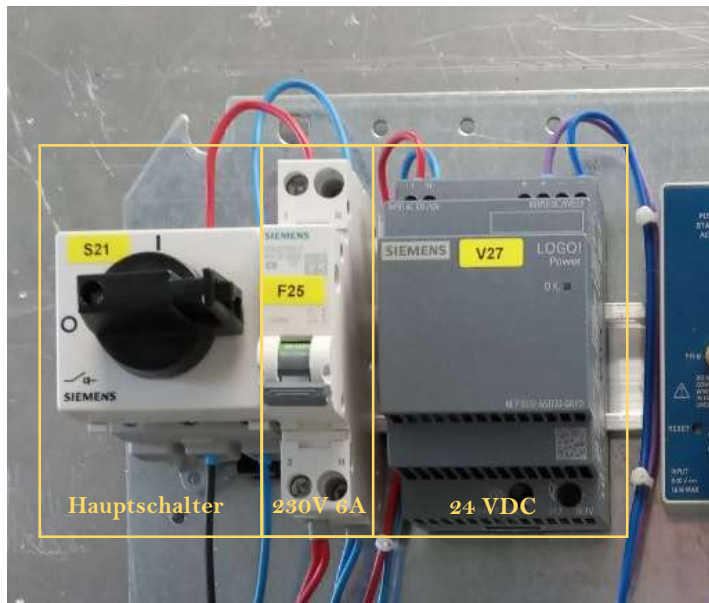


Abbildung 42 Spannungsversorgung

7.1.3 Schnittstellen zu Abwassersteuergerät

Der HiL-Simulator muss alle Schnittstellen der Abwassersteuerung abdecken. In der nachfolgenden Abbildung sind die erforderlichen Schnittstellen dargestellt.



Abbildung 43 Schnittstellen Abwassersteuerung

Spezifikation

Die digitalen Ausgänge dürfen maximal mit 250VAC / 6A beschalten werden, die Ausgänge sind potentialfrei.

Die analogen Ausgänge müssen galvanisch getrennt ausgewertet werden.

Für die analogen Eingänge wird ein 4–20 mA Signal benötigt.

Die digitalen Eingänge der Steuerung müssen potentialfrei geschalten werden. An den Ausgangsklemmen liegt eine Spannung von 24 VDC an. Bei geschlossenem Kontakt fließt ein Strom von 5 mA.

Die Kontakte werden mit Kopplungsrelais von Siemens geschalten. Sie wurden wegen ihrer kleinen Bauform und weil sie schon in anderen Biral Projekten verwendet wurden ausgewählt.

Spezifikation des Relais:

Siemens Koppelrelais: RQ3018-2AB01

Steuerspannung: 24V AC/DC

Betriebsstrom: 1.0 A

Das Kopplungsrelais wird in der Abbildung 42 dargestellt.



20

Abbildung 44 Kopplungsrelais Siemens

Das Relais wird auch als Variante mit 230VAC Steuerspannung (3RQ3018-2AF00) eingesetzt, um den analogen Ausgang der Abwassersteuerung zu erfassen (230VAC Alarmhorn).

²⁰ (Siemens, 2020)

7.1.4 Schnittstellen zu Test-Infrastruktur

Die Schnittstelle zur bestehenden Test-Infrastruktur (Build Agent) erfolgt mit einem Ethernetkabel, welches direkt an das NI Chassis angeschlossen wird. Auf die NI Hardware wird im nächsten Abschnitt eingegangen.

7.1.5 National Instruments DAQ System

Im Abschnitt 6.3 wurde die Variante mit LabVIEW ausgewählt. Da es sich bei LabVIEW um die Entwicklungssoftware handelt, braucht es für ein System noch die entsprechende Hardware. Diese kann auch bei National Instruments bezogen werden. Die gewählte Hardware wird in den folgenden Abschnitten genauer erläutert.

NI cDAQ-9185 CompactDAQ Chassis

Als Chassis wurde ein cDAQ-9185 gewählt. Es steuert das Timing, die Synchronisierung und die Datenübertragung zwischen einem externen Host und I/O-Modulen der C-Serie von NI.²¹ Es bietet zwei Ethernet-Anschlüsse und vier Slots für verschiedene I/O-Module. In der Abbildung 43 ist das Chassis ersichtlich.



Abbildung 45 NI cDAQ-9185 CompactDAQ Chassis

²¹ (National Instruments, 2020)

NI 9375 Modul für digitale Signale

Das NI 9375 ermöglicht digitale Ein- und Ausgabefunktionen auf CompactDAQ Systemen.²² Es bietet 16 digitale Eingänge und 16 digitale Ausgänge. In der nachfolgenden Abbildung ist es ersichtlich.



Abbildung 46 NI 9375 Modul für digitale Signale

NI 9265 Stromausgabemodul

Das NI 9265 ermöglicht analoge Ausgabefunktionen auf CompactDAQ Systemen. Es bietet 4 Kanäle, welche jeweils ein 0-20mA Signal liefern können und eine Auflösung von 16Bit hat. Es ist in der Abbildung 45 dargestellt.



Abbildung 47 NI 9265 Stromausgabemodul

²² (National Instruments, 2020)

7.1.6 Übersicht der Hardware

Die verdrahtete Hardware ist in der nachfolgenden Abbildung ersichtlich.

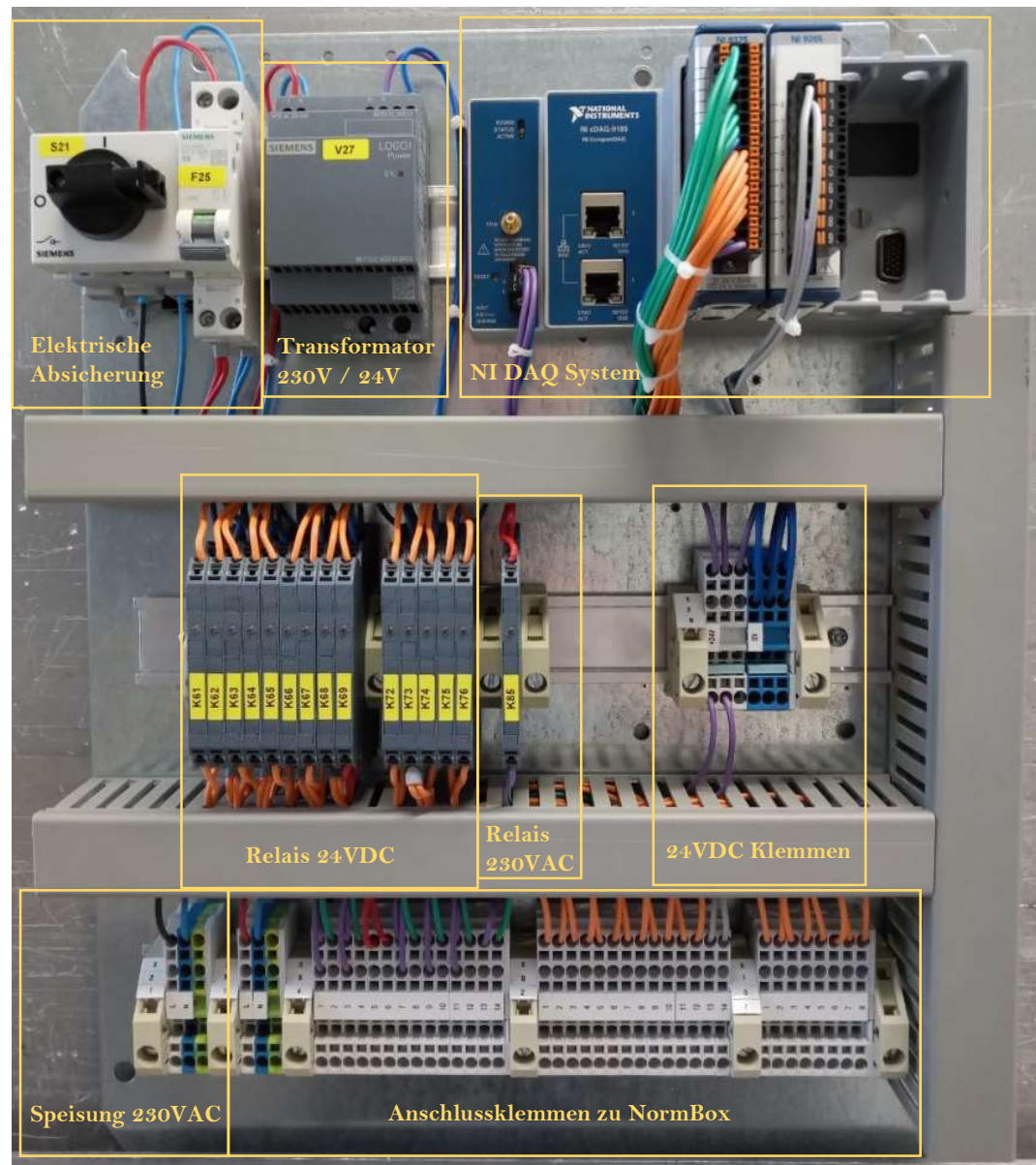


Abbildung 48 Übersicht der Hardware

7.1.7 Anschluss an die NormBox

Die Anschlüsse an die Abwassersteuerung wurden auf Federzugklemmen geführt, um das Anschliessen zu erleichtern. Alle Klemmen wurden nummeriert und beschriftet, um das korrekte Anschliessen gewährleisten zu können. Die Anschlüsse sind der Abbildung 47 zu entnehmen.

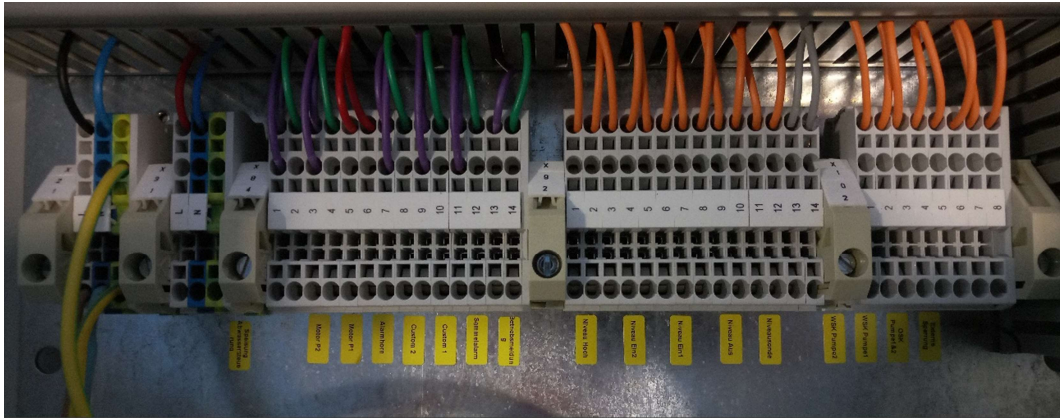


Abbildung 49 Anschluss an die NormBox

7.2 Elektroschema

Um die Verdrahtung des HiL-Simulators zu erleichtern und im späteren Gebrauch Fehler schneller identifizieren zu können, wurde ein Elektroschema mit der verbauten Hardware gezeichnet. In der nachfolgenden Abbildung ist eine Übersicht der gezeichneten Blätter des Schemas zu finden. Im Anhang (A.4 Elektroschema) befindet sich das komplette Schema.

Inhaltsverzeichnis
Einspeisung
Speisung Abwassersteuerung
NI cDAQ-9185
cDAQ Ein-/Ausgänge
DO0-8
DO9-12
DI0-7
Niveauregler
Anschlüsse für Abwassersteuerung

Abbildung 50 Aufbau Elektroschema

7.3 Software

Dieser Abschnitt befasst sich mit der Planung und Entwicklung der Software des HiL-Simulators. Die Entwicklung wird in einzelne Etappen aufgeteilt, damit Zwischenziele erreicht und somit Schritt für Schritt die Software erarbeitet werden kann. Die Etappen bauen aufeinander auf, weshalb sie während der Entwicklung auch bereits getestet werden müssen. Dadurch werden Fehler frühzeitig erkannt und können behoben werden.

7.3.1 Initialisierung

Damit die Software immer aus dem gleichen Grundzustand gestartet wird, wird eine Initialisierung durchgeführt. In dieser werden die Ausgänge der Steuerung auf null gesetzt. Somit ist gewährleistet, dass beim Softwarestart immer die gleichen Bedingungen anzutreffen sind.

7.3.2 Schnittstellen zu Abwassersteuergerät

Um gleichzeitig auch die Hardware auf ihre Funktionalität testen zu können, werden zuerst die Schnittstellen zur Abwassersteuerung programmiert. Damit effizient und lösungsorientiert programmiert werden kann, wurde zuerst ein Blockdiagramm erarbeitet. In der nachfolgenden Abbildung ist dies dargestellt.

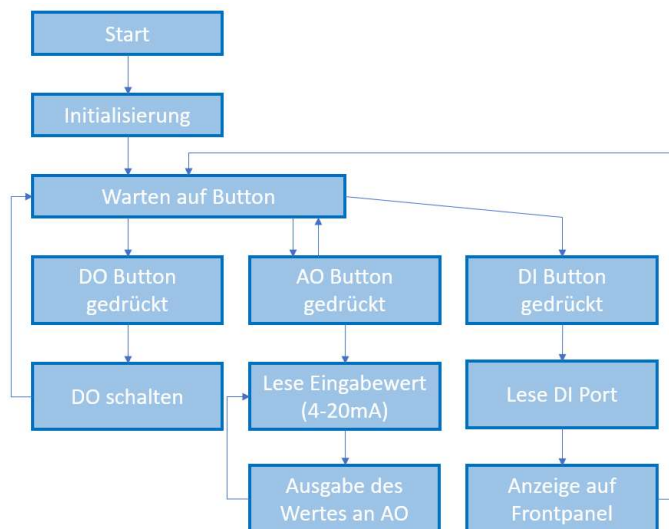


Abbildung 51 Blockdiagramm Schnittstellen zu Abwassersteuergerät

Die Software konnte umgesetzt und getestet werden. Die komplette Hardware läuft einwandfrei und braucht keine Nacharbeit. Somit steht dem weiteren Vorgehen nichts mehr im Weg. Das LabVIEW Frontpanel der geschriebenen Software ist in der Abbildung 50 dargestellt.

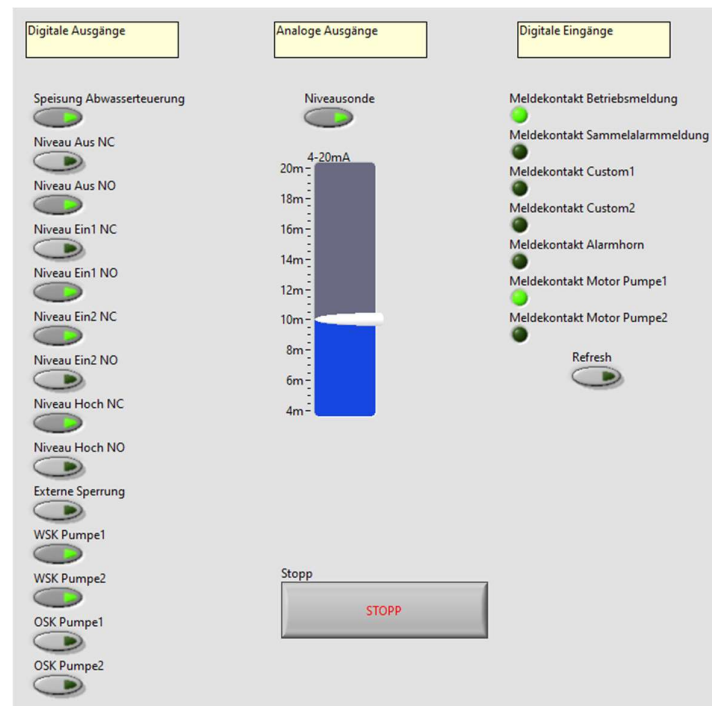


Abbildung 52 LabVIEW Frontpanel Schnittstellen zu Abwassersteuergerät

Die digitalen Ausgänge wurden mit Schaltern auf dem Frontpanel verknüpft. Wird einer der Schalter betätigt, wird der entsprechende Ausgang geschaltet und damit das jeweilige Relais angesprochen. Der analoge Ausgang wird mit einem Schalter ein- und ausgeschaltet. Wenn er eingeschaltet ist, wird der eingestellte Wert am Schieber ausgelesen und am analogen Ausgang ausgegeben. Die digitalen Eingänge wurden zuerst dauerhaft, das heisst in einem Loop, eingelesen. Da das Programm aber so sehr träge wurde, schuf die Programmierung eines Refresh Tasters Abhilfe. Nun werden die digitalen Eingänge nur noch ausgelesen, wenn dieser betätigt wird.

Das LabVIEW Blockdiagramm zu der geschriebenen Software ist der nachfolgenden Abbildung zu entnehmen.

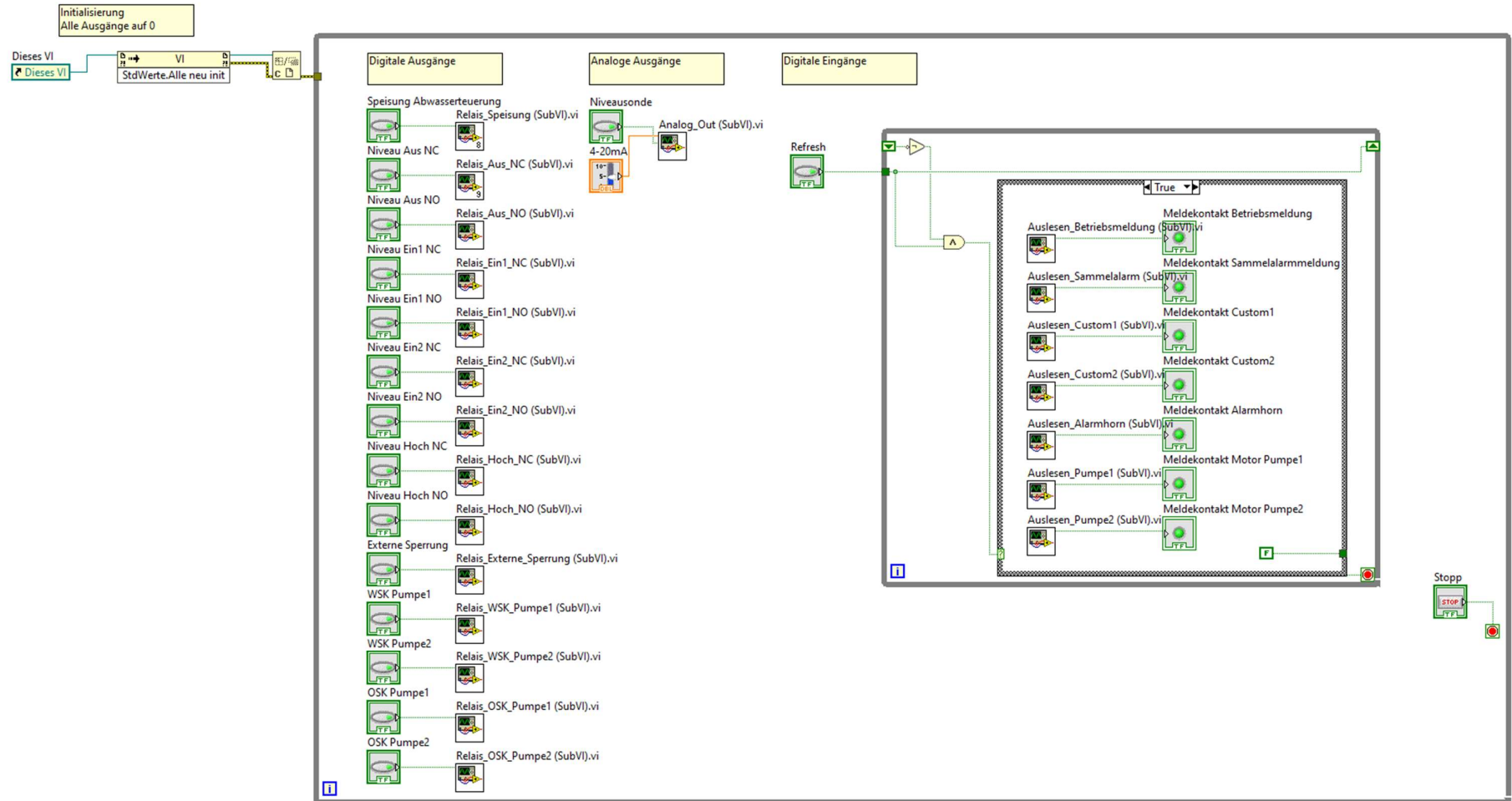


Abbildung 53 LabVIEW Blockdiagramm Schnittstellen zu Abwassersteuergerät

Das Schalten der Ausgänge und das Lesen der Eingänge wurden jeweils in SubVI's verpackt, damit die Software übersichtlich bleibt.

7.3.3 Schnittstellen zu Test-Infrastruktur

Auch für die Programmierung der Schnittstellen zur bestehenden Test-Infrastruktur wurde zuerst ein Blockdiagramm erstellt. Dieses ist der Abbildung 52 zu entnehmen.



Abbildung 54 Blockdiagramm Schnittstellen zu Test-Infrastruktur

Da der HiL-Simulator die Rolle als TCP/IP Server übernimmt, wurde zuerst dieser programmiert. In der nachfolgenden Abbildung ist das LabVIEW-Blockdiagramm des Verbindungsaufbaus des Servers ersichtlich.

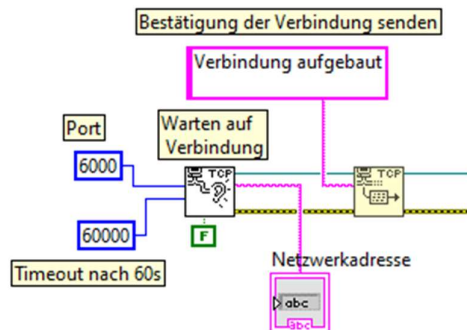


Abbildung 55 LabVIEW-Blockdiagramm Verbindungsaufbau des Servers

Der Port wurde auf 6000 eingestellt und der Server stoppt nach 60 Sekunden ohne Verbindung mit einem Timeout. Wird jedoch eine Verbindung hergestellt, wird dies vom Server bestätigt.

Um den Verbindungsaufbau zu testen, wird das freie Softwaretool «PuTTY» verwendet. Dieses dient als Client und kann eine Verbindung zu einem Server aufbauen, indem die IP-Adresse und der Port angegeben werden. In der nachfolgenden Abbildung wird die Konfigurationsseite des Tools dargestellt.

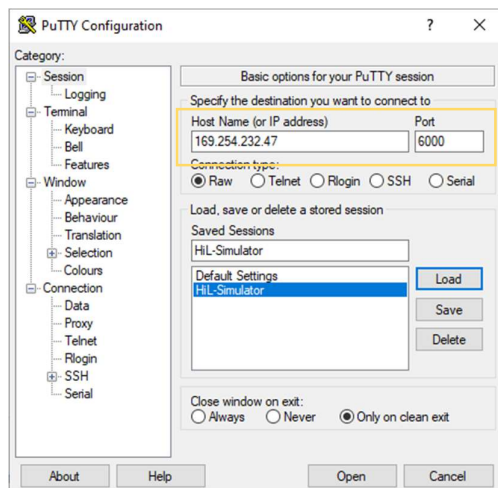


Abbildung 56 PuTTY Konfigurationsseite

Mit einem Klick auf Open wird die Verbindung aufgebaut und es öffnet sich ein Terminalfenster.

Ist die Verbindung erfolgreich, sendet der TCP/IP Server eine Bestätigung. Das Terminalfenster mit der erwähnten Bestätigung ist in der nachfolgenden Abbildung ersichtlich.



Abbildung 57 PuTTY Terminalfenster

Im Terminalfenster können ASCII-Zeichen eingegeben werden und mit der Entertaste werden diese an den Server gesendet.

Da die Anzahl an empfangenen Bytes flexibel bleiben muss, wartet der Server bis er ein CRLF (Zeilenumbruch) detektiert und gibt die vorangehenden Bytes bis und mit dem CRLF anschliessend weiter. Das Lesen der empfangenen Zeichen ist in der Abbildung 56 dargestellt.

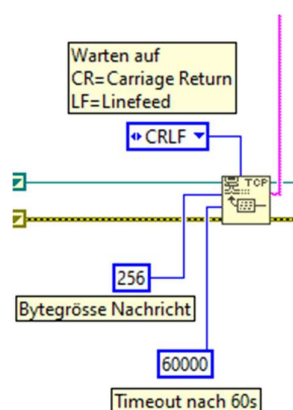


Abbildung 58 LabVIEW Lesen der empfangenen Zeichen

Die maximale Bytegrösse der Nachricht wurde auf 256 Bytes festgelegt. Auch hier wurde wieder eine Timeoutfunktion implementiert, die den Server nach 60 Sekunden ohne den Empfang einer Nachricht beendet.

In der Abbildung 57 werden die Steuerzeichen genauer erläutert.

Zeichensatz	Abkürzung	Beschreibung	Zeichenkombination die nicht als Text erkannt wird
ASCII	CR	Carriage Return, in Deutsch «Wagenrücklauf»	\r
	LF	Line Feed, in Deutsch «Zeilenvorschub»	\n
	CRLF	Zeilenumbruch	\r\n

Abbildung 59 ASCII Steuerzeichen

Mit der Entertaste wird im Terminalfenster jeweils ein Zeilenumbruch an den Server gesendet. Somit ist das Befehlssende immer ein CRLF.

Kommunikationsprotokoll

Damit die Kommunikation im späteren Gebrauch möglichst einfach ist, wurde zusammen mit den Entwicklern ein Kommunikationsprotokoll erarbeitet. In der nachfolgenden Abbildung ist der Befehlsaufbau anhand eines Beispiels dargestellt.

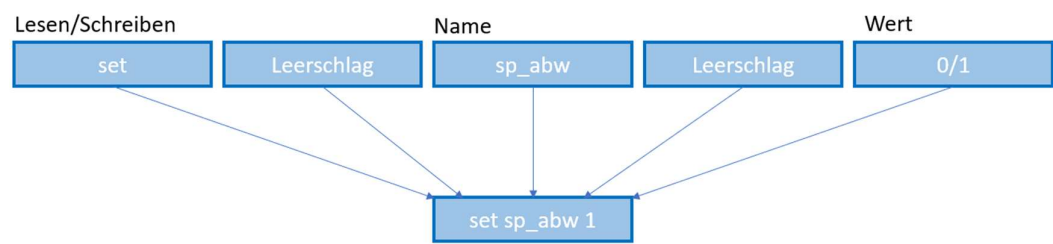


Abbildung 60 Beispiel Befehlsaufbau

Nach oben dargestelltem Aufbau wurde eine Tabelle mit dem Kommunikationsprotokoll erstellt. Dieses ist der Abbildung 59 zu entnehmen.

Kommunikationsprotokoll						
Beschreibung	Anschluss (HiL-Simulator)	Lesen/Schreiben	Name	Wert	Befehl(Beispiel)	Bestätigung (ACK)
Digitale Eingänge	Meldekontakt Betriebsmeldung	get	mk_betr	x	get mk_betr	mk_betr 1
Digitale Eingänge	Meldekontakt Sammelalarm	get	mk_sam	x	get mk_sam	mk_sam 0
Digitale Eingänge	Meldekontakt Custom1 Steuerung	get	mk_cu1	x	get mk_cu1	mk_cu1 1
Digitale Eingänge	Meldekontakt Custom2 Steuerung	get	mk_cu2	x	get mk_cu2	mk_cu2 1
Digitale Eingänge	Meldekontakt Alarmhorn	get	mk_ala	x	get mk_ala	mk_ala 1
Digitale Eingänge	Meldekontakt Motor Pumpe1	get	mk_mo_p1	x	get mk_mo_p1	mk_mo_p1 1
Digitale Eingänge	Meldekontakt Motor Pumpe2	get	mk_mo_p2	x	get mk_mo_p2	mk_mo_p2 1
Digitale Ausgänge	Speisung Abwassersteuerung ON/OFF	set	sp_abw	0/1	set sp_abw 1	ACK sp_abw 1
Digitale Ausgänge	Niveau Aus NC	set	ni_aus_nc	0/1	set ni_aus_nc 0	ACK ni_aus_nc 0
Digitale Ausgänge	Niveau Aus NO	set	ni_aus_no	0/1	set ni_aus_no 1	ACK ni_aus_no 1
Digitale Ausgänge	Niveau Ein1 NC	set	ni_ein1_nc	0/1	set ni_ein1_nc 1	ACK ni_ein1_nc 1
Digitale Ausgänge	Niveau Ein1 NO	set	ni_ein1_no	0/1	set ni_ein1_no 1	ACK ni_ein1_no 1
Digitale Ausgänge	Niveau Ein2 NC	set	ni_ein2_nc	0/1	set ni_ein2_nc 1	ACK ni_ein2_nc 1
Digitale Ausgänge	Niveau Ein2 NO	set	ni_ein2_no	0/1	set ni_ein2_no 1	ACK ni_ein2_no 1
Digitale Ausgänge	Niveau Hoch NC	set	ni_hoch_nc	0/1	set ni_hoch_nc 1	ACK ni_hoch_nc 1
Digitale Ausgänge	Niveau Hoch NO	set	ni_hoch_no	0/1	set ni_hoch_no 1	ACK ni_hoch_no 1
Digitale Ausgänge	Externe Sperrung	set	ext_sper	0/1	set ext_sper 1	ACK ext_sper 1
Digitale Ausgänge	WSK Pumpe1	set	wsk_p1	0/1	set wsk_p1 1	ACK wsk_p1 1
Digitale Ausgänge	WSK Pumpe2	set	wsk_p2	0/1	set wsk_p2 1	ACK wsk_p2 1
Digitale Ausgänge	OSK Pumpe1	set	osk_p1	0/1	set osk_p1 1	ACK osk_p1 1
Digitale Ausgänge	OSK Pumpe2	set	osk_p2	0/1	set osk_p2 1	ACK osk_p2 1
Analoge Ausgänge	Niveausonde 4-20mA	set	ni_sonde	0.00 bis 20.00	set ni_sonde 17.25	ACK ni_sonde 17.25mA

Abbildung 61 Kommunikationsprotokoll

Nachdem der TCP/IP-Server programmiert und das Kommunikationsprotokoll erarbeitet wurde, folgt die Verknüpfung mit der in Abschnitt 7.3.3 geschriebenen Software zur Ansteuerung der Abwassersteuerung. Die Befehle des Kommunikationsprotokolls wurden dafür unterteilt in «set» und «get» Befehle.

Set-Befehle

Wird ein set-Befehl detektiert, landet der Befehlsstring in der set Case Struktur. Dort wird er weiter aufgeschlüsselt und der entsprechende Ausgang wird, entsprechend dem mitgegebenen Wert, ein- oder ausgeschaltet. Danach wird zur Bestätigung ein ACK (Acknowledgement) an den Client geschickt. In der nachfolgenden Abbildung ist dies anhand eines Beispielbefehls ersichtlich.

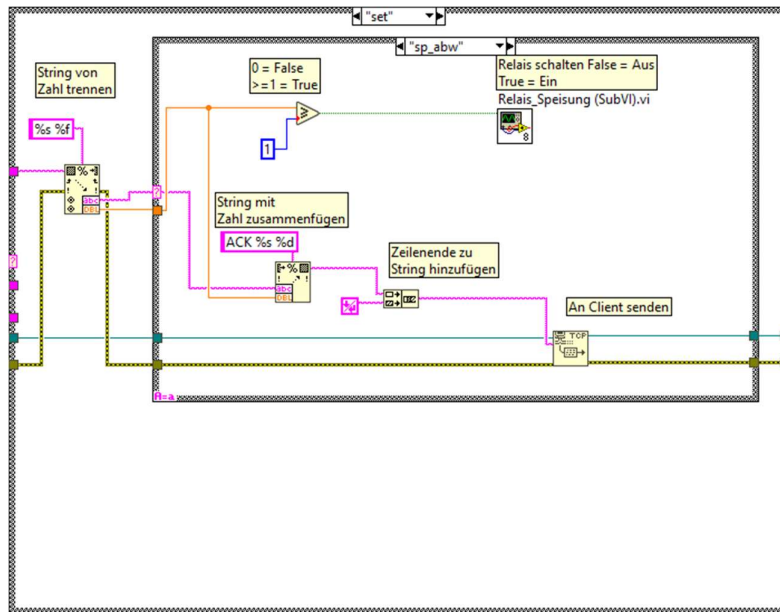


Abbildung 62 LabVIEW Set Befehl Beispiel

Get-Befehle

Wird ein get-Befehl detektiert, landet der Befehlsstring in der get Case Struktur. Dort wird der entsprechende Eingang gelesen und der Wert an den Client geschickt. Dies ist der Abbildung 61 zu entnehmen.

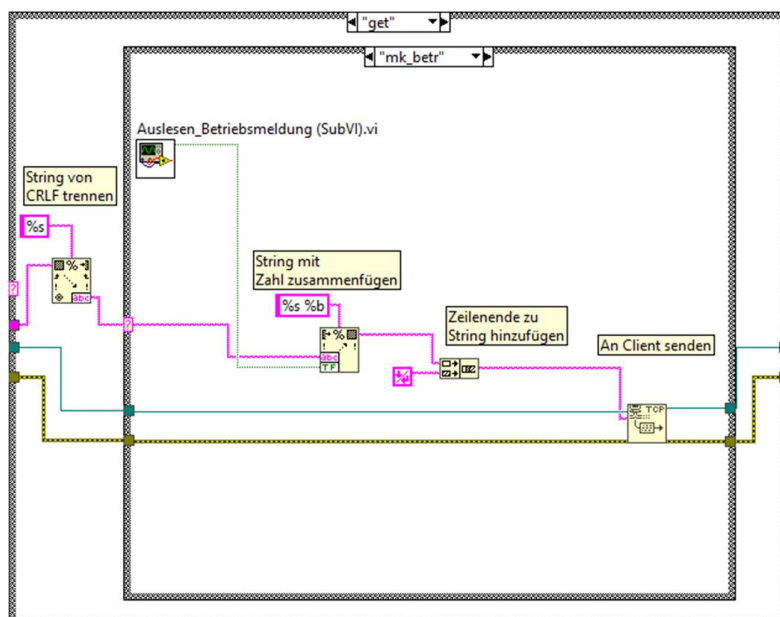
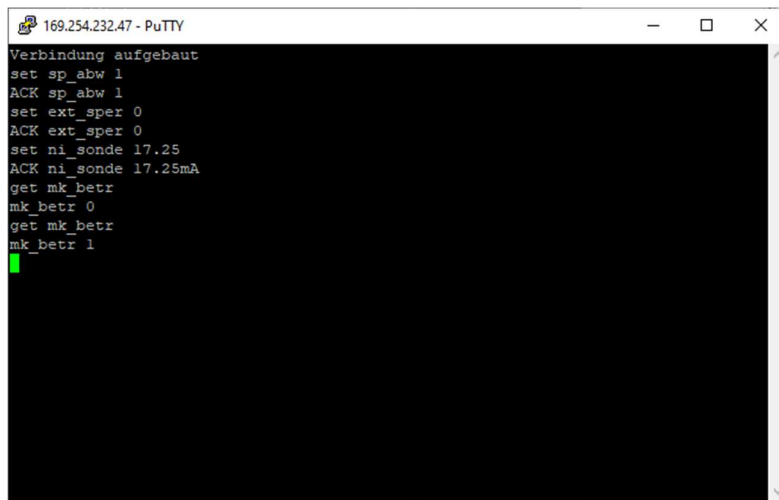


Abbildung 63 LabVIEW Get Befehl Beispiel

Die Kommunikation mit den Befehlen aus dem Kommunikationsprotokoll wurde ausgiebig getestet. In der nachfolgenden Abbildung ist ein Testlauf mit verschiedenen Befehlen und Rückmeldungen ersichtlich.



```
169.254.232.47 - PuTTY
Verbindung aufgebaut
set sp_abw 1
ACK sp_abw 1
set ext_sper 0
ACK ext_sper 0
set ni_sonde 17.25
ACK ni_sonde 17.25mA
get mk_betr
mk_betr 0
get mk_betr
mk_betr 1
█
```

Abbildung 64 Testlauf Kommunikationsprotokoll

7.3.4 Fehlerinjektion

Da der HiL-Simulator keine logischen Abläufe abbilden oder Zustände des Systems interpretieren soll, erübrigt sich die Implementierung der Fehlerinjektion in der Software.

Gleichzeitig ist zu erwähnen, dass mit der Erarbeitung der Abschnitte 7.3.2 und 7.3.3 bereits eine Fehlerinjektion möglich ist, da jeder einzelne Ausgang entsprechend geschaltet werden kann. Die Testprozedur wird von den Entwicklern geschrieben und diese können die bekannten Fehler aus dem Abschnitt 5.4 bei Bedarf somit selbst injizieren.

8 Testing

8.1 Hardware

Nachdem die Hardware aufgebaut und verdrahtet wurde, begann die Entwicklung der Software. Während der Programmierung der Schnittstellen zum Abwassersteuergerät (siehe Abschnitt 7.3.2) wurde die Hardware ausgiebig getestet. Die digitalen Ausgänge konnten dabei via Schalter auf dem Frontpanel ein- und ausgeschaltet werden. Hierbei wurden stets die entsprechenden Relais geprüft. Der analoge Ausgang konnte auch mit einem Schalter eingeschaltet werden und mit einem Schieber auf dem Frontpanel wurde der Wert eingestellt. Dieser wurde mit einem Multimeter Fluke 1587 gemessen und überprüft. Die digitalen Eingänge wurden mit Schaltern simuliert und auf dem Frontpanel angezeigt. Während des Hardwaretests wurden keine Fehler entdeckt und somit ist auch keine Nacharbeit nötig.

8.2 Software

Die Software wurde während der Entwicklung der einzelnen Etappen stets geprüft. Somit wurden Fehler frühzeitig erkannt und konnten behoben werden.

8.3 HiL-Simulator

Nach der Fertigstellung der Hard- und Software des HiL-Simulators wird das komplette System nochmals auf die korrekte Funktion überprüft. Dabei kommt ein DUT-Dummy zum Einsatz, da die Abwassersteuerung zur Zeit der Diplomarbeit noch nicht weit genug entwickelt wurde, um sie bereits zu testen. Der DUT-Dummy wurde gemäss den Anforderungen des Pflichtenhefts (A.2 Pflichtenheft DUT-Dummy) von einem Elektronikerlehrling der Biral AG entwickelt. Dabei handelt es sich um eine Platine, auf der die Signale mit LED's angezeigt und mit Schaltern gesteuert werden können. In der Abbildung 63 ist der DUT-Dummy dargestellt.

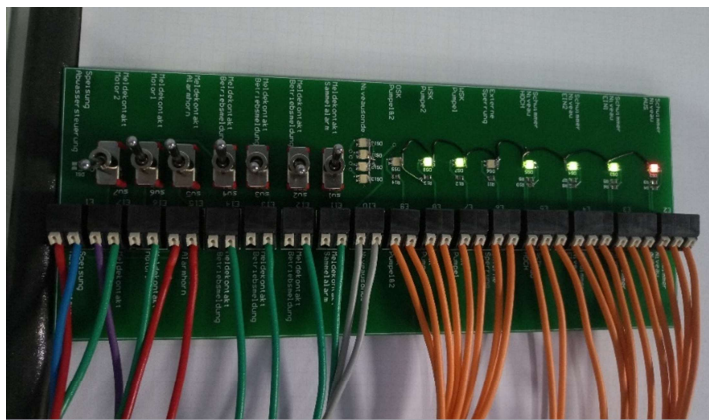


Abbildung 63 DUT-Dummy

8.3.1 Funktionstest

Da der Diplomand die Hardware aufgebaut, verdrahtet und auch die Software dazu entwickelt hat, kennt er das System sehr gut. Dies birgt die Gefahr, dass er gewisse Punkte, wenn auch unbewusst, vernachlässigt hat. Damit die Funktionen ohne Vorwissen ausgiebig getestet werden, wird der HiL-Simulator durch eine zweite Person, welche nicht am Projekt beteiligt ist, getestet. Das detaillierte Ergebnis dieses Funktionstests ist dem Anhang (A.4 Funktionstest Protokoll) zu entnehmen. Zusammengefasst kann gesagt werden, dass alle Funktionen erfolgreich getestet und keine Fehler ermittelt wurden. Festgestellt wurde jedoch, dass das VI bei einem Timeout besser komplett geschlossen werden sollte, damit es durch den Build Agent dann neu ausgeführt werden kann. Dies wurde in der Software noch angepasst.

9 Kontrolle Zielerreichung

Nachdem das ganze System erfolgreich durch eine zweite Person getestet wurde, welche nicht am Projekt beteiligt war, werden nun die erfüllten Ziele ermittelt. Hierfür wird wiederum in Muss- und Kannziele unterteilt.

9.1 Grundfunktionen (Mussziele)

Die Zielerreichung der Mussziele ist in der folgenden Abbildung ersichtlich.

Ziel	Erläuterung	Status
Hardware Aufbau	Die Hardware des HiL-Simulators muss so aufgebaut werden, dass er alle Ein- und Ausgänge des Abwassersteuergerätes abdeckt. Zudem muss er eine Schnittstelle zur bestehenden Test-Infrastruktur haben. Die Speisung des Interfaces (24V) muss den Dauerbetrieb (24/7) gewährleisten können.	Die Hardware deckt alle Ein- und Ausgänge des Abwassersteuergerätes ab. Sie bietet eine Schnittstelle zur bestehenden Test Infrastruktur und ein Dauerbetrieb (24/7) ist gewährleistet.
Schnittstellen zu Abwassersteuergerät	Der HiL-Simulator muss alle «Systemschnittstellen» der Abwassersteuerung abdecken. Alle Eingänge müssen gemäss Vorgaben simuliert werden können und die Ausgänge entsprechend ausgewertet werden. Die Stromversorgung der Abwassersteuerung muss über den HiL-Simulator geschaltet werden können.	Die Systemschnittstellen der Abwassersteuerung werden abgedeckt und es können alle Ein- und Ausgänge simuliert und ausgewertet werden. Die Stromversorgung kann durch den HiL-Simulator geschaltet werden.
Schnittstellen zu Test-Infrastruktur	Die Test-Infrastruktur muss über TCP/IP oder Serial-Interface(USB) den HiL-Simulator ansteuern können (die Kommunikation wird im Zuge des Projekts bestimmt).	Die Test-Infrastruktur kann via TCP/IP den HiL-Simulator ansteuern und es wurde ein Kommunikations-protokoll zusammen mit den Stakeholdern (Kunden) erarbeitet.
Fehlerinjektion	Der HiL-Simulator muss in der Lage sein, bei allen überwachten Systemkomponenten resp. Schnittstellen gezielt Fehler zu injizieren.	Da der HiL-Simulator jeden Ein- und Ausgang einzeln ansteuern bzw auslesen kann, ist eine Fehlerinjektion durch die Test Prozedur möglich.
Business-Logic	Der HiL-Simulator bildet nur eine Schnittstelle zwischen DUT und der Test-Infrastruktur. Die Testprozedur muss über die Test-Infrastruktur definiert werden können. Es sollen keine logischen «Testabläufe» im Interface abgebildet werden oder irgendwelche Zustände des Systems interpretiert werden.	Der HiL-Simulator führt nur die Befehle aus, welche ihm über TCP/IP übermittelt werden (mit entsprechendem Feedback) Somit werden keine logischen Testabläufe abgebildet und auch keine Zustände des Systems interpretiert.

Abbildung 66 Zielerreichung Mussziele

9.2 Optionale Erweiterungen (Kannziele)

Die Zielerreichung der Kannziele ist in der folgenden Abbildung dargestellt.

Ziel	Erläuterung	Bestandteil
Programmierschnittstelle	Die Abwassersteuerung wird über ein JTAG-Programmier-Interface verfügen. Dieses soll für das Programmieren der Firmware verwendet werden. Der HiL-Simulator übernimmt dabei eine Controller-Aufgabe – das eigentliche Programmieren soll über ein SEGGER J-Link erfolgen.	Vorstudie abgeschlossen. Realisierung aus zeitlichen Gründen noch offen.
Parametrierschnittstelle	Es wird möglich sein, die Abwassersteuerung mithilfe einer Parametrierschnittstelle zu parametrieren. Dazu wird eine serielle Verbindung (UART) eingesetzt, über die durch das ProtoBuf-Protokoll kommuniziert wird. Der HiL-Simulator soll, anhand von Angaben der Test-Infrastruktur, das Abwassersteuergerät parametrieren können.	Vorstudie abgeschlossen. Realisierung aus zeitlichen Gründen noch offen.
Motoren simulieren	Mit einem externen Lastwiderstand soll ein externer Motor simuliert werden, was den Betrieb einer Pumpe abbildet. Da die Abwassersteuerung über ein breites Leistungsspektrum verfügen wird, soll der gewählte Lastwiderstand einstellbar sein.	Vorstudie abgeschlossen. Realisierung aus zeitlichen Gründen noch offen.
Strommessung	Auf der Abwassersteuerung wird die Stromaufnahme der Motoren/Pumpen überwacht. Um den gemessenen Strom der Steuerung prüfen zu können, muss der HiL-Simulator auch über diese Möglichkeit verfügen.	Vorstudie abgeschlossen. Realisierung aus zeitlichen Gründen noch offen.

Abbildung 67 Zielerreichung Kannziele

10 Abschluss

10.1 Berechnungen/Wirtschaftlichkeit

Im Abschnitt 6.1 wurden die Zeit- und Kosteneinsparnisse durch den HiL-Simulator bereits aufgezeigt. Dennoch müssen auch die Kosten für die Hardware und die Zeit für die Entwicklung des Simulators ermittelt werden. In der nachfolgenden Abbildung werden die Hardwarekosten tabellarisch dargestellt.

Hardware	Stk	Preis Total	
NI 9265, Analogausgangsmodul, 4 Kanäle	1	520	CHF
NI 9981, Zugentlastung und Bedienschutz	1	41	CHF
Digitalausgangsmodul NI 9375, Federklemme	1	635	CHF
Backshell NI 9940 für 36-pol. Zugfederklemme	1	46	CHF
CompactDAQ-Chassis NI cDAQ-9185	1	1530.00	CHF
Ausgangskoppler 1 Wechsler hartvergoldet AC/DC 24 V	15	180.68	CHF
Ausgangskoppler Relaiskoppler, 1 Wechsler AC/DC 230 V	5	61.94	CHF
LEITUNGSSCHUTZSCHALTER 230V 6KA, 1+N	1	29.16	CHF
SETRON, Hauptschalter, 3-polig, Iu: 16 A	1	13.37	CHF
SIPLUS LOGO! Power 24V 2,5A	1	144.86	CHF
ARRETIERSTÜCK 35 MM	12	7.68	CHF
REIHENKLEMME BL 2.5MM2	5	3.10	CHF
REIHENKLEMME ERDE 2.5MM2	2	4.10	CHF
REIHENKLEMME GR 2.5MM2	38	23.18	CHF
TRENNWAND 2.5MM2	6	1.56	CHF
Kleinmaterial(Beschriftung, Kleber etc.)	1	10.00	CHF
Total	Ist	3251.63	CHF

Budget	Soll	10000	CHF
Differenz	Dif	6748.37	CHF

Abbildung 68 Hardwarekosten

Es ist gut zu erkennen, dass das Budget, trotz der eher teuren Hardware von National Instruments, nicht voll ausgeschöpft wurde.

Die Entwicklungskosten können der Abbildung 67 entnommen werden.

	Zeit/Kosten	
Totalzeit für Diplomarbeit ca.	216.00	h
Kosten MA pro Stunde ca.	35.00	CHF
Total	7560.00	CHF

Abbildung 69 Entwicklungskosten

Die Entwicklungskosten sind im Verhältnis zu den Hardwarekosten eindeutig höher. Es ist jedoch zu beachten, dass diese einmalig anfallen und bei einem weiteren HiL-Simulator nur die Kosten der Hardware anfallen würden.

10.2 Reflexion

In der folgenden Reflexion geht es um die Erarbeitung der Diplomarbeit, welche ich nach dem Abschluss des letzten Semesters der Weiterbildung zum Elektrotechniker HF in einem Zeitrahmen von sechs Wochen geschrieben habe.

Das Thema der Diplomarbeit war die Erarbeitung eines produktiv einsetzbaren HiL-Simulators. Hierfür habe ich zuerst eine Vorstudie zur Informationssammlung durchgeführt. Anschliessend konnte ich ein Elektroschema erarbeiten und die Hardware aufbauen und verdrahten. Danach habe ich die Software des HiL-Simulators geschrieben und abschliessend das komplette System getestet.

Durch die Vorstudie habe ich mir sehr viel Wissen angeeignet, welches ich in der späteren Arbeit gut anwenden konnte. Der Aufbau der Hardware hat mir viel Spass gemacht, da ich gerne mit den Händen arbeite. Die Entwicklung der Software war anspruchsvoll, da ich vorher nie richtig mit LabVIEW gearbeitet habe, aber vielleicht war genau deshalb meine Motivation sehr hoch, dies zu schaffen. Ich habe während der Softwareprogrammierung sehr viel über die Programmierung mit LabVIEW gelernt. Auch das Thema TCP/IP war für mich Neuland und ich habe mich sehr gefreut, als ich die ersten Befehle senden und empfangen konnte. Ich konnte alle Mussziele, welche im Pflichtenheft definiert wurden, erfolgreich abschliessen. Der HiL-Simulator kann mit Befehlen, via TCP/IP, alle Ein- und Ausgänge der Abwassersteuerung simulieren und auswerten. Die optionalen Ziele waren jedoch aus Zeitgründen nicht mehr realisierbar. Die Vorstudie dazu konnte ich dennoch abschliessen. Meine Erwartungen an die Diplomarbeit haben sich erfüllt und ich konnte sehr viele neue und spannende Themen erlernen.

Den Aufbau der Hardware und die Verdrahtung konnte ich ohne Schwierigkeiten durchführen. Auch das Programmieren der Schnittstellen zur Abwassersteuerung verlief ohne grössere Probleme. Ein wenig Bedenken hatte ich bei der Programmierung des TCP/IP Servers und der Schnittstelle zur bestehenden Test-Infrastruktur, da dies ein neues Gebiet für mich war. Ich konnte aber auch dies durch gute Recherche und Ausprobieren relativ rasch lösen. Da ich den Zeitplan sehr gut einhalten konnte und keine gravierenden Probleme aufgetreten sind, würde ich in einem zukünftigen Projekt nicht viel anders machen. Einziger Kritikpunkt ist für mich, dass ich keines der Optionalen Ziele realisieren konnte. Dies hätte eventuell besser gemacht werden können, wenn ich für die Vorstudie im Allgemeinen weniger Aufwand und Zeit investiert hätte.

Ich habe für die Zukunft gelernt, mich nicht von grösseren Arbeiten wie einer Diplomarbeit abschrecken zu lassen. Durch mein Studium in Elektrotechnik habe ich mir ein sehr grosses Fachwissen angeeignet und damit kann ich auch komplexere Probleme und Projekte in Angriff nehmen.

10.4 Fazit

In der Diplomarbeit wurden die verschiedenen Projektphasen des Basis-Phasenkonzept erarbeitet. In der Initialisierung wurden die Randbedingungen der Arbeit analysiert und die Zielsetzung definiert. Zusätzlich wurden die Risiken betrachtet und mit entsprechenden Massnahmen minimiert. Damit das Projekt von den Anspruchsgruppen akzeptiert und aktiv unterstützt wird, wurde eine Stakeholder-Analyse durchgeführt und entsprechende Kommunikationsmassnahmen ergriffen. Damit möglichst effektiv und strukturiert gearbeitet werden konnte, wurde ein Zeitplan mit wichtigen Terminen und Entwicklungsphasen erstellt.

Während der Vorstudie wurden viele Informationen zur Vorbereitung des Konzepts gesammelt. Unter anderem wurde der Hardware Aufbau und die Schnittstellen zur Abwassersteuerung analysiert. Weiter wurden die Schnittstellen zur bestehenden Test-Infrastruktur angeschaut. Auch die Fehlerinjektion wurde thematisiert und die Business-Logic angeschaut. Des Weiteren wurde auch die Vorstudie der optionalen Erweiterungen durchgeführt. Aufbauend auf der Vorstudie wurde ein Konzept erarbeitet. In diesem Abschnitt wurden auch die Einsparnisse gegenüber dem Istzustand ermittelt. Es wurden Lösungsvarianten entwickelt und schlussendlich ein Lösungsentscheid gefällt.

In der Realisierungsphase wurde ein Elektroschema gezeichnet und die Hardware entsprechend zusammengebaut und verdrahtet. Danach wurden zuerst die Schnittstellen zur Abwassersteuerung programmiert und dabei auch die Hardware getestet. Bei der Schnittstelle zur Test-Infrastruktur wurde der HiL-Simulator als Server programmiert und die TCP/IP-Kommunikation realisiert. Mithilfe der Entwickler wurde ein Kommunikationsprotokoll erarbeitet, um den späteren Gebrauch möglichst einfach zu gestalten.

Nach Abschluss der Software wurde das ganze System nochmals auf die korrekte Funktion getestet. Dabei kam ein DUT-Dummy zum Einsatz, da die Abwassersteuerung zur Zeit der Diplomarbeit noch nicht weit genug entwickelt wurde. Damit die Funktionen ohne Vorwissen ausgiebig getestet werden konnten, wurde der HiL-Simulator durch eine zweite Person, welche nicht am Projekt beteiligt war, getestet. Anschliessend wurde die Zielerreichung kontrolliert und der Projektabschluss fertiggestellt.

Im Zuge der Diplomarbeit konnten alle Grundfunktionen erarbeitet werden. Die Hardware des HiL-Simulators deckt alle Ein- und Ausgänge des Abwassersteuergerätes ab. Sie bietet eine Schnittstelle zur bestehenden Test-Infrastruktur und ein Dauerbetrieb (24/7) ist gewährleistet. Die Systemschnittstellen der Abwassersteuerung werden abgedeckt und es können alle Ein- und Ausgänge simuliert und ausgewertet werden. Auch die Stromversorgung kann durch den HiL-Simulator geschaltet werden. Die Test-Infrastruktur kann via TCP/IP den HiL-Simulator ansteuern und es wurde ein Kommunikationsprotokoll zusammen mit den Stakeholdern (Kunden) erarbeitet. Da der HiL-Simulator jeden Ein- und Ausgang der Abwassersteuerung einzeln ansteuern bzw. auslesen kann, ist eine Fehlerinjektion durch die Testprozedur möglich. Der HiL-Simulator führt nur die Befehle aus, welche ihm über TCP/IP übermittelt werden (mit entsprechendem Feedback). Somit werden im Simulator keine logischen Testabläufe abgebildet und auch keine Zustände des Systems interpretiert. Die optionalen Erweiterungen wurden jedoch aus zeitlichen Gründen nicht mehr realisiert. Die Vorstudie konnte dennoch abgeschlossen werden. Es wäre aber denkbar den HiL-Simulator in Zukunft auch noch mit diesen Features weiter zu entwickeln.

Zusammengefasst kann gesagt werden, dass mit dem produktiv einsetzbaren HiL-Simulator ein Produkt entwickelt wurde, mit dem ein komplettes System in einer definierten Umgebung automatisch getestet werden kann. Der HiL-Simulator entlastet und unterstützt somit die Entwickler beim Testing der neuen Abwassersteuerung. Dadurch können viel Zeit und hohe Kosten eingespart werden und gleichzeitig gibt es dem Entwickler eine erweiterte Sicherstellung, dass sein Code mit der Hardware funktioniert. Dazu kommt, dass er die gewonnene Zeit verwenden kann, um weitere Entwicklungen zu erarbeiten. Somit können die Entwicklungszeiten zusätzlich verkürzt werden.

11 Literaturverzeichnis

- Aucotec AG. (2020). *ELCAD - Planungs- und Dokumentationssystem - AUCOTEC*. Abgerufen am 22. September 2020 von <https://www.aucotec.com/de/produkte/elcad-aucoplan/elcad-planungs-und-dokumentationssystem-aucotec/>
- Dirk-Jan. (26. Oktober 2018). *Was ist Continuous Integration(Kontinuierliche Integration)?* Abgerufen am 28. September 2020 von <https://www.it-talents.de/blog/it-talents/was-ist-continuous-integration-kontinuierliche-integration>
- Energie Wasser Bern. (2020). *Wissen.Wasser Wasserverbrauch - Energie Wasser Bern*. Abgerufen am 23. Oktober 2020 von <https://www.ewb.ch/wissen/wissen/wissen-wasser-wasserverbrauch>
- FRIZLEN. (13. August 2001). *14s0136_Grundsaltungen Model (1)*. Abgerufen am 29. Oktober 2020 von https://www.frizlen.com/fileadmin/user_upload/Product_Database/Schaltbilder/14s0136_Grundsaltungen.pdf
- Führer, A., & Züger, R.-M. (2017). *Projektmanagement für Führungsfachleute* (5., überarbeitete Auflage Ausg.). Schweiz: Compendio Bildungsmedien AG.
- mikrocontroller.net wiki. (29. Februar 2020). *JTAG - Microcontroller.net*. Abgerufen am 29. Oktober 2020 von <https://www.mikrocontroller.net/articles/JTAG>
- National Instruments. (26. März 2019). *What is Measurement & Automation Explorer (MAX)?* Abgerufen am 22. September 2020 von <https://knowledge.ni.com/KnowledgeArticleDetails?id=kA00Z000000P9KBSA0&l=de-CH>
- National Instruments. (2020). *CompactDAQ-Systeme - NI*. Abgerufen am 22. September 2020 von <https://www.ni.com/de-ch/shop/compactdaq.html>
- National Instruments. (2020). *Modul der C-Serie für digitale Signale*. Abgerufen am 11. Oktober 2020 von <https://www.ni.com/de-ch/shop/hardware/products/c-series-digital-module.html?modelId=122208>
- National Instruments. (2020). *Was ist LabView? - NI*. Abgerufen am 22. September 2020 von <https://www.ni.com/de-ch/shop/labview.html>
- SEGGER. (2020). *Interface Description - SEGGER - The Embedded Experts*. Abgerufen am 29. Oktober 2020 von <https://www.segger.com/products/debug-probes/j-link/technology/interface-description/>
- Shiklo, B. (6. Oktober 2019). *Vorgehensweise der Softwareentwicklung im Vergleich*. Abgerufen am 24. September 2020 von <https://www.scnsoft.de/blog/vorgehensmodelle-der-softwareentwicklung>
- Siemens. (2020). *Produkt Details - Industry Mall - Siemens Schweiz*. Abgerufen am 06. Oktober 2020 von <https://mall.industry.siemens.com/mall/de/ch/Catalog/Product/3RQ3018-2AB01>
- Wikipedia. (8. Mai 2019). *Geschäftslogik*. Abgerufen am 28. September 2020 von <https://de.wikipedia.org/wiki/Gesch%C3%A4ftslogik>
- Wikipedia. (13. September 2019). *Joint Test Action Group - Wikipedia*. Abgerufen am 29. Oktober 2020 von [https://de.wikipedia.org/wiki/Joint_Test_Action_Group#:~:text=Joint%20Test%20Action%20Group%20\(kurz,also%20Hardware%20auf%20Leiterplatten%2C%20beschreibt.](https://de.wikipedia.org/wiki/Joint_Test_Action_Group#:~:text=Joint%20Test%20Action%20Group%20(kurz,also%20Hardware%20auf%20Leiterplatten%2C%20beschreibt.)
- Wikipedia. (18. Juni 2020). *Abwasser - Wikipedia*. Abgerufen am 22. September 2020 von <https://de.wikipedia.org/wiki/Abwasser>

12 Danksagung

An dieser Stelle möchte ich allen danken, die durch ihre fachliche und persönliche Unterstützung zum Gelingen dieser Diplomarbeit beigetragen haben.

Um das erlernte Wissen der Weiterbildung zum Elektrotechniker HF praktisch anwenden zu können, musste ich in einem Zeitrahmen von sechs Wochen eine Aufgabenstellung selbstständig, umfassend und zweckmässig lösen sowie sauber dokumentieren und präsentieren. In Zusammenarbeit mit meinem Vorgesetzten wurde das Thema für die Diplomarbeit bestimmt und das Pflichtenheft geschrieben.

Für die hilfreichen Tipps und Erklärungen möchte ich mich herzlich bei Simon Grossenbacher bedanken.

Des Weiteren möchte ich mich bei Jan Segessenmann für die gute Projektbetreuung bedanken. Er hat mir stets mit hilfreichen Tipps weitergeholfen.

Mein Dank geht auch an Loris Knutti, welcher sich die Zeit genommen hat, um den ausführlichen Funktionstest durchzuführen.

Weiter möchte ich mich auch bei der Firma Biral bedanken, die das spannende Projekt ermöglicht und mir die nötige Zeit und finanziellen Mittel zur Verfügung gestellt hat.

Zuletzt möchte ich noch all denjenigen danken, die in der Zeit der Erstellung dieser Arbeit für mich da waren, insbesondere meiner Freundin.

13 Eigenständigkeits-Erklärung

Die untenstehende Eigenständigkeitserklärung ist auf der letzten Seite der Diplomarbeit zu formulieren, mit dem Fertigstellungsdatum zu versehen und zu unterschreiben:

Eigenständigkeitserklärung

Arbeiten, die nachweisbar in vollen Umfang oder in den wesentlichen Teilen unverändert oder ohne korrekte Quellenangabe übernommen werden, gelten als vorfabriziert und werden nicht bewertet.

Ich bestätige, dass ich die vorliegende Diplomarbeit selbstständig verfasst und alle benutzten Quellen gekennzeichnet habe. Diese Arbeit wurde weder in gleicher noch in ähnlicher Form bereits einer Prüfungskommission vorgelegt.

Name / Vorname:

Langenegger Fabian

Ort / Datum / Unterschrift:

Münsingen 02.11.2020



F. Langenegger