



# Afstudeerdossier Simulatieomgeving SIMIT

*Datum: 08-01-2021*

*School: De Haagse Hogeschool*

*Project: Simulatieomgeving SIMIT*

*Document: Afstudeerdossier*

*Auteur: J. Mank*

*Bedrijf: DWG Schiedam*

*Versie: 1.0*

# Voorwoord

In dit document leest u het afstudeerdossier van Jelle Mank naar aanleiding van de afstudeeropdracht simulatieomgeving SIMIT bij DWG Schiedam. Deze opdracht is als afstudeeronderzoek ter afsluiting van de opleiding Network & Systems Engineering op de Haagse Hogeschool te Delft. Het afstudeertraject heeft plaatsgevonden in de periode van 31 augustus 2020 tot en met 8 januari 2021.

Tijdens dit traject is onderzocht wat de mogelijkheden zijn voor DWG indien het softwareprogramma SIMIT wordt gebruikt bij industriële toepassingen. De resultaten van het onderzoek zijn vervolgens omgezet in een adviesrapport, een functioneel & technisch ontwerp en een proof of concept.

Dit project is uitgevoerd onder begeleiding van Lorenzo Daamen en opdrachtgever Alex van der Pluijm, zij zijn werkzaam bij DWG. Door middel van adviezen, kritische feedback en goede samenwerking hebben zij een grote bijdrage geleverd aan het resultaat van de afstudeeropdracht. Ook wil ik graag mijn examinatoren dhr. Wieringa en dhr. van Doorn bedanken voor het begeleiden en beoordelen van mijn afstudeerdossier.

Ik wens u veel plezier met het lezen van mijn afstudeerdossier.

Jelle Mank

Alphen aan den Rijn, 8 januari 2021

# Kenmerken

**Document**      **Afstudeerdossier**  
**Simulatieomgeving**  
**SIMIT**

**Versie**            **1.0**

## Versiebeheer

Versie	Datum	Status	Opmerkingen
0.1	14-10-2020	Concept	Afstudeerverslag doorlopen met de stagebegeleider
0.2	28-10-2020	Concept	Eerste concept afstudeerverslag voor de schoolbegeleider
0.3	2-12-2020	Concept	Feedback schoolbegeleider doorvoeren en oplevering voor het tussentijdse assessment
1.0	8-1-2020	Definitief	Oplevering scriptie

# Inhoudsopgave

<b>Voorwoord .....</b>	<b>2</b>
<b>Kenmerken .....</b>	<b>3</b>
<b>1 Afkortingen .....</b>	<b>6</b>
<b>2 Samenvatting .....</b>	<b>7</b>
<b>3 Inleiding .....</b>	<b>8</b>
<b>4 Organisatiebeschrijving.....</b>	<b>9</b>
4.1 Algemene omschrijving .....	9
4.2 Projecten softwareafdeling .....	9
4.3 Afdelingen .....	10
<b>5 De afstudeeropdracht .....</b>	<b>11</b>
5.1 Probleemdomein .....	11
5.2 Aanleiding.....	11
5.3 Probleemstelling .....	12
5.4 Doelstelling .....	12
5.5 Resultaat.....	12
5.6 Projectmethode .....	12
5.7 Projectfasen .....	13
5.8 FTO .....	14
<b>6 Plan van aanpak.....</b>	<b>15</b>
6.1 Onderzoeksmethoden.....	15
6.2 Werkzaamheden.....	15
6.3 Globale planning .....	18
6.4 Risico's.....	19
6.5 Wijzingen.....	19
<b>7 Vooronderzoek .....</b>	<b>20</b>
7.1 De simulatieomgeving.....	20
7.2 Vervanging lagen in SIMIT .....	22
7.3 Libraries.....	23
7.4 Siemens applicaties met SIMIT .....	24
7.5 Geautomatiseerd testen .....	25
7.6 Rapportagemogelijkheden .....	26
7.7 Operator training DWG .....	26
7.8 Tijdsbesparing.....	26

7.9	Interview .....	27
7.10	Conclusie vooronderzoek .....	30
7.11	User-requirements.....	32
<b>8</b>	<b>Functioneel &amp; Technisch ontwerp.....</b>	<b>33</b>
8.1	Functionele ontwerpen .....	33
8.2	Opstelling proof of concept.....	35
8.3	Technische oplossingen.....	36
8.4	Automatisch testen met SIMIT .....	39
8.5	Device laag.....	43
8.6	Proces laag.....	44
<b>9</b>	<b>Proof of concept.....</b>	<b>46</b>
9.1	Operator handelingen .....	46
9.2	Routes testen en rapporteren .....	46
9.3	Device laag.....	47
9.4	Proces laag.....	48
9.5	Testplan .....	49
<b>10</b>	<b>Vervolgstappen .....</b>	<b>50</b>
10.1	Het advies.....	50
<b>11</b>	<b>Beroepstaken .....</b>	<b>52</b>
<b>12</b>	<b>Evaluatie en reflectie .....</b>	<b>54</b>
12.1	Evaluatie aanpak.....	54
12.2	Evaluatie tussenproducten .....	54
12.3	Reflectie en leerdoelen .....	55
<b>13</b>	<b>Conclusie.....</b>	<b>57</b>
<b>14</b>	<b>Bibliografie.....</b>	<b>58</b>

# 1 Afkortingen

Afkorting	Volledige omschrijving
FAT	Factory Acceptance Test
SAT	Site Acceptance Test
PA	Process automation
FA	Factory automation
AS	Automation system
SUS	Software update service
PLC	Programmable logic controller
I/O	Input & output
VC	Virtuele controller
OB	Organization block
CTE	Component type editor
CFC	Continuous Function Chart
SFC	Sequential Function Chart
PAA	Plant automation accelerator
PCS7	Process control system 7
S7	Step 7
IT	Informatietechnologie
COO	Chief operational officer
CTO	Chief technology officer
CEO	Chief executive officer
CTE	Component type editor
KTP	Koole tankstorage Pernis
CSV	Comma Separated Value
XML	Extensible Markup Language

## 2 Samenvatting

### Het advies voor het toepassen van een SIMIT simulatieomgeving binnen DWG

Ter afronding van het afstuderen van de opleiding Network & Systems Engineering aan de Haagse Hogeschool te Delft heeft dit onderzoek plaatsgevonden in de periode van september 2020 tot februari 2021. Tijdens deze periode is een advies uitgebracht voor het toepassen van een SIMIT simulatieomgeving intern binnen DWG door middel van een vooronderzoek en een proof of concept. DWG wenst een hoger kwalitatieve simulatieomgeving om toekomstige projecten zorgvuldiger te testen voordat de applicatie geïmplementeerd wordt bij de opdrachtgevers. Vanwege het gebrek aan informatie over de mogelijkheden en technische uitdagingen van SIMIT, wordt dit middels een vooronderzoek en een proof of concept voor het bedrijf verduidelijkt. Dit project is volgens het V-model uitgewerkt.

Tijdens het vooronderzoek is door middel van interviews en onderzoeksstrategieën de benodigde informatie verzameld om de vraagstellingen te beantwoorden en de simulatieomgeving SIMIT te onderzoeken. Gebaseerd op het resultaten van het vooronderzoek is besloten dat de onderdelen geautomatiseerd routes testen, rapporteren en operator trainingen (optioneel) te gaan implementeren in het proof of concept. Deze onderdelen werden in het hoofdstuk 'user-requirement' vastgelegd met de technische oplossingen die tijdens het vooronderzoek zijn onderzocht.

Via een script bestand is een programma geschreven die ervoor zorgt dat een signaal afgegeven kan worden naar de virtuele controller (software-in-the-loop oplossing), zodat via de SFC van route control een route kan worden gestart en getest. Vervolgens wordt de geteste route teruggekoppeld naar SIMIT om te kunnen bepalen of de route succesvol is doorlopen. In de testomgeving worden ook de testresultaten van de route gerapporteerd middels een zelfontworpen control type editor (CTE) blok in SIMIT.

Na het uitwerken van de functionaliteiten zijn deze onderdelen in het proof of concept gerealiseerd en gepresenteerd aan de opdrachtgever middels een combinatie van een site acceptance test (SAT) en een factory acceptance test (FAT). De resultaten die gebaseerd zijn op het vooronderzoek zijn positief beoordeeld en er kan worden geconcludeerd dat het proof of concept is geslaagd. Uit de resultaten is een positief advies uitgebracht middels vervolgstappen om het proof of concept in de toekomst verder uit te werken naar een volledig product zodat dit toegepast kan worden in de toekomstige tankterminal projecten van DWG.

### 3 Inleiding

Dit afstudeerverslag is geschreven naar aanleiding van het afstudeertraject van Jelle Mank bij DWG Schiedam in de periode van 31 augustus 2020 tot en met 8 januari 2021. Voor het bedrijf is een proof of concept in de simulatieomgeving SIMIT gerealiseerd gebaseerd op een intern vooronderzoek en de eisen en wensen van de interne opdrachtgever DWG. Tijdens het vooronderzoek is onderzocht wat de mogelijkheden zijn voor DWG indien er wordt geïnvesteerd in een SIMIT-softwarepakket en op welke manier de gewenste functionaliteiten toegepast kan worden in de softwareomgeving.

SIMIT is een softwareprogramma van Siemens met als doel om industriële automatiseringsprocessen te testen op programmeer, ontwerp en denkfouten om tijdens de laatste fases van een project tijdwinst en kosten te besparen. Ook staat SIMIT bekend als ondersteunende tool om operator trainingen te geven bij industriële automatiseringsprojecten.

Dit document bevat een beschrijving van het algehele proces van de verschillende opgeleverde producten voor het bedrijf. Daarbij wordt in het verslag behandeld hoe dit project tot stand is gekomen, wat de resultaten zijn en welke vervolgstappen door het bedrijf kunnen worden genomen na het afstudeertraject. Het afstudeerdossier is beoordeeld door de examinatoren van de Haagse Hogeschool te Delft.



## 4 Organisatiebeschrijving

### 4.1 Algemene omschrijving

DWG Schiedam is een bedrijf dat opgericht is in 2014. Momenteel bestaat het bedrijf uit meer dan 100 medewerkers en is gevestigd in Schiedam en Zaandam. Het bedrijf levert geautomatiseerde oplossingen binnen de industriële automatisering en onderscheidt zich door de disciplines process & safety, electrical, industrial automation en IT uit één hand te leveren.

DWG richt zich niet alleen op automatiseringssystemen, maar ook op innovatie, duurzaamheid en veiligheid binnen de technische branche. DWG houdt zich aan de BRZO-wetgeving voor bedrijven die werkzaam zijn voor gevaarlijke stoffen en voldoet aan proces en/of machine veiligheidseisen via behaalde certificaten. Daarnaast staat DWG bekend om het leveren van 24/7 onderhoudsservice, energiemanagement, cyber security, big data-analyse, virtualisatie en standaardisatie. DWG richt zich op de havengebieden in de Regio's Amsterdam, Rotterdam en Antwerpen en heeft zich ontwikkeld tot de totale asset- en process safety lifecycle van een installatie. DWG is vooral gespecialiseerd in de industrie, energy, infra en tank terminals.

### 4.2 Projecten softwareafdeling

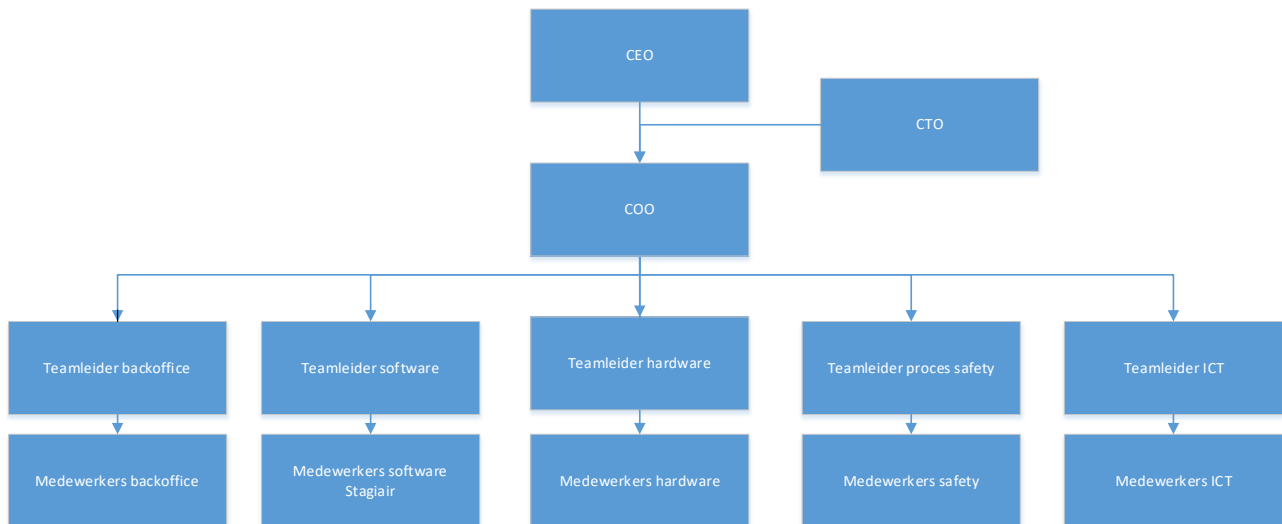
Zoals in de vorige paragraaf staat beschreven specialiseert DWG zich in de industrie, energy, infra en tank terminals sectoren. Het bedrijf heeft al meerdere grote projecten opgeleverd waaronder:

- Software (PCS7, safety) om vanaf verschillende centrale punten bruggen te kunnen bedienen in de omgeving Rotterdam met onder andere het Erasmusbrug.
- Software voor het reguleren van de stadswarmte Rotterdam afkomstig van onder andere Shell en AVR.
- Software van Tankterminals Koole om vloeistoffen te transporteren van schepen naar andere vervoersmiddelen doormiddel van opslagtanks.
- Productieautomatiseringsprojecten van onder andere energie, chemie, stadsverwarming, metaal en food.
- Energiebeheer waaronder ovenbesturingen en netbalancing.
- Het ombouwen van bestaande oude installaties naar een vernieuwd automatiseringssysteem.

De meeste projecten worden uitgevoerd met de softwareprogramma's PCS7 en TIA-portal van Siemens waarin het bedrijf is gespecialiseerd.

### 4.3 Afdelingen

Tijdens mijn afstudeerperiode ben ik werkzaam op de softwareafdeling gespecialiseerd in de industriële automatisering. Verder heeft het bedrijf nog een aantal afdelingen waaronder de backoffice voor de administratie, hardware afdeling voor de elektrische tekeningen, proces & safety om de veiligheid te kunnen waarborgen van de procesinstallaties en ook beschikt het bedrijf over de ICT-afdeling. Boven alle afdelingen bestaat het bedrijf uit een COO (operationeel directeur), een CTO (technisch directeur) en CEO (algemeen directeur). Dit resulteert in het volgende organogram in Figuur 1.



*Figuur 1 organisatiestructuur DWG*

## 5 De afstudeeropdracht

Dit hoofdstuk bevat de opdrachtomschrijving van de afstudeeropdracht. Hierin worden de volgende onderwerpen besproken: de aanleiding van dit project, het probleem en doelstelling, de resultaten, projectmethode en tussenproducten.

### 5.1 Probleemdomein

Op de afdeling software engineering zijn PLC en SCADA specialisten werkzaam die software ontwerpen en realiseren voor diverse opdrachtgevers. De automatiseringssoftware is vooral gebaseerd op Siemens PCS7, TIA-portal en Step 7. Het bedrijf is verantwoordelijk voor een aantal diensten, waaronder process engineering & safety, functional safety, industrial automation, electrical engineering en IT-engineering. Daarnaast beschikt het bedrijf over 24 uur maintenance engineers, hierdoor is het bedrijf dag en nacht bereikbaar voor het afhandelen van storingen in de industriële automatisering.

Recent heeft de afdeling software engineering van DWG onderzocht hoe het bedrijf efficiënter kan werken en ongewenste situaties in een systeem sneller kan opsporen. Hieruit bleek een simulatieomgeving benodigd te zijn voor het ondersteunen van de validatie en verificatie factory acceptance test (FAT). Het doel van de simulatieomgeving is om programmeer, denk en ontwerpfouten die mogelijk kunnen optreden bij het implementeren van een systeem eerder op te sporen. Hierdoor kunnen ongewenste situaties in een live installatie worden voorkomen en tijd en/of productieverlies worden bespaard. Echter, het bedrijf mist de benodigde informatie en kennis om een simulatieomgeving op een goede manier te ontwikkelen. De eisen en wensen van de simulatieomgeving zijn complex. Hierdoor moet eerst onderzocht worden welke mogelijkheden beschikbaar zijn in het Siemens softwareprogramma SIMIT.

### 5.2 Aanleiding

DWG heeft een laagdrempelige simulatieomgeving om producten te testen. Hierdoor is de werking van een ontworpen systeem moeilijk te beoordelen en kan het leiden tot ongewenste situaties. Een laagdrempelige simulatie houdt in dat de software momenteel in de huidige applicatie wordt getest middels het aanpassen van de software in PCS7, TIA-portal of Step7. Er wordt geen externe simulatieomgeving toegepast zoals SIMIT met als gevolg dat niet alle configuratiefouten voor de implementatie worden verholpen. Daarnaast is de omvang van DWG projecten te groot om alle elementen in een project te testen. Een voorbeeld hiervan is de tankterminal projecten van DWG waarbij door middel van de applicatie Route Control miljoenen routes worden gegenereerd en niet één voor één kunnen worden getest.

Er is meer kennis nodig binnen DWG om een simulatieomgeving te ontwikkelen waarin automatiseringssystemen gesimuleerd kunnen worden. Er moet worden onderzocht welke Siemens applicaties die veel toegepast worden in de projecten van DWG, zoals onder andere Route Control, geïmplementeerd kunnen worden in het softwareprogramma SIMIT en er moet een kostenanalyse worden gemaakt. SIMIT is een vereist softwareprogramma vanuit DWG om de simulatieomgeving te kunnen realiseren omdat veel Siemens applicaties toegepast worden binnen DWG.

### 5.3 Probleemstelling

DWG heeft momenteel een laagdrempelige simulatie om producten in te testen, waardoor het gedrag van systeemfuncties niet volgens de werking van een in bedrijf gestelde project is gesimuleerd. Hierdoor kunnen zowel software programmeer, denk en ontwerpfouten vaak niet voor de implementatie worden opgespoord.

### 5.4 Doelstelling

Dit onderzoek heeft als doel een advies te geven aan DWG hoe de simulatie in SIMIT ontwikkelt kan worden zodat systeemfuncties waarvan niet zeker is over de werking van het product gesimuleerd kunnen worden en waarin zowel software programmeer, denk en ontwerpfouten sneller worden opgespoord.

### 5.5 Resultaat

De afstudeeropdracht resulteert in de volgende producten en wordt opgeleverd aan de opdrachtgever:

- Tussenproducten: plan van aanpak, het vooronderzoek, functioneel & technisch ontwerp en het testplan.
- Proof of concept in de simulatieomgeving SIMIT. De eisen en wensen die volgen uit de resultaten van het vooronderzoek worden hierin verwerkt.
- Een adviesrapport.

Op basis van de proof of concept in de simulatieomgeving SIMIT worden de volgende stappen bepaald om het product in de toekomst uit te breiden.

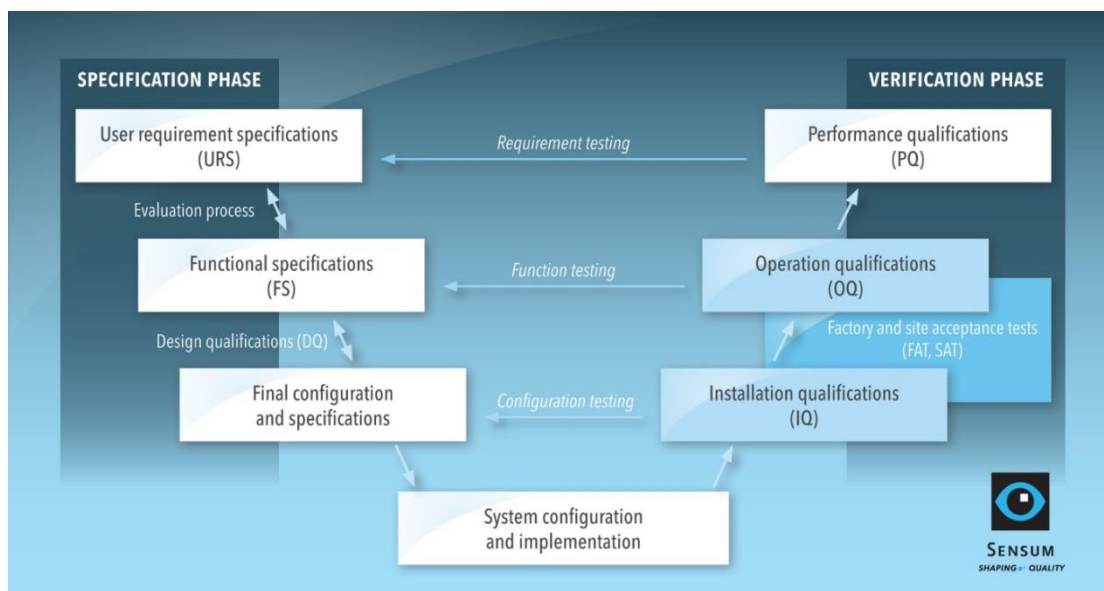
### 5.6 Projectmethode

In de volgende paragrafen worden de werkmethodes beschreven voor het algehele project. Om een gestructureerde aanpak te gebruiken worden een aantal methodes behandeld om te kunnen bepalen welke methodes toegepast kunnen worden voor het project. DWG gebruikt intern in principe twee werkmethodes waaronder het V-model en Scrum.

#### 5.6.1 V-model

Het V-model is opgedeeld in een aantal fases. Iedere fase wordt afgesloten met een document of product. Dit document of product zijn vervolgens het startpunt voor de volgende fase. Een volgende fase kan pas gestart worden indien de vorige fase is opgeleverd en goedgekeurd door de opdrachtgever. Per projectfase worden de producten en verantwoordelijkheden vastgelegd. Deze methode sluit goed aan op mijn project, omdat de opdrachtgever wenst gebruik te maken van de simulatieomgeving. Met deze methode is het mogelijk om de simulatieomgeving op een structurele werkwijze middels het V-model te gebruiken. Het V-model is als volgt is opgebouwd:

- De linkerzijde van het V-model heeft betrekking op alle activiteiten en documenten van het ontwerp.
- Tijdens de middelste fase wordt de hard- en software van het gewenste systeem gebouwd.
- De rechterzijde heeft betrekking op de verificatie- en validatieactiviteiten en wordt teruggekoppeld naar de specificatie fase.



Figuur 2 V-model (Actemium, sd)

Scrum daarentegen wordt toegepast om delen van een applicatie te realiseren. In deze methode worden sprints toegepast die elk bestaan uit een kleine V-model waardoor de opdrachtgever vaak wordt betrokken, namelijk na iedere fase van een sprint. In de scrum methode hoeft het eindresultaat nog niet duidelijk te zijn. Het eindresultaat is van tevoren al vastgelegd, namelijk een simulatieomgeving middels een vooronderzoek om de functionaliteiten te bepalen. Daarom is het toepassen van het V-model hierbij het meest efficiënt.

## 5.7 Projectfases

### 5.7.1 Vooronderzoek

Allereerst wordt een plan van aanpak van het project gemaakt, waarin de planning van het vooronderzoek en het proof of concept worden beschreven. Vervolgens wordt er in het vooronderzoek een uitgebreide productanalyse uitgevoerd van het softwareprogramma SIMIT. Hierin wordt onderzoek gedaan naar de mogelijkheden van het softwareprogramma en of er in de toekomst een SIMIT licentie moet worden aangeschaft. De productanalyse leidt vervolgens tot het beantwoorden van de volgende vraagstellingen die door DWG voor het afstudeertraject zijn opgesteld:

- Wat is de winst in o.a. kosten en tijd van de simulatieomgeving binnen het bedrijf?
- Wat is de relatie met uitwisseling configuratie/gegevens COMOS/PAA/Route Control/Safety?
- Is geautomatiseerd testen met route control mogelijk in SIMIT?
- Wat is de connectiviteit van de automatiseringsomgeving?
- Wat is de functionaliteiten/complexiteit van de beschikbare library functies?
- Welke rapportages kunnen we uit het simulatiesysteem verkrijgen?
- Wat is ervoor nodig om het simulatiesysteem te onderhouden?

Het is noodzakelijk om deze vraagstellingen te beantwoorden om te bepalen of de toegepaste applicaties in de DWG projecten met SIMIT kunnen worden getest. Verder wordt het vooronderzoek verduidelijkt middels een interview. Het interview leidt tot het beantwoorden van ten minste de onderstaande vragen:

- Aan welke eisen moet de simulatieomgeving voldoen?
- Welke systemen moeten in de toekomst gesimuleerd worden?
- Kunnen de eisen en wensen toegepast worden in het softwareprogramma SIMIT?

Op basis van het resultaat van de gesprekken met de opdrachtgever en de resultaten van het vooronderzoek wordt een requirementsanalyse opgesteld. Het plan van aanpak, het vooronderzoek en de requirementsanalyse vormen het URS welke wordt beschreven in het vooronderzoek.

## 5.8 FTO

In het document van het functioneel & technisch ontwerp (FTO) wordt de simulatie ontworpen welke gebaseerd is op de resultaten van het vooronderzoek. Hierin wordt beschreven welke technische aspecten worden toegepast en in welke omgangsvorm de simulatie in SIMIT wordt geprogrammeerd. Dit document behoort in het V-model in zowel het functioneel als het technisch deel van de methode. De functioneel & technische aspecten is in dit project samengevoegd aangezien al snel wordt ingegaan op het technische aspect van de simulatie. Daarnaast heeft de opdrachtgever de benodigde technische kennis om beide aspecten in één keer te beoordelen.

### 5.8.1 Realisatiefase

In de realisatiefase wordt het simulatiesysteem ontwikkeld gebaseerd op het functioneel & technisch ontwerp in het softwareprogramma SIMIT. Dit is de volgende stap in de methode volgens het V-model.

### 5.8.2 FAT & SAT

In deze fase wordt de simulatieomgeving getest samen met de opdrachtgever. Hierbij wordt gebruik gemaakt van een testplan. Na het testen volgt een testrapport waarnaar de eventuele aanpassingen worden doorgevoerd. Dit onderdeel valt onder de laatste twee stappen van het V-model. De factory acceptance tests (FAT) & site acceptance tests (SAT) is in dit project ook samengevoegd aangezien het project niet daadwerkelijk intern geïmplementeerd gaat worden maar als proof of concept wordt gepresenteerd.

## 6 Plan van aanpak

Dit hoofdstuk bevat de aanpak van het afstudeertraject. Dit document is voor mij de basis om een strategische planning te kunnen volgen om met de beschikbare tijd, die bepaald is door de Haagse Hogeschool, de documenten en eindproduct te kunnen opleveren. Daarbij is het van belang dat de opdrachtgever en begeleiders een globale planning ontvangen zodat ze kunnen verwachten wanneer en welke producten opgeleverd worden gedurende het afstudeertraject (17 weken en 2 weken extra voor uitloop).

### 6.1 Onderzoeksmethoden

Om bij het vooronderzoek de benodigde data te kunnen verzamelen voor het proof of concept en het adviesrapport is het noodzakelijk om onderzoeksmethodes te gebruiken. In de volgende paragrafen worden een aantal van deze methodes uitgewerkt.

#### 6.1.1 Kwalitatief

Voor het vooronderzoek wordt een kwalitatieve onderzoeksmethode toegepast. Deze onderzoeksmethode wordt gebruikt, omdat de antwoorden van de onderzoeksvragen zijn gebaseerd op interpretaties, ervaringen en betekenissen. Data wordt onder andere verzameld middels interviews intern op de softwareafdeling. Daarnaast wordt data geanalyseerd op het heden en wordt toegepast om testen in de toekomst te verbeteren (Onderzoeksmethoden, sd).

#### 6.1.2 Deskresearch

Daarnaast wordt een deskresearch methode toegepast in het vooronderzoek. In deze methode wordt informatie verzameld om te bepalen wat de mogelijkheden zijn in de simulatieomgeving. De resultaten van deze onderzoeksmethode worden vervolgens toegepast om de user-requirements te bepalen voor het proof of concept (Onderzoeksmethoden, sd).

#### 6.1.3 Exploratief onderzoek

Bij een exploratief onderzoek wordt verkennend onderzoek gedaan. Deze methode wordt toegepast, als de onderzoeker nog weinig of geen kennis heeft over de resultaten en welke richting het onderzoek opgaat. Hiervoor worden alle interessante gegevens verzameld en worden de resultaten in het onderzoek uitgewerkt (Onderzoeksmethoden, sd). Door deze methode toe te passen is het mogelijk om informatie over SIMIT te verzamelen en de mogelijke functionaliteiten in beeld te krijgen.

### 6.2 Werkzaamheden

Tijdens het afstudeertraject worden een aantal tussenproducten opgeleverd om uiteindelijk tot de gewenste proof of concept en adviesrapport van de simulatieomgeving te komen. In Tabel 1 tussenproducten worden de fases beschreven van het gehele project gebaseerd op het V-model. Hierbij is te vermelden dat alle fases tot en met de SAT (site acceptance test) de opdrachtnemer verantwoordelijk is voor de werkzaamheden. Echter, de opdrachtgever bepaalt uiteindelijk het resultaat van elke fase en geeft een laatste definitief oordeel voordat naar de volgende fase wordt gegaan. De laatste twee fases wordt door zowel de opdrachtnemer als de opdrachtgever uitgevoerd om tijdens de werkzaamheden het proof of concept te kunnen beoordelen.

Fase	Werkzaamheden
Plan van aanpak	Plan van aanpak opstellen
Vooronderzoek	Vooronderzoek opstellen
Functioneel & Technisch ontwerp	FTO document opstellen
Proof of concept	Proof of concept ontwikkelen in SIMIT
Intern testen	Proof of concept testen
Factory Acceptance Test	Met de opdrachtgever het proof of concept doorlopen
Site Acceptance Test	Definitieve proof of concept presenteren
Adviesrapport	Adviesrapport schrijven

*Tabel 1 tussenproducten*

### 6.2.1 Planning vooronderzoek

In het opgeleverde vooronderzoek document zijn de onderzoeksresultaten onderbouwd en de vraagstellingen in paragraaf 5.7.1 beantwoord. Hierin is onderzocht welke mogelijkheden kan worden toegepast in het proof of concept en worden de resultaten afgestemd met de opdrachtgever middels het opstellen van user-requirements. Een indicatie van tijdsduur om het product te realiseren is twintig werkdagen. Tijdens het vooronderzoek is een interview afgenomen met Thijs van Giesbergen, een gespecialiseerde PA (process automation) en FA (factory automation) werknemer bij DWG zodat werkervaringen met projecten en de simulatieomgeving SIMIT kan worden meegenomen in het onderzoek. Ook de gesprekken met de opdrachtgever worden hierin meegenomen.

### 6.2.2 Functioneel & Technisch ontwerp

In het functioneel & technisch ontwerp wordt de simulatieomgeving ontworpen gebaseerd op de resultaten van het vooronderzoek en de gemaakte keuzes met de opdrachtgever. De volgende onderwerpen worden hierin behandeld.

- De visuele weergave van de simulatie om de opdrachtgever een preview te geven op welke manier de simulatie eruit komt te zien.
- Functionele uiteenzetting van de user-requirements en de daarvoor benodigde technische oplossingen.
- Welke project(en) en softwareapplicatie(s) worden toegepast voor het proof of concept.



### **6.2.3 Proof of concept**

In de realisatiefase wordt de simulatie ontwikkeld gebaseerd op de resultaten van het functioneel & technisch ontwerp. De simulatie wordt in het softwareprogramma SIMIT ontwikkeld, waarin de gekozen oplossingen en de functionaliteiten worden meegenomen. De simulatie wordt opgeleverd als proof of concept, om een indicatie te geven welke toepassingen in de simulatieprogramma mogelijk zijn. Een indicatie voor het opleveren van het proof of concept zijn 30 werkdagen. Het aantal werkdagen is gebaseerd op de omvang van de functionaliteiten van het proof of concept en zijn te lezen in paragraaf 7.11.

### **6.2.4 Intern testen**

Na het ontwikkelen van de software worden de functionaliteiten getest middels een testplan. Voordat de FAT plaatsvindt wordt eerst een interne test gedaan door mijzelf. Daarnaast wordt het testplan beoordeeld door de opdrachtgever voor de FAT plaatsvindt.

### **6.2.5 FAT & SAT**

De FAT & SAT moet in de week volgens de planning in paragraaf 6.2 worden ingepland met de opdrachtgever. Dit zal ongeveer twee weken voor de afgesproken week worden ingepland. Tijdens deze fase worden de functionaliteiten gedemonstreerd middels het gerealiseerde testplan. Dit wordt uitgevoerd op een interne locatie bij DWG samen met de opdrachtgever. Tijdens deze fase zal het proof of concept worden beoordeeld en eventuele feedback worden vastgelegd.

### 6.3 Globale planning

In Figuur 3 weeknummers fases staat de planning voor het realiseren van het proof of concept. Hierbij zijn de weeknummers als uitgangspunt genomen. Realisatie van het project moet eind dit jaar afgerond zijn (week 51). Dit in verband met het inleveren van het eindrapport in de eerste week van januari. De planning is gebaseerd op de zeventien weken dat opgesteld is door de Haagse Hogeschool. Bij een eventuele uitloop van een fase is aan het einde van het traject uitloop ruimte gereserveerd.

Weeknummer	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	1
Fase																			
Plan van aanpak																			
Vooronderzoek																			
FTO																			
Proof of concept																			
Intern testen																			
FAT & SAT																			
Adviesrapport																			
Uitloop																			

Figuur 3 weeknummers fases

Werkzaamheden	Startdatum	Einddatum
Plan van aanpak	01-09-2020	03-09-2020
Vooronderzoek	03-09-2020	01-10-2020
Functioneel & Technisch ontwerp	02-10-2020	23-10-2020
Simulatieomgeving ontwikkelen	26-10-2020	04-12-2020
Intern testen	07-12-2020	10-12-2020
Factory Acceptance Test	11-12-2020	15-12-2020
Site Acceptance Test	16-12-2020	18-12-2020
Adviesrapport	18-12-2020	23-12-2020
Uitloop	23-12-2020	04-01-2021

Tabel 2 globale planning

## 6.4 Risico's

Vanwege een risico op uitloop is het van belang dat opgeleverde documenten behandeld worden door DWG om binnen een tijdsperiode van 3 dagen om indien nodig aanpassingen te kunnen doorvoeren. Daarnaast moet per week bekeken worden hoe het project ervoor staat om de planning te kunnen waarborgen. Dit wordt gedaan middels gesprekken met mijn stagebegeleider om de voortgang te kunnen bepalen.

Risico's	Hoe waarborgen
Een projectfase loopt uit	Bij het uitlopen van een projectfase wordt eerst bekeken of dit gecompenseerd kan worden met de volgende fases voordat de uitloop ruimte wordt toegepast.
SIMIT Licentie vertraagd	De licentie is nodig bij het proof of concept, vanaf eind oktober. In het begin van het traject moet de licentie geregeld worden. Hierdoor is 8 weken speling om de licentie te verkrijgen. Eventueel als alternatief heeft de Haagse Hogeschool een licentie in bezit waar gebruik van kan worden gemaakt.
Vooronderzoek resultaten niet als verwacht	Indien vooronderzoek resultaten niet zijn zoals wordt verwacht, dan wordt er met de opdrachtgever naar een alternatief gezocht.
Proof of concept mislukt	Bij eventueel een mislukte proof of concept wordt alsnog een adviesrapport opgesteld om vervolgstappen te kunnen ondernemen na de afstudeerperiode.
Thuiswerken geen optie	Indien het project vastloopt vanwege de coronamaatregelen, wordt tijdig met de HHS en DWG naar een alternatief gezocht. Verlenging van het afstudeertraject is een mogelijkheid.

Tabel 3 risico's

## 6.5 Wijzingen

Bij wijzingen in de globale planning moet de situatie bespreekbaar worden gemaakt met de opdrachtgever, stagebegeleider en schoolbegeleider. Bij grote veranderingen is het realiseren van een wijzigingsvoorstel noodzakelijk.

## 7 Vooronderzoek

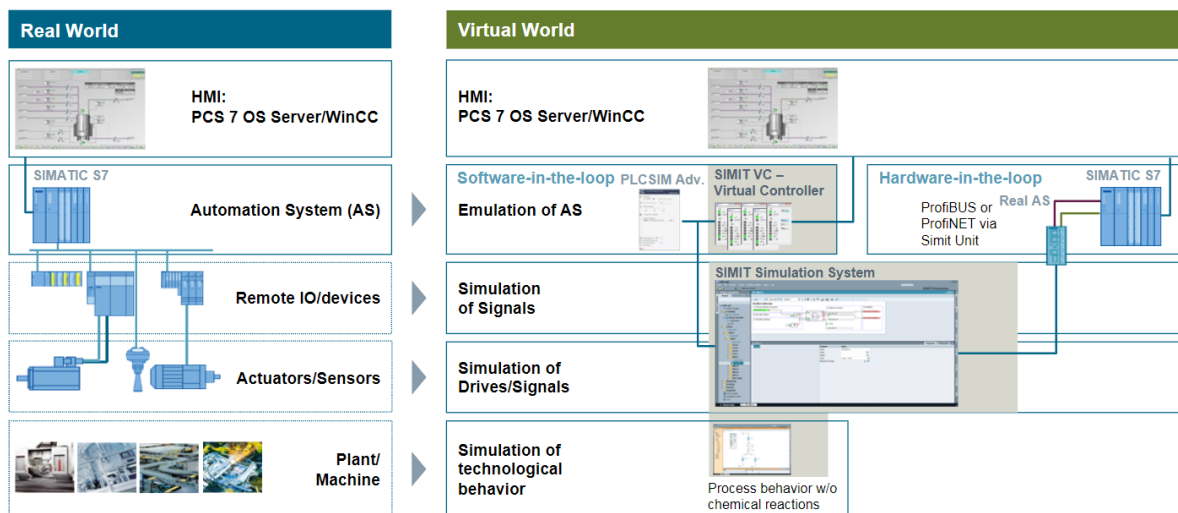
In het vooronderzoek worden de SIMIT mogelijkheden onderzocht en wordt bepaald welke functionaliteiten in het proof of concept worden gerealiseerd. Het vooronderzoek wordt uitgevoerd middels de onderzoeksmethodes die beschreven zijn in paragraaf 5.6 en volgens de planning in paragraaf 6.3. In dit hoofdstuk worden alleen de belangrijkste resultaten behandeld van het vooronderzoek document. In het vooronderzoek wordt nog geen conclusies getrokken voor het proof of concept. Dit wordt pas in hoofdstuk 8 Functioneel & Technisch ontwerp uitgevoerd en beargumenteerd gebaseerd op de onderzoeksresultaten van het vooronderzoek.

### 7.1 De simulatieomgeving

Voordat naar de mogelijkheden wordt gekeken is het belangrijk om de structuur van SIMIT te onderzoeken. Hierdoor kan worden bepaald op welke manier de simulatie moet worden ontwikkeld en welke lagen worden vervangen ten opzichte van een echte installatie. Daarnaast kan worden bepaald welke oplossing toegepast kan worden voor het vervangen van de PLC (hardware-in-the-loop of software-in-the-loop).

Het softwareprogramma SIMIT biedt de mogelijkheid om projecten te simuleren en het gedrag van systemen te rapporten zonder dat hierbij systeemcomponenten en procesinstallaties nodig zijn. Hierdoor worden sneller programmeer, denk en ontwerpfouten van een systeem vooraf aan de implementatie opgespoord. Ook biedt het softwareprogramma een ondersteunende tool om operatortrainingen te geven in een virtuele omgeving.

Figuur 4 overzicht lagen geeft een overzicht welke lagen SIMIT vervangt ten opzichte van een echte installatie (Siemens, 2017). In plaats van dat een OS-server connectie maakt met een PLC gebeurt dit met SIMIT via een virtual controller (software-in-the-loop). Het is ook mogelijk om de virtual controller te vervangen middels een SIMIT Unit dat vervolgens connectie maakt met SIMIT en de PLC (hardware-in-the-loop). De overige drie lagen van een industrieel automatiseringsproject worden vervangen door de simulatieomgeving. Hierdoor zijn remote I/O devices, actuatoren, sensoren en installaties niet meer nodig (Siemens, 2017). De software-in-the-loop en hardware-in-the-loop oplossingen worden later in dit hoofdstuk besproken.

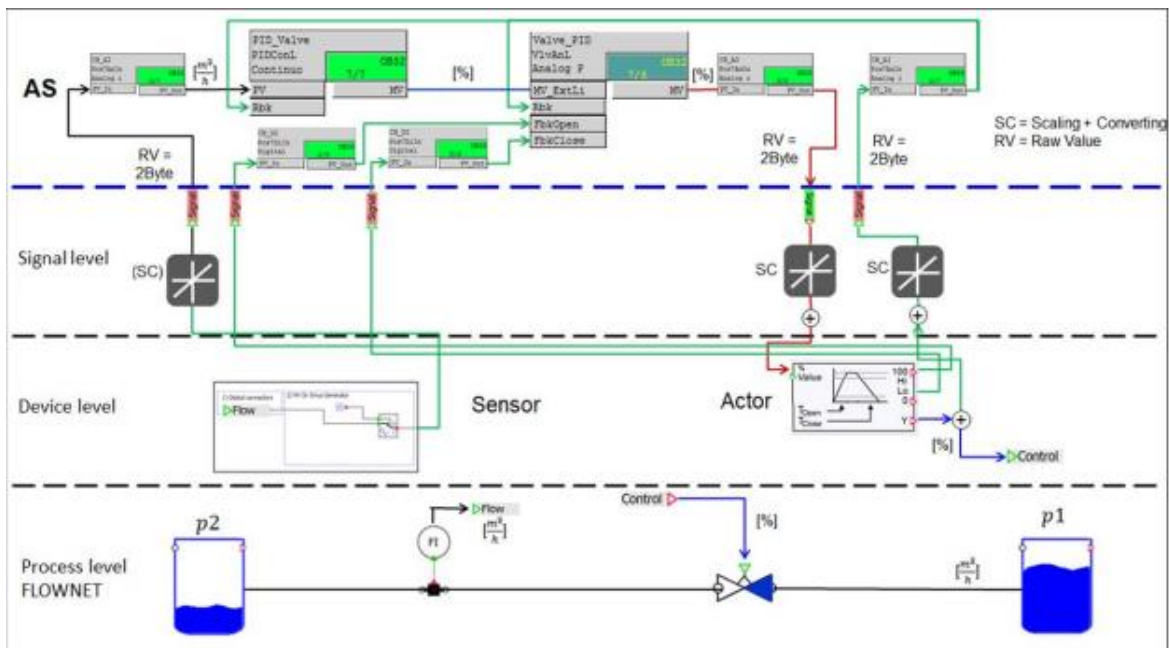


Figuur 4 overzicht lagen (Siemens, 2017)

### 7.1.1 Structuur SIMIT

Door de structuur van SIMIT te onderzoeken, is bekend op welke manier de simulatie moet worden ontwikkeld van zowel in de functioneel & technisch ontwerp fase en het proof of concept. De SIMIT structuur bestaat uit de volgende drie levels:

- **Signal level:** In de signal level (ook wel virtual controller) worden de I/O signalen geïmporteerd. In deze laag komen de signalen binnen van zowel SIMIT als de OS. Signalen kunnen worden veranderd door bijvoorbeeld in SIMIT kleppen of alarmeren te forceren.
- **Device level:** SIMIT voert een bepaalde actie uit op basis van de binnengekomen signalen. Het gedrag van de simulatie wordt in de device laag verwerkt middels een actor of een sensor en wordt vervolgens doorgestuurd naar de proces laag en er kan eventueel een feedback signaal teruggestuurd worden naar de OS. Hierdoor kan het proces worden gemanipuleerd.
- **Proces level:** In deze laag wordt het gedrag van de simulatie gesimuleerd. Afhankelijk op de output signalen van de device laag reageert de simulatie in de proces laag door bijvoorbeeld de flow via pijpleidingen te simuleren of een tank te vullen indien de kleppen zijn geopend (Siemens, 2017).



Figuur 5 structuur SIMIT (Siemens, 2017)

### 7.1.2 OPC

OPC is een protocol in SIMIT dat zorgt voor de communicatie tussen SIMIT en een third party application. Dit protocol kan een overweging zijn om toe te passen in de simulatieomgeving om routes geautomatiseerd te kunnen rapporteren via een externe applicatie. Hierdoor is het mogelijk om informatie te delen van en naar de simulatieomgeving en kan dus aan een externe applicatie worden gekoppeld. Hierdoor kan de simulatie altijd nog uitgebreid worden naar een externe applicatie waardoor meer toepasbare functionaliteiten mogelijk is (Siemens, 2018).

Een nadeel van het toepassen van een externe applicatie is dat het extra tijd en geld kost om het naast de simulatieomgeving te gaan ontwikkelen. Daarnaast is extra kennis benodigd om de externe applicatie te ontwikkelen. Desondanks kan dit protocol wel een goede optie zijn om in de toekomst toe te passen voor verdere uitbreiding van de simulatieomgeving.

## 7.2 Vervanging lagen in SIMIT

In de volgende paragrafen worden twee verschillende oplossingen uitgeschreven voor een software-in-the-loop of een hardware-in-the-loop oplossing. Deze opties zijn nodig om de verschillende beschikbare oplossingen te vergelijken en gebaseerd op de informatie een keuze kan worden gemaakt voor het realiseren van het proof of concept.

### 7.2.1 Hardware-in-the-loop

Bij een hardware-in-the-loop oplossing bestaat er geen verschil tussen een echt device en een SIMIT Unit. Dit betekent dat de PLC uit een SIMIT-module bestaat, welke wordt ondersteund door het Profinet of Profibus protocol. De SIMIT-module vervangt de I/O laag en communiceert met de simulatieomgeving via een interne coupling in de simulatieomgeving. Het toepassen van deze oplossing zorgt voor een realistischer weergave in de simulatie aangezien de communicatie met echte modules werkt. Hierdoor zijn metingen die gebaseerd zijn op tijd nauwkeuriger te meten dan bij een

software-in-the-loop concept die afhankelijk is van de een virtuele CPU. Het nadeel van deze oplossing is dat hardwarecomponenten nodig zijn tijdens het testen van het systeem. Hierdoor kunnen de kosten oplopen als meerdere SIMIT Unit en componenten benodigd zijn (Siemens, 2017).

### 7.2.2 Software-in-the-loop

Bij een software-in-the-loop oplossing wordt de SIMIT Unit en de PLC vervangen voor één of meerdere virtuele controller(s) of met PLCSIM. De virtuele controller ondersteund IP-communicatie over ethernet en de volgende protocollen: S7, TCP, UDP en RFC (request for comments). RFC is alleen te gebruiken middels communicatie tussen virtuele controllers en is daarbij een technisch document die protocollen en andere aspecten van het internet beschrijft. Voor een virtuele controller is, naast de simulatieomgeving, ook een licentie nodig om een connectie tot stand te kunnen brengen tussen de OS-server en de simulatieprogramma SIMIT. In Bijlage II: Vooronderzoek is te zien welke licenties hiervoor nodig zijn. Het aantal benodigde virtuele controllers wordt per project bepaald door het aantal PLC's. De snelheid van het verwerken van de processen in de virtuele controller(s) is gebaseerd op de kwaliteit van de CPU. Ook is het mogelijk om in plaats van een virtuele controller de PLCSim te gebruiken. In Figuur 6 PLCSIM vs VC is te zien wat de verschillen zijn tussen virtuele controller en PLCSim (Siemens, 2017).

	PLCSIM	SIMIT VC
Emulated controllers	S7-300 and S7-400	S7-300 and S7-400
Access point	Proprietary PLCSIM interface	Original PCS 7 interface, pc internal
Communication protocol	TCP/IP	TCP/IP
Max. among of emulated controllers	8 instances on one PC-Station. No distribution	32 instances per SIMIT-Station, distribution on other PC possible
Communication between virtual controllers, on one pc	Yes	yes
Communication to external partners, outside the pc (OS-Server, real plc)	No	yes
Snapshot function	No	Yes
Support PROFINET	No	Yes
Accelerate and reduce simulation time	No	yes

Figuur 6 PLCSIM vs VC

## 7.3 Libraries

In het softwareprogramma SIMIT zijn libraries beschikbaar voor het simuleren van verschillende installatiesystemen in de proces laag die te zien zijn in Figuur 5 structuur SIMIT. Voor de meeste libraries in SIMIT is een licentie benodigd om de componenten te kunnen gebruiken. De uitleg van de licenties zijn te lezen in het opgeleverde vooronderzoek document in Bijlage II: Vooronderzoek. Een aantal libraries worden in de volgende paragrafen uitgeschreven die worden toegepast in het proof of concept.

### 7.3.1 FLOWNET

De meeste projecten binnen DWG zijn gebaseerd op het transporteren van vloeistoffen via tankterminals. Het project dat voor het proof of concept toegepast gaat worden is ook hierop gebaseerd. Daarom is het noodzakelijk om deze library te onderzoeken om te kunnen bepalen wat de mogelijkheden zijn zodat in de proces laag de vloeistoffen gevisualiseerd kan worden in de simulatieomgeving.

Met de FLOWNET library is het mogelijk om procesmodellen te realiseren en waardes te meten waaronder: druk, temperatuur en flow van pijpleiding installaties. De toepassingsmogelijkheden zijn: water, stoom, gassen en verschillende vloeistoffen. Hiermee is het ook mogelijk om een visueel beeld te krijgen van de beschikbare componenten, zoals gebruik te maken van: pompen, kleppen, pijpleidingen, flenzen, tanks, flow/level indicators, elektrische heater en heat exchangers. Middels het binnenkomende of het forceren van signalen in SIMIT is het mogelijk om de componenten aan te sturen in FLOWNET (Siemens, Tutorials SIMIT, 2020).

### 7.3.2 Component Type Editor (CTE)

Om mogelijkheden voor de simulatieomgeving te kunnen uitbreiden is het noodzakelijk om de CTE (component type editor) te onderzoeken. In de CTE is het mogelijk om eigen componenten aan te maken en toe te voegen in een library. Voor deze editor is naast de simulatieomgeving een aparte licentie benodigd. Deze CTE applicatie is noodzakelijk wanneer componenten moeten worden aangemaakt die niet beschikbaar zijn in één van de libraries (Siemens, Tutorials SIMIT, 2020). Voor de overige informatie over de licenties verwijst ik u door naar Bijlage II: Vooronderzoek.

## 7.4 Siemens applicaties met SIMIT

In het vooronderzoeksdocument worden een aantal Siemens applicaties behandeld om te kunnen onderzoeken welke mogelijke toepassingen zijn binnen SIMIT. Deze Siemens applicaties worden veel toegepast bij de DWG projecten. In de volgende paragrafen worden alleen de resultaten behandeld van applicaties die daadwerkelijk toegepast gaan worden in het proof of concept. De overige onderzoeksresultaten zijn in het vooronderzoek document te lezen.

### 7.4.1 Route control

Route control is een Siemens applicatie die in projecten van DWG wordt toegepast. Tijdens dit onderzoek is het belangrijk om kennis te verkrijgen over hoe deze Siemens applicatie werkt binnen de projecten van DWG en zodat deze applicatie toegepast kan worden in SIMIT. Route control wordt binnen DWG toegepast op tankterminal projecten om vloeistoffen te verplaatsen van verschillende transportmogelijkheden zoals bijvoorbeeld schip of vrachtwagen naar een tankopslag in een tankterminalpark. Een tankterminal park bestaat uit een groot hoeveelheden opslagtanks die aan elkaar zijn verbonden met pijpleidingen om vloeistof naar elkaar te kunnen verpompen. In een tankterminal park zijn miljoenen routes mogelijk die ergens opgeslagen moeten worden in het project om routes te kunnen gebruiken. Deze routes worden in de applicatie route control opgeslagen en beheert.

De vloeistof moet een bepaalde route doorlopen om van de source naar de destination tank te kunnen verpompen. Er zijn grote hoeveelheden beschikbare routes, welke specifieke route de



vloeistof moet doorlopen wordt ook door deze applicatie bepaald. Ook weet de applicatie welke pijpleidingen bezet zijn zodat tegelijk verpompen van verschillende vloeistoffen door één pijpleiding wordt voorkomen.

Tijdens het ontwikkelen van route control applicaties kunnen configuratiefouten ontstaan. Om deze fouten voor de implementatie op te sporen is het noodzakelijk om te onderzoeken op welke manier route control in SIMIT toegepast kan worden. Omdat de configuratiefouten alleen te vinden zijn tijdens het testen van elke route, is geautomatiseerd testen noodzakelijk om grote hoeveelheden routes te kunnen testen.

Om routes te kunnen testen moet worden onderzocht op welke manier gecommuniceerd kan worden met route control. In Figuur 7 communicatie Route control staat weergegeven wat noodzakelijk is om routes geautomatiseerd te kunnen testen. Via een script bestand moet een programma geschreven worden die ervoor zorgt dat een signaal afgegeven kan worden naar de virtuele controller (PLC laag), zodat via de SFC van route control een route kan worden gestart. Vervolgens moet deze route teruggekoppeld worden naar SIMIT om te kunnen bepalen of de route succesvol is doorlopen.



*Figuur 7 communicatie Route control*

## 7.5 Geautomatiseerd testen

Voordat de routes geautomatiseerd kunnen worden getest, moet de communicatie vastgelegd worden met route control via een S7 communicatie. Verder moet nagedacht worden op welke manier de grote hoeveelheden routes binnen de projecten van DWG worden getest. Vanwege de miljoenen routes mag een route niet langer dan een bepaalde tijd in bedrijf zijn om de route in de simulatieomgeving volledige te kunnen testen. In real-time kan dit een paar uur duren maar in de simulatie moet een versneld proces voor worden opgesteld. Dit kan middels het proces in de simulatieomgeving en het verpompen van de vloeistof te versnellen.

Middels het aanleggen van een S7 communicatie tussen de OS-server en route control en een communicatie tussen de virtual controller en de OS-server kan informatie gedeeld worden tussen SIMIT en route control. Om vervolgens geautomatiseerd testen mogelijk te maken moet in SIMIT de routes worden ingeladen via een CSV of XML-bestand. Een andere mogelijkheid is om de routes af te handelen via de communicatieprotocol OPC naar een third party application of via de TMS-applicaties die binnen de tankterminal projecten van DWG worden toegepast. TMS is een applicatie die door DWG is ontwikkeld om routes te kunnen managen. De werking van de TMS applicatie valt buiten de

scope van het project indien niet voor deze oplossing wordt gekozen. Ook moet er een rapportageomgeving worden gerealiseerd waarin geteste routes worden gerapporteerd.

## 7.6 Rapportagemogelijkheden

Bij de functionaliteit automatisch testen is het noodzakelijk om belangrijke gegevens te rapporteren gedurende de testprocedure. Hierdoor hoeft een software engineer niet aanwezig te zijn tijdens de test aangezien mislukte routes automatisch worden gerapporteerd. In SIMIT is het mogelijk om scripting bestanden aan te maken waarin een if, then, else syntax wordt toegepast. Deze scripting functie geeft de mogelijkheid om projectsignalen te forceren vanaf SIMIT naar de PLC en de dialog-based functie te gebruiken. Via de dialog-based functie is het mogelijk om testresultaten weg te schrijven naar een tekstbestand voor rapportagemogelijkheden. Daarbij is het mogelijk om met de `snapshot` functie de status op te slaan van de simulatie. Met deze functie kan de engineer na het testen van de routes de rapportages inzien om te achterhalen welke route is mislukt en op welke manier dit opgelost kan worden.

### 7.6.1 PCS7

Een andere mogelijkheid is om rapportages van testresultaten te rapporteren in PCS7. Als alternatief kan ook de PCS7 alarminterface worden toegepast om deze gegevens te bewaren. In deze interfaces is filtering mogelijk, zodat de mislukte routes eenvoudig kan worden weergegeven indien miljoenen routes geautomatiseerd worden getest. In het hoofdstuk functioneel & technisch ontwerp wordt verder onderbouwd welke keuzes hiervoor worden gemaakt.

## 7.7 Operator training DWG

Momenteel worden PCS7 applicaties en WinCC interfaces als ondersteuning toegepast voor operator trainingen binnen DWG. Hierdoor kan je de logica van de applicatie op een eenvoudige manier laten functioneren. Echter, dit heeft wel beperkingen ten opzichte van een simulatieomgeving. Zonder SIMIT is het ingewikkeld om real-time situaties te creëren waarop de operator moet reageren aangezien de software aangepast moet worden waardoor wordt afgeweken van de daadwerkelijke geïmplementeerde applicatie. Ook biedt SIMIT een overzichtelijke omgeving om gegevens uit te lezen van verschillende componenten waardoor het voor de operator eenvoudiger te volgen is.

### 7.7.1 Toekomst met SIMIT

De bijdrage van SIMIT moet ervoor zorgen dat het softwareprogramma een ondersteuning biedt voor het geven van operator trainingen. Het is de bedoeling dat scenario's worden ontworpen, zodat indien ongewenste situaties in het proces ontstaan, de operator de benodigde kennis krijgt om de juiste handelingen uit te voeren. In het softwareprogramma moet een real-time simulatie worden gerealiseerd voor het bijbehorende project. Dit kan ontworpen worden door het importeren van de applicatie en middels de FLOWNET library het proces te simuleren.

## 7.8 Tijdsbesparing

Gebaseerd op het V-model in paragraaf 5.6.1 wordt bepaald hoeveel tijdsbesparing SIMIT kan opleveren indien een simulatieomgeving niet buiten de scope valt. Afhankelijk van het project verschilt per fase hoeveel winst en kosten hiermee kan worden bespaard. De volgende factoren bepalen hoeveel tijd en kosten hiermee worden bespaard: de grote van het project, het verloop van

het project, het aantal storingen, ontwerp en programmeerfouten, ongewenste gedragsveranderingen en de kwaliteit van de testfase zijn factoren waarin tijd een grote rol speelt. Om in algemene vorm te kunnen bepalen hoeveel tijdsbesparing de simulatie gaat opleveren worden situaties beschreven dat regelmatig voorkomt in het bedrijf. Hierbij wordt het V-model toegepast om dit per fase uit te werken.

SIMIT heeft in de URS-fase geen impact op de tijdsbesparing. Tijdens het opstellen van de user-requirements kan SIMIT geen bijdrage leveren. Wel kan alvast nagedacht worden of SIMIT een mogelijkheid is om toe te passen in het bijbehorende project.

Voor het functioneel & technisch ontwerp kan SIMIT toegepast worden om de simulatieomgeving te ontwerpen. Hoeveel tijd er nodig is om de simulatieomgeving te ontwerpen is afhankelijk van de volgende factoren: beschikbare templates, functionaliteiten en wijzigingen. In deze fase kost het alleen maar tijd om een simulatieomgeving te ontwerpen en zal het geen tijdswinst opleveren.

Tijdens het realiseren van de applicatie kan de simulatieomgeving tijd besparen. Ten eerste moet de simulatieomgeving worden gerealiseerd. Vervolgens kan de simulatie toegepast worden om de gerealiseerde functionaliteiten in het PCS7 project te testen. Hierdoor weet de engineer direct of de werking van het systeem naar wens is voordat de FAT plaatsvindt met de opdrachtgever. De meeste tijdswinst kan worden behaald tijdens de FAT, IBS en SAT. In deze fases wordt het project getest met de simulatieomgeving. Tijdens de FAT kan SIMIT een ondersteuning bieden om het proces van de simulatie te simuleren. Verder biedt SIMIT als ondersteuning om software fouten te beperken. Dit zorgt voor tijdswinst en beperkingen van fysieke schade tijdens het implementeren van systemen.

Indien systemen geïmplementeerd worden kost dit stilstaande productie bij de opdrachtgever. Tijd speelt hierbij een belangrijke rol want hoe langer het proces stilstaat, des te minder producten geproduceerd kunnen worden. Indien storingen en gedragsproblemen zoveel mogelijk worden beperkt, wordt stilstaande productie voorkomen. Ook veroorzaken vertragingen personeelskosten, welke zoveel mogelijk kan worden beperkt middels de simulatieomgeving.

## 7.9 Interview

Tijdens het vooronderzoek is een interview gehouden met Thijs van Giesbergen, een gespecialiseerde PA (process automation) en FA (factory automation) collega, werkzaam bij DWG en al enige ervaring met SIMIT waardoor al de benodigde informatie intern verzameld kon worden. Hieronder een opsomming van de gestelde vragen en antwoorden:

- In welke fase valt tijdens projecten de meeste tijdswinst te behalen wanneer een simulatie wordt toegepast denk je? Is dat pas bij de SAT of eerder in het project?
  - o SIMIT zou al nuttig kunnen zijn wanneer functionaliteiten direct na het realiseren getest kunnen worden. Eerder in het project heeft het niet echt een toegevoegde waarde.
- Wat verwacht je over het algemeen van een testsimulatie?
  - o Een I/O simulatie waarin actie/reactie gesimuleerd kan worden. Ook kan het erg handig zijn voor het opschonen van systemen of om alarmen te testen. Verder kan met de library FLOWNET vloeistoffen gevisualiseerd worden waardoor een

actie/reactie van het system erg overzichtelijk maakt. Daarnaast zal het natuurlijk handig zijn indien een simulatieomgeving tijd en kosten oplevert. Het proces simuleren kan ik afraden aangezien het extra tijd en geld kost om het in de simulatieomgeving te realiseren.

- Hoe worden COMOS/PAA ontwerpen gecontroleerd en getest?
  - o SIMIT is al eerder toegepast bij tankterminal projecten. Hierin konden COMOS en PAA ontwerpen getest worden middels control module types. De ontwerpen kunnen geëxporteerd en geïmporteerd worden via XML bestanden.
- Wat zou je graag gerapporteerd willen zien in een simulatie en wat zijn de belangrijkste rapportages?
  - o Looptijden van kleppen, alarmen, grensinstellingen, meettrance van analoge signalen (aantal milliampère).
- Op welke manier wordt over het algemeen operator trainingen gegeven, enige softwareprogramma's?
  - o Meestal gebeurt dit via de PCS7 applicatie of WinCC waarin de logica van de applicatie op een laagdrempelige manier gedemonstreerd kan worden.
- Wat verwacht je van een simulatieomgeving dat eventueel toegepast kan worden voor operatortrainingen?
  - o Reactie van operators bij ongewenste gedragsveranderingen in een systeem om de juiste informatie te kunnen aanleren bij een operator.
- Heeft u overige informatie over SIMIT?
  - o CTE kan toegepast worden om template blokken aan te maken en upgraden naar een nieuwere versie is niet noodzakelijk.

### 7.9.1 Gesprekken opdrachtgever

Met de opdrachtgever zijn een aantal vergadersessies geweest om de voortgang van de opdracht te bespreken. Vooraf aan het afstudeertraject heeft er een vergadersessie plaatsgevonden waarin de opdracht en vraagstellingen van het vooronderzoek zijn besproken. Vervolgens zijn de bevindingen van het vooronderzoek besproken in een volgende vergadersessie, waarin naar voren kwam dat de focus gelegd ging worden op zowel route control als het rapporteren van de routes. Daarna zijn de user-requirements samen met de opdrachtgever naar aanleiding van de resultaten van het vooronderzoek opgesteld en is begonnen met het functioneel & technisch ontwerp. Tijdens het proof of concept zijn een aantal vergadersessies geweest om de voortgang in SIMIT te bespreken en werkende functionaliteiten te laten zien.

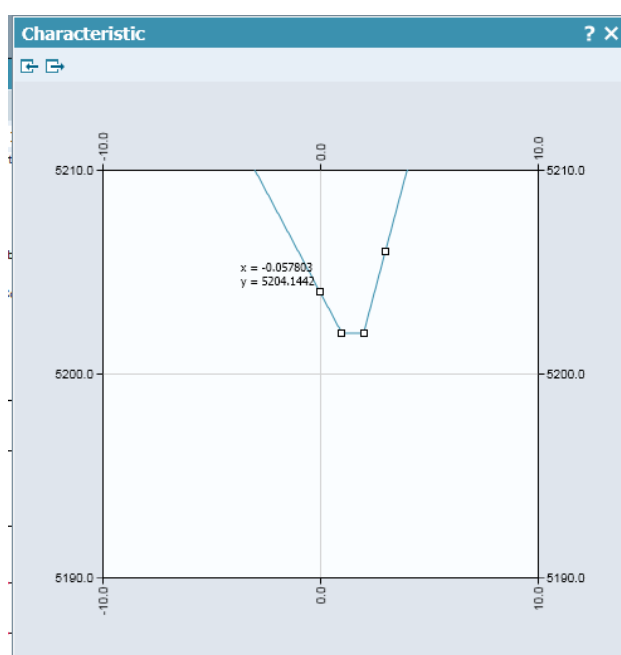
### 7.9.2 Siemens

Tijdens het afstudeertraject zijn drie sessies geweest met het bedrijf Siemens om de afstudeeropdracht te bespreken en te bekijken welke bijdrage Siemens kan leveren aan het realiseren van de simulatieomgeving. Tijdens de eerste sessie vond een kennismaking en introductie plaats. Hierin is de afstudeeropdracht besproken en bekendgemaakt welke vraagstukken beantwoord gaan worden tijdens het vooronderzoek. Ook is afgesproken om voor de volgende sessie een aantal functionele ontwerpen te demonsteren om ideeën voor het proof of concept te kunnen bespreken. Ook heb ik van Siemens de licentie van SIMIT ontvangen met daarbij een virtuele controller (software-in-the-loop) en toegang tot de CTE.

Tijdens de tweede sessie zijn de functionele ideeën besproken die in Bijlage III: Functioneel & Technisch ontwerp zijn weergegeven. In dit gesprek bleek dat de gewenste eisen en wensen technisch mogelijk te zijn indien het OPC ontwerp toegepast wordt, om naast SIMIT een externe applicatie te ontwikkelen voor het rapporteren en inladen van routes. Onze voorkeur ging uit om alles intern in SIMIT op te lossen en daarbij een aantal technische mogelijkheden in SIMIT te onderzoeken, aangezien extra tijd, kosten en kennis benodigd is om naast de simulatieomgeving een externe applicatie te ontwikkelen. Door een aantal factoren was het niet met zekerheid te zeggen dat alles intern in SIMIT zou kunnen worden opgelost. Hierdoor is besloten om tijdens de onderzoeksfase de volgende vraagstellingen te beantwoorden:

- Is het mogelijk om de logica van de routes op een vaste plek op te slaan, zodat de routes kunnen worden ingeladen en opgehaald?
- Is het mogelijk om via de scripting functie geautomatiseerd te rapporteren in CTE?

Voor deze nieuwe vraagstellingen heb ik mijzelf een week de tijd gegeven voordat de volgende sessie met Siemens plaatsvond. Tijdens het onderzoek heb ik bekeken naar de beschikbare componenten in SIMIT. Hierin heb ik een component gevonden voor een mogelijke oplossing om de routes te kunnen inladen. Dit component bestaat uit een simpele formule functie die kan worden toegepast om routes in te laden middels een diagram met X en Y punten. Deze punten worden uitgelezen middels een ID (X-as, afgerond 0) die gekoppeld is aan een tanknummer (Y-as, afgerond 5204). Hierdoor is het mogelijk om van elke geteste route het benodigde tanknummer te gebruiken. Hieronder in Figuur 8 diagram component staat een weergave van het diagram component.



*Figuur 8 diagram component*

De component heb ik getest in SIMIT, met als conclusie dat de routes zowel kunnen worden ingeladen als worden opgehaald. Daarnaast heb ik bekeken wat er in de scripting functie mogelijk is door middel van het schrijven van een script om routes te rapporteren in een tekstbestand. Tijdens het scripten

ben ik tot de conclusie gekomen dat er wel gerapporteerd kan worden, maar dat er vrij beperkte mogelijkheden zijn. Het was namelijk niet mogelijk om in vergelijking met andere programmeertalen een cyclisch programma te maken en er moest na een geteste route opnieuw op een knop gedrukt worden om het script te kunnen starten. Dit is niet wenselijk omdat grote hoeveelheden routes moeten worden getest.

Met deze resultaten en een voorbeeld in SIMIT zijn we de volgende sessie ingegaan met Siemens om kunnen worden in SIMIT, moest alleen nog een alternatief gevonden worden om geautomatiseerd te kunnen rapporteren. Hierbij adviseerde Siemens om de CTE (component type editor) te gebruiken aangezien een soortgelijke programmeertaal C is ingebouwd. Hierdoor is het mogelijk om een eigen in cyclus component te ontwikkelen om routes geautomatiseerd te kunnen rapporteren en een programma continue te doorlopen. Dit is vervolgens getest in SIMIT waardoor de technische onzekerheden zijn verholpen en kan dit middels het functioneel ontwerp worden toegepast voor het proof of concept.

## 7.10 Conclusie vooronderzoek

In de volgende twee alinea's bestaan uit de conclusie van het opgeleverde vooronderzoek document in Bijlage II: Vooronderzoek.

Voor het investeren in een SIMIT-pakket moet de hoeveelheid benodigde tags worden bepaald. Elk component dat in SIMIT wordt toegepast heeft een aantal hoeveelheden tags. Hoe meer tags wordt gebruikt in een SIMIT project, hoe groter de aanschaf van de licentie. Indien niet bekend is hoeveel tags nodig zijn, is het advies om te investeren in een S of M licentie. Het upgraden naar een grotere licentie brengt geen extra kosten met zich mee dan wanneer direct wordt geïnvesteerd in een grotere licentie. Hiermee wordt voorkomen dat direct in een te grote licentie wordt geïnvesteerd. Indien het aantal benodigde tags wel bekend is, dan is het voordeliger om ook voor de benodigde licentie te kiezen aangezien om kosten te besparen. Hierbij moet uiteraard wel rekening gehouden worden met uitloop.

De kosten van de licenties staan in het vooronderzoek document in Bijlage II: Vooronderzoek. Bijna elk jaar worden updates uitgebracht door Siemens met nieuwe functies, maar het is niet noodzakelijk om na elke update te investeren in een nieuwe licentie. Daarnaast is het na twee jaar voordeliger om apart een upgrade uit te voeren in plaats van te kiezen voor een jaarlijkse SUS. Het upgraden van virtual controllers brengt wel kosten met zich mee wanneer gekozen wordt voor losse upgrades. Indien de simulatieomgeving SIMIT wordt geüpgraded, moet de virtuele controller(s) ook met dezelfde versie meegaan aangezien de oude controllers niet meer worden ondersteund. Ook ondersteund SIMIT vaak niet meer de oudere PCS7 versies waardoor huidige projecten niet meer toegepast kunnen worden in SIMIT. Daarom is het advies om pas te upgraden wanneer dit noodzakelijk is. Bijvoorbeeld wanneer nieuwe PCS7 versies worden uitgebracht en deze bij nieuwe DWG-projecten worden toegepast.

Om te bepalen of geïnvesteerd moet worden in een hardware of software-in-the-loop is het aantal PLC's in een project een belangrijke factor. Indien meerdere PLC's benodigd zijn moeten worden geïnvesteerd in meerdere SIMIT Units of virtual controllers. Het investeren in virtual controllers zijn

vergeleken met SIMIT Units goedkoper en ondersteunen zowel Profibus als Profinet. Voor de SIMIT Units moeten aparte modules worden geïnvesteed waardoor meer geïnvesteed moet worden. Mocht de opdrachtgever niet in bezit zijn van 410 CPU componenten moet ook hierin worden geïnvesteed. De virtual controller vervangt deze modules bij een software-in-the-loop concept. Aangezien projecten van DWG vaak uit meerdere PLC's bestaat en niet specifieke tijdseenheden moet meten, is het advies om te investeren in virtual controllers.

Voor het toepassen van libraries binnen DWG-projecten is een FLOWNET library benodigd om de proces laag te kunnen simuleren. Aangezien de meeste projecten van DWG bestaat uit tankterminals en vloeistofprocessen is deze library noodzakelijk. Hierdoor is het mogelijk om flowprocessen te simuleren en het gegevens uitlezen van tankterminals, zoals: druk, temperatuur en flow van pijpleiding installaties. Om deze library in gebruik te kunnen nemen is een FLOWNET-licentie benodigd. De overige libraries zoals CHEM BASIC en CONTEC zijn overbodig en op dit moment niet van toepassing voor de meeste projecten van DWG. Daarom is het advies om alleen te investeren in deze licenties wanneer dat noodzakelijk is.

Daarnaast is het mogelijk om componenten aan te maken met de component type editor. Aangezien DWG zelfontworpen FB-blokken gebruikt, is het advies om eenmalig deze componenten te realiseren en als template te gebruiken voor de projecten die hiervoor van toepassing zijn. Een CTE-licentie is benodigd om deze componenten te kunnen aanmaken.

De simulatie levert een efficiëntere FAT, IBS en SAT op aangezien programmeer/denk fouten zoveel mogelijk worden voorkomen. Onnodig stilstaand productie wordt voorkomen middels het sterk verminderen van storingen tijdens de in bedrijfstelling. Bij de simulatieomgeving kan worden aangetoond dat de software geen fouten bevat. Fabrieksinstallaties hoeven hierdoor minder stil te staan waardoor kosten worden bespaard. Deze bespaarde personeelskosten (engineers en productiepersoneel) en de extra productie opbrengsten kunnen worden toegepast om het ontwikkelen van een simulatieomgeving rendabel te maken.

## 7.11 User-requirements

In de requirementsanalyse staan de user requirements voor het proof of concept van de simulatieomgeving. De user-requirements zijn gebaseerd op het vooronderzoekdocument, een interview met een interne collega en het advies van het bedrijf Siemens. De uitkomst van deze informatiebronnen zijn besproken met de opdrachtgever en heeft geresulteerd in de volgende user requirements voor het proof of concept.

1. Geautomatiseerd testen met route control
  - a. Er moet een Communicatie met S7 verbindingen tussen SIMIT en de AS (o.a. route control) gerealiseerd worden om informatie te kunnen delen. Ook is een interne koppeling in SIMIT benodigd voor een verbinding tussen de virtuele controller en de simulatieomgeving.
  - b. De flow van de route kunnen simuleren met FLOWNET zodat de route van de source naar de destination kan worden gesimuleerd en dat de route op de correcte manier kan worden uitgevoerd volgens de SFC van route control.
  - c. Via een file (nader te bepalen) bron en doel locaties kunnen inladen. Hierna moet deze applicatie het bron en doel (samen een route) één voor één kunnen doorgeven aan de AS welke als startpunt fungeert voor het oplijnen van de route middels route-control.
  - d. Een applicatie moet de eerste doorgegeven route kunnen starten nadat een engineer op start heeft gedrukt. Nadat deze klaar is moet de applicatie de volgende route automatisch starten totdat alle routes zijn getest.
2. Rapportages aanmaken voor het geautomatiseerd testen met route control
  - a. Er moet een applicatie gerealiseerd worden welke kan controleren, middels beschikbare data in Simit /de AS, of een route succesvol is opgelijnd en doorlopen.
  - b. Er moet een applicatie gerealiseerd worden welke niet geslaagde routes kan rapporteren middels een leesbaar en deelbaar bestandstype.
  - c. Er moet een applicatie gerealiseerd worden welke een rapport kan genereren over de geslaagde en niet geslaagde routes welke reeds getest zijn zodat deze kan worden overhandigd aan de opdrachtgever.
  - d. Optioneel: via de scripting functie in SIMIT geautomatiseerd snapshots maken om de status van de ongewenste routes in zowel SIMIT als de virtual controller te kunnen terugzien, zodat het probleem van de route kan worden vastgelegd.
3. Optioneel: SIMIT als ondersteunende tool voor operator trainingen
  - a. Communicatie tussen de applicatie van de operator en SIMIT zodat de operator achter zijn eigen applicatie de handelingen kan uitvoeren indien in SIMIT een ongewenste situatie wordt geforceerd.
  - b. Het ontwerpen en realiseren van ongewenste scenario's in SIMIT zodat de operator voor zijn eigen procesinstallatie de benodigde informatie krijgt om storingen op te kunnen lossen.



## 8 Functioneel & Technisch ontwerp

Zowel het functioneel als het technisch aspect wordt in één hoofdstuk behandeld omdat na het functioneel ontwerp al snel wordt ingegaan op het technische aspect. Tijdens het vooronderzoek is met zowel de opdrachtgever als met Siemens de technische mogelijkheden besproken om een definitief besluit te kunnen maken welk functioneel ontwerp toegepast gaat worden voor het proof of concept. De functionele onderbouwing wordt verder uitgewerkt in de volgende paragrafen. Vervolgens volgt het technische gedeelte en visuele weergaves om alvast een indruk te geven van de simulatie. Zoals te zien is in de user-requirements wordt vooral gericht op de onderdelen rapportages en route control. Voor het testen van het proof of concept wordt het DWG PCS7 KTP tankterminal project toegepast.

### 8.1 Functionele ontwerpen

Om te kunnen bepalen welke technische oplossingen toegepast gaat worden voor het proof of concept zijn er verschillende ontwerpen gerealiseerd om tijdens de vergadersessies met Siemens te kunnen oriënteren over de mogelijkheden. De resultaten van deze vergadersessies zijn te lezen in paragraaf 7.9.2 en de vier verschillende functionele ontwerpen zijn weergegeven in Bijlage V: Functioneel ontwerpen. Voor een overzicht van de ontwerpverschillen zie tabel hieronder.

Funcities	Ontwerp 1	Ontwerp 2	Ontwerp 3	Ontwerp 4
Routes inladen in SIMIT	✓	✓		
Routes inladen via OPC			✓	
Routes inladen in TMS				✓
Rapportages in SIMIT	✓			
Rapportages via OS		✓		
Rapportages via OPC			✓	
Rapportages via TMS				✓

Tabel 4 verschil ontwerpen

#### 8.1.1 Omschrijving ontwerp 1

Tijdens de vergaderingen met Siemens zijn we tot de conclusie gekomen dat het eerste ontwerp het meest tijdbesparend en technisch mogelijk zou moeten zijn. Het functioneel ontwerp bestaat uit de volgende oplossingen:

- De logica van de volgorde voor het testen van routes zit in de simulatieomgeving SIMIT.
- De route moet worden getest in de simulatieomgeving SIMIT en wordt ingeladen en verstuurt via een interne coupling naar de virtual controller. Ook wordt via SIMIT een startsignaal gegeven en operator handelingen toegevoegd.
- De gegevens van de route wordt doormiddel van de PLC via een S7 communicatie door gecommuniceerd naar route control.

- Route control start de route doormiddel van de SFC en koppelt de gegevens terug naar de PLC.
- De gegevens worden verwerkt in SIMIT en via FLOWNET gevisualiseerd en gesimuleerd. Dit wordt vervolgens weer teruggekoppeld aan de PLC en route control.
- Rapportages worden in een tekstbestand verwerkt en gelogd via de scripting functie. Alleen de mislukte routes worden gerapporteerd.

### 8.1.2 Omschrijving ontwerp 2

Het tweede ontwerp in de bijlage is vergelijkbaar met het eerste ontwerp. De verschillen zit in dit ontwerp bij de functionele oplossingen voor de rapportage. Hierin is een aparte interface toegepast vanaf de OS server om gegevens van mislukte routes te kunnen bewaren. Het voordeel van deze oplossing is dat in PCS7 een overzichtelijke interface gerealiseerd kan worden in plaats van dat de gegevens van een route in een notepad bestand worden opgeslagen. Het nadeel van deze oplossing is dat de PCS7 applicatie aangepast moet worden om de gegevens te kunnen doorsturen naar de interface. Dit is niet wenselijk aangezien de software dan gaat afwijken van de daadwerkelijke geïmplementeerde variant.

### 8.1.3 Omschrijving ontwerp 3

Het derde ontwerp is een grote verandering in vergelijking met de eerste twee ontwerpen. Hierin wordt de OPC protocol toegepast tussen SIMIT en een third party application. Hierdoor is het mogelijk om in grote omvang een externe applicatie te koppelen en via deze applicatie de logica van routes te kunnen verwerken. Hierdoor kan ook een uitgebreide applicatie ontwikkeld worden voor het verwerken van informatie van routes. Het voordeel van de OPC protocol is dat je hierdoor een groter bereik krijgt met mogelijkheden om routes en rapportages functioneel op te lossen. Het nadeel van dit ontwerp is dat een volledig nieuwe applicatie moet worden ontwikkeld waardoor deze oplossing niet tijd en kostenbesparend is. Er is namelijk kennis te kort, meerdere applicaties moeten worden onderhouden en andere engineers zijn nodig om deze oplossing te kunnen realiseren.

### 8.1.4 Omschrijving ontwerp 4

Het vierde ontwerp is ook weer een ander functioneel ontwerp dan de vorige drie ontwerpen. In dit ontwerp wordt de applicatie TMS toegepast dat momenteel door de tankterminal projecten binnen DWG wordt toegepast. Met deze managementapplicatie worden alle routes en opdrachten beheert en naar onder andere route control gekoppeld via CFC's in PCS7. Ook dit zou technisch mogelijk moeten zijn om te kunnen koppelen met SIMIT. Echter, bij deze oplossing is het probleem dat DWG afhankelijk wordt van deze applicatie. Ook moet in dit ontwerp de PCS7 software worden aangepast aangezien data van routes moeten worden gecommuniceerd met deze applicatie.

### 8.1.5 Conclusie ontwerpen

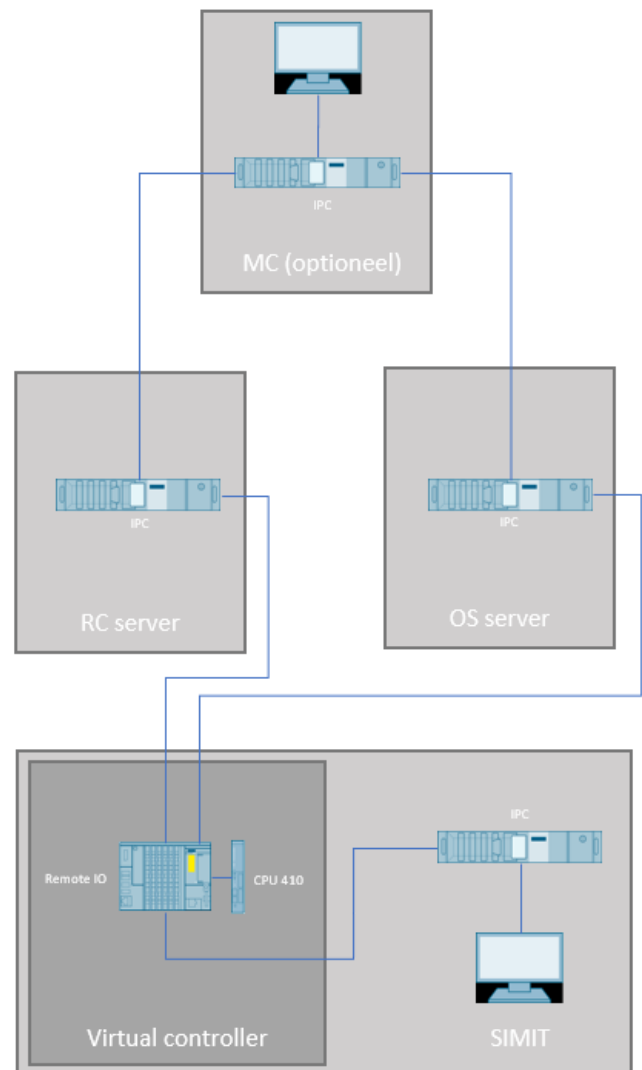
Om een keuze te kunnen maken tussen de technische oplossingen wordt er gekozen voor het eerste functioneel ontwerp. Dit resultaat is gebaseerd op de voor- en nadelen van de functionele ontwerpen en gesprekken met zowel de opdrachtgever als met Siemens. Hierdoor kan intern de logica van route control voor het automatisch testen worden verwerkt in SIMIT. Er is geen externe applicatie benodigd en de huidige PCS7 applicatie hoeft voor het testen van de routes niet worden aangepast. Dit is niet wenselijk omdat dan afgeweken wordt van de daadwerkelijke geïmplementeerde applicatie.

## 8.2 Opstelling proof of concept

Voor het realiseren van het proof of concept is een opstelling benodigd. De opstelling is als volgt opgebouwd en te zien in Figuur 9 opstelling . De opstelling is gebaseerd op het functioneel ontwerp en bestaat uit de onderdelen; SIMIT, OS server, Route control server en een Multi-client die verbonden zijn met de OS en RC servers. Deze servers zijn vervolgens weer gekoppeld aan de PLC met dit geval een gesimuleerde virtuele omgeving in SIMIT. Het toevoegen van een Multi-client is optioneel maar opgebouwd volgens het tankterminal project om eenvoudig via deze server een directe verbinding te kunnen maken met de client servers (OS en RC).

De volgende softwareprogramma's worden voor het proof of concept toegepast:

- Siemens PCS7 v8.1 engineering station
  - STEP 7
  - Route control
  - WinCC
- Siemens Route control server
  - PCS7 RC-server
  - KTP-project
- Siemens OS
  - PCS7 WinCC server applicatie
  - KTP-project PCS7
- Siemens OS MC
  - PCS7 WinCC client applicatie
- Siemens SIMIT
  - SIMIT SP v10.2
  - SIMIT CTE
  - Virtual controller
  - FLOWNET
- Benodigde licenties
  - PCS7 v8.1
  - SIMIT SP v10.2 size M
  - SIMIT CTE
  - SIMIT FLOWNET
  - SIMIT 1 virtual controller



*Figuur 9 opstelling PoC*

## 8.3 Technische oplossingen

Om het functioneel ontwerp technisch te kunnen oplossen wordt in de volgende paragrafen de technische mogelijkheden voor het proof of concept behandeld. Door deze technische mogelijkheden toe te passen, kunnen de functionaliteiten van de user-requirements worden gerealiseerd in het proof of concept.

### 8.3.1 De virtual controller (PLC)

Voor de virtual controller zijn een aantal stappen noodzakelijk om deze omgeving in te richten voor het proof of concept. In de virtual controller (ook wel de PLC) moeten de benodigde inputs en outputs van het PCS7 project worden geïmporteerd om data te kunnen communiceren tussen de OS-server en SIMIT, om het gedrag van devices zoals pompen en kleppen te kunnen simuleren. Daarnaast worden PCS 7 datablokken in de virtual controller toegepast om operatorhandelingen te automatiseren zodat de benodigde SFC van het PCS7 project doorlopen kan worden. Aangezien operatorhandelingen op het werkveld plaatsvinden moeten deze handelingen tijdens het testen worden geautomatiseerd. De verschillende commando's om operator handelingen te automatiseren worden later in dit hoofdstuk behandeld. De toegepaste SFC voor het proof of concept is de SFC die zorgt voor de vloeistof transportatie tussen twee opslagtanks.

### 8.3.2 Opbouw routes

Om een route geautomatiseerd te testen moet er kennis zijn om te weten hoe de route is opgebouwd. Een route is opgebouwd uit een source en destination nummer. Deze nummers bestaan uit integers en zijn gekoppeld aan bijvoorbeeld een tank in het terminalpark. Met deze gegevens is bij elke route bekend waar de vloeistof begint en eindigt. Daarnaast bestaat een route uit een CS-nummer. Dit nummer is benodigd om aan te geven uit wat voor control strategie de route bestaat. Aangezien er verschillende strategieën beschikbaar zijn in het project, wordt voor elke type route een andere SFC toegepast om de route succesvol te kunnen doorlopen. In het proof of concept wordt eerste instantie één strategie (van tank naar tank) getest zodat van de source tank naar de destination tank kan worden verpompt.

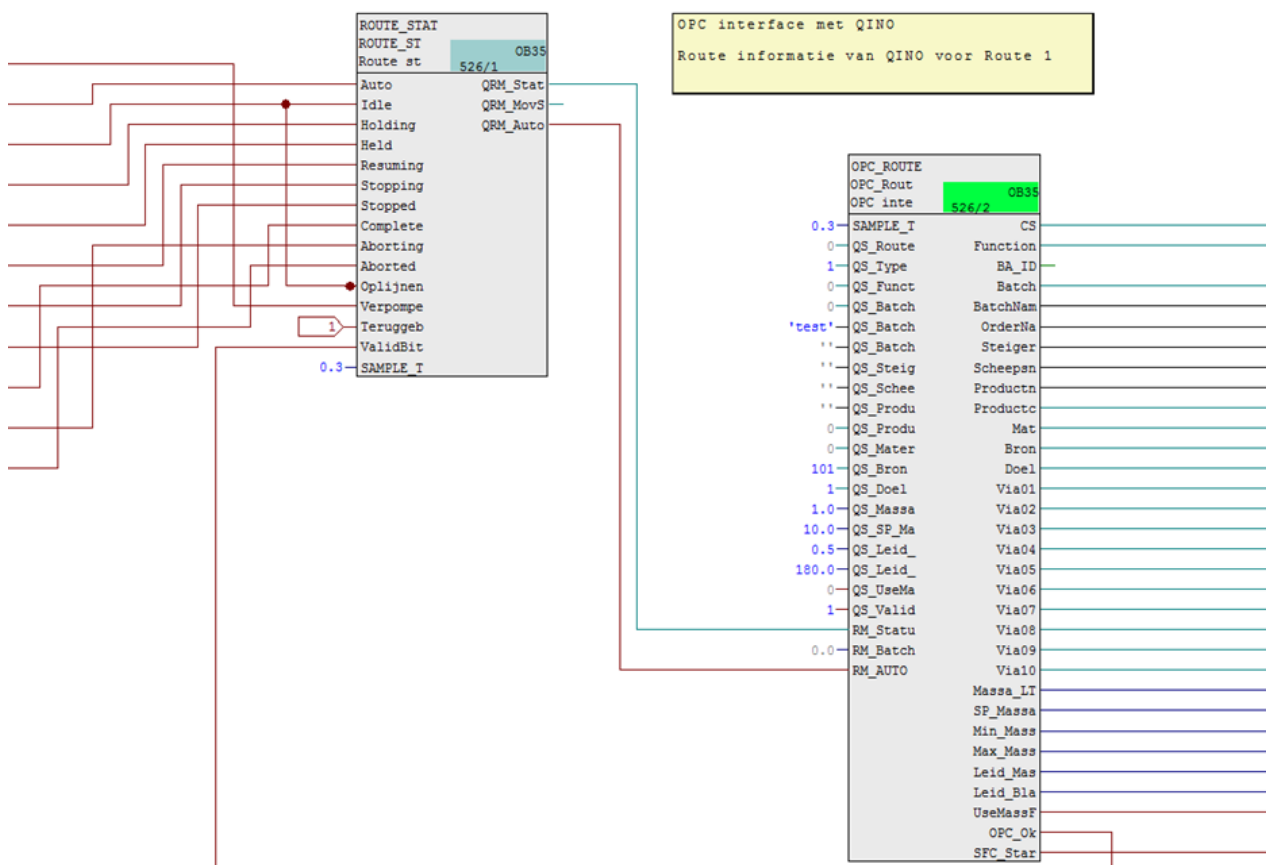
### 8.3.3 Een route starten

Om routes te starten moet een aantal commando's worden geforceerd door middel van signalen vanaf SIMIT te versturen naar de virtual controller (PLC). In de virtual controller moeten de DB's (datablokken) worden geïmporteerd van het KTP-project en er moeten S7 verbindingen worden opgezet naar de route control server. De CFC blokken worden vervolgens vanuit PCS7 de virtual controller bereikt om de geforceerde signalen van SIMIT te kunnen verwerken. In Figuur 10 zijn de CFC blokken van de OPC route uit te lezen. Om een route te starten moeten de volgende signalen worden geforceerd in het datablok van de OPC blok door middel van de signalen vanaf SIMIT te versturen naar de PLC. In Bijlage VI: Signalen staan de signalen weergegeven met de bijbehorende datatypes en adressen.

- Source van de route (QS\_bron) in datatype integer om de source van de vloeistof te kunnen aangeven.
- Destination van de route (QS\_doel) in datatype integer om de destination van de vloeistof te kunnen aangeven.

- CS-nummer (CS) in datatype integer om de soort route te kunnen bepalen.
- De massa van de vloeistof (QS\_Massa) om de massa te bepalen.
- Routes in automaat zetten (RM\_Auto).
- Via's, om vloeistoffen over een specifieke pijpleiding te kunnen transporteren.

Er wordt voor het OPC blok in het PCS7 project gekozen aangezien het gekoppeld is aan zowel het SFC blok als met route control. Hierdoor is het eenvoudig om in één blok verschillende signalen te forceren en is maar één datablok benodigd.



Figuur 10 OPC route

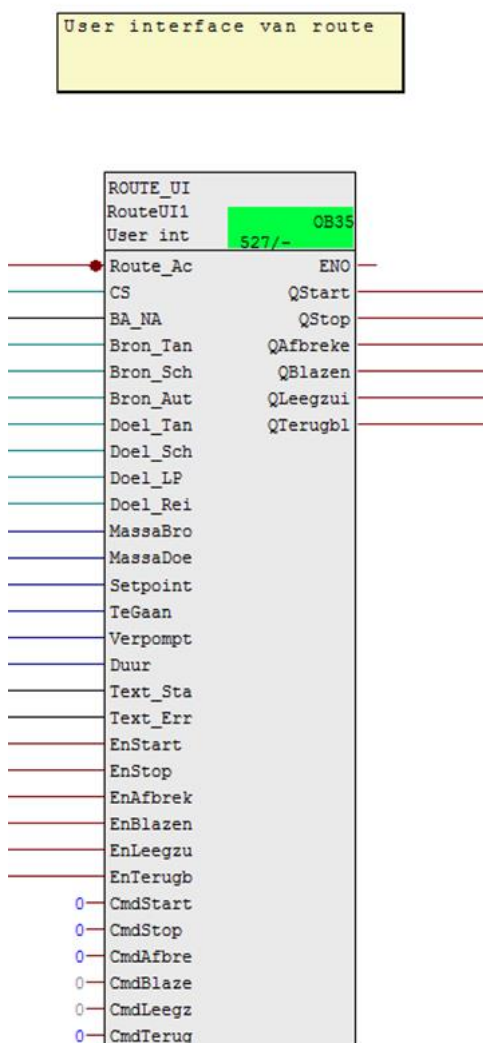
De verschillende mogelijke statussen van de route zijn uit te lezen in de ROUTE\_STAT blok in Figuur 11 route UI. Het uitlezen van de status wordt via het input signaal RM\_Status van de OPC\_route blok uitgelezen. Door het terugkoppelen van het input signaal via de PLC wordt de status uitgelezen van de route. Hierdoor weet SIMIT in welke status de route zich bevindt. Ook de massa en de productietijd van de OPC\_Route blok wordt teruggekoppeld naar SIMIT.

Om een route te kunnen starten moet operator signalen die normaal op het werkveld worden uitgevoerd worden geautomatiseerd. Deze signalen worden naar de ROUTE\_UI blok verzonden waarin

operator commando signalen worden verwerkt. Het is noodzakelijk om dit blok te gebruiken voor het proof of concept om operator handelingen te automatiseren en de SFC te kunnen doorlopen.

Omschrijving gebruikte signalen in het ROUTE\_UI blok in het proof of concept zijn als volgt:

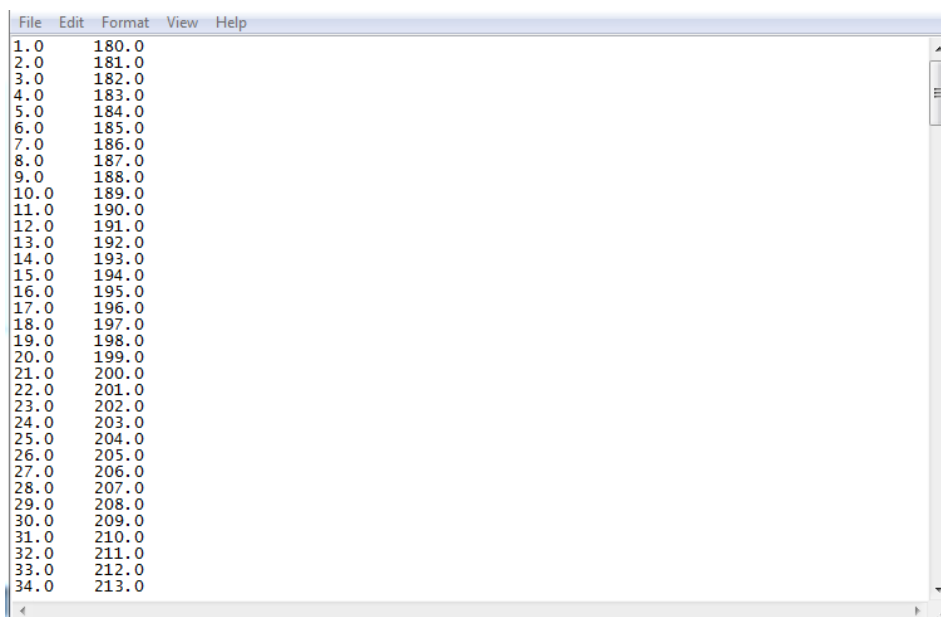
- Indien het starten gereed is middels het signaal (EnStart), dan kan een startcommando gegeven worden via CmdStart.
- Indien benodigd het stoppen van de route (CmdStop).



Figuur 11 route UI

## 8.4 Automatisch testen met SIMIT

Om routes geautomatiseerd te kunnen testen is onderzocht op welke manier dit technisch moet werken in SIMIT. Om een route te starten moeten de routes via een tekstbestand worden ingeladen in een SIMIT-component, om de gegevens van alle routes te kunnen verwerken en de routes één voor één kan worden getest. Om de routes te kunnen inladen moet de source en destination nummer van een route worden gekoppeld aan een unsigned integer. Hierdoor weet SIMIT in welke volgorde de routes worden getest en welke source en destination nummers op dat moment doorgegeven moeten worden naar de PLC om de juiste route te kunnen testen. De routes worden geïmporteerd in drie SIMIT diagram componenten. Het geïmporteerde bestand bevat een ID van de route en de source of destination nummer en is te zien in Figuur 12 routes importeren. Door middel van een unsigned integer kunnen de source en destination nummers van een route worden opgehaald.

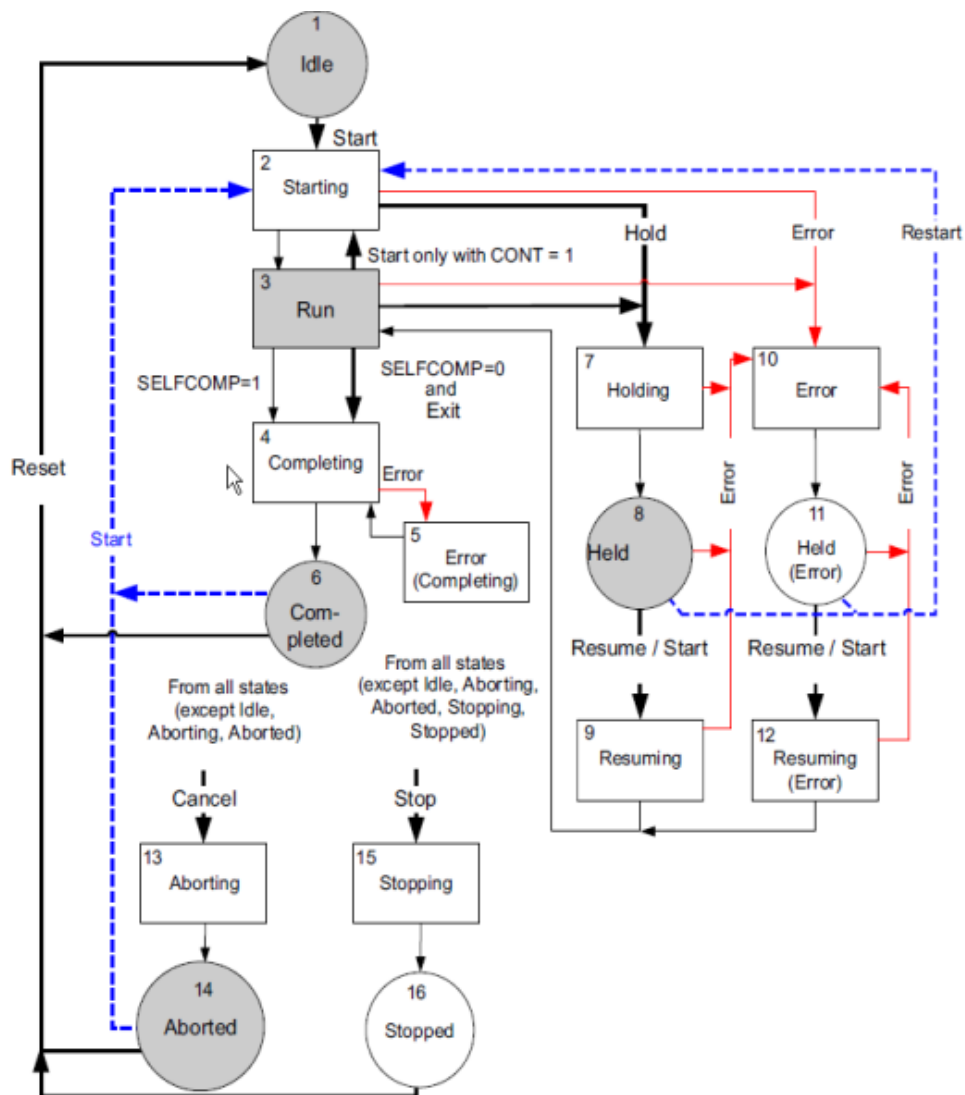


1.0	180.0
2.0	181.0
3.0	182.0
4.0	183.0
5.0	184.0
6.0	185.0
7.0	186.0
8.0	187.0
9.0	188.0
10.0	189.0
11.0	190.0
12.0	191.0
13.0	192.0
14.0	193.0
15.0	194.0
16.0	195.0
17.0	196.0
18.0	197.0
19.0	198.0
20.0	199.0
21.0	200.0
22.0	201.0
23.0	202.0
24.0	203.0
25.0	204.0
26.0	205.0
27.0	206.0
28.0	207.0
29.0	208.0
30.0	209.0
31.0	210.0
32.0	211.0
33.0	212.0
34.0	213.0

Figuur 12 routes importeren

Op de volgende pagina in Figuur 13 schema statussen staat een schematische weergave van alle statussen van de SFC en route control zonder SIMIT inbegrepen. Hierin is te uit te lezen welke statussen actief worden indien een route wordt doorlopen. Deze informatie is belangrijk aangezien vervolgens SIMIT gekoppeld moet worden aan het PCS7 KTP project.

Volgens het flowdiagram wordt een normale route doorlopen via de volgende doorlopende stappen: Idle > starting > Run > completing > completed > idle. Indien een operator op de knop holding drukt zal het proces in de status 'holding' gaan en na een bepaalde tijd naar de 'held' status. Vanaf dit punt kan het proces weer verder worden doorlopen indien op de 'resume' knop wordt gedrukt. Dit geldt ook indien een error ontstaat tijdens de route. Vanaf dit punt kan het proces weer gestart worden wanneer dit door de operator aangegeven wordt. Ook is het als operator altijd mogelijk om tijdens het proces op stop of cancel te drukken om het proces volledig af te breken.

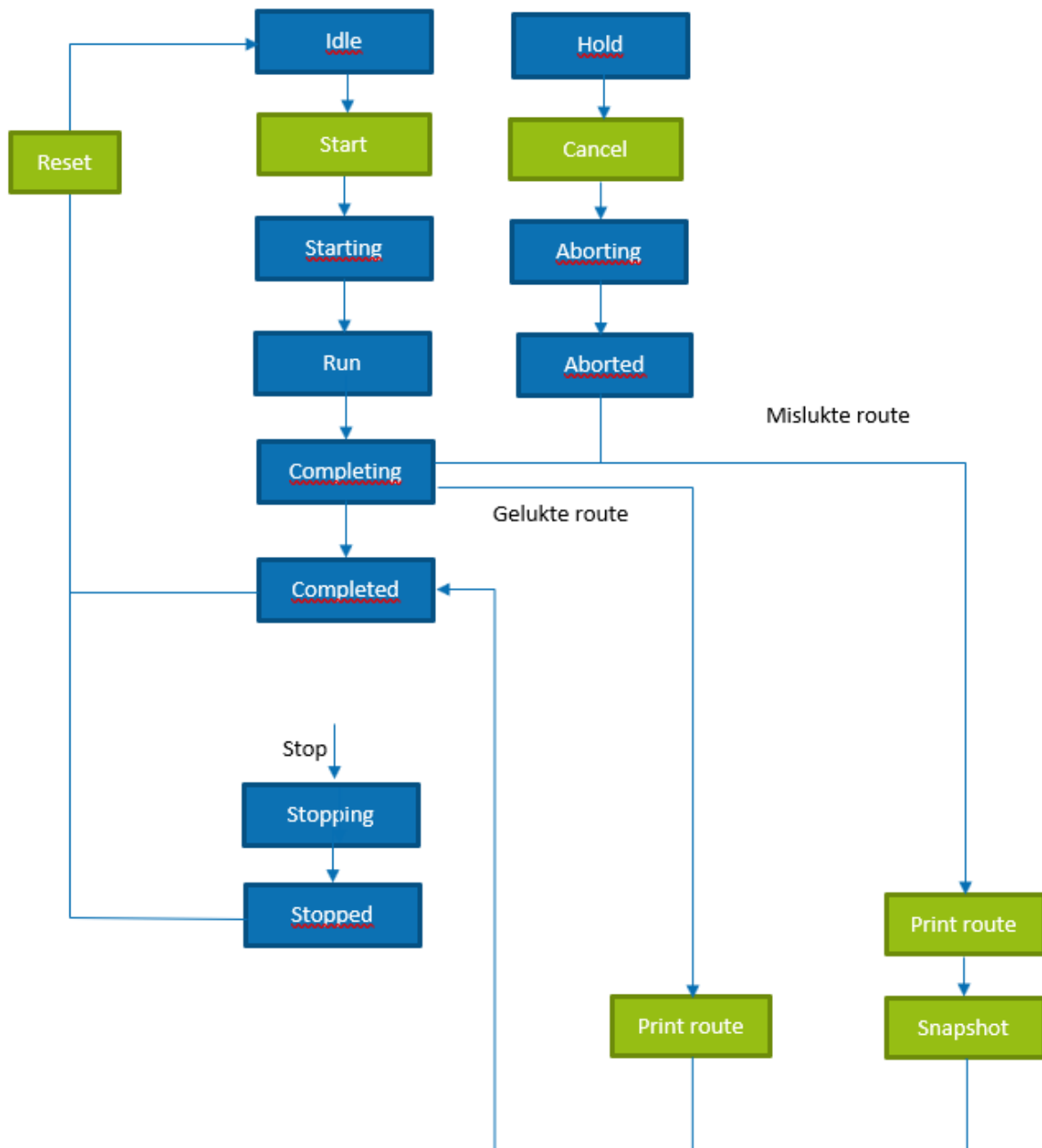


Figuur 13 schema statussen (Automation, 2018)

Door het toepassen van SIMIT in het flowdiagram is het noodzakelijk om te bepalen op welke status van een route een operator handeling uitgevoerd moet gaan worden voor het automatisch testen. In de volgende alinea is gekozen voor de volgende oplossing aangezien dit het meest efficiënt is om te controleren of de route volledig is doorlopen.

Indien vloeistof een route heeft gevolgd en komt in status 'completing', dan zal SIMIT deze status uitlezen en bepalen of de route is gelukt of mislukt. Vervolgens worden de routes gerapporteerd in een tekstbestand en zal er een snapshot worden gemaakt bij een mislukte route. Dit gebeurt middels het controleren of de juiste destination tank is gevuld. Indien een route wordt 'aborted' dan zal dit resulteren in een mislukte route door middel van een error tekst. Bij een status stop wordt het testen gestopt. Dit resulteert in het volgende flowdiagram op de volgende pagina. De blauwe acties is geprogrammeerd in het PCS7 KTP project en de groene acties worden uitgevoerd in SIMIT.





Figuur 14 overzicht met SIMIT

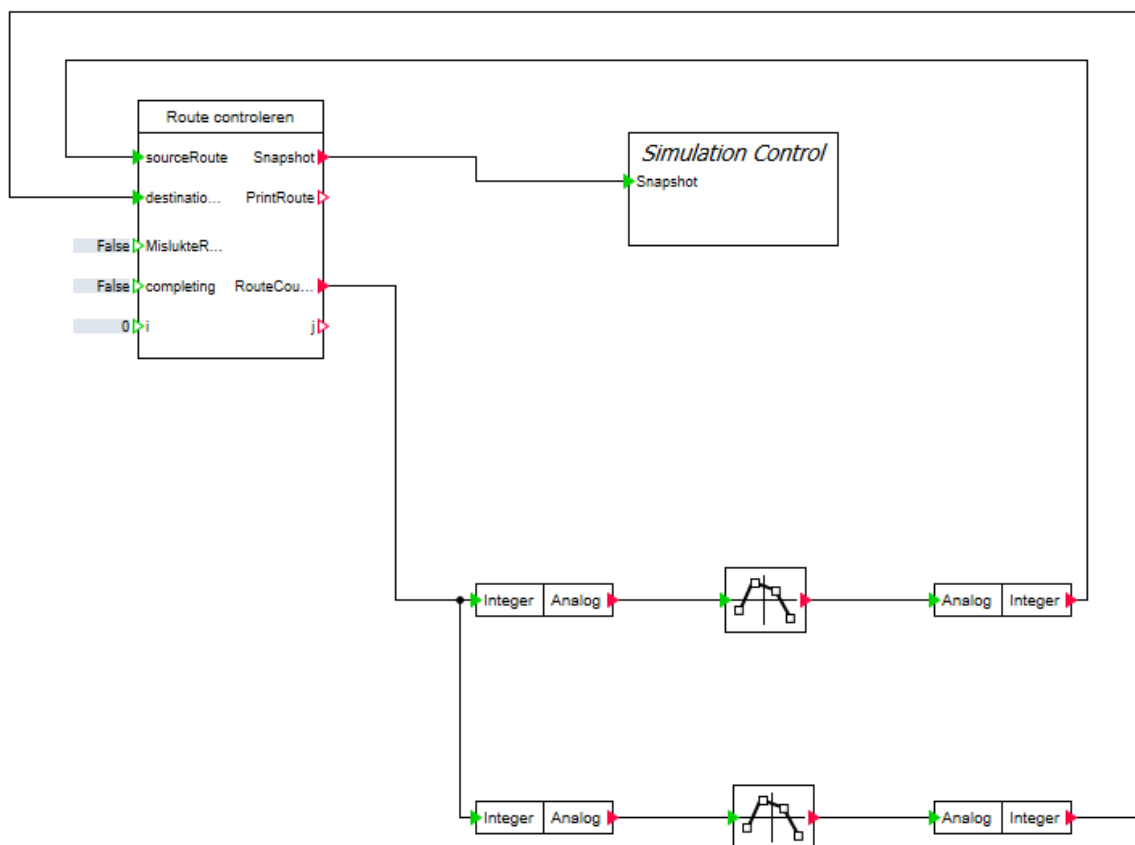
#### 8.4.1 Resetten

Na het verpompen van de vloeistof moet na deze fase de SFC van de route worden gereset. Dit wordt uitgevoerd door middel van een 'aborted' signaal te verzenden naar de PLC zodat de route afgebroken wordt en de SFC teruggezet wordt naar het startpunt. In verband met een functionaliteit 'terugblazen' in elke route, is gekozen om deze functionaliteit niet te testen zodat tijdwinst wordt behaald en dus het programma af te breken in plaats van volledig te doorlopen. Daarnaast moet de volgende route worden gebruikt in het diagram component in Figuur 15 ontwerp rapportage door het optellen van een integer om de volgende route te kunnen starten.

### 8.4.2 Routes testen en rapporteren

Indien een route mislukt moet in SIMIT een signaal afgegeven worden naar een aangemaakte 'Route controleren' CTE-blok dat hieronder is weergegeven in Figuur 15 ontwerp rapportage. In dit datablok wordt in de CTE-cyclus functie een klein script geschreven waarin de routes worden gelogd in een tekstbestand.

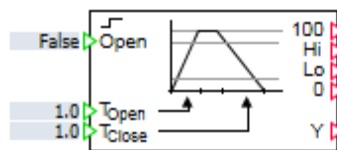
Ook wordt bij een mislukte route een snapshot aangemaakt in de simulation control blok zodat de engineer kan inzien wat er mis is gegaan bij de route. Met een snapshot is het mogelijk om in zowel SIMIT als het PCS7 project terug te gaan naar het moment dat een mislukte route wordt gerapporteerd. De snapshot wordt aangemaakt indien de route mislukt is en de status op 'completing' staat. Tenslotte wordt de volgende route gepakt door middel van het optellen van de RouteCounter output om de gegevens van de volgende route te kunnen gebruiken. Een route mislukt indien vloeistof in een niet gewenste destination tank is gevuld. Dit is bijvoorbeeld mogelijk indien tijdens het testen de verkeerde kleppen worden aangestuurd of een error tekst zichtbaar wordt.



Figuur 15 ontwerp rapportage

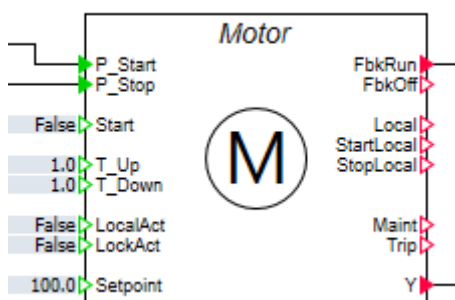
## 8.5 Device laag

Het device level is een laag tussen het proces level en de signal level (PLC). De alarmen en sensoren van de gehele testomgeving moeten worden voorzien van een gezonde status middels deze objecten te voorzien van een waarde om routes te kunnen testen. Ook wordt in deze laag het gedrag van verschillende devices gesimuleerd zoals pompen en kleppen die toegepast worden in het KTP testproject. In het proof of concept moeten een aantal kleppen open en dicht kunnen door middel van de signalen uit de PLC. Dit zal door middel van een drive component worden gesimuleerd dat hieronder in Figuur 16 drive component staat weergegeven. Elke klep en pomp heeft zijn eigen drive in SIMIT dat ervoor zorgt dat componenten run-time kan worden gesimuleerd. Bij de inputs worden de output signalen van PCS7 aangesloten en de output zorgt voor de benodigde feedback dat teruggestuurd wordt naar de PLC.



Figuur 16 drive component

Om de pomp te kunnen simuleren is een motorblok benodigd in het device level. Dit motorblok wordt gestart indien een signaal wordt ontvangen vanaf de virtual controller (PLC). Hierop wordt gereageerd in het device level om de pomp te kunnen starten. De uitgangen van het motorblok wordt vervolgens toegepast om een feedbacksignaal terug te sturen naar de PLC en de waarde (Y) voor het FLOWNET pump component.

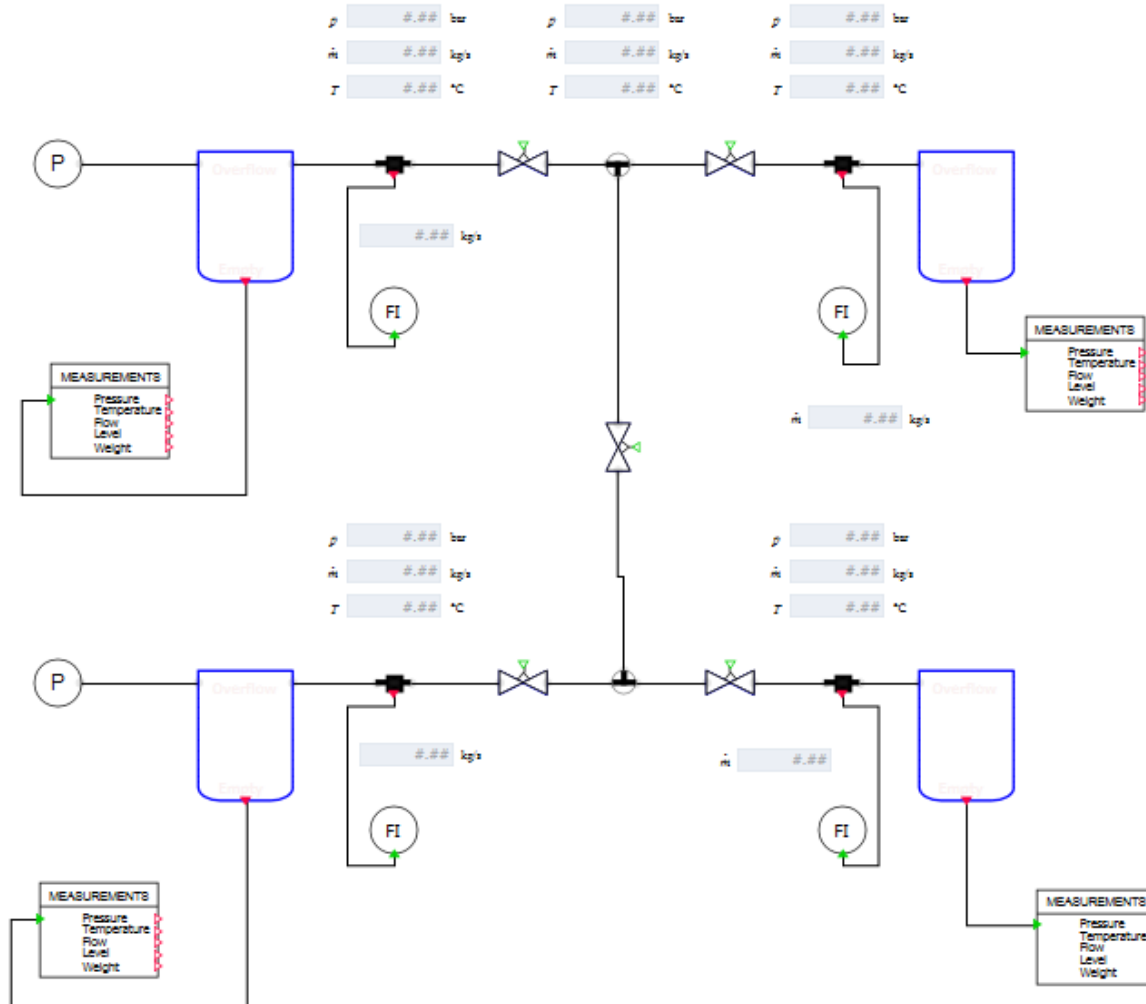


Figuur 17 motor pump component

## 8.6 Proces laag

In de proces laag wordt de flow van de vloeistof gevisualiseerd met de FLOWNET componenten. Hieronder in Figuur 18 ontwerp FLOWNET is een ontwerp te zien die voor het proof of concept wordt toegepast. De blauwkleurige symbolen zijn de opslagtanks. Hierin wordt de druk, temperatuur, flow, level, en gewicht van de tank gemeten via het measurement component uit de FLOWNET library. De informatie van de opslagtanks worden teruggestuurd naar de virtual controller.

Indien de simulatie wordt gestart worden de bovenste opslagtanks gevuld om de vloeistof van de source naar de destination tank te kunnen verpompen. Tussen de opslagtanks zijn de kleppen weergegeven met als doel om deze te openen indien de juiste kleppen worden ingeschakeld. Verder zijn via de zwarte measurements componenten de druk te meten in de pijpleidingen. Het gedrag van het ontwerp is gebaseerd op de resultaten van route control. Deze signalen worden gecommuniceerd naar de PLC en zijn toegankelijk voor SIMIT om in de P&ID te kunnen visualiseren.

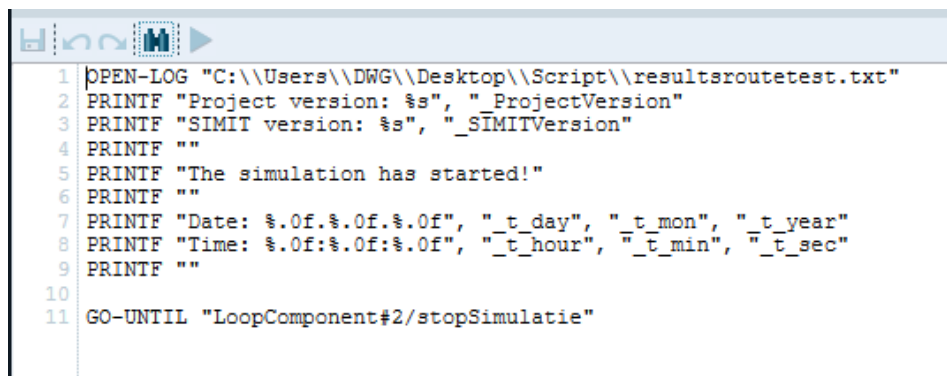


Figuur 18 ontwerp FLOWNET

### 8.6.1 Scripting

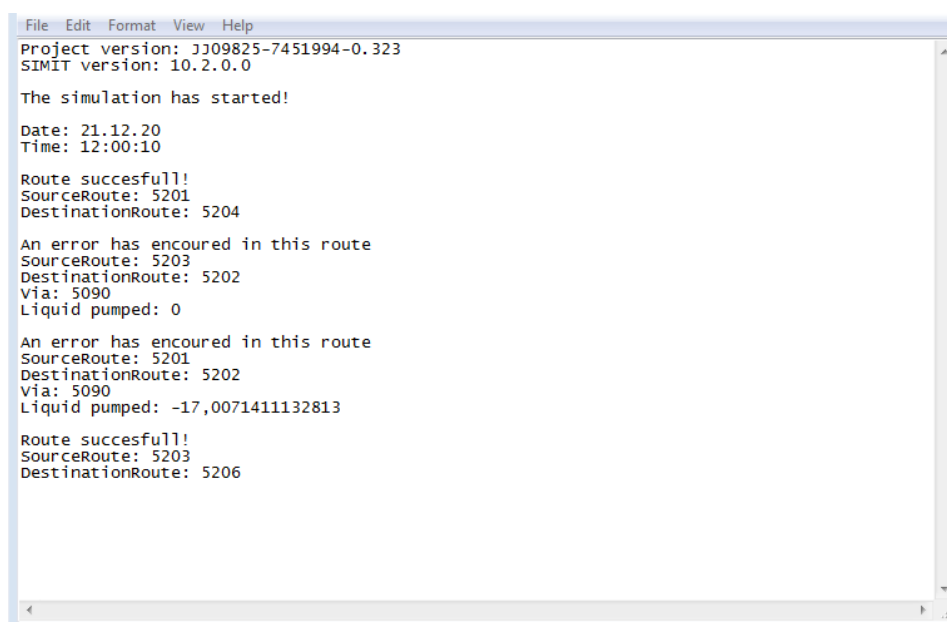
Tijdens het testen van de routes moet de communicatie naar het tekstbestand altijd actief blijven zodat het resultaat van de routes direct via het 'routes controleren' blok geschreven kan worden. In SIMIT is een optie waarin één script tegelijk uitgevoerd kan worden voor een onbepaalde tijd. Dit zorgt ervoor dat tijdens het testen van de routes altijd een tekstbestand geopend kan blijven waarin het resultaat van de route kan worden weggeschreven.

Het openen van het tekstbestand werkt via de scripting functie in SIMIT en blijft altijd actief gedurende het testen van de routes. Dit werkt via een loop om de routes in één tekstbestand te kunnen loggen. Hierdoor is het altijd mogelijk om een nieuwe route toe te voegen in een tekstbestand en zullen de geteste routes onder elkaar komen te staan. Om dit te kunnen realiseren wordt de OPEN-LOG functie toegepast om een bestand te kunnen openen. Het rapporteren van routes wordt via een eigen CTE component geprogrammeerd en wordt in dezelfde notepad bestand gerapporteerd. Dit zelfgemaakte component is in paragraaf 8.4.2 beschreven. Indien een route correct of niet correct is getest worden beide gevallen gerapporteerd in het bestand.



```
1 OPEN-LOG "C:\\Users\\DWG\\Desktop\\Script\\resultsroutetest.txt"
2 PRINTF "Project version: %s", "_ProjectVersion"
3 PRINTF "SIMIT version: %s", "_SIMITVersion"
4 PRINTF ""
5 PRINTF "The simulation has started!"
6 PRINTF ""
7 PRINTF "Date: %.0f:%.0f:%.0f", "_t_day", "_t_mon", "_t_year"
8 PRINTF "Time: %.0f:%.0f:%.0f", "_t_hour", "_t_min", "_t_sec"
9 PRINTF ""
10
11 GO-UNTIL "LoopComponent#2/stopSimulatie"
```

Figuur 19 openen/aanmaken tekstbestand



```
File Edit Format View Help
Project version: JJ09825-7451994-0.323
SIMIT version: 10.2.0.0

The simulation has started!

Date: 21.12.20
Time: 12:00:10

Route succesfull!
SourceRoute: 5201
DestinationRoute: 5204

An error has encoured in this route
SourceRoute: 5203
DestinationRoute: 5202
Via: 5090
Liquid pumped: 0

An error has encoured in this route
SourceRoute: 5201
DestinationRoute: 5202
Via: 5090
Liquid pumped: -17,0071411132813

Route succesfull!
SourceRoute: 5203
DestinationRoute: 5206
```

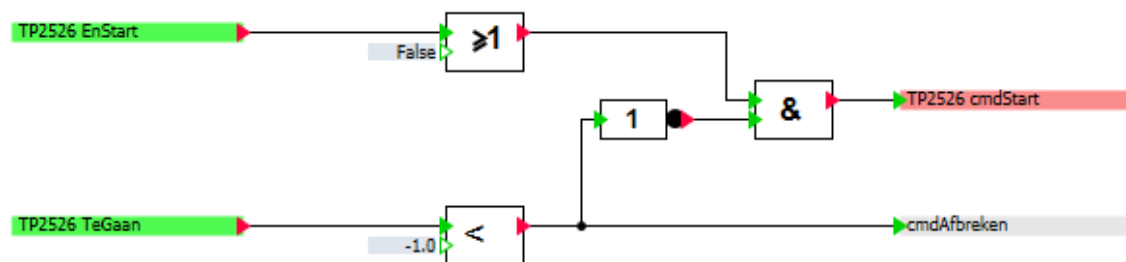
Figuur 20 rapportage tekstbestand

## 9 Proof of concept

In de fase van het proof of concept is de applicatie gerealiseerd volgens het functioneel & technisch ontwerp. In dit hoofdstuk zijn naast het realiseren van de applicatie ook te lezen welke onderdelen zijn toegevoegd ten opzichte van het ontwerp in het functioneel en technisch document. Deze veranderingen zijn tijdens de realisatie van de applicatie voortgekomen en worden in dit hoofdstuk met argumentatie onderbouwd.

### 9.1 Operator handelingen

Het verpompen van vloeistof wordt in het werkveld door operator handelingen aangestuurd. Tijdens het geautomatiseerd testen van routes moeten deze operator handelingen ook worden geautomatiseerd. Dit wordt in de simulatieomgeving uitgevoerd middels commando signalen te versturen naar de PLC. Hieronder in Figuur 21 worden twee verschillende operator handelingen aangestuurd indien de voorwaarden waar is. CmdStart wordt waar wanneer EnStart (de start knop) zichtbaar wordt en indien het aantal verpompingen (in waarde aantal tons) niet kleiner is dan negatieve getal -1. Aangezien het aantal liters niet stopt wanneer TeGaan een nul waarde heeft moet dit afgehandeld worden in SIMIT. In het werkveld wordt na het verpompen nog de leiding schoongebazen. Echter, is besloten om na het verpompen de route af te breken zodat tijd wordt bespaard en een nieuwe route kan worden gestart.



Figuur 21 commando signalen

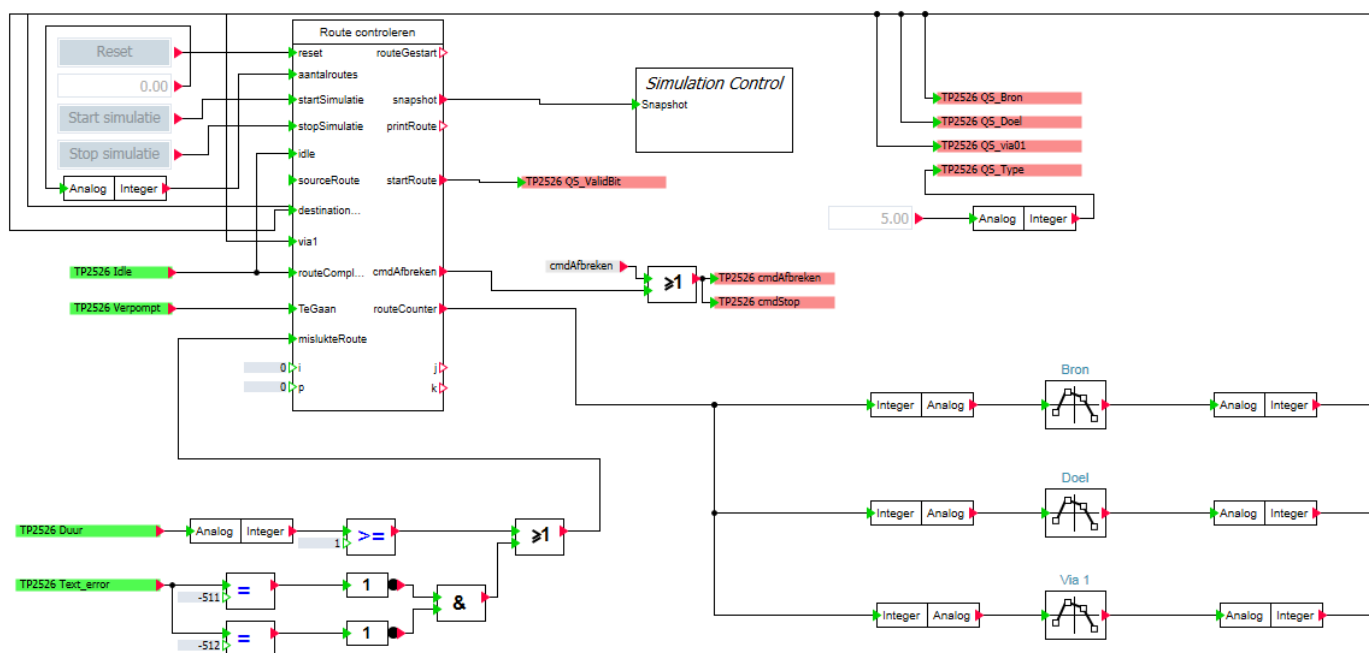
### 9.2 Routes testen en rapporteren

Het resultaat van routes testen en rapporteren is ook uitgebreider in vergelijking met het functioneel & technisch ontwerp. Er is besloten om het starten, stoppen, resetten van parameters en het selecteren van aantal routes te verwerken in het route controleren blok. Hierdoor is het mogelijk om eenvoudig aan alle voorwaarden in te voldoen voordat de simulatie kan worden gestart. Het route controleren blok is ontwikkeld in de CTE (component type editor) omgeving. Om te reageren op de binnenkomende input signalen is dit in een SIEMENS taal geprogrammeerd. De code bevindt zich in Bijlage VII: Route controleren blok.

Daarnaast zijn scenario's toegevoegd om mislukte routes te kunnen ontdekken en rapporteren. Onder het route controleren blok staan twee voorwaarden voor een mislukte route. Ten eerste is een route mislukt indien er een flowfout ontstaat in de geteste route, waardoor niet via de pomp vloeistof kan

worden verpompt. Dit is niet opgevangen in het PCS7 project waardoor gekozen is om de route te laten afbreken na een tijdslimiet van één minuut. De tijdsduur van het simuleren van het verpompen van vloeistof bij elke route is altijd onder de één minuut. Daarnaast wordt een mislukte route gerapporteerd indien de route een tekst error weergeeft. Aangezien op het SCADA systeem geen strings uit de PLC gehaald kan worden is het omgezet naar een integer en wordt de waarde hiervan vergeleken. In Figuur 22 aanpassingen routes inladen/controleren staat onder het route controleren blok een tekst error output waarin wordt vergeleken of de lengte van de string niet de waarde -511 of -512 heeft zodat kan worden bepaald of een tekst error is weergegeven.

Ook is besloten om in dit ontwerp de via's toe te voegen. Met via's is het mogelijk om de vloeistof een bepaalde route van de installatie te laten verpompen. Minstens een via bleek nodig te zijn in het ontwerp om een juiste test uit te kunnen voeren. Aangezien bij elke route verschillend is welke via toegepast moet worden is dit net als de bron en doel in een diagram component opgevangen en moet dit voor het testen van de routes geïmporteerd worden via een notepad bestand om de routes te kunnen testen.



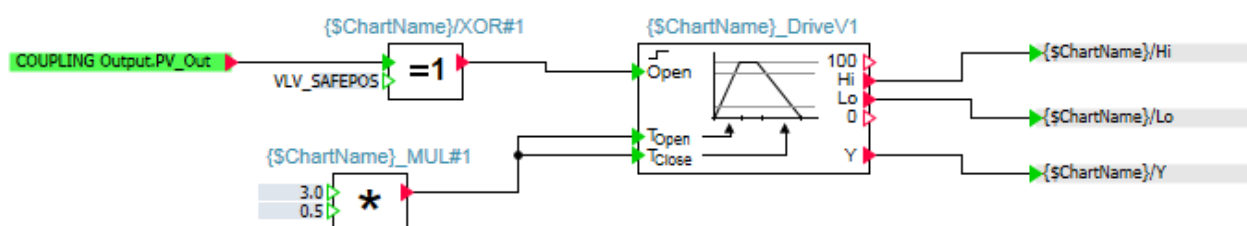
Figuur 22 aanpassingen routes inladen/controleren

### 9.3 Device laag

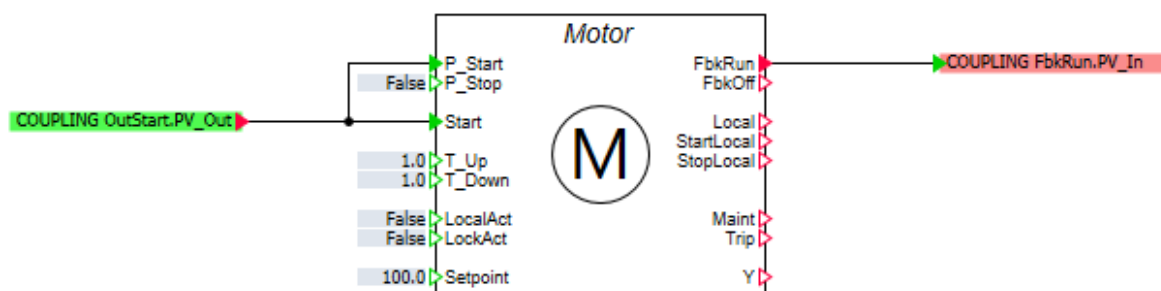
Zoals te lezen was in het functioneel & technisch ontwerp wordt in het device laag twee componenten toegepast om kleppen en pompen te kunnen visualiseren. Deze componenten zijn tijdens het proof of concept in een template ontworpen. Hierdoor kunnen de hoeveelheden kleppen en pompen worden geïmporteerd in SIMIT met de juiste output en feedback signalen. Door middel van de brackets en coupling bij de signalen weet SIMIT dat deze signalen moeten worden vervangen naar de naam en signaal van de klep. Hierdoor kunnen de hoeveelheden kleppen in één keer in SIMIT worden

geïmporteerd. Dit zal een engineer tijdens het proces veel tijd besparen aangezien bij een tank terminal gebruik wordt gemaakt van vele honderden kleppen en pompen.

In dit project wordt niet gebruik gemaakt van de feedback signaal aan de uitgang van de drive blok aangezien de werkelijke kleppen van dit testproject geen feedback signaal benodigd heeft. Wel kunnen door middel van de Topen en Tclose input signalen van het drive blok de klep open en dicht worden geforceerd binnen een bepaalde tijdsperiode. Ook heeft de klep een veiligheidssignaal (safe position) die ervoor zorgt dat de klep niet geopend kan worden indien deze input waar is bij een ongewenste situatie. Het feedback signaal wordt door SIMIT wel toegepast om de proces laag aan te kunnen sturen.



Figuur 23 template kleppen



Figuur 24 template pomp

## 9.4 Proces laag

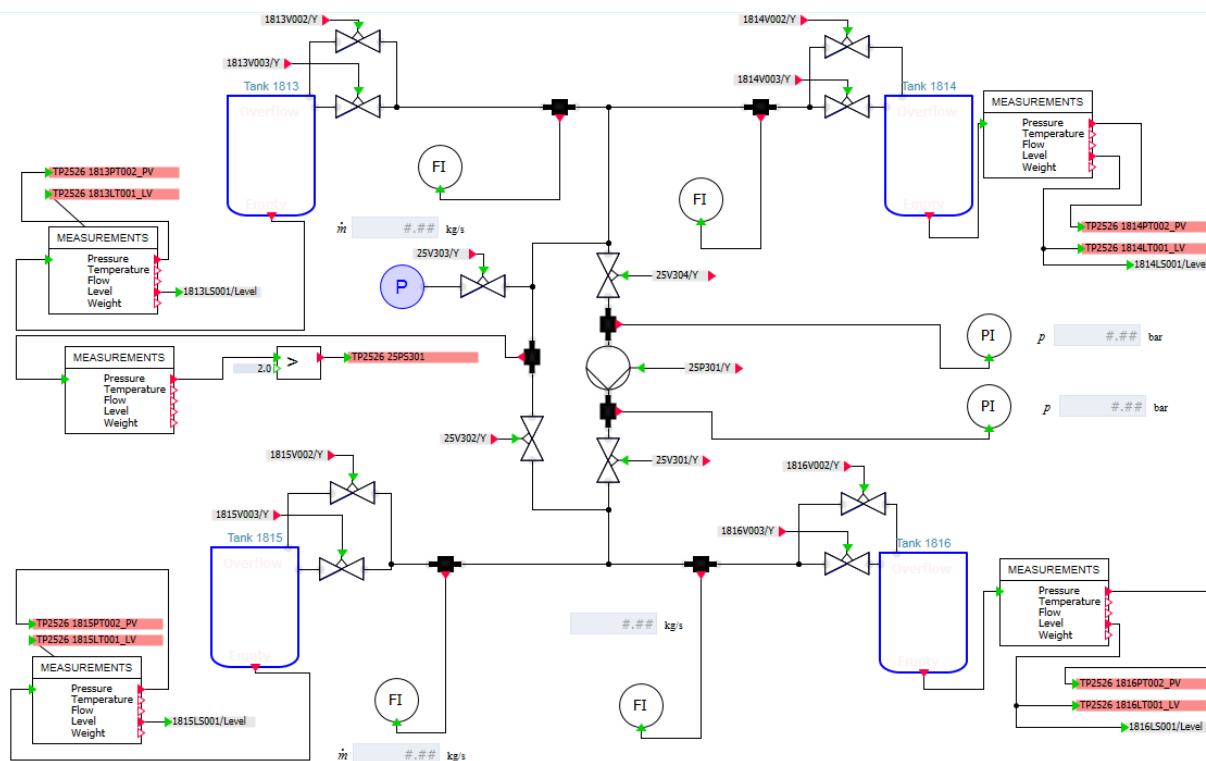
Het FLOWNET P&ID ontwerp is ten opzichte van het functioneel & technisch ontwerp uitgebreid. Het ontwerp is aangepast volgens het geteste PCS7 tankterminal project. Hiervoor is gekozen om een zo realistisch mogelijke weergave in SIMIT te creëren. Het ontwerp is opgebouwd uit twee kleppen per tank om de vloeistof in en uit de tank te kunnen verpompen. Daarnaast moet een route altijd door de koppelplaats worden verpompt om de vloeistof van de source naar de destination tank te kunnen verpompen. De koppelplaats is na het functioneel & technisch ontwerp toegevoegd aangezien tijdens het proof of concept van testproject werd veranderd. De reden van deze verandering is omdat dit testproject al direct klaarstond om in gebruik te kunnen nemen. (Automation, 2018)

De koppelplaats is het middelste gedeelte van het ontwerp en is aangepast volgens de WinCC weergave op de OS server van het PCS7 project. Hierin is ook een Pnode (blauwe p symbool) te zien dat stikstof in de leiding verpompt om twee druktesten uitvoeren te voeren voordat vloeistof door de leiding wordt verpompt. Deze druktesten test of er geen lekkage in de leiding is door middel van de



druk te verhogen voor een tijdsperiode van 30 seconde. Indien de druktest is geslaagd zal de druk worden verlaagd en de vloeistof volgens de route worden verpompt.

Om vloeistof te kunnen verpompen wordt in SIMIT de opslagtanks gevuld middels het invoeren van startwaardes. Hierin wordt een percentage vloeistof meegegeven zodat de route succesvol de vloeistof van de source naar de destination kan verpompen. Indien de simulatie in status online staat kunnen de gegevens van de tank worden uitgelezen via de measurement component zodat de informatie doorgestuurd kan worden naar de PLC en de tank virtueel is gevuld. Dit wordt gedaan via de input signalen van de virtual controller die gekoppeld zijn aan het measurement component.



Figuur 25 aanpassingen FLOWNET

## 9.5 Testplan

Om de applicatie op een zorgvuldige manier te kunnen testen is er een testplan gerealiseerd en doorlopen met de opdrachtgever tijdens de FAT (factory acceptance test). Het testplan is te vinden in Bijlage VIII: Resultaten FAT. Hierin staat beschreven wat de testresultaten zijn met daarbij de opmerkingen van de opdrachtgever. Uit het testplan kan geconcludeerd worden dat het proof of concept is geslaagd. De opdrachtgever is tevreden met het resultaat en er worden vervolgstappen ondernomen om het proof of concept tot een echt product te laten ontwikkelen.

## 10 Vervolgstappen

In dit hoofdstuk worden de vervolgstappen beschreven van de resultaten van het algehele afstudeertraject. De documenten zijn afgerond en het proof of concept is gerealiseerd. Middels het advies wordt beschreven welke vervolgstappen benodigd zijn om het proof of concept verder uit te werken naar een compleet product zodat de testsimulatie toegepast kan worden voor de toekomstige DWG projecten.

### 10.1 Het advies

Nadat het vooronderzoek was afgerond is besloten dat de focus gelegd ging worden op de functionaliteiten geautomatiseerd testen van route control en het genereren van rapportages. Deze functionaliteiten zijn vervolgens functioneel en technisch opgelost in een document en gerealiseerd in SIMIT als een proof of concept. Uit de resultaten van het testplan kan geconcludeerd worden dat het proof of concept is geslaagd. Hierdoor kunnen vervolgstappen worden ondernomen en is hieronder uitgewerkt in het volgende advies:

1. Het P&ID ontwerp bestaat momenteel uit vier opslagtanks waaruit vier verschillende routes kan worden getest. Echter, er moet tijd gereserveerd worden om de gehele tankterminal in SIMIT te realiseren. Er moet een nieuw P&ID ontwerp worden ontworpen en de SIMIT componenten moeten met de juiste instellingen worden voorzien om grote hoeveelheden routes te kunnen testen. Daarnaast moeten de alarmen en sensoren van de gehele tankput worden voorzien van een gezonde status.
2. Om het proces van het vorige advies versnelt te kunnen uitwerken, is het wenselijk om te onderzoeken of het mogelijk is om templates te maken van gedeeltelijke FLOWNET ontwerpen met bijvoorbeeld COMOS. Indien dit mogelijk is kan er veel tijd bespaard worden om volledige tankputten te importeren in SIMIT.
3. Momenteel biedt het proof of concept verschillende scenario's om configuratiefouten te achterhalen. Echter, is het noodzakelijk om te investeren in het uitwerken van meerdere mislukte scenario's om de kwaliteit van de software te verhogen. Deze verschillende scenario's moeten worden onderzocht om vervolgens toe te passen in SIMIT. De technische uitdagingen om dit te realiseren zijn opgelost maar de tijd die benodigd is om het uit te werken moet hiervoor worden gereserveerd.
4. Voor het rapporteren van routes is gekozen voor een oplossing om de resultaten op te slaan in een tekstbestand. Echter, dit is niet optimaal indien in grote omvang routes worden getest. Het is wenselijk om de testresultaten in een CSV-bestand te exporteren of een eigen rapportage interface te ontwikkelen via het OPC protocol om de testresultaten beter te kunnen uitlezen en de scripting snapshot optimaal te kunnen gebruiken bij een mislukte route. Hierdoor kan de engineer dit makkelijk omzetten in een excel format en zo kan sorteren op mogelijke afwijkingen.

5. Momenteel is één VIA toegevoegd en wordt de type strategie statisch met een integer ingevuld. Om in de toekomst verschillende strategieën te testen om in plaats van tank > tank naar schip > tank te testen, is het wenselijk om het type strategie in het route controleren blok te verwerken zodat in het SIMIT diagram component de strategieën geïmporteerd kunnen worden. Dit is benodigd om meerdere via's te kunnen toevoegen in de applicatie. Dit betekent ook dat de operator handelingen in andere route strategieën moeten worden aangepast in SIMIT.
6. Operator trainingen is meegenomen in de user-requirement als optioneel maar was vanwege de tijdslimiet niet mogelijk om verder uit te werken. SIMIT is ontworpen om operator trainingen te geven en is technisch mogelijk om verder in de toekomst uit te werken. De functionaliteit die nog toegevoegd moet worden voor operator trainingen, is ervoor te zorgen dat tijdens de operator training in SIMIT scenario's worden geforceerd waarop de operator in zijn eigen applicatie moet kunnen reageren. Hierdoor kan in SIMIT een realistische weergave worden gemaakt van de werkelijkheid in het werkveld.

# 11 Beroepstaken

## 5. Beroepstaken

- **A-1 Analyseren probleemdomein & opstellen probleemstelling**
  - o **Actie:** het opstellen van documenten & verslagen.

Voor het gehele afstudeertraject zijn documenten opgeleverd aan de opdrachtgever volgens de planning die beschreven staat in het plan van aanpak paragraaf 6.3 Globale planning. De documenten zijn in de volgende volgorde opgeleverd:

- Een plan van aanpak
- Het vooronderzoeksdocument
- Het Functioneel & Technisch ontwerp
- Een testplan
- Een adviesrapport
- Afstudeerdossier

Verder is een proof of concept opgeleverd in SIMIT met de user-requirements die zijn opgesteld in overleg met de opdrachtgever.

- **B-5 Adviseren over inrichting ICT-gerelateerde oplossingen en processen**
  - o **Acties:** Het maken van afwegingen tijdens het vooronderzoek, en ontwerpen van de simulatieomgeving met hierbij een gemotiveerde onderbouwing.

In het afstudeertraject zijn een aantal afwegingen gemaakt om tot het resultaat van het proof of concept te kunnen komen. Ten eerste het vooronderzoek waarin de meeste afwegingen zijn gemaakt. In deze fase zijn de vraagstellingen beantwoord en de mogelijkheden in SIMIT onderzocht. Uit deze resultaten ben ik tot de conclusie gekomen dat de onderdelen route control en rapportage de beste mogelijkheden hadden om het proof of concept te laten slagen. De onderbouwing hiervoor is dat ik tijdens het vooronderzoek te weten ben gekomen dat de connectie vanaf SIMIT versie 9.0 naar route control gesupport wordt. Ook ben ik tot de conclusie gekomen dat het mogelijk is om CFC blok signalen te forceren in een PCS7 project middels de virtual controller in SIMIT waardoor een groot deel geautomatiseerd testen mogelijk werd. Daarnaast kon ik concluderen dat het op verschillende manieren mogelijk was om in SIMIT of een externe applicatie data te rapporteren. Dit kon ik ook concluderen middels de onderbouwing van de Siemens sessies en interne interview met een collega.

Tijdens de fase functioneel & technisch ontwerp heb ik met Siemens besproken over de ideeën en technische mogelijkheden. In deze sessies is een functioneel ontwerp gekozen met als onderbouwing dat deze oplossing technisch mogelijk is en het beste toepasbaar voor het proof of concept. Om het functioneel ontwerp technisch te kunnen oplossen heb ik tijdens het tweede onderzoek een week de tijd genomen om te kunnen onderzoeken of het volledig mogelijk is om de functionaliteiten technisch te realiseren. Dit werd door mij uiteindelijk geadviseerd en is na dit onderzoek samen met de opdrachtgever besloten dat gestart kan worden met het proof of concept.

- **C-6 Ontwerp software**

- **Actie:** het ontwerpen van de software voor de simulatieomgeving in het softwareprogramma SIMIT.

Het ontwerpen van de software heeft tijdens het functioneel & technische fase plaatsgevonden. Hierin is een functioneel ontwerp gerealiseerd van de simulatieomgeving, een flowdiagram gemaakt om het proces te ontwerpen en een FLOWNET diagram om de flow te visualiseren voor het automatisch testen van route control. Ook is in SIMIT een eigen component ontworpen om de routes te kunnen controleren op configuratiefouten. Dit is te lezen in hoofdstuk 8 Functioneel & Technisch ontwerp.

- **D-14 Realiseren van software**

- **Acties:** het realiseren van de simulatieomgeving in het Siemens softwareprogramma SIMIT.

Na de goedkeuring van het functioneel & technisch ontwerp is de simulatieomgeving gerealiseerd volgens de eisen en wensen van de opdrachtgever in hoofdstuk 9 Proof of concept. De ontwerpen zijn in SIMIT gerealiseerd en de servers zijn gekoppeld met de virtual controller (PLC).

- **D-15 Testen**

- **Actie:** het testen van de simulatieomgeving gebaseerd op een testplan.

Het testen van de simulatieomgeving is ten eerste doorlopen zonder de opdrachtgever om de functionaliteiten te testen. Daarna is er een SAT & FAT ingepland waarin de opdrachtgever de simulatieomgeving heeft beoordeeld. In de resultaten zijn te lezen in paragraaf 9.5 Testplan.

- **G-c Kritisch, onderzoekend & methodisch werken**

- **Actie:** het uitvoeren van de opdracht volgens de twee methodische modellen: waterval en het V-model.

Het afstudeertraject is volgens het V-model verlopen. Alle fases zijn doorlopen en de documenten zijn door de opdrachtgever goedgekeurd. Verder is tijdens het vooronderzoek methodisch te werk gegaan door het toepassen van verschillende onderzoekstechnieken. Door het toepassen van deze technieken is het afstudeertraject succesvol doorlopen en kon de juiste informatie worden verzameld. Deze technieken zijn te lezen in paragraaf 5.6 Projectmethode.

- **G-f Leren leren: voorbereiden op volgende studiefase en beroep**

- **Actie:** het stellen van leerdoelen in de scriptie.

De leerdoelen staan beschreven in paragraaf 12.3 met daarbij de competenties en uitwerking tijdens de periode van het afstudeertraject. De volgende competenties zijn uitgewerkt: communiceren, ondernemen, analyseren en besluitvorming.

## 12 Evaluatie en reflectie

In dit hoofdstuk wordt de afstudeerperiode geëvalueerd. Hierin is te lezen hoe de periode is verlopen en wat er verbeterd zou kunnen worden voor de aankomende projecten. Het hoofdstuk is onderverdeeld in het evalueren van de aanpak en tussenproducten. Daarnaast wordt de zelfreflectie en een aantal leerdoelen behandeld.

### 12.1 Evaluatie aanpak

In deze paragraaf evalueer ik de aanpak van het afstudeertraject. Voor deze tijdsperiode is gebruikgemaakt van een aantal vooronderzoeksmethodes om informatie te verkrijgen over de mogelijkheden van de simulatieomgeving SIMIT. Door het toepassen van deze methodes heb ik veel profijt gehad tijdens het vooronderzoek. Ik heb tijdens het onderzoek de juiste informatie kunnen vinden mede door de planning van mijn vooronderzoek en methodes die ik van tevoren had opgesteld. Hierdoor kon ik in een korte periode de simulatieomgeving verkennen en de vraagstellingen beantwoorden.

Tijdens de afstudeerperiode heb ik de V-model projectmethode toegepast om de tussenproducten op te leveren. Deze methode was achteraf een goede keuze om het project op een methodische manier te kunnen uitvoeren. Echter, heb ik wel voor een klein deel afgeweken van de methode om tijd te besparen. Tijdens het beoordelen van de tussenproducten ben ik alvast begonnen met de nieuwe fase aangezien ik alleen werkzaam was van één project. Hierdoor heb ik voorkomen dat ik moest wachten op de reactie van de opdrachtgever. De beoordelingen van de opdrachtgever heeft verder geen invloed gehad op de volgende fase. Wel heb ik de feedback verwerkt in de tijdsperiode van de volgende fase en was dit uiteindelijk de beste keuze.

### 12.2 Evaluatie tussenproducten

Tijdens de Afstudeerperiode zijn een aantal producten opgeleverd waaronder een plan van aanpak, vooronderzoeksdocument, functioneel & technisch ontwerp, proof of concept, testplan en een advies. In het plan van aanpak is vooraf een planning gemaakt waarin een vaste datum is vastgesteld om de tussenproducten op te leveren.

De tussenproducten zijn regelmatig met afwijking van de geplande datum opgeleverd. Het vooronderzoek liep een week uit dan de afgesproken datum vanwege de hoeveelheid informatie voor het beantwoorden van de vraagstellingen dat verwerkt moest worden in het document. Daarnaast kwam uit de gesprekken van Siemens nog technische vervolgvragen over route control die beantwoord moesten worden voordat naar de volgende fase kon worden gegaan. Daarnaast had ik niet direct beschikking over een SIMIT licentie om de softwareomgeving te onderzoeken. Aangezien het vooronderzoek een belangrijke prioriteit heeft voor het vervolg van het afstudeertraject, heb ik ervoor gekozen om een week extra tijd te investeren.

Ook het functioneel & technisch ontwerp is een week uitgelopen ten opzichte van de planning. Dit kwam mede door de uitloop van het vooronderzoek en omdat ik in de planning te weinig tijd had ingepland. Door middel van extra tijd te investeren in deze fase is het document uiteindelijk wel met

kwaliteit opgeleverd en werd het eenvoudiger om tijdens het proof of concept de functionaliteiten volgens het document te realiseren. Hierdoor heb ik tijdens deze fase tijd bespaard en gecompenseerd met de vertragingen die ik had opgelopen. Uiteindelijk zijn ondanks de vertraging alle tussenproducten door middel van het compenseren in de fase van het proof of concept binnen de afgesproken datum opgeleverd.

### 12.3 Reflectie en leerdoelen

- **Competentie: Communiceren**
  - **Leerdoel:** Tijdens de afstudeerperiode heb ik op beroepsniveau kunnen communiceren met de opdrachtgever, Siemens, de begeleider en collega's.
  - **Uitwerking:** Dit leerdoel is toegepast middels verschillende vergaderingen en interviews met instanties van zowel intern als met Siemens. Tijdens deze gesprekken kon ik op zowel non-technisch als op technisch niveau communiceren met de verschillende personen in het afstudeertraject. Achteraf kan ik tevreden terugkijken op de interviews en gesprekken. Wel merkte ik in het begin van de afstudeerperiode dat het niet makkelijk was om de afstudeeropdracht op te pakken. Ik moest verschillende technische termen leren die tijdens de gesprekken voortkwamen en ik miste nog de benodigde kennis van verschillende Siemens applicaties. Echter, merkte ik dat later in het traject dat het eenvoudiger ging en de gesprekken vloeiender verliepen.
- **Competentie: Ondernemen**
  - **Leerdoel:** Tijdens de afstudeerperiode heb ik zelfstandig acties ondernomen middels doorzettingsvermogen en het nemen van verantwoordelijkheden.
  - **Uitwerking:** Het project heb ik zoveel als mogelijk zelfstandig toegepast, aangezien ik niet in een teamverband werkte. Ik heb in deze periode opgemerkt dat ik veel technische problemen zelf kon oplossen middels zelf onderzoek te doen voordat ik advies ging vragen aan collega's. Ook nam ik de volledige verantwoordelijkheid van mijn eigen project en communiceerde ik direct met mijn begeleider om advies te vragen indien ik vastliep tijdens het project. Door direct actie te ondernemen kon ik veel tijd besparen en het project volgens planning afronden.
- **Competentie: Analyseren**
  - **Leerdoel:** Tijdens de afstudeerperiode heb ik tussentijds geanalyseerd over de status van mijn project en de kwaliteit van de documenten gewaarborgd voordat deze werden opgeleverd.
  - **Uitwerking:** Ik heb meerdere keren de status van het project geanalyseerd en bekeken of de planning te veel ging afwijken. Dit heb ik vooral in de periode uitgevoerd toen de planning vanaf het vooronderzoek voor een week ging uitwijken. Dit heb ik gemeld bij de opdrachtgever maar er hoefde nog geen acties ondernomen te worden. Ik heb toen aangegeven dat de uitloop nodig was om later in het project tot een hoger kwalitatief resultaat te kunnen komen. Ook heb ik toen aangegeven dat het proof of concept ruim is gepland en er eventueel nog twee weken extra was voor noodzakelijke werkzaamheden. Hierdoor is de uitloop van het vooronderzoek gecompenseerd tijdens het proof of concept en is de benodigde tijd gecompenseerd.

- **Competentie: Besluitvorming**
  - **Leerdoel:** Tijdens het afstudeertraject heb ik keuzes gemaakt en een advies geschreven gebaseerd op de resultaten van het vooronderzoek en de eisen en wensen van de opdrachtgever.
  - **Uitwerking:** Tijdens de fase van het vooronderzoek zijn er verschillende keuzes gemaakt om het proof of concept te realiseren. Deze keuzes zijn gemaakt in overleg met technische SIMIT specialisten van Siemens en de opdrachtgever van DWG. De eisen en wensen zijn in het proof of concept gerealiseerd en gebaseerd op deze resultaten heb ik een advies geschreven om het product verder uit te werken. Dit advies zijn door mij als engineer keuzes uit de ervaringen die ik met de simulatieomgeving heb opgedaan.



## 13 Conclusie

In dit afstudeerdossier was de doelstelling: ‘Dit onderzoek heeft als doel een advies te geven aan DWG hoe ze de simulatie in SIMIT kunnen ontwikkelen zodat systeemfuncties waarvan niet zeker is over de werking van het product gesimuleerd kunnen worden en waarin zowel software programmeer, denk en ontwerpfouten sneller worden opgespoord’. Hiervoor is een kwalitatief vooronderzoek uitgevoerd, een functioneel & technisch ontwerp opgeleverd en een gerealiseerde applicatie als proof of concept.

Er kan geconcludeerd worden dat het proof of concept is geslaagd. Hierbij zijn vervolgstappen geadviseerd om het proof of concept uit te werken naar een compleet product voor de tankterminal projecten binnen DWG. SIMIT is een applicatie om de kwaliteit van je product te verhogen. Middels het testen en rapporteren van routes is het voor DWG in de toekomst mogelijk om applicaties te realiseren waarin zowel software programmeer, denk en ontwerpfouten in de software wordt opgelost. Daarnaast kan dit worden bewezen bij de opdrachtgevers voordat het project in bedrijf gesteld wordt. Echter, moet wel tijd worden geïnvesteerd om het proof of concept uit te breiden naar een volledig product.

Uit de vraagstellingen van het vooronderzoek is gebleken dat het geautomatiseerd testen van routes en het rapporteren hiervan de beste optie was om toe te passen in de simulatieomgeving SIMIT. Deze functionaliteiten zijn toegepast in het proof of concept en gedemonstreerd. Hierdoor kunnen in de toekomst routes op grootschalige grote worden getest en kan worden bewezen naar de opdrachtgever dat de routes volledig werken indien de toekomstige projecten in bedrijf gesteld wordt. Ook is geconcludeerd dat SIMIT een bijdrage kan leveren voor operator trainingen. Echter, is dit niet bewezen tijdens het proof of concept maar is tijdens het vooronderzoek wel gebleken dat SIMIT ontwikkeld is om operator trainingen te geven en in de toekomst kan worden toegepast.

Daarnaast is geconcludeerd dat SIMIT niet tijdbesparend is op korte termijn aangezien veel tijd moet worden geïnvesteerd in het ontwerpen en realiseren van de testomgeving. De hoeveelheid tijd dat daadwerkelijk geïnvesteerd moet worden is afhankelijk per project. De factoren zoals: de grote van het project, het verloop van het project, het aantal configuratiefouten in een testomgeving, ontwerp en programmeerfouten, ongewenste gedragsveranderingen en de kwaliteit van de testfase zijn factoren waarin tijd een grote rol speelt.

## 14 Bibliografie

- Actemium. (sd). *V-model*. Opgehaald van <https://www.actemium.nl/solutions-services/gamp/v-model/>
- Automation, D. (2018). *Functionele specificatie - Tankpunt automatisering 2.0*. Schiedam: DWG.
- Onderzoeksmethoden. (sd). Opgehaald van <https://www.scribbr.nl/category/onderzoeksmethoden/>
- Siemens. (2017, July). Opgehaald van [https://support.industry.siemens.com/dl/dl-media/289/109754289/att\\_1001386/v1/20191029\\_Partner-Platform\\_web/content/assets/12101.pdf](https://support.industry.siemens.com/dl/dl-media/289/109754289/att_1001386/v1/20191029_Partner-Platform_web/content/assets/12101.pdf)
- Siemens. (2018, 06). Opgehaald van <https://support.industry.siemens.com/cs/mdm/93763144?c=113418584075&lc=en-US>
- Siemens. (2018, 06). *SIMIT Simulation Platform*. Opgehaald van [https://support.industry.siemens.com/cs/document/109759317/simatic-simit-simulation-platform-\(v10-0\)?dti=0&lc=en-WW#!](https://support.industry.siemens.com/cs/document/109759317/simatic-simit-simulation-platform-(v10-0)?dti=0&lc=en-WW#!)
- Siemens. (2018, September). *SIMIT, COMOS & PAA*. Opgehaald van <https://www.youtube.com/watch?v=20RNdjRcaTs>
- Siemens. (2020). *Tutorials SIMIT*. Opgehaald van [https://support.industry.siemens.com/dl/dl-media/289/109754289/att\\_1001386/v1/20191029\\_Partner-Platform\\_web/start.htm?lang=en#!/us/11225](https://support.industry.siemens.com/dl/dl-media/289/109754289/att_1001386/v1/20191029_Partner-Platform_web/start.htm?lang=en#!/us/11225)

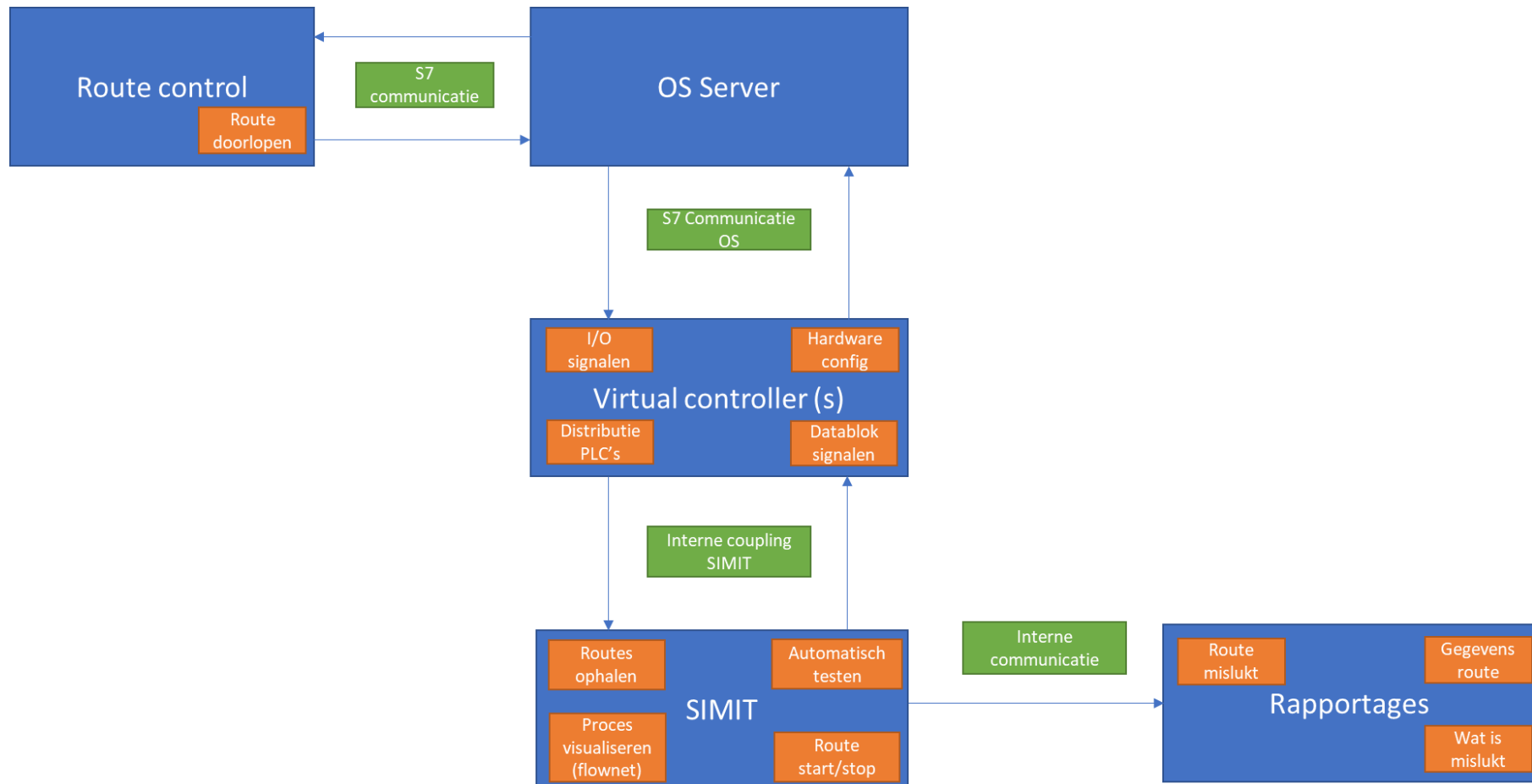
# Bijlage I: Plan van aanpak

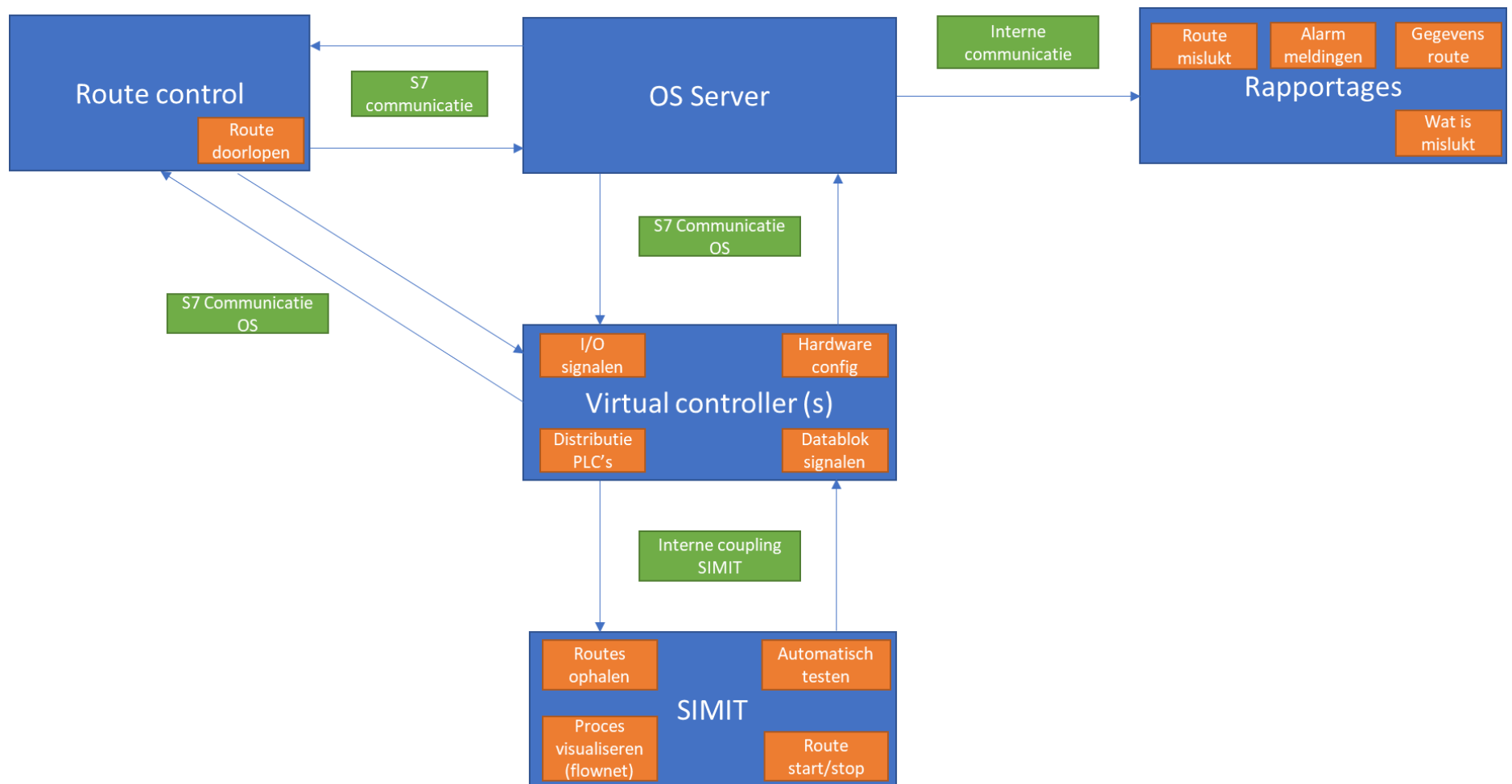
## Bijlage II: Vooronderzoek

## Bijlage III: Functioneel & Technisch ontwerp

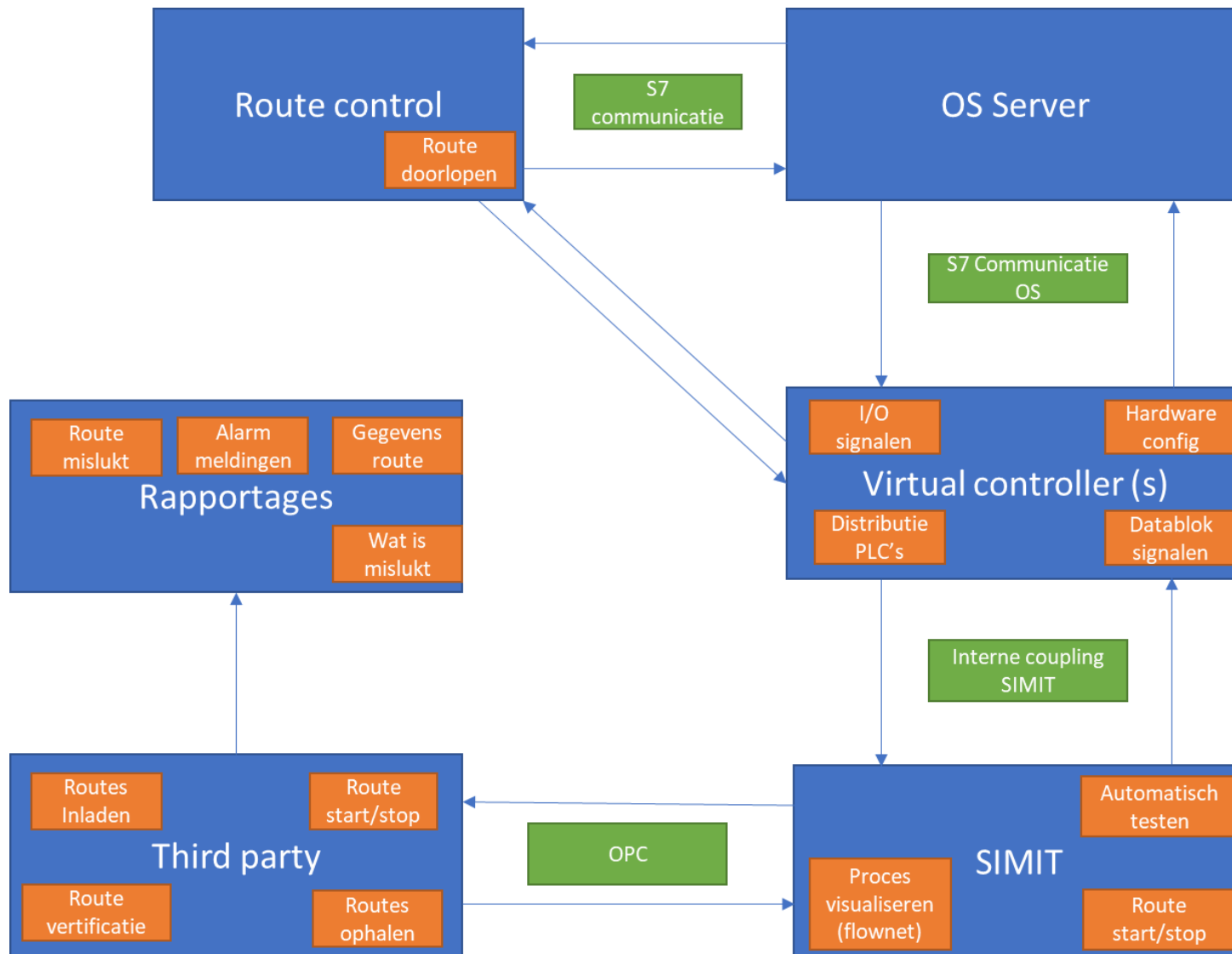
## Bijlage IV: Testplan

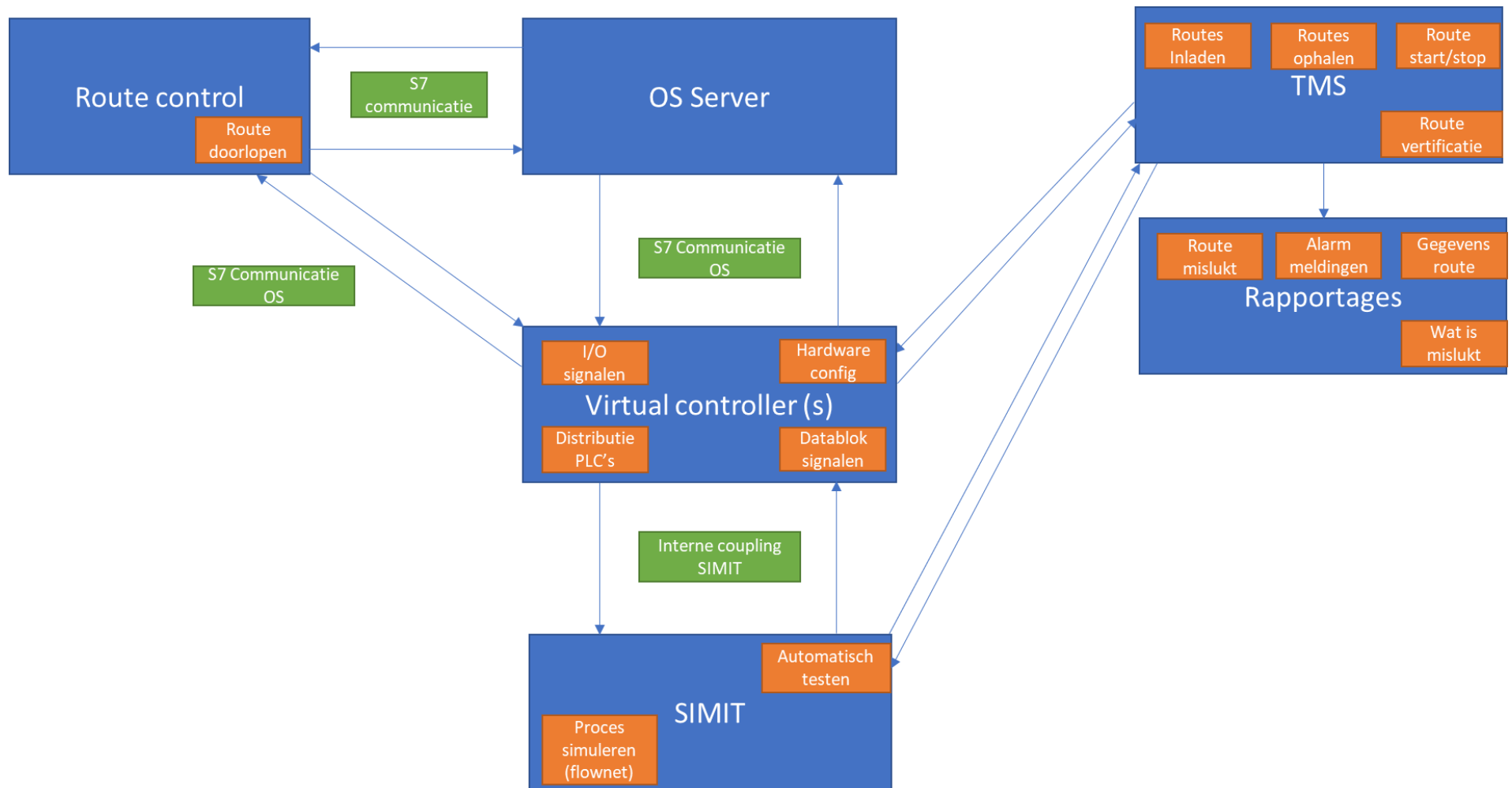
## Bijlage V: Functioneel ontwerpen











## Bijlage VI: Signalen

OPC blok:

Signaal	Datatype	DB adres
QS_bron	INT	DB1606.DBD414
QS_doel	INT	DB1606.DBD418
QS_Type	INT	DB1606.DBD6
QS_via01	DINT	DB1606.DBD422
QS_via02	DINT	DB1606.DBD426
QS_via03	DINT	DB1606.DBD430
QS_via04	DINT	DB1606.DBD434
QS_via05	DINT	DB1606.DBD438
QS_via06	DINT	DB1606.DBD442
QS_via07	DINT	DB1606.DBD448
RM_Auto	Bool	DB1606.DBX494
QS_SP_Massa	INT	DB1606.DBD747

Route statussen:

Signaal	Datatype	DB adres
Auto	BOOL	DB2621.DBX0.0
Idle	BOOL	DB2621.DBX0.1
Holding	BOOL	DB2621.DBX0.2
Held	BOOL	DB2621.DBX0.3
Resuming	BOOL	DB2621.DBX0.4
Stopping	BOOL	DB2621.DBX0.5
Stopped	BOOL	DB2621.DBX0.6
Completed	BOOL	DB2621.DBX0.7
Aborting	BOOL	DB2621.DBX1.0
Oplijnen	BOOL	DB2621.DBX1.1
Verpompen	BOOL	DB2621.DBX1.2
Terugblazen	BOOL	DB2621.DBX1.3
QRM_Stat	INT	DB1517.DBW8

Operator signalen:

Signaal	Datatype	DB adres
EnStart	BOOL	DB1517.DBX604.0
EnStop	BOOL	DB1517.DBX604.1
EnAfbreken	BOOL	DB1517.DBX604.2
CmdStart	BOOL	DB1517.DBX608.0
CmdStop	BOOL	DB1517.DBX608.1
CmdAfbreken	BOOL	DB1517.DBX608.2

## Bijlage VII: Route controleren blok

```
BLOCK
//simulatie starten
if(startSimulatie){
    startRoute = 1;
}

//simulatie stoppen
if(stopSimulatie){
    cmdAfbreken = 1;
}

//start een route indien in idle
if(idle == 0){
    routeGestart = 1;
}

//reset start route indien route is gestart
if(routeGestart){
    startRoute = 0;
}

//print route mislukt
if(routeCompleted == 0){
    if(mislukteRoute == 1){
        if(k == p){
            snapshot = 1;
            printRoute = 1;
            cmdAfbreken = 1;
            _printlog("An error has encoured in this route");
            _printlog("SourceRoute: " + sourceRoute);
            _printlog("DestinationRoute: " + destinationRoute);
            _printlog("Via: " + vial);
            _printlog("Liquid pumped: " + TeGaan);
            _printlog("");
            k = k + 1;
        }
    }
}
else{
    k = p;
}

//print route succesvol en reset alle outputs.
if(routeCompleted == 1){
    if(j == i){
        if(printRoute == 0){
            _printlog("Route succesfull!");
            _printlog("SourceRoute: " + sourceRoute);
            _printlog("DestinationRoute: " + destinationRoute);
            _printlog("");
        }
        cmdAfbreken = 0;
        snapshot = 0;
        printRoute = 0;
        routeGestart = 0;
        routeCounter = routeCounter + 1;

        if(aantalroutes > routeCounter){
            startRoute = 1;
        }
        j = j + 1;
    }
}
else{
    j = i;
}

if(reset){
    startRoute = 0;
    routeCounter = 0;
    routeGestart = 0;
}

END_BLOCK
```

## Bijlage VIII: Resultaten FAT

---

**Paragraaf:** 1 Route control geautomatiseerd testen

**Doel:** In SIMIT route control geautomatiseerd testen

---

---

**Procedure 1**  
1.1 Communicaties

**Ref:** Functioneel & Technisch  
ontwerp V1.0

**Criterium**

Er moet een Communicatie met 57 verbindingen tussen SIMIT en de AS (o.a. route control) gerealiseerd worden om informatie te kunnen delen. Ook is een interne koppeling in SIMIT benodigd voor een verbinding tussen de virtuele controller en de simulatieomgeving.

**Naam en paraaf:**

Jelle Mank

**Datum:**

21-12-2020

**Opmerkingen:**

Er is een virtual controller toegevoegd en de hardware configuraties en I/O signalen van het KTP project zijn geïmporteerd in de virtual controller. Verder is een IP ingesteld om connectie te kunnen maken met de OS.

**Afwijking:**

**Status:** Afgerond

**OVERALL:** AKKOORD

**Criterium**

De flow van de route kunnen simuleren met FLOWNET zodat de route van de source naar de destination kan worden gesimuleerd en dat de route op de correcte manier kan worden uitgevoerd volgens de SFC van route control.

<b>Naam en paraaf:</b> Jelle Mank	<b>Datum:</b> 21-12-2020
<b>Opmerkingen:</b>  Er is een P&ID gerealiseerd met FLOWNET om de flow te kunnen visualiseren in SIMIT. De componenten zoals kleppen, tanken en een pomp zijn toegevoegd in zowel de device level als de process level zodat in FLOWNET met de juiste instellingen routes doorlopen kunnen worden en vloeistof te verpompen volgens de SFC's van route control.	<b>Afwijking:</b>
<b>Status:</b> Afgerond	
<b>OVERALL:</b> AKKOORD	

**Criterium**

Een applicatie moet de eerste doorgegeven route kunnen starten nadat een engineer op start heeft gedrukt. Nadat deze klaar is moet de applicatie de volgende route automatisch starten totdat alle routes zijn getest.

<b>Naam en paraaf:</b> Jelle Mank	<b>Datum:</b> 21-12-2020
<b>Opmerkingen:</b> Middels het doorgeven van locaties en commando's van SIMIT naar de PLC kan routes geautomatiseerd doorlopen worden. Het inladen van enumereringen van zowel de source als de destination worden door meerdere SIMIT diagram componenten opgeslagen. Hierdoor is het mogelijk om per route de source en destination van de tank te achterhalen. Indien de engineer op start heeft gedrukt wordt door middel van een CTE route controleren blok de testsimulatie gestart. De aantal routes kan ook in dit blok worden aangegeven. Hierdoor kan na een geteste route de nieuwe route met de nieuwe source en destination nummer worden gestart door middel van het optellen van een counter. Operator handelingen worden geautomatiseerd middels het forceren van commando's naar de PLC	
<b>Afwijking:</b>	
<b>Status:</b> Afgerond	
<b>OVERALL:</b> AKKOORD	



**Criterium**

Er moet een applicatie gerealiseerd worden welke kan controleren, middels beschikbare data in Simit/de AS, of een route succesvol is opgelijnd en doorlopen.

<b>Naam en paraaf:</b>  Jelle Mank	<b>Datum:</b>  21-12-2020
<b>Opmerkingen:</b>  Er zijn verschillende scenario's opgezet om (configuratie) fouten te achterhalen. Scenario 1: de route kan niet gevonden worden waardoor een error gegenereerd wordt. Scenario 2: de flow wordt niet door een pomp bereikt waardoor de vloeistof niet verpompt wordt naar de destination tank. Deze Scenario's zijn opgevangen in SIMIT middels het versturen van een signaal naar de CTE blok indien een text error ontstaat of indien het verpompen niet na een minuut is voltooid.	
<b>Afwijking:</b>	
<b>Status:</b> Afgerond	
<b>OVERALL:</b> AKKOORD	

## 2.1 Rapporteren van routes

**Criterium**

Er moet een applicatie gerealiseerd worden welke niet geslaagde routes kan rapporteren middels een leesbaar en deelbaar bestandstype.

<b>Naam en paraaf:</b> Jelle Mank	<b>Datum:</b> 21-12-2020
<b>Opmerkingen:</b> Door middel van de scripting functie in SIMIT wordt een tekstbestand aangemaakt zodat geslaagde en niet geslaagde routes gelogd kunnen worden.	
<b>Afwijking:</b>	
<b>Status:</b> Afgerond	
<b>OVERALL:</b> AKKOORD	

**Criterium**

Er moet een applicatie gerealiseerd worden welke een rapport kan genereren van niet geslaagde routes welke reeds getest zijn zodat deze kan worden overhandigd aan de opdrachtgever.

<b>Naam en paraaf:</b> Jelle Mank	<b>Datum:</b> 21-12-2020
<b>Opmerkingen:</b> Door middel van het CTE blok wordt via PRINT_F functies de informatie in het tekstbestand gezet van geslaagde en niet geslaagde routes.	
<b>Afwijking:</b>	
<b>Status:</b> Afgerond	
<b>OVERALL:</b> AKKOORD	

#### Criterium

Via de scripting functie in SIMIT geautomatiseerd snapshots maken om de status van de ongewenste routes in zowel SIMIT als de virtual controller te kunnen terugzien, zodat het probleem van de route kan worden vastgelegd.

<b>Naam en paraaf:</b>	<b>Datum:</b>
Jelle Mank	21-12-2020
<b>Opmerkingen:</b>	<b>Afwijking:</b>
<p>Na een mislukte route wordt een snapshot gegenereerd middels een SIMIT blok zodat de status van de route kan worden bekeken op het moment dat een route is mislukt. Hierdoor kan in de OS achterhaald worden welke fout is gegenereerd.</p>	
<b>Status:</b> Afgerond	
<b>OVERALL:</b> AKKOORD	

**Criterium**

Communicatie tussen de applicatie van de operator en SIMIT zodat de operator achter zijn eigen applicatie de handelingen kan uitvoeren indien in SIMIT een ongewenste situatie wordt geforceerd.

<b>Naam en paraaf:</b> Jelle Mank	<b>Datum:</b> 21-12-2020
<b>Opmerkingen:</b> Deze functionaliteit kon niet binnen de tijdsbestek worden gerealiseerd.	
<b>Afwijking:</b>	
<b>Status:</b> N.V.T	
<b>OVERALL:</b> N.V.T	

## 3.1 Ongewenste scenario's (optioneel)

**Criterion**

Bet ontwerpen en realiseren van ongewenste scenario's in SIMIT zodat de operator voor zijn eigen procesinstallatie de benodigde informatie krijgt om storingen op te kunnen lossen.

<b>Naam en paraaf:</b> Jelle Mank	<b>Datum:</b> 21-12-2020
<b>Opmerkingen:</b> Deze functionaliteit kon niet binnen de tijdsbestek worden gerealiseerd.	
<b>Afwijking:</b>	
<b>Status:</b> N.V.T	
<b>OVERALL:</b> N.V.T	