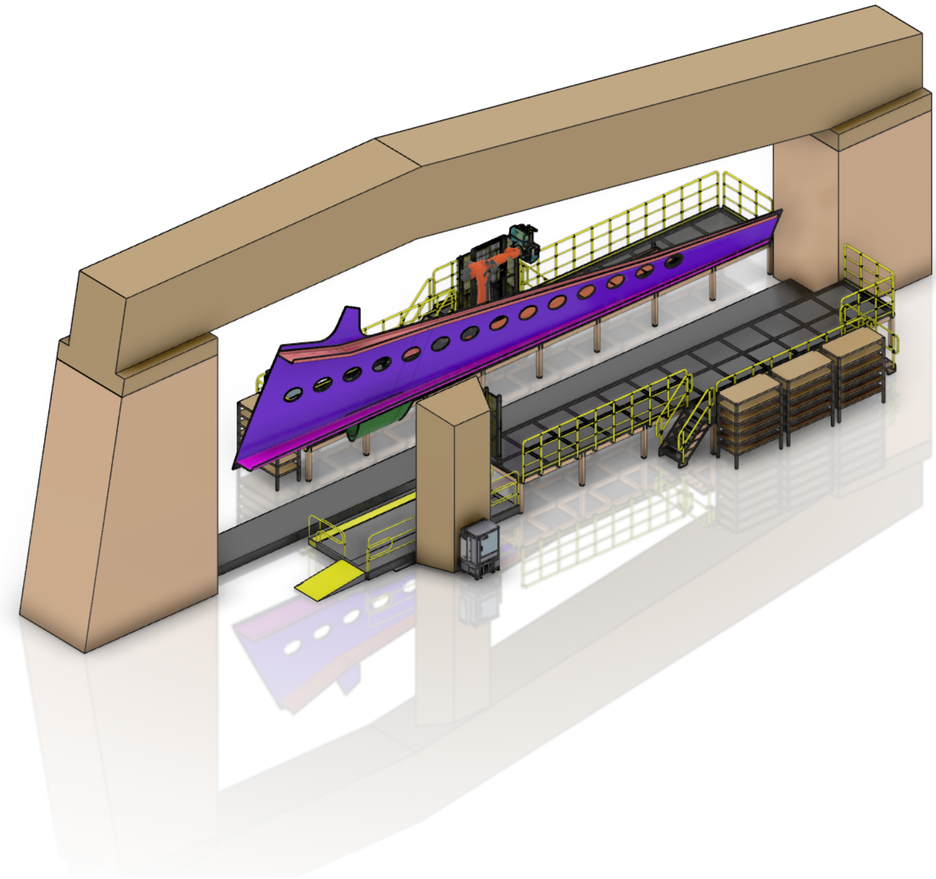
CHALMERS
UNIVERSITY OF TECHNOLOGY



Virtual Commissioning as a service: Standardizing modern co-simulation

A Deployment Framework For The Aerospace Industry

Master's thesis in Production Engineering

ALBIN JONSSON

DEPARTMENT OF ELECTRICAL ENGINEERING
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2024
www.chalmers.se

Master's thesis 2024

Virtual Commissioning as a service: Standardizing modern co-simulation

A Deployment Framework For The Aerospace Industry

ALBIN JONSSON

CHALMERS
UNIVERSITY OF TECHNOLOGY

Department of Electrical Engineering
Division of Systems and Control
Automation
Chalmers University of Technology
Gothenburg, Sweden 2024

Virtual Commissioning As A Service: Standardizing Modern Co-simulation
A Deployment Framework For The Aerospace Industry
ALBIN JONSSON

Supervisor: ANTON ALBO, Ind. PhD Student, Chalmers+Volvo Cars
Examiner: Prof. PETTER FALKMAN, Department of electrical engineering

Master's Thesis 2024
Department of Electrical Engineering
Chalmers University of Technology
SE-412 96 Gothenburg
Telephone +46 31 772 1000

Cover: Manufacturing cell created for virtual commissioning demonstration.

Typeset in LaTeX
Printed by Chalmers Reproservice
Gothenburg, Sweden 2024

Virtual Commissioning as a Service: Standardizing modern co-simulation
A Deployment Framework For The Aerospace Industry
ALBIN JONSSON
Department of Electrical Engineering
Chalmers University of Technology

Abstract

This thesis explores the deployment of Virtual Commissioning (VC) as a service within the aerospace industry. It aims to standardise co-simulation to enhance manufacturing efficiency and reduce commissioning times. The study introduces a deployment framework leveraging the PLM platform 3DEXPERIENCE and the control software ControlBuild, focusing on the complexities of VC technologies and the relationship between simulation fidelity and business value. Key findings demonstrate the framework's effectiveness in standardising VC implementation and deployment, enabling interoperability between isolated simulation systems and elevating simulation fidelity through the use of external applications. The framework's deployment was validated through controlled environment tests, indicating potential applicability to other industries and manufacturing settings.

The research highlighted Scalability, Interoperability and Versatility as selling points of the proposed framework using a shared simulation memory. The research makes theoretical contributions by increasing the understanding of VC in labour-intensive settings and offering a flexible framework adaptable to various simulation applications while aligning with industry 4.0 practices. Practically, it provides a blueprint for OEMs and suppliers to adopt VC with fewer prerequisites, potentially leading to cost savings and new business opportunities for VC service providers. Future research could explore the integration of AI into the proposed framework, investigating AI's potential to evaluate control logic and improve simulation model realism, thereby enhancing the long-term comprehensiveness of VC solutions.

Keywords: Virtual Commissioning, Industry 4.0, Digital Manufacturing, 3DEXPERIENCE, ControlBuild, Co-Simulation.

Acknowledgements

I would like to express to a number of individuals whose guidance and expertise have been invaluable throughout this thesis.

First and foremost, I am grateful to my supervisor, Anton Albo, for his guidance and state-of-the-art knowledge. His insightful feedback, published papers and profound competence in the field of Virtual Commissioning have been fundamental to the success of this research. His mentorship has not only shaped this work but also my future as a scholar in this exciting field of manufacturing.

I extend my heartfelt thanks to the director of Prodtex LTD, Peter Helgosson, for his generous hospitality and care during my research in Bristol. I am grateful for the opportunity he provided me of doing my thesis in the UK, at a pioneering where personal growth is encouraged. His excellent leadership and responsiveness to both his employees and myself inspired me, and his support made it possible for me to take my project further than what was originally planned.

My sincere appreciation also goes to Dr. Henrik Kihlman for introducing me to Prodtex during my bachelor's thesis and for sparking my interest in Robotics and Digital Manufacturing. His role as an examiner during my bachelor's thesis on digital manufacturing left me with an unshakable curiosity to explore the subject further.

I'm also immensely grateful to the entire staff at Prodtex UK and also the few I met in the Gothenburg office for their warmth spirit and friendship. Being far from home, they created a welcoming and fun environment that made my stay in Bristol memorable and enriched me as a person. I have made many invaluable friends and created memories for life.

This thesis would not have been possible without the contributions and encouragement of these individuals, to whom I owe a great deal of gratitude. Thank you for your belief in my potential and for providing me with the opportunity to grow both personally and professionally. I think my most important lesson from these 6 months is well sad by Mark Twain - *"The secret of getting ahead is getting started."*

Albin Jonsson, Bristol, April 2024

List of Acronyms

Below is the list of acronyms that have been used throughout this thesis listed in alphabetical order:

3DX	3DEXPERIENCE
CB	Control Build
DT	Digital Twin
FMI	Functional Mock-up Interface
FMU	Functional Mock-up Unit
HiL	Hardware-in-the-loop
HMI	Human machine interface
IO	Inputs and outputs
MBOM	Manufacturing bill of material
MiL	Model-in-the-loop
OEM	Original equipment manufacturer
OPC UA	OPC unified architecture
PLC	Programmable Logic Controller
SiL	Software-in-the-loop
ST	Structured Text
TIA Portal	Totally Integrated Automation Portal
VC	Virtual Commissioning
VR	Virtual Reality

Contents

1

Introduction

This chapter provides an overview of the importance of digital manufacturing and virtual commissioning and states the project challenges and goals.

1.1 Background

Manufacturing industries are facing increasingly higher demand for product variety and quantity [1], which results in high requirements for more flexible production systems and shorter commissioning times [2]. The increasing demands on large manufacturers translate into pressing requirements for the original equipment manufacturers (OEM) and contractors who are often expected to have state-of-the-art knowledge to solve complex tasks in manufacturing scenarios and are using digital manufacturing tools to deal with advanced production challenges [2]. Digital manufacturing refers to improving production systems with the help of computer-aided technologies and is nowadays a function found in major cloud-based PLM platforms, offering higher performance and lower cost of ownership to achieve benefits faster [3]. Recent advancements in the use of PLM platforms are increasingly allowing digital manufacturing engineers to utilise *digital representations* of manufacturing assets [4], a term often confused with its bidirectionally interconnected form; digital twin (DT) [5], [6], [7]. Digital representations are highly relevant for Commercial OEMs and tier 2 suppliers in many manufacturing industries. In the Aerospace Industry in particular, however, digital representations have huge potential, mainly because of each part's high standard quality requirements in the complicated products [4]. One reason for investing in creating expensive digital representations is the possibility of manufacturing simulations [5], which allows the manufacturer to recreate existing manufacturing scenarios virtually or design new ones.

1.1.1 Virtual Commissioning

Virtual Commissioning (VC) is the process of combining digital representations of manufacturing assets and control equipment to detect technical issues before the actual commissioning of a new production line [2] [8]. Initially, VC methods were only adopted by larger corporations, mainly because of the large and complex costs associated with model creation implementation [5] [8]. The methods significantly reduced the overall commissioning time, software quality and safety [5] [7] and are used frequently by more prominent manufacturers today. Fundamentally it connects a control system, hardware or software to a simulation model to replicate an

existing or planned system [5] [8]. Besides enabling more predictable physical commissioning, VC allows developers to modify and verify code without pausing the development process or production. This results in more time and tools to optimise the automation code.
As demonstrated in Figure 1.1, the overall commissioning time decreases, and the active production time increases when preparing commissioning steps like PLC programming, IT systems and Operator training alongside live production. Overall, the VC approach consumes more engineering hours than the traditional approach. Although digital modelling can be expensive, the potential savings from longer production runs can outweigh the costs in large-scale manufacturing, making it cost-effective [2].

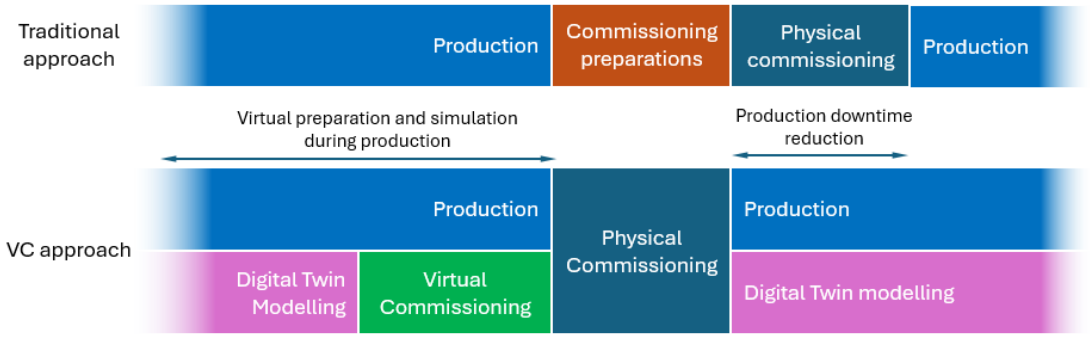


Figure 1.1: Shortened commissioning time demonstrated [9]
.

1.2 Frame of reference

Successful VC implementations usually require high simulation expertise [10], which may not be available to all companies. To overcome this challenge, suppliers nowadays provide this service to manufacturers [10]. Because of this recent opportunity for smaller companies to take on sizable projects, this master's thesis project will explore how the collaboration between manufacturer and supplier can work when implementing VC. The project will also explore how to design a framework that enables such a collaboration. The primary workload will be spent on understanding the characteristics of VC, when and how VC can be applied, and its critical factors for creating business value.

1.2.1 Prodtex perspective

Prodtex LTD, a Bristol-based Tier 2 supplier [11], is currently working with *Airbus* [12] and their suppliers in the UK with manufacturing solutions including robotics, fixturing, design and specialised knowledge. Airbus's new requirement is that suppliers deliver VC-compatible solutions. They emphasise that virtual commissioning is not 'just' an additional requirement but rather a tool to remove work pressure during project phases where the stress is high on both Airbus employees and integrators. Prodtex works with Dassault and is currently providing *3DEXPERIENCE* competence for Airbus. Still, for upcoming VC implementations, there is a need for simulation models with the option to be co-simulated with 3DX and control devices. Prodtex needs to develop a framework to build VC solutions with Dassault software like the 3DX platform and ControlBuild to highlight the opportunities that come with VC and create a demonstrational manufacturing scenario that Prodtex can use to prove a successful deployment is realistic.

1.2.2 Challenges

Despite VC's promising nature and recent advancements, successful implementations have substantial challenges. The most notable challenge in this project's scope is that for VC to become financially viable, companies need standardised implementation approaches for VC solutions to be cost-effective [13].

Modelling labour-intensive manufacturing scenarios also presents a challenge for VC [5]. The aerospace industry has exceptional quality demands, which makes the required level of detail in a simulation model just as high. Other possible labour-intensive scenarios include small-scale production that lacks the budget for expensive automation.

Simulation engineers face a challenge regarding communication protocols between tools from different industrial software providers. When a Tier 2 supplier contract with multiple other OEMs using separate VC tools, interoperability issues arises [14]. The licenses to different platforms are very costly, and having access to all major platforms is often not an option for OEMs. This results in an urgent need for VC interoperability in manufacturing sectors such as aeronautics and automotive [15].

Lastly, to make a business case out of a VC implementation, the benefits of simulations need to be demonstrated to the manufacturer to highlight potential savings and safety improvements in the commissioning process. One possible effective way of promoting VC capabilities in this scenario is through virtual reality (VR) technologies [16] [17]. VR can, in theory, improve a VC setup by interactively testing control logic and getting unique insights into the 3D model's intended functionality [17]. Since VC is not yet a mature technology for most manufacturers, one major challenge is identifying the key factors that define a VC case that brings value and not just unnecessary complexity to the commissioning project.

1.2.3 Aim

The thesis aims to approach the challenges mentioned in section 1.2.2 with a solution-oriented demonstration of VC's potential in the labour-intensive aerospace industry. The thesis will also investigate how VC as a service can be a competitive alternative to in-house development. The following sub-goals are defined to achieve the overall goal.

- G1: Create a virtual manufacturing scenario with relevant functionality in the aerospace industry.
- G2: Virtually commission the demonstrational manufacturing cell.

- G3: Develop a standard implementation framework for VC as a service and its deployment at Prodtex.
- G4: Summarise and demonstrate VC as a service capability and selling point in the aerospace industry.

1.2.4 Delimitations

The thesis will last 20 weeks and only consider creating a manufacturing cell for demonstrational purposes. Implementation will, therefore, not be attempted in real industry. The scope does not include an analysis of how the cell is designed, meaning that the manufacturing scenario's performance in lead time, safety or accuracy will not be considered. Chalmers University and Prodtex's overlapping expectations for this project define what is included in the scope, which only extends to creating and analysing a VC solution with a corresponding framework. Any activities not contributing to this are not considered. Although cyber security is generally important to consider in projects regarding industry 4.0-related topics, this thesis does not have cyber-security as a main concern.

The following software will be used for evaluation, while other solutions will only be considered in section 5:

- CATIA ControlBuild
- 3DEXPERIENCE
- PLCSim Advanced

1.2.5 Specification of the issue being investigated

Relating to the problem description and the challenges, three research questions can be formulated. As Prodtex is working heavily with Dassault Systèmes products and wants to upscale its VC capability, it must overcome the interoperability gap between simulation software. The first question is therefore:

- *R1: How can tools from Dassault Systèmes be used to demonstrate a Virtual Commissioning business case?*

Prodtex does not have a standardised implementation approach for VC and needs a framework to be developed for upcoming projects. Projects for Prodtex UK are usually related to labour-intensive industries. Therefore R2 is formulated:

- *R2: How can a standardised VC as a service approach be developed to utilise simulation tools efficiently in a labour-intensive industry setting?*

For Prodtex to successfully demonstrate VC's potential, the benefits must be known. They should also be able to determine a suitable level of detail for every business case. For this, R3 is formulated:

- *R3: What factors and level of detail result in business value for VC solutions?*

2

Preliminaries

This chapter serves to explain software, concepts and other technical terms used in this report

2.1 Dimensions of Virtual Commissioning

Since VC applications can be applied to a broad range of situations, [2] presents a model with three dimensions, illustrated in Figure 2.1, that the author uses to ensure the solution is proportionate to the problem. The first dimension explores the commissioning main objectives, whether introducing a new product or enhancing production rates and how the project's scale impacts technical decisions. The second dimension examines the technical solutions needed to adapt to the current and future demands of a more intelligent, fully connected factory, focusing on smart functionality and cost-effectiveness. The third dimension considers whether the project is a greenfield (starting from scratch) or brownfield (upgrading or integrating with existing systems) scenario, each offering different advantages and challenges. Greenfield projects allow more freedom in setting standards and creating a modern production environment but can be costly and difficult to validate without historical data. Brownfield scenarios leverage existing knowledge and infrastructure to reduce costs but may face limitations in system lifespan, quality, and integration challenges with existing systems.

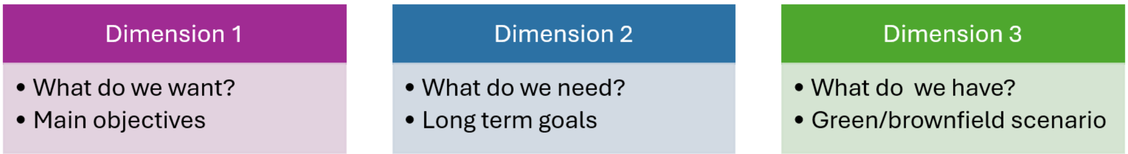


Figure 2.1: The tree dimensions of virtual commissioning [2]

2.2 Levels of Complexity

Levels of detail in a VC implementation are determining factors for the model fidelity [18]. However, finding the optimal complexity level [19] can be challenging. Regarding 3D representations, a higher degree of detail does not necessarily mean a better model. High complexity is not something to strive for in itself, but rather a model that produces accurate simulation results with just enough effort put into it [19].

VC, on the other hand, is different when considering complexity. As described by [2], VC can be categorised into various levels of detail, spanning from basic control logic to advanced control with a dynamic representation of the target production system. [2] introduces a classification hierarchy with five distinct levels displayed in Figure 2.2, each prodding specific functionalities or guidance. The initial level focuses on emulating actual controllers or PLCs for mechanic systems, highlighting software used for offline programming. The subsequent level deals with signal properties and communication protocols, employing SiL or HiL setups for testing logic performance. This level builds upon recent co-simulation standards like the Functional Mock-up Interfaces (FMI) [5]. The next level advances to sensors, devices, and actuators; the third level emphasises using simulation software to model actuator behaviour and introduces the concept of dynamic and visual responses for enhanced verification. The fourth level requires resource modelling, demanding a comprehensive analytical understanding of each component and introducing kinematic relationships and partial visualisation for more complex simulations. The final level expands to include multiple connected systems, aiming for a comprehensive digital twin or a collection of twins forming a virtual factory, integrating human models and educational tools to facilitate interaction and learning.

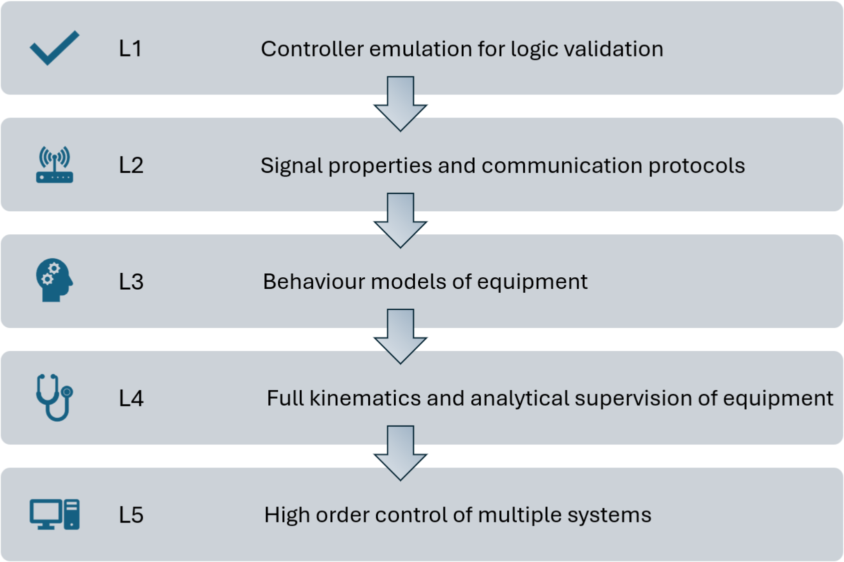


Figure 2.2: Levels of complexity as defined by [2]

2.3 X-in-the-Loop

X-in-the-Loop is a term used primarily in the automotive industry describing the state-of-the-art simulation and model-based-development techniques that consist of mainly three methods [9], [20]. Model-in-the-loop (MiL) is the least demanding method, utilising only modelled components [15], [20]. This type of simulation is already well used in the early stages of manufacturing planning [15], [20]. This approach does not utilise any PLC but an in-built control compiler inside the simulation environment. To elevate the simulation's complexity level, an emulated

PLC can control virtual assets like robots, sensors and conveyors, referred to as Software-in-the-loop (SiL) [15]. This additional layer of complexity more accurately represents the system and, therefore, will have a higher chance of detecting issues with the commissioning [9]. The final step towards simulation accuracy is switching out the emulated PLC for a physical one and running the code like it would have done after commissioning. Using a real PLC to control the virtual assets, one introduces noise, signal delays and other physical sources of uncertainty to the simulation model. This step is known as Hardware-in-the-loop (HiL) [15], [20], and has the most demanding setup, but also better reflects the characteristics of an actual manufacturing system [9]. Figure 2.3 displays the current capabilities of the different VC providers. SIEMENS has virtual and physical PLCs along with a simulation environment for verification. Dassault does not provide PLCs and instead focuses more on tools for designers to create the virtual assets used in the simulation model.

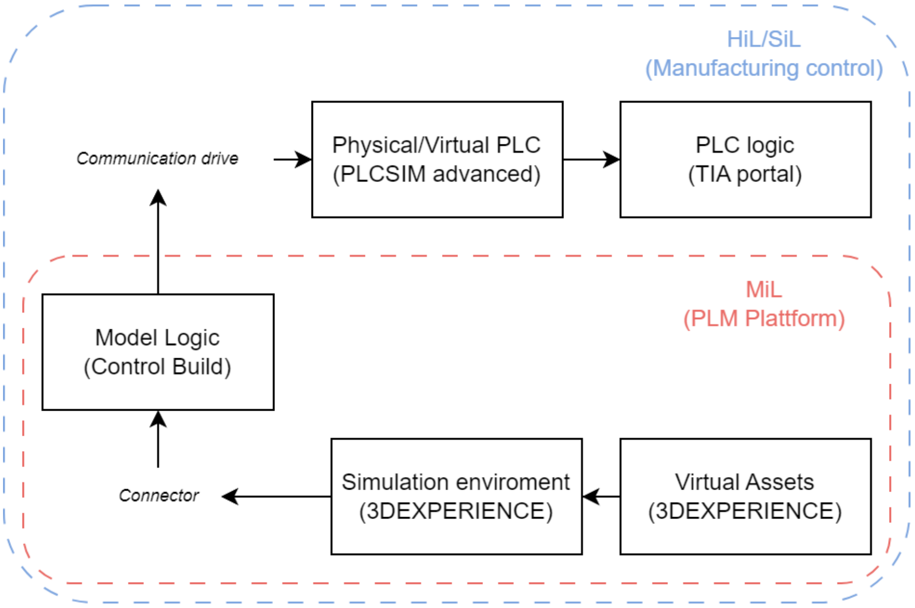


Figure 2.3: Representation of a digital manufacturing system

2.4 Software and communication protocols

The software available for this project includes tools from Dassault and SIEMENS. They are both large actors in the manufacturing scene with extensive experience in 3D modelling [21] and have a broad range of software solutions for a wide range of different production applications [22] [23].

2.4.1 3DEXPERIENCE

The 3DEXPERIENCE (3DX) platform by Dassault offers a unified environment that integrates multiple disciplines to streamline the development of products and services [22]. It emphasises a robust data model, collaboration tools and a unified user experience to enable secure, real-time collaboration [22]. Dassault aims to reduce development costs, speed up market launch times and minimise errors in concurrent work [22] through 3DX. Figure 2.4 demonstrates that 3DX extends a traditional PLM system. However, simulating only with 3DX is classified as MiL since the platform has no application for building robust control logic.

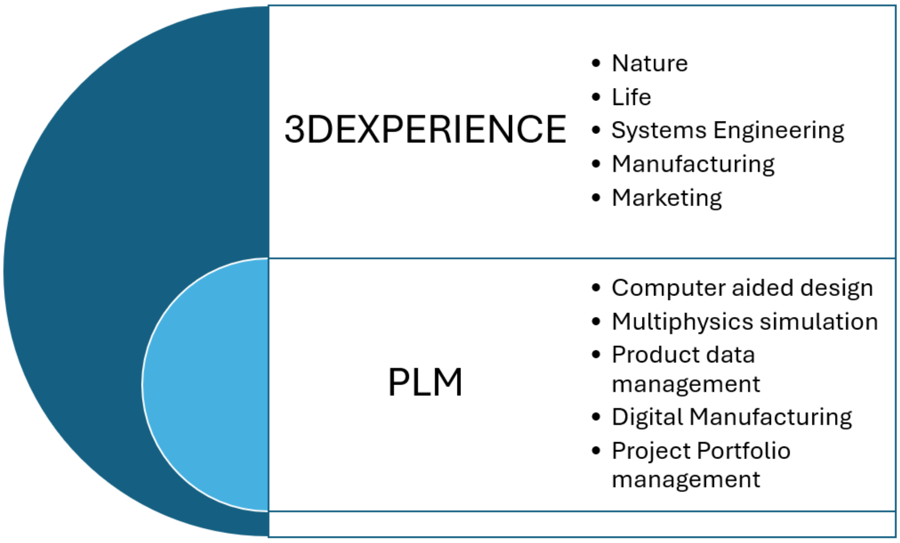


Figure 2.4: Illustration of 3DEXPERIENCE intended functionality [22]

2.4.2 PLCSim Advanced

Totally Integrated Automation Portal (TIA Portal) from Siemens offers a comprehensive set of digitalised automation services, from digital planning and integrated engineering to transparent operation [23] [24]. The TIA portal is commonly used to write control logic for SIEMENS PLCs in the actual plant, but to simulate this code before commissioning, SIEMENS has also developed a PLC software called PLCsim Advanced. This program is referred to as an *emulated PLC* because it contains the same connections and system architecture as a real PLC but can, for example, be run on a simulation computer [24].

2.4.3 FMI Interface

The Functional Mock-up Interface (FMI) is a widely adopted standard for model exchange and co-simulation of dynamic systems, promoting interoperability between simulation tools [25] [26]. A Functional Mock-up Unit (FMU) is a critical component of FMI, storing a dynamic model using the FMI standard, allowing it to be shared and integrated across various simulation environments [25] [26]. In practice, this means allocating memory on the host machine's physical hard drive to which multiple simulation environments can read/write. This architecture facilitates diverse simulation tasks and is currently the standard for communication between ControlBuild and 3DX.

2.4.4 UPC UA architecture

The OPC unified architecture (OPC UA) is a platform-independent service-oriented architecture that bridges the gap between the execution and simulation layer [15]. The communication protocol is commonly used to connect emulated PLCs to a simulation controller [20] and various drivers.

2.4.5 ControlBuild

CATIA ControlBuild (CB) by Dassault is a software environment for designing and validating control and monitoring systems aimed at industries like transportation, energy and manufacturing [27]. The tool adheres to safety standards 61508 and EN-50128 and aims to reduce development time and improve system performance [27]. However, most notable in a VC setting is the ControlBuild shared memory, which allows for co-simulation by utilising both the FMI interface and the OPC UA architecture. ControlBuild shared memory has the potential to serve as a "bridge" between simulation environments, web clients, controllers and PLCs. Depending on the setup, ControlBuild's shared memory can be used as the controller for a simulation environment or the actual simulation environment.

3

Methodology

This chapter explains the methods used through the project and how they contribute to answering research questions stated in Section 1.2.5. This chapter also states what criteria define a desirable system that meets the goals.

3.1 Procedure

The project involves collecting data to define a system that fulfils the goals stated in section 1.2.3. Following that is the creation of a framework for implementation and the proposal of deployment strategies for the frameworks that collectively can provide answers to the research questions stated in Section 1.2.5.

3.1.1 Data collection

Starting off relating to *R3*, data is collected in the form of instructional videos, models and instructions on digital manufacturing and simulation from field experts at Prodtex. This data is used as a foundational understanding of the industry, and the customers Prodtex currently has. The data regarding the three dimensions of VC presented in Figure 2.1 is then analysed to identify what critical assets and tasks must be present in the demonstrational manufacturing cell to be perceived as meaningful. These are stated in Section 3.2. Additional data is collected from a range of articles, ensuring a comprehensive and unbiased collection process. Keywords in the search for relevant articles include virtual commissioning, digital manufacturing and digital twin.

Qualitative methods are used to evaluate the collected data. Content analysis is conducted to identify key themes and patterns within the articles. The data is coded and categorised, allowing for an in-depth examination of the concept of VC. Thematic analysis is performed to interpret the findings further, providing a nuanced understanding of the differences between VC in different industries.

3.1.2 Simulation asset creation

The digital representation is created once the defining parameters are determined, providing answers to *R1*. Starting with creating simple models and then increasing model complexity [28] makes it easier to find the appropriate level of complexity. Most 3D models are imported virtual assets from Prodtex's asset library. Following that, necessary IOs like requests, emergency stops, and sensors are added to the model to allow the cell to be run by a controller.

Furthermore, a CB project that initially serves as the cell controller is created. Ladder logic and ST are used when building the controller program to allow for the migration of the logic programs easily in the following steps. The created model is set up for simulation with a 3DX environment using FMI interface [29]. Preparing the model requires setting up rules in the simulation environment that use IO's to determine the model behaviour, thereby further answering *R1*. However, the internal process should not be confused with a logic controller, an external device that manages IOs.

3.1.3 System definition

The system perspective for the co-simulation, displayed in Figure 3.1, is defined with regard to the software available and similar projects found in the literature. The system thereby fully answers *R1* and therefore serves as an outline for the project. It contains a controller from CB to run the simulation environment from 3DX, a model expansion that elevates business value, and the essential PLC that is tested by the rest of the system.

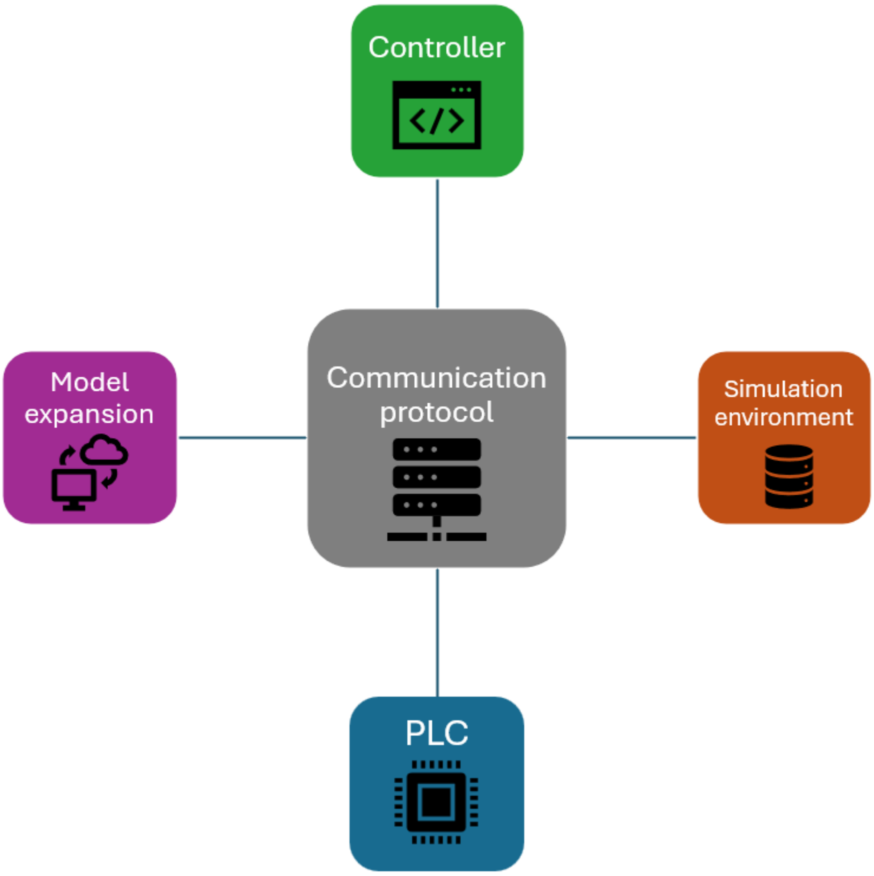


Figure 3.1: System perspective of Co-simulation

As shown in Figure 3.2, the information flow is determined to achieve an effective communication protocol. The model displays how the information from the virtual assets in the simulation model is communicated through the FMI interface and onward to communication drivers. Data reaching the PLC implies a HiL/SiL simulation, while data reaching a custom controller setup for high-fidelity simulation results in an elevated state of the simulation. This state will be referred to as Shared-Memory-in-the-Loop (SMiL) in future sections.

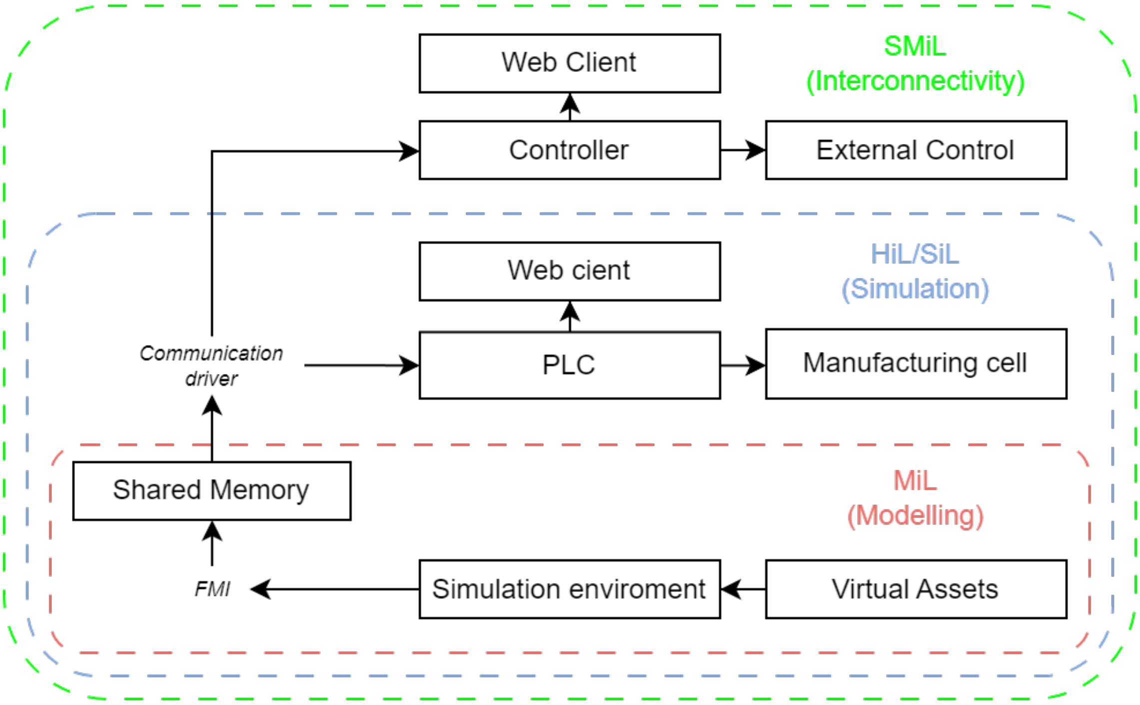


Figure 3.2: Information flow form MiL to SMiL

The last part of acknowledging *R1* is to expand the simulation model to include a web client to access the model from external controllers and mobile devices. This step involves dealing with different data types and communication protocols and is where most of the project's time is spent. This interoperability issue leads to a reevaluation of the traditional VC system architecture.

3.1.4 Virtual Reality

The cell is also evaluated in Virtual Reality (VR) as a complementary step to increase overall fidelity and see small details in the simulations, which relates to *R3*. For example, verifying that the robot tasks look correct and that sensors are correctly placed by entering the cell in VR. This method of evaluation also enables the manufacturer to see the intended functionality of the new cell and may be more comfortable with investing in VC.

3.1.5 Framework creation

Relating to *R2*, after the assessment, a framework for implementation is iteratively formed to be used as an outline for future greenfield VC projects in labour-intensive industry applications. The first concept study involves documentation, evaluation, and information collection about real industrial cases and their current approaches. With the framework determined, the final obstacle and part of *R2* is the VC solution's actual deployment, including simulation models, control logic and applications. Existing systems in the industry were observed continuously to refine the framework's final state.

3.1.6 Deployment strategy

The deployment of the framework has to be determined to verify that it is relevant and could be used in a real industrial setting. Verification is done by installing a mock customer computer with 3DX and CB and importing the control program, FMU, and simulation model.

3.2 Use case

To fulfil all four goals stated in section 1.2.3, the system design must achieve appropriate fidelity, meaning the level of detail that contributes to business value rather than just adds extra complexity [14]. As stated in *G1*, the model should have meaningful functionality in the context of the aerospace industry. This goal means that the simulation model should resemble a scenario found in real aerospace factories and be recognisable and relatable to personnel from aerospace manufacturers. The model should include critical safety assets like guards, gates and emergency stops and mainly focus on manual work to make the scenario realistic. The model should represent an actual manufacturing cell by providing feedback to the controller via sensory data and boolean variable communication. To increase complexity and reach at least the third level of VC relating to section 1.1.1 and [2], the control program should include at least one behaviour model or internal kinematic for some equipment. Again, relating to *G1*, the third level is chosen as a benchmark because it is not too technically complicated for a labour-intensive setting but complex enough to elevate fidelity. Alongside the control software, an intuitive HMI should be available that provides information about the cell status, can safely affect the environment through the controller, and has a good user experience [8].

Relating to *G2*, a few criteria must be met for the manufacturing cell to be considered virtually commissioned. The simulation model should be controlled from control software outside the 3DX platform, such as a custom controller or a real PLC. The co-simulation of the model and the control software proves that the control software works as intended, thereby fulfilling *G2*. Furthermore, *G3* requires a standardised implementation framework for VC as a service and its deployment. An implementation framework consists of a system architecture that displays how different components of a VC setup can be used together, forming one interconnected system [30]. Deployment strategies should display how VC as a service is a viable alternative to in-house development under the right conditions. Lastly, relating to *G4*, the business value-increasing factors of VC as a service should be explored in the project.

4

Results

This chapter presents the results, including the manufacturing cell, framework and deployment strategies.

4.1 Defining workspace

Digital representations from the Prodtex aerospace manufacturing library portray the third dimension of VC in Figure 2.1, *"what do we have"*. It is a greenfield scenario with the opportunity to introduce new equipment and working tasks. As its foundation, the manufacturing cell utilises non-disclosure-agreement-free assets from Prodtex, including fixturing, aerospace wings, and manufacturing staging. The first dimension of VC, *"what do we need"* is, in this case, VC as a service provided to Airbus. Because the VC solution should be a service, a requirement is to exclusively use equipment that Airbus could realistically adopt fast and reliably without reorganising the manufacturing procedure. Therefore, it is decided to develop an automated staging equipment system that improves workers' mobility. In the second dimension *"what do we want"*, the repetitive riveting task is identified as an appropriate subject for VC in a future scenario. Figure 4.1 displays the result of using the three dimensions of VC to determine what the simulation model should contain.

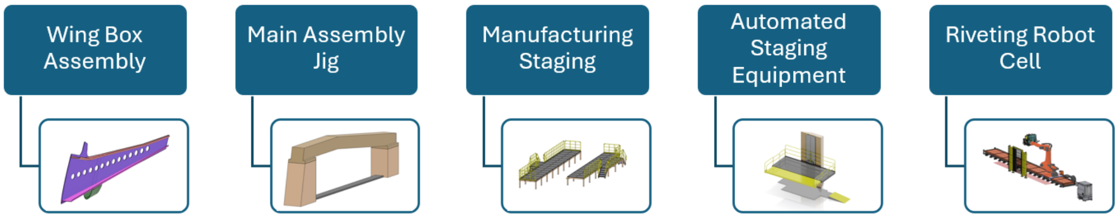


Figure 4.1: A collection of the resulting components of the simulation model

4.2 Simulation model

The first step of the cell creation is to import the templates to the wing box assembly, main assembly jig and manufacturing staging and scale them to each other. The result is a realistic static setting to the simulation model with designs similar to the industry standard. Figure 4.2 shows how the static simulation components look together.

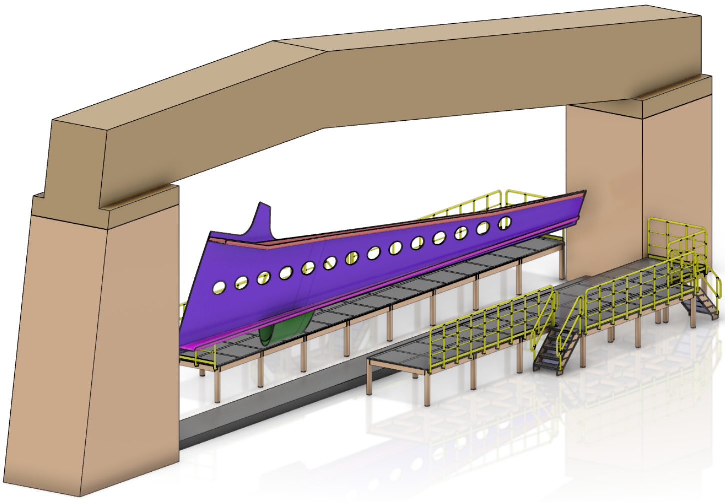


Figure 4.2: Static components in place

After importing the provided assets, the automated staging is designed from scratch, visible in Figure 4.3. The device has a six-meter-tall base with a cart attached to it. The cart can move vertically like an elevator, but it also has an extension that can move in the horizontal direction, allowing workers to come close to the wing. This improves safety since the gap between staging and wing decreases, improving ergonomic posture. The automated staging device has guards following the extension and two programmable safety gates. The base assembly also has a ramp leading up to the lowest position of the cart. Lastly, four predetermined heights are defined, and corresponding sensors are added to the four heights along with sensors for both gates and the extension. This simulation model fulfils the criteria specified in section 3.2.

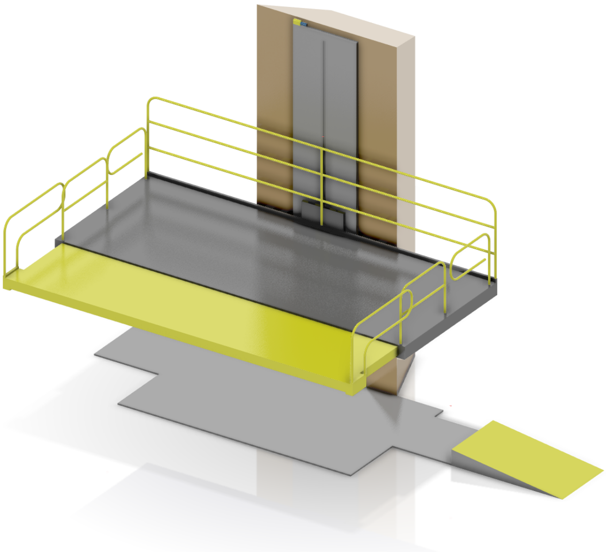


Figure 4.3: Automated staging equipment

Furthermore, the automated staging is imported to the simulation model, and the riveting robot cell is imported as an asset, visible in Figure 4.4. A KUKA robot on a rail with a riveting tool can be imported directly from the available simulation assets library at Prodtex. The robot can perform drilling and riveting operations along several wing box assembly points.

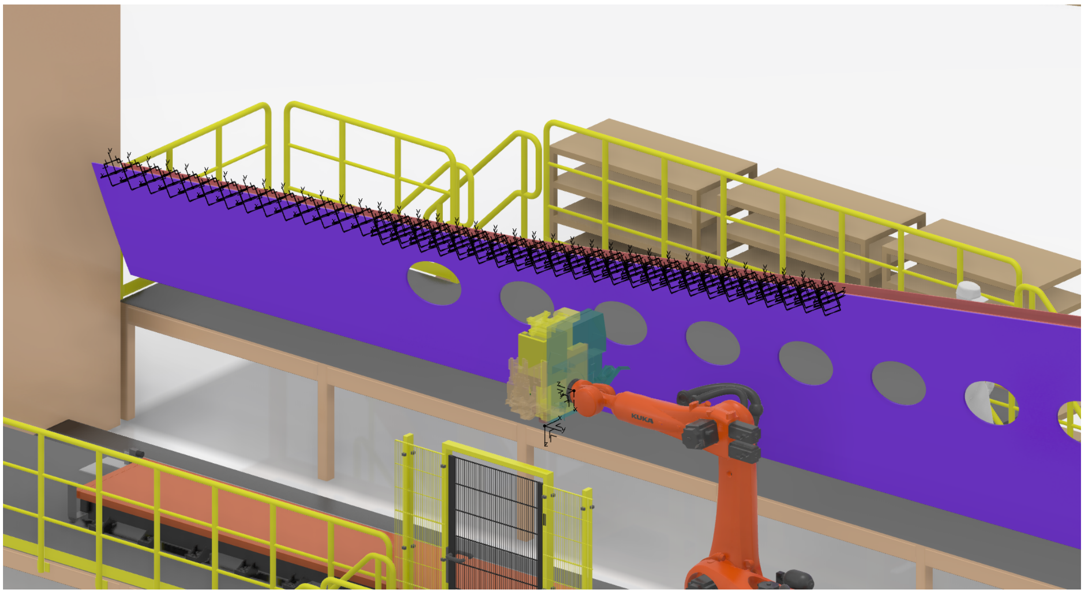


Figure 4.4: KUKA riveting cell

4.3 Logic Controller

With the simulation model complete, a CB controller is developed. The simple logic for tasks running in the automated staging equipment is created with ladder logic, displayed in Figure A.3. The controller sends requests to the staging to perform tasks displayed in Figure 4.5 and then waits for a response in the form of a sensor confirming the task to be completed. These tasks include moving to a specific floor, extending/retracting the extension platform, and opening/closing the gates. A mode selection variable allows the controller to be switched to a state where the controller models the staging kinematic behaviour instead of requesting tasks through a regulator in ST code, displayed in Appendix A. The controller continuously calculates the current position in every simulation time step and passes the staging its current height and length with the datatype "REAL". The regulator is created by defining a p-regulator, constraining the increments to a max/min speed, and skipping to the target position within a small distance. The resulting controller containing the ladder block to the left in Figure 4.5 and the ST block to the right is complemented with an emergency stop and set up with an FMU containing all variables.

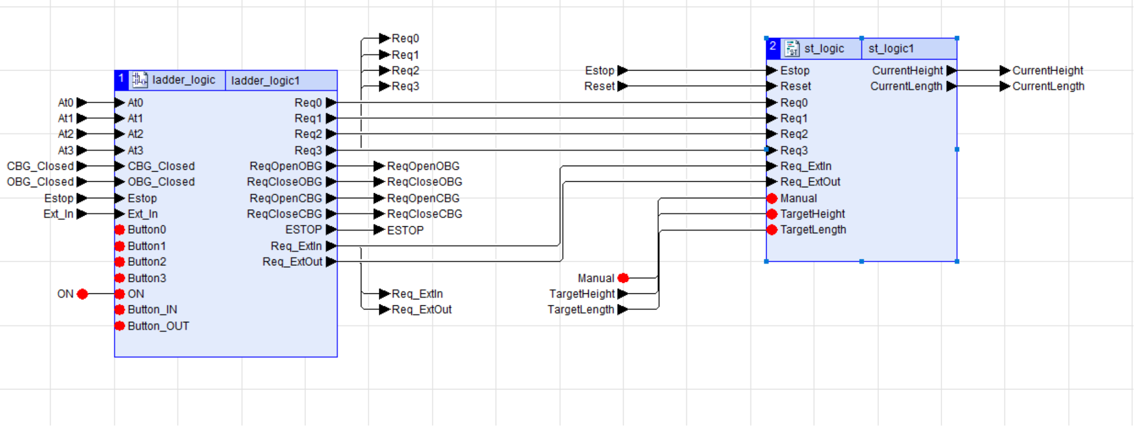
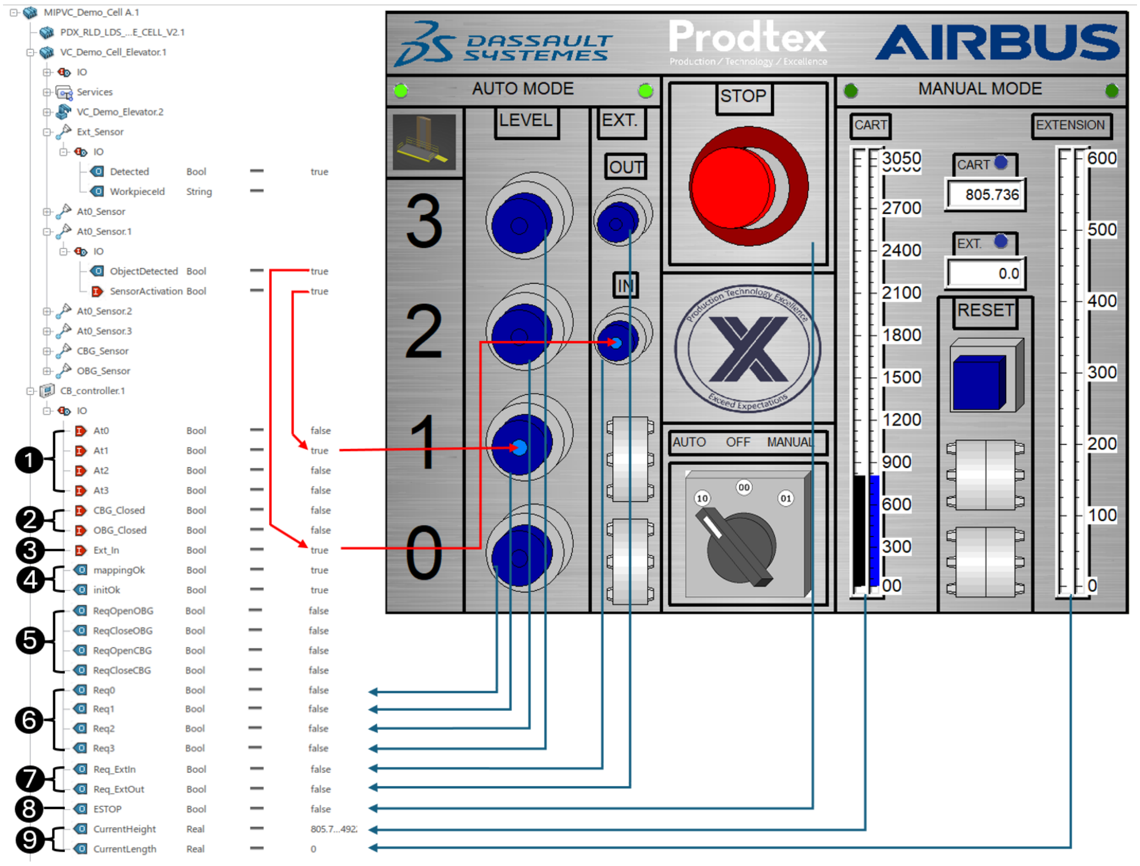


Figure 4.5: Logic controller components with IOs

When the controller adopts the intended behaviour, an HMI is created for the controller in CB. The controller in Figure 4.6 has an intuitive design with a large E-stop in the middle and a mode switch with the possible values Auto, Off and Manual, which can be changed anytime. The Estop will automatically reset the mode to Off. To the HMI's left are four light buttons for the predetermined levels. The HMI lights the outer parts of the level buttons when a request is active, while the inner light of the buttons is connected to the level sensors. The same functionality is adopted for the extension buttons. To the right of the HMI, there is a manual mode that allows the user to set a specific height by pressing on the desired height/length. The black bar will instantly confirm user selection, along with a display showing the exact target height. The controller gives live feedback on the cart position displayed in the blue bar. A reset button also sets the target height to the minimum allowed value. The modes are fully interoperable, so pushing a level button will update the manual selection and position feedback. The other way around, manually inputting a position will update the inner position lights, and correcting a manual input with a preset button is possible. Note that there is no way for the user to control the gates individually. This is for safety reasons since it is safer not to allow operators to open a gate when it could result in a fall from the elevator. This controller fulfils the criteria specified in the use case in section 3.2.

4.4 Model Simulation

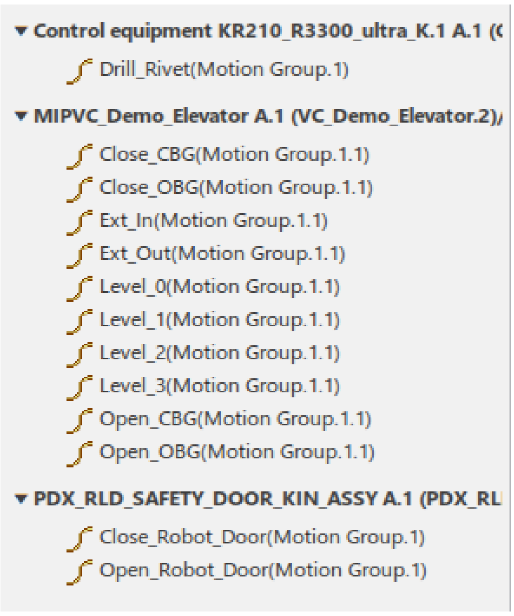
The next step in the VC process is to bridge the interoperability gap and connect the logic controller described in section 4.3 with the simulation model described in section 4.2. This step is done by creating an FMU and associating that with a control resource in 3DX, continuously passing variables to the simulation. As demonstrated in Figures 4.6 and A.1, the IOs are accessed in 3DX through ControlBuild's memory allocation on the local hard drive. This memory is central to the resulting framework and will be referred to as the *shared memory*.



```
MIPVC_Demo_Cell A.1
  PDX_RLD_LDS_...E_CELL_V2.1
  VC_Demo_Cell_Elevator.1
    IO
    Services
    VC_Demo_Elevator.2
    Ext_Sensor
      IO
        Detected            Bool     —    true
        WorkpieceId         String   —
    At0_Sensor
    At0_Sensor.1
      IO
        ObjectDetected      Bool     —    true
        SensorActivation    Bool     —    true
    At0_Sensor.2
    At0_Sensor.3
    CBG_Sensor
    OBG_Sensor
  CB_controller.1
    IO
      At0                   Bool     —    false
      At1                   Bool     —    true
      At2                   Bool     —    false
      At3                   Bool     —    false
      CBG_Closed            Bool     —    false
      OBG_Closed            Bool     —    false
      Ext_In                Bool     —    true
      mappingOk             Bool     —    true
      initOk                Bool     —    true
      ReqOpenOBG            Bool     —    false
      ReqCloseOBG           Bool     —    false
      ReqOpenCBG            Bool     —    false
      ReqCloseCBG           Bool     —    false
      Req0                  Bool     —    false
      Req1                  Bool     —    false
      Req2                  Bool     —    false
      Req3                  Bool     —    false
      Req_ExtIn             Bool     —    false
      Req_ExtOut            Bool     —    false
      ESTOP                 Bool     —    false
      CurrentHeight         Real     —    805.7...492
      CurrentLength         Real     —    0
```

Figure 4.6: Logic controller HMI with corresponding simulation IOs

Co-simulation can be executed when the IO communication between the controller and the simulation environment has been set up. First, tasks for the programmable resources like the automated staging is created. In 3DX this means programming joint values with targets and motion profiles, determining speed and acceleration for the motions. As shown in Figure 4.7, there are tasks to be run when the corresponding requests in IO groups 5,6 and 7 in Figure 4.6 are active. There are also tasks for the robot riveting station that opens and closes the safety door and a collection of every sub-task required to complete the riveting task.



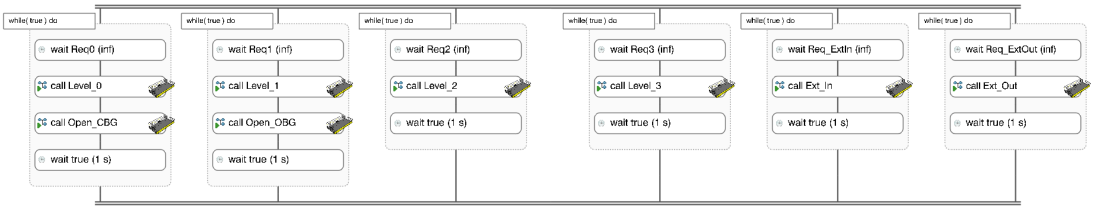
▾ **Control equipment KR210_R3300_ultra_K.1 A.1 (**
- Drill_Rivet(Motion Group.1)

▾ **MIPVC_Demo_Elevator A.1 (VC_Demo_Elevator.2)**
- Close_CBG(Motion Group.1.1)
- Close_OBG(Motion Group.1.1)
- Ext_In(Motion Group.1.1)
- Ext_Out(Motion Group.1.1)
- Level_0(Motion Group.1.1)
- Level_1(Motion Group.1.1)
- Level_2(Motion Group.1.1)
- Level_3(Motion Group.1.1)
- Open_CBG(Motion Group.1.1)
- Open_OBG(Motion Group.1.1)

▾ **PDX_RLD_SAFETY_DOOR_KIN_ASSY A.1 (PDX_RL**
- Close_Robot_Door(Motion Group.1)
- Open_Robot_Door(Motion Group.1)

Figure 4.7: Automated staging tasks

The automated staging tasks shown in Figure 4.7 are set up to run in parallel, like in Figure 4.8, so they run instantly when the controller requests them. In this case, the simulation environment controls the gates entirely, not the controller. This is because they are set up in a series of tasks for going to floor 0 and floor 1, where they are allowed to be opened. Taking the logic away from the controller and into the simulation environment results in the controller not waiting for the gate to be closed before moving but instead performing the gate-closing operation alongside the next instruction. This effect is not desirable in itself, but it is used to demonstrate a possible scenario that would have been detected with co-simulation, underscoring the importance of VC.



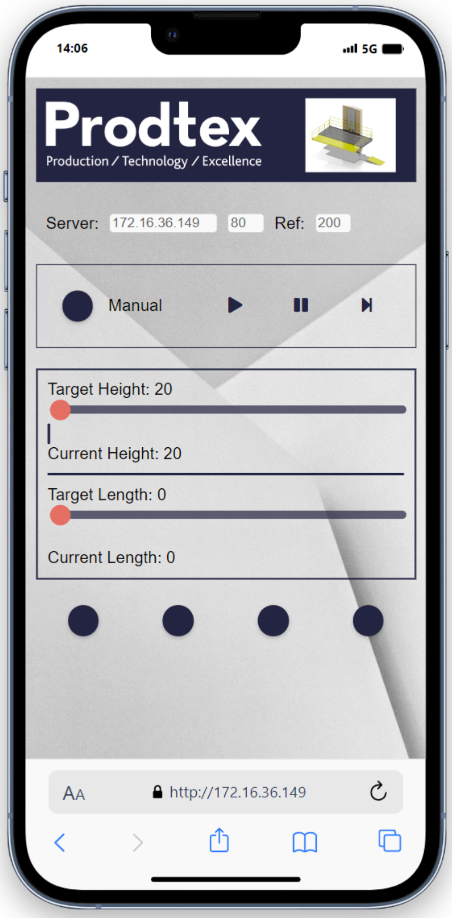
```
while( true ) do
    wait Req0 (inf)
    call Level_0
    call Open_CBG
    wait true (1 s)

while( true ) do
    wait Req1 (inf)
    call Level_1
    call Open_OBG
    wait true (1 s)

while( true ) do
    wait Req2 (inf)
    call Level_2
    wait true (1 s)

while( true ) do
    wait Req3 (inf)
    call Level_3
    wait true (1 s)

while( true ) do
    wait Req_ExtIn (inf)
    call Ext_In
    wait true (1 s)

while( true ) do
    wait Req_ExtOut (inf)
    call Ext_Out
    wait true (1 s)
```

Figure 4.8: Simulation tasks set up in parallel

The last setup needed for the simulation environment is to create variables that control individual joint values in the automated staging, thereby allowing kinematics to be modelled in the logic controller instead of the simulation environment. By connecting *Current Height* and *Current Length* to the controller, like in Figure 4.5, the automated staging kinematics are now modelled in the controller instead of the simulation environment. Relating to the use case in section 3.2, the internal controller kinematics allows the system to achieve VC level 3, which is the requirement.

4.5 Logic Control Expansion

To elevate business value and make this VC solution more relevant to the modern industry, a new node is connected to ControlBuild's shared memory, pushing the VC solution further. The node is connected by setting up a web client with a *CB web server*. This web server can be complemented with functions in *JavaScript* that allow for read and write operations to shared memory. The JavaScript functions are used alongside a *HTML* web page that had access to some of the variables in the shared memory; it is decided not to give the web client access to all variables for safety reasons. The HTML page visible in Figure 4.9 is developed to serve as an extension of the CB controller. When the CB controller is switched to manual mode, the web page will display that manual mode is active, and the user can change the target position with sliders and display the current height with a bar underneath. There are also lights at the bottom of the page displaying if the cart is currently at one of the predetermined heights. This web page controller is effectively an external version of the manual mode on the CB controller that can run on any connected device.



Server: 172.16.36.149 80 Ref: 200

Target Height: 20

Current Height: 20

Target Length: 0

Current Length: 0

Figure 4.9: Shared memory in web client running on a mobile phone

To further underscore the potential of having a running web client during simulation. The *Gamepad API* is utilised in JavaScript to connect a physical gamepad to the web client, allowing the user to adjust the target position precisely with two analogue sticks. It can be noted that the latency from the analogue stick to visual feedback from the 3DX simulation is barely noticeable, and it appears like this is a perfectly viable option to control the automated staging accurately. Additional possibilities with physical input devices take advantage of VR demonstration, where the user can virtually stand beside the staging and control it using a gamepad, displayed in Figure A.2.

The fourth and last entry to the shared memory for this VC is created by setting up in CB with the available communication driver *PLCsim Advanced driver*. This driver gives PLCsim Advanced access to the shared memory. Usually, when connecting ControlBuild to an emulated PLC, like in this case, the purpose is to turn CB into a black box that only passes the variables from the logic program in the PLC to the simulation environment. But in this case, the logic is kept in CB, and the PLC is instead the black box that gathers sensory data and sends requests from CB to the fictional actual manufacturing cell.

4.6 Framework Creation

The resulting system architecture is used to create a standard for future implementations. The framework will be referred to as Shared-Memory-in-the-Loop (SMiL), displayed in Figure 4.10. SMiL is centred in a shared memory, allocated on the simulation machine's physical drive. All data is centralised, fully accessible and manipulated from the four nodes: PLC, FMI, Logic Controller and Web Client. The shared memory is asynchronous and continuously reads/writes the different "nodes" corresponding to the simulation functionality.

The PLC node represents a real or emulated PLC, exchanging all IOs needed for commissioning, thereby setting the state of the real manufacturing cell. This node's purpose is to pass IOs between the PLC and the shared memory, but it does not specify where the main control logic originates. The three alternatives are:

1. The PLC is only passing on IOs to the actual manufacturing cell, and the main logic exists in a logic controller node.
2. The PLC contains some control logic, for example, safety functions, while the rest exists in the logic controller node.
3. The PLC contains all control logic, excluding the logic controller node from the shared memory.

The first sub-node to the PLC cell represents the actual manufacturing cell with tooling, robots, actuators, and sensors continuously generating data. The other sub-node is Industrial HMI, used alongside the actual manufacturing cell and requires high robustness and integration with the PLC. This HMI is usually hardware from the same provider as the PLC. Relating to the use case in section 3.2, this demonstrational setup uses PLCsim advanced as the PLC node with no control logic in the emulated PLC.

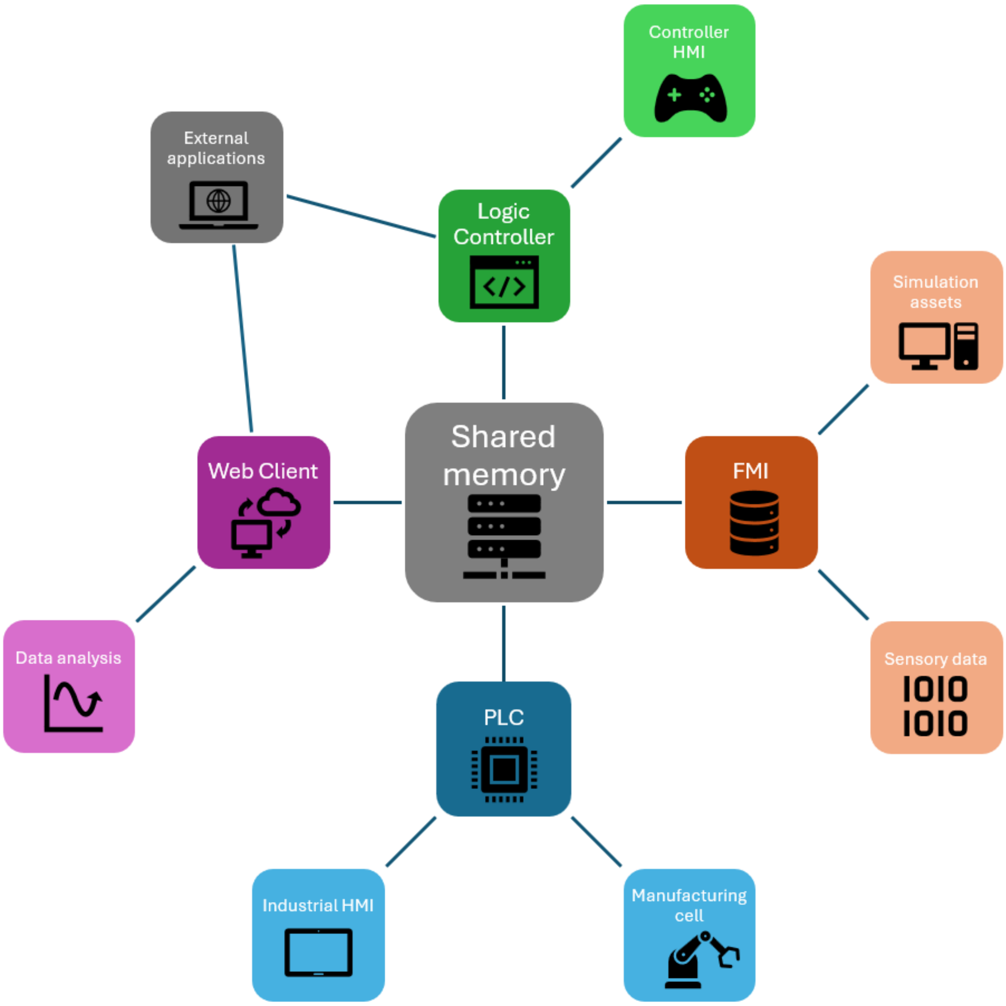


Figure 4.10: Proposed system architecture SMiL

The optional FMI node represents the communication protocol as a bridge between the shared memory and the simulation model. The two sub-nodes contain sensory data and simulation assets. The two sub-nodes in this node are simulation assets and sensory data, which are included in the simulation environment but are separated because of their different functionalities. Sensory data can be excluded in some simpler VC applications, for example, where the number of sensors is small, like in many labour-intensive applications. Relating to the use case, the FMI node in this project represents an FMU from the shared memory to the simulation model and simulation IOs, both in 3DX. However, it is possible to create a low-fidelity simulation model consisting of a logic controller, replicating the manufacturing cell's IO behaviour.

The logic controller is another optional node representing a control application that is not a PLC, containing control logic like a ladder, sequential function charts and ST. This node contrasts traditional VC approaches in that the logic is centralised and accessible rather than a hidden and closed PLC process. This controller can contain various levels of VC, from simple conditions to entire kinematic models. From the direct controller node is the controller HMI, a tool that interacts with the logic controller. The controller HMI differs from an industrial HMI in that it does not require the same level of robustness and reliability, which opens up new possibilities in a pure simulation environment without sensitive or dangerous hardware. The other sub-node from the logic controller controller is the external application, a multipurpose node representing every imaginable addition to the logic controller. Relating to the use case, this project utilises the logic controller node as the main control logic. For kinematic modelling, using ControlBuild components for both the logic controller and the controller HMI, in a scenario where all logic exists in the PLC node and the simulation environment in the FMI node, it is possible to exclude the logic controller node.

The optional web client node is another node missing from many VC applications today. This node represents hosting a server with access to the shared memory and setting up infrastructure to read and write to the shared memory through remote devices on the network. This node provides a new dimension of possibilities since it provides connectivity that can be used to analyse all data flowing through the system. Data analysis is a sub-node because it is crucial to elevate fidelity and accuracy in the VC setup. The other sub-node is shared with the logic controller, and its connections to the web client are described as every possible addition to the web client that uses the accessible shared memory and performs calculations or manipulations. Relating to the use case, this framework fulfils the criteria and completes the intended use case.

4.7 Deployment

When a customer orders a VC solution from a supplier, Three different deployment strategies are displayed in Figure 4.11. The first deployment strategy, Data feed, means that the customer iteratively feeds the developer with simulation assets like models, sensors and IOs. Since this data feed usually occurs during the planning phase of the manufacturing cell, it is optimal in a greenfield scenario. It allows the VC developer to provide feedback about the cell design as the VC developer can simulate each iteration to find improvement potential. Therefore, VC with this method involves inventing control logic and exploring alternative cell designs. The final delivery to this customer is simply a data folder containing the SM behaviour, usually as a controller. The advantage of this method is that it allows for a developer at an early stage, which can result in a quality improvement for the final result. However, this method is very time-consuming since it requires close collaboration between the customer, who feeds constant updates, and the developer has to adapt at the same rate. A successful deployment to a mock customer verified this strategy. The co-simulation could be executed without the need for any configuration.

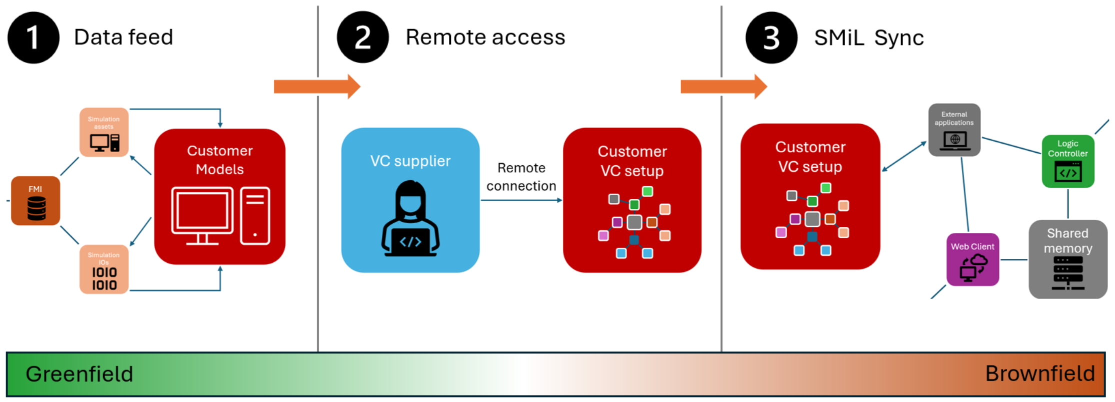


Figure 4.11: Three deployment methods of SMiL

The following strategy is referred to as remote access. This means keeping the entire system architecture on the customer's machines and letting the contractor connect to them with secure, remote access. This way, the sensitive information will never leave the secure drives and is therefore easier for customers to authorise. From the developer's perspective, this currently means some latency affecting the development process negatively. There is also the need to install all necessary software on the host computer, which could potentially cause issues like licence costs and firewall clashes. However, this deployment strategy runs the system on a single machine, just like the data drop strategy, but on the customer computer, allowing for efficient development in greenfield and brownfield scenarios. This strategy is verified by remotely connecting to the mock customer computer and setting up the VC solution through the remote connection.

The third strategy, SMiL sync, is the idea of having an SMiL framework running on a machine connected to the actual manufacturing cell and via an external application connected to a secure web client, letting the shared memory synchronise to another SMiL setup containing the logic to be tested. For example, the customer might already have a VC setup, and a contractor might want to demonstrate an improvement in the logic. The customer can set up a web client to which the customer can connect their shared memory and sync their own to it. This approach can potentially become an extremely efficient way to test, especially brownfield scenarios. SMiL sync is perfectly suitable for a fast-changing environment where a quick and somewhat accurate demonstration of a control improvement proposal will be run. This strategy is not verified in this project due to a lack of hardware.

5
Discussion

This chapter discusses the results chapter and identifies selling points and problem areas with the results.

5.1 Fidelity and Complexity

This project explores 3DX's and CB's capabilities in the field of VC. It demonstrates how fidelity can be elevated using extensive tools contributing to a streamlined 3D model development process. The standardised file system for models comes with the opportunity to co-simulate a manufacturing scenario with high fidelity. The literature collectively sheds light on the potential of VC in manufacturing and the challenges of implementing VC on a complexity level that generates fidelity. One of these challenges that could also be identified in this project was the interoperability gap, which was overcome by using the shared memory model. CB is hosting the shared memory and communication drivers in this setup, making the application the simulation's centre. However, having the logic controller outside the simulation environment increases simulation complexity without increasing fidelity. 3DX currently lacks a dedicated controller and relies on external applications for simple MiL simulations, resulting in complex setups even when low-fidelity setups are sufficient. An alternative to using an external controller for simple simulations, with lower prerequisites in terms of experience and preparation time, is to have proper control software built into the PLM platform. CB is a powerful tool for creating models, controllers, and HMIs, but being a standalone software that does not qualify as an emulated PLC takes away some of its usefulness. Interoperability in VC requires simplicity and standardisation, which generally are strengths with 3DX, but in this particular case, it is not utilised fully.

5.2 Framework evaluation

The VC process steps differed heavily in time consumption and complexity, which is essential to consider in a high-value project with narrow time and budget constraints. Therefore, technical competency alongside the right tools and deployment strategy is critical to a complex VC implementation. In section 1.2.2, interoperability is presented as a main challenge for VC and, therefore, naturally a selling point. Furthermore, the thematic analysis in section 3.1.1 suggests that successful VC solutions have scalability and versatility in common, as shown in Figure 5.1 along with challenges that may come with the selling points.

5.2.1 Selling points

Three selling points can be identified from framework evaluation, as displayed in Figure 5.1. Scalability in a VC setting implies the possibility of taking a simple VC demonstration and expanding the solution with complexities and controller applications without changing the controller logic. Interoperability implies co-simulation capability with different software providers, which the other nodes in the framework represent. Versatility suggests a variety of industrial applications that can use SMiL, from automated manufacturing settings focusing on logic behaviour and sequencing to a fully manual work setting concentrating on data analysis and equipment safety. The scalable and interoperable nature of SMiL also allows for industry 4.0 concepts like digital twins, IIoT, cloud computing and AI to play critical roles in the solution. For example, the complexity of a digital twin can be harnessed by connecting a web client to the sensory data in the IIoT.

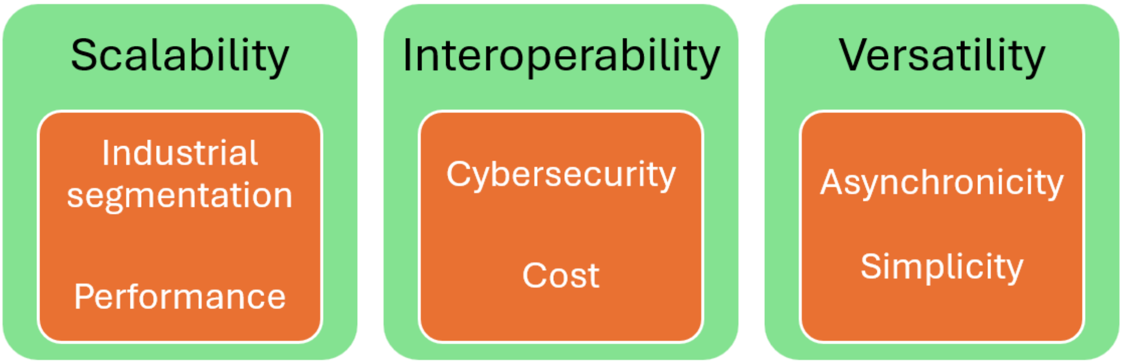


Figure 5.1: Selling points with respective challenges

SMiL possesses these three attributes, increasing the framework's impactfulness. Because SMiL can theoretically have a large number of nodes connected to it, it is scalable. Thanks to its scalable nature, SMiL is especially advantageous in a green-field scenario where new ideas frequently need assessment. Furthermore, because the nodes can use a variety of communication standards from different automation software providers, the framework is interoperable. Interoperability can be seen as a requirement for even a very basic simulation, but the shared memory in SMiL offers interoperability beyond co-simulation. Lastly, the framework is versatile due to all the varying nodes that can be attached to the shared memory. For example, it can be set up to test logic code by simply manipulating variables in the PLC, or it can be used to analyse data from a simulation environment in real time. The use of AI is a possibility, given the versatility of SMiL. In theory, it could control a whole manufacturing cell by itself. Ultimately, the SMiL model expands SiL/HiL simulation and could potentially provide new business opportunities for VC service providers.

5.2.2 Framework challenges

The selling points indirectly bring problem areas, stated in Figure 5.1. Generally, the literature is united in the opinion that the strength of VC is found in automation.

The industry segment that benefits from virtually commissioning their manufacturing, usually large and mature manufacturers, will likely have come further in their VC journey and might be prevented from switching VC platform providers effectively. In such cases, scalability is not a valued trait but only brings unnecessary complexity. Performance is another question that arises in a complex VC setup because every new node the data has to travel through to reach its destination introduces latency. In this system, a signal from the mobile device encounters a web client, a communication driver and then an FMU that both delay it slightly, although this has little to no effect on the demonstrations cell in this project, a substantially more complex setup can potentially have a latency affecting the logic functionality.

Moreover, SMiL's interoperability largely depends on the centralised information stored in the shared memory, which raises the question of cyber security. As cyber security is not within this project's scope, it is simply pointed out as a challenge. Implementation cost, however, is highly relevant for this project and must be considered when deploying the framework. Interoperability requires communication standards and standardisation, which in turn requires engineering hours. On the other hand, a standardised framework that already has these key components in place will not necessarily have to observe cost increments with interoperability.

The versatility of SMiL is enabled through the shared memory's ability to read/write from different types of nodes continuously. Communication of this kind is usually handled in an industrial PLC with a fixed clock cycle time. Instead, multiple nodes read and write the shared memory whenever new information is available. This can lead to different interpretations of system states when updating the shared memory, making the nodes asynchronous. Lastly, simplicity scales with versatility, making the framework harder to understand and implement. Simplicity is identified as the primary challenge with the framework because it is necessary for a manufacturer to have a foundational understanding of VC before SMiL, an expansion of a VC setup, can be explained. Manufacturers do not always have experience with digital manufacturing, which could make it hard for them to see the benefits of SMiL. Perhaps as the industry grows mature with industry 4.0 concepts, VC will be more common knowledge, making SMiL easier to motivate.

These three problem areas should be considered before implementing the full SMiL. However, the optional nodes FMI, logic controller, and web client can be excluded from a particular setup, making it identical to traditional VC setups with a PLC and a simple simulation environment. Therefore, this challenge should not be seen as a reason not to implement SMiL but rather as a guide for what nodes should be included in the setup.

5.3 Example use cases

SMiL is a versatile framework that can be modified to fit many industrial applications by adjusting what nodes to include in the setup. Below, two examples of alternative implementations will be presented.

5.3.1 Labour intense industry VC as a service

This example mainly aims at suppliers delivering a greenfield scenario VC. In this particular case, there is no need for a logic controller since the labour-intensive workstation will have a simple PLC code that can be communicated straight to the FMI in a 3D environment. For the same reason, data analysis or external applications via a web client will not increase fidelity enough to justify the extra complexity. Figure 5.2 displays the resulting alternative framework.

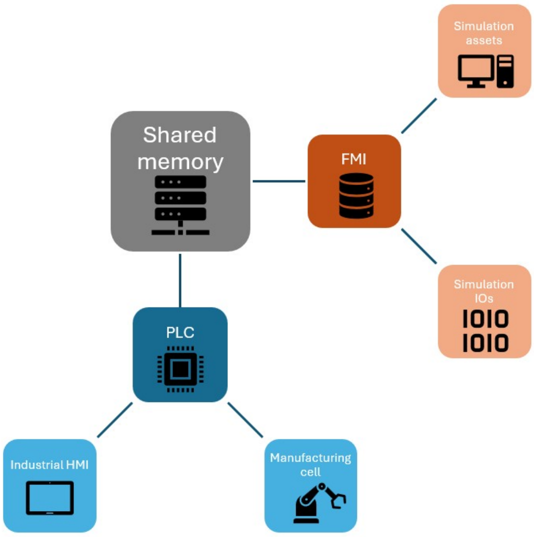


Figure 5.2: SMiL alternative 1

5.3.2 High-level automation industry as a service

In highly automated manufacturing with potentially hundreds of IOs, a heavy focus will be on evaluating the logic regarding interlocking, sequencing and performance. Therefore, a 3D simulation might be too big of a scope and cost more than the worth of extra intelligence acquired from a 3D simulation. Thus, the FMI can be excluded, as displayed in Figure 5.3, and a simplified simulation model can be created in the logic controller. This way, PLC debugging can be conducted more simply, streamlining the process.

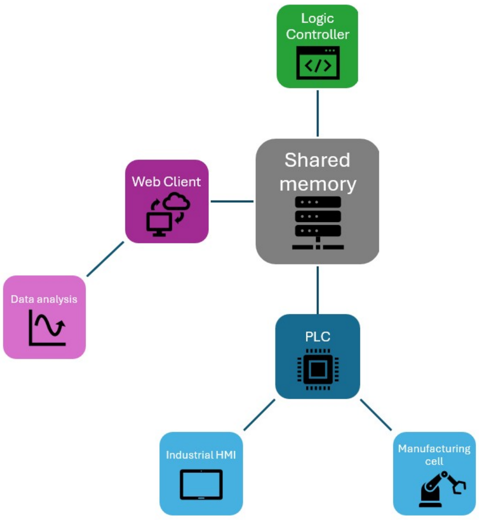


Figure 5.3: SMiL alternative 2

5.4 Deployment evaluation

When evaluating deployment strategies, one issue with the first strategy is that transferring a data system from one company to another, especially a large customer, means there will be sensitive assets on company drives/servers that cannot always be used externally. Allowing external users to access live simulation models in a cooperative environment is not an option for most manufacturing companies due to the risk of data loss or unwanted model changes. 3DX has the option to share released, read-only versions of 3D models that can rule out the risk of file corruption but have no effect on confidential assets that can still prevent data sharing. However, to virtually commission the sensitive digital assets, the contractor needs the simulation model details to develop the control logic correctly. This contradiction has no straightforward solution that is effortless for all parties. Still, the framework developed in this project allows for three different approaches to the interoperability issue, as displayed in Figure 4.11. A third option, not as competitive as the previous two in the current maturity state of the industry but holding great potential for the future of highly interconnected digital twins, is the deployment strategy referred to as SMiL Sync. This approach's strength lies in its possibility of allowing multiple different VC setups to be developed in parallel and used interchangeably. This deployment strategy can, in future research, be verified by setting up a third computer hosting a server to synchronise both shared memories.

6

Conclusion

This chapter serves to concisely answer the research questions presented.

6.1 Summary

This thesis explored the deployment of VC as a service within the aerospace industry, aimed at standardizing co-simulation processes to enhance manufacturing efficiency and reduce commissioning times. The study introduced deployment frameworks built on extensive simulation using the PLM platform 3DEXPERIENCE and the control software ControlBuild. The study emphasised understanding the complexities of VC technologies, identifying the necessary software/hardware integration and exploring the relationship between simulation fidelity and deployment complexity.

6.1.1 Key findings

The developed framework proved effective in standardizing the deployment of VC across simulation environments used within aerospace manufacturing. It facilitated the interoperability of isolated simulation systems and, by example, demonstrated that the framework can improve future VC solutions, thereby increasing the predictability of commissioning projects. One of the central challenges addressed was the interoperability between simulation tools and platforms. Bridging the interoperability gaps was done using shared memory and communication protocols. These drivers are crucial for the scalable deployment of VC services. The deployment of the framework was validated through tests that demonstrated its effectiveness in a controlled environment, suggesting that it can be generalized and applied to other industries and manufacturing settings and provide new business opportunities for VC service providers.

6.1.2 Research questions

- *R1: How can tools from Dassault Systèmes be used to demonstrate a Virtual Commissioning business case?*

Engineers can use Dassault Systèmes' PLM platform 3DEXPERIENCE and logic creation software ControlBuild to create simulation environments and co-simulate them with PLCs and external applications. The manufacturing cell in this project demonstrated that high fidelity could be achieved with the toolset available from Dassault Systèmes software, proving that the tools offered are sufficient to set up a VC solution for business demonstrations.

- *R2: How can a standardised VC as a service approach be developed to utilise simulation tools efficiently in a labour-intensive industry setting?*

The framework SMiL, developed in this project, provides a standardized approach for multiple VC applications, including VC as a service in the labour-intensive aerospace industry. A framework can be developed by setting up a demonstration VC setup and generalising the functionality of every system node to allow others to replicate similar setups.

- *R3: What factors and level of detail result in business value for VC solutions?*

This project presents Scalability, Interoperability and Versatility as selling points for VC applications, all connected to fidelity. A VC solution with a simulation model complexity reflecting necessary detail without being overly complex for the context generates business value. A VC solution with the appropriate simulation complexity while being scalable for the future, interoperable for the different software providers, and versatile for the upcoming commissioning projects generates even more business value. Ultimately, a VC solution that has all these attributes but is also standardised in development and deployment, accessible across the company and well documented has the potential of generating a substantial business value both short term and long term.

6.2 Theoretical and Practical Contributions

The theoretical implications of this research extend to the enhancement of VC as a service by visualizing a generalized system architecture with possibilities for industry 4.0 practices. By advancing the understanding of VC possibilities in a labour-intensive setting, the thesis contributes to the broader field of digital manufacturing, offering a flexible framework that can be adapted for various simulation applications. Practically, this thesis provides a blueprint for OEMs and suppliers to adopt VC with fewer prerequisites than traditional VC setups, potentially leading to significant cost savings for overall commissioning and an opportunity for contractors to sell a valued service.

6.3 Future work

This thesis demonstrates some of the substantial benefits of virtual commissioning as a service within the aerospace industry, providing a scalable, interoperable and versatile framework for its implementation. Among them is the scalable nature of shared memory, allowing for industry 4.0 concepts like AI to be used during commissioning. There is undoubtedly potential in combining AI and manufacturing, which suggests that an AI-assisted shared memory might also have potential. Two future research topics could be how AI can evaluate control logic for better PLC code or how AI can improve simulation models by giving them more realistic characteristics.

Bibliography

[1] W. L. Berry and M. C. Cooper, “Manufacturing flexibility: methods for measuring the impact of product variety on performance in process industries,” *Journal of Operations Management*, vol. 17, 1999.

[2] A. Albo and P. Falkman, “A standardization approach to virtual commissioning strategies in complex production environments.,” *ScienceDirect*, 2020.

[3] S. Singh and S. C. Misra, “Exploring the challenges for adopting the cloud plm in manufacturing organizations,” *IEEE Transactions on Engineering Management*, vol. 68, no. 3, 2021.

[4] F. G. Medina, A. W. Umpierrez, V. Martínez, and H. Fromm, “A maturity model for digital twin implementations in the commercial aerospace oem industry,” in *2021 10th International Conference on Industrial Technology and Management (ICITM)*, 2021.

[5] N. Striffler and T. Voigt, “Concepts and trends of virtual commissioning – a comprehensive review,” *Journal of Manufacturing Systems*, vol. 71, 2023.

[6] R. Carlier, J. Gillis, E. Rademakers, G. Borghesan, P. De Clercq, C. Ganseman, K. Stockman, and J. D. M. De Kooning, “A digital twin framework for virtual re-commissioning of work-drive systems using cad-based motion co-simulation,” in *2023 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*, 2023.

[7] T. Lian, M. Ran, W. Zhao, and P. Liu, “Design of virtual commissioning system for intelligent manufacturing production line based on digital twin,” 2022.

[8] B. Davis, “How to validate machines with virtual commissioning: Virtual commissioning begins with a vision of the desired machine behavior and sequence of operation.,” *Plant Engineering*, vol. 74, no. 7, 2020.

[9] D. Siegrist, A. Matsikis, S. Dirbach, A. Volz, and F. Bellalouna, “A virtual commissioning selection approach for machine automation,” in *2022 IEEE 27th International Conference on Emerging Technologies and Factory Automation (ETFA)*, 2022.

[10] M. Dahl, K. Bengtsson, P. Bergagård, M. Fabian, and P. Falkman, “Integrated virtual preparation and commissioning: Supporting formal methods during automation systems development,” *IFAC-PapersOnLine*, vol. 49, no. 12, 2016.

[11] Prodtex, “Prodtex home.” https://www.prodtex.com/, 2024. Accessed: 2024-03-01.

[12] Airbus, “Airbus home.” https://www.airbus.com/, 2024. Accessed: 2024-03-11.

[13] L. Svedlund, A. Albo, and B. Lennartson, “Co-simulation of rigid interactions using differential algebraic equations,” in *2021 IEEE 26th International Conference on Emerging Technologies and Factory Automation (ETFA)*, 2021.

[14] R. Raffaeli, P. Bilancia, F. Neri, M. Peruzzini, and M. Pellicciari, "Engineering method and tool for the complete virtual commissioning of robotic cells," *Applied Sciences*, vol. 12, 2022.

[15] R. B. Hill, J. Delbos, S. Trebosc, J. Tsague, G. Feroldi, J. Martin, T. Master, and N. Lassabe, "Improving interoperability of virtual commissioning toolchains by using opc-ua-based technologies," in *2021 26th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*, 2021.

[16] H. Xu, D. Wu, Y. Zheng, Q. Yu, and Y. Zhao, "Augmented reality-based virtual-real fusion commissioning: a novel approach to production commissioning," *The International Journal of Advanced Manufacturing Technology*, Aug 2023.

[17] M. Dahl, A. Albo, J. Eriksson, J. Pettersson, and P. Falkman, "Virtual reality commissioning in production systems preparation," in *2017 22nd IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*, 2017.

[18] A. Albo and P. Falkman, "Integrating efficient virtual preparation and commissioning into traditional production system development work flow," 2022.

[19] P. Puntel-Schmidt and A. Fay, "Levels of detail and appropriate model types for virtual commissioning in manufacturing engineering," *IFAC-PapersOnLine*, vol. 48, no. 1, 2015.

[20] J. H. Lugo, A. Abassi, R. Balderas, and N. Lassabe, "Digitalization of aeronautic painting shop floors for improved commissioning activities," *Sensors & Transducers*, vol. 261, 6 2023. Received: 24 April 2023; Accepted: 23 June 2023; Published: 26 June 2023.

[21] G. Acosta-Zazueta, J. Alcaide-Marzal, and J. A. Diego-Más, "Generative design software and different approaches," in *Proceedings of DARCH 2021 1st International Conference on Architecture & Design*, (València, Spain), Universitat Politècnica de València, 2021.

[22] Dassault Systèmes, "Transform your business with 3dexperience platform." https://www.3ds.com/3dexperience, 2023. Accessed: 2024-02-08.

[23] "Totally integrated automation portal - automation software - siemens global website." `https://www.siemens.com/global/en/products/automation/industry-software/automation-software/tia-portal.html`, 2024. Accessed: 2024-02-08.

[24] D. R. Delgado Sobrino, R. Ružarovský, R. Holubek, and K. Velíšek, "Into the early steps of virtual commissioning in tecnomatix plant simulation using s7-plcsim advanced and step 7 tia portal," *MATEC Web of Conferences*, vol. 299, 2019.

[25] J. Bastian, C. Clauß, S. Wolf, and P. Schneider, "Master for co-simulation using fmi," *Proceedings of the 8th Modelica Conference*, 2011.

[26] A. Junghanns, T. Blochwitz, C. Bertsch, T. Sommer, K. Wernersson, A. Pillekeit, I. Zacharias, M. Blesken, P. R. Mai, K. Schuch, C. Schulze, C. Gomes, and M. Najafi, "The functional mock-up interface 3.0 - new features enabling new applications," in *Proceedings of the 14th International Modelica Conference*, (Linköping, Sweden), Modelica Association, 2021.

[27] "Controlbuild | catia - dassault systèmes." `https://www.3ds.com/products/catia/controlbuild`, 2023. Accessed: 2024-02-08.

[28] G. Barbieri, A. Bertuzzi, A. Capriotti, *et al.*, "A virtual commissioning based methodology to integrate digital twins into manufacturing systems," *Prod. Eng. Res. Devel.*, vol. 15, 2021.

[29] Dassault Systèmes, *How to Create a FMU that Allows Any Other Host to Read-/Write User's Variables of ControlBuild.* Dassault Systèmes, 2018. Tutorial.

[30] A. Bendjelloul, B. Mihoubi, M. Gaham, B. Bouzouia, and M. Moufid, "A framework for an effective virtual commissioning of agent-based cyber-physical production systems integrated into manufacturing facilities.," *Concurrent Engineering Research and Applications*, 2022.

A

Appendix 1

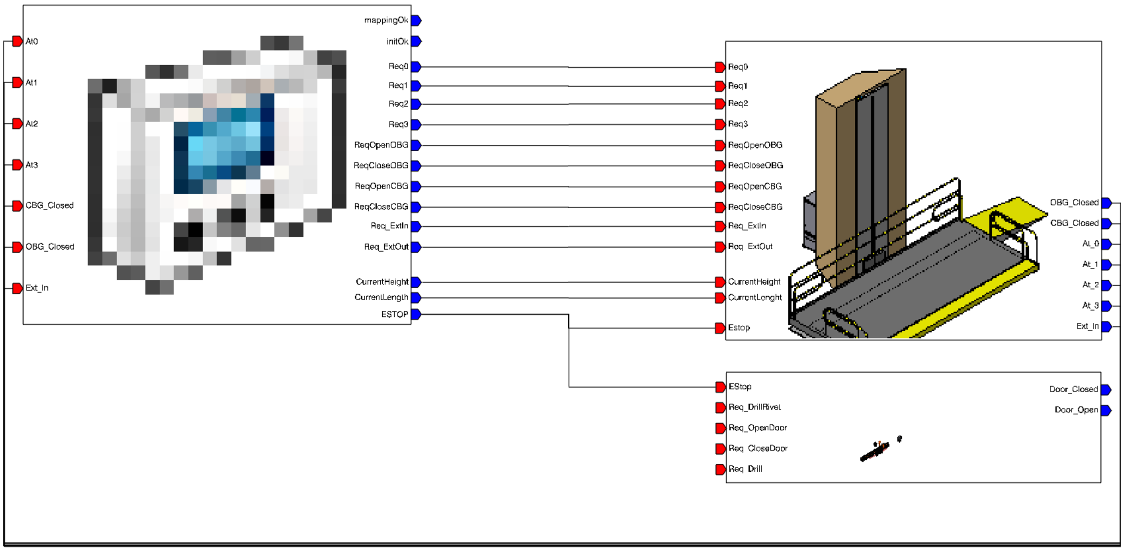


Figure A.1: IO mapping in 3DX

Figure A.2: VR demonstration of the manufacturing cell

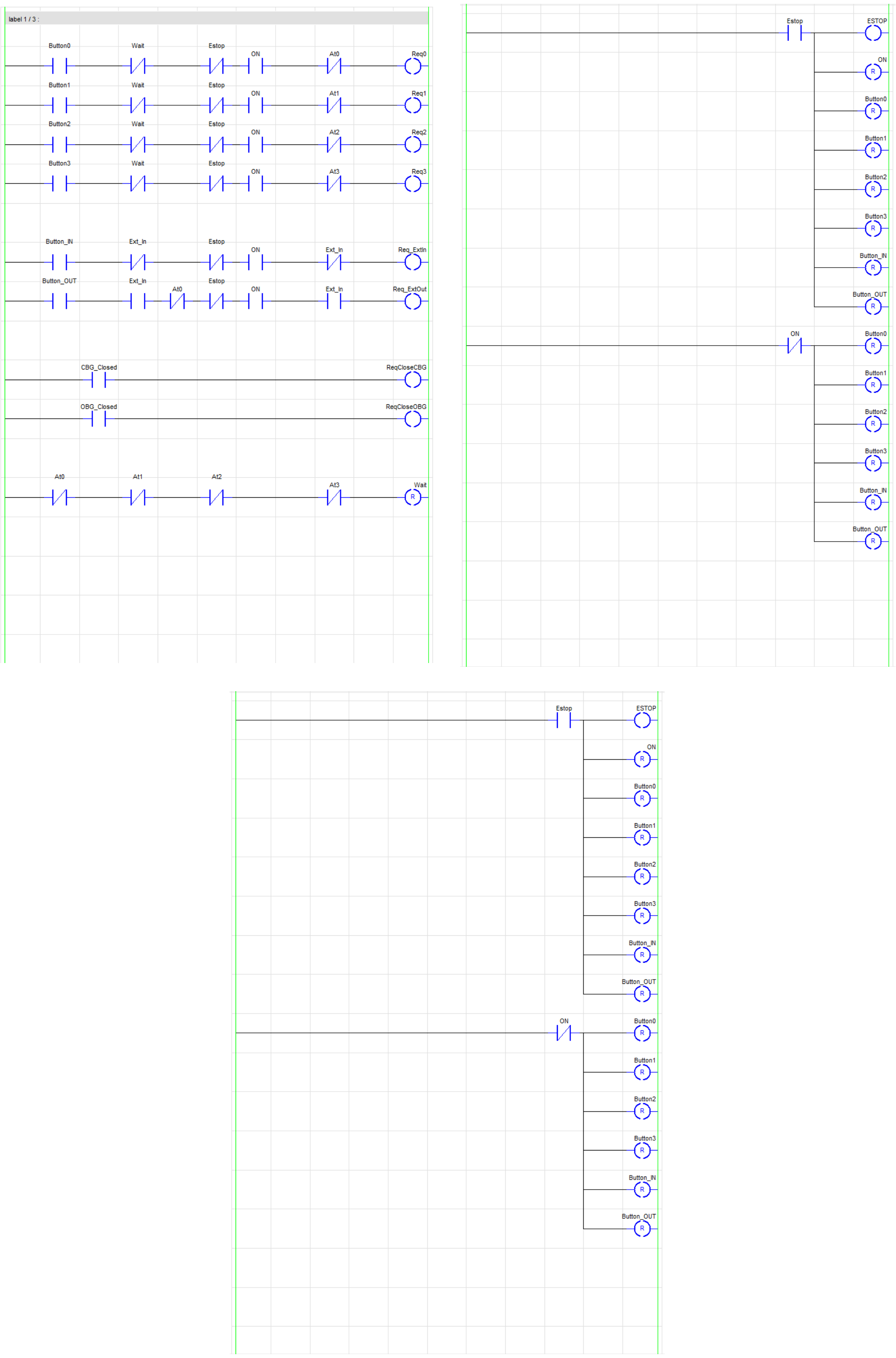


Figure A.3: Ladder code for automated staging

```
(*------------------REGULATOR--------------------*)
IF Manual AND NOT Estop THEN

(* Calculate error *)
ErrorHeight:= TargetHeight-CurrentHeight;
ErrorLength:= TargetLength-CurrentLength;

(* Calculate delta with maximum limit enforcement *)
DeltaHeight:=  ErrorHeight;
DeltaLength:=  ErrorLength;

(* Limit DeltaHeight to MaxDeltaHeight *)
IF DeltaHeight > MaxDeltaHeight THEN
    DeltaHeight := MaxDeltaHeight;
ELSIF DeltaHeight < -MaxDeltaHeight THEN
    DeltaHeight := -MaxDeltaHeight;
END_IF;

(* Limit DeltaLength to MaxDeltaLength *)
IF DeltaLength > MaxDeltaLength THEN
    DeltaLength := MaxDeltaLength;
ELSIF DeltaLength < -MaxDeltaLength THEN
    DeltaLength := -MaxDeltaLength;
END_IF;

(* Apply filter to smoothly adjust height *)
IF (DeltaHeight >= 0 AND DeltaHeight <= Tolerance) OR
 (DeltaHeight < 0 AND DeltaHeight > -Tolerance) THEN
    CurrentHeight := TargetHeight;
    AtHeigthTarget := true;
ELSE
    CurrentHeight := CurrentHeight + K*DeltaHeight;
    AtHeigthTarget := false;
END_IF;

IF (DeltaLength >= 0 AND DeltaLength <= Tolerance) OR
 (DeltaLength < 0 AND DeltaLength > -Tolerance) THEN
    CurrentLength := TargetLength;
    AtLengthTarget := true;
ELSE
    CurrentLength := CurrentLength + K*DeltaLength;
    AtLengthTarget := false;
END_IF;
END_IF;
```

```
(*----------------HMI FUNCTIONS-------------------*)
IF Reset THEN
TargetHeight:= 20;
TargetLength:= 0;
END_IF;

IF Estop THEN
Manual:= false;

END_IF;


(*---------------AUTO/MANUAL TRANSITION---------------*)
IF Req0 THEN
    CurrentHeight:= Level0;
    TargetHeight:= Level0;

    CurrentLength:= LengthIn;
    TargetLength:= LengthIn;
END_IF;

IF Req1 THEN
    CurrentHeight:= Level1;
    TargetHeight:= Level1;

    CurrentLength:= LengthIn;
    TargetLength:= LengthIn;
END_IF;

IF Req2 THEN
    CurrentHeight:= Level2;
    TargetHeight:= Level2;

    CurrentLength:= LengthIn;
    TargetLength:= LengthIn;
END_IF;

IF Req3 THEN
    CurrentHeight:= Level3;
    TargetHeight:= Level3;

    CurrentLength:= LengthIn;
    TargetLength:= LengthIn;
END_IF;
```

```
IF Req_ExtOut THEN
    CurrentLength:= LengthOut;
    TargetLength:= LengthOut;
END_IF;

IF Req_ExtIn THEN
    CurrentLength:= LengthIn;
    TargetLength:= LengthIn;
END_IF;
```

DEPARTMENT OF ELECTRICAL ENGINEERING
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden
www.chalmers.se

CHALMERS
UNIVERSITY OF TECHNOLOGY