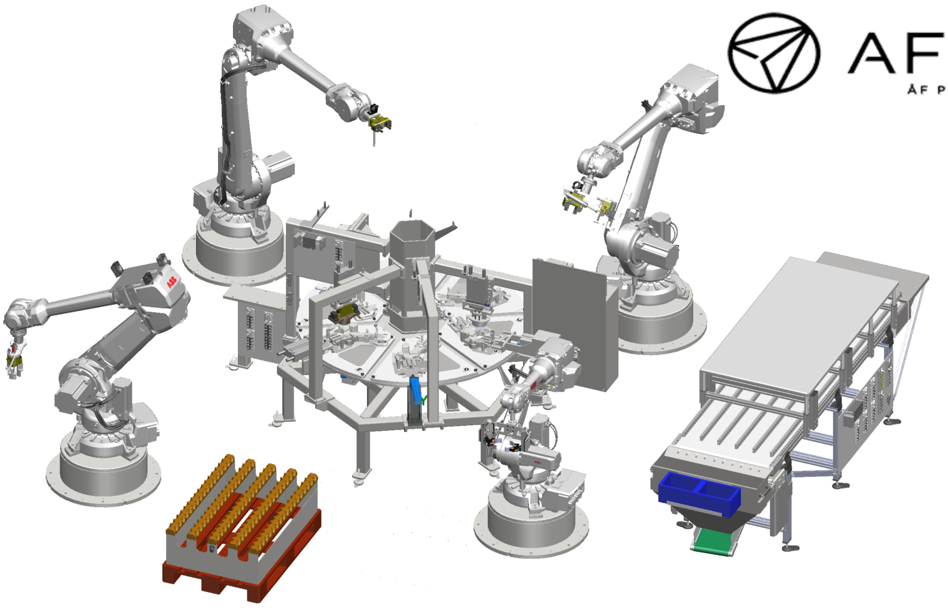


# Migration of 3D Simulation software in a Real Digital Twin framework

## Moving from ABB RobotStudio to Process Simulate and an exploratory study for possible AI integration

Master's thesis in Production Engineering

José Alberto Valdez Becerril
Harshith Kumar Prasanna

**Department of Electrical Engineering**
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2021

Master's thesis 2021:EENX-30

# Migration of 3D Simulation software in a Real Digital Twin framework

## Moving from ABB RobotStudio to Process Simulate and an exploratory study for possible AI integration

José Alberto Valdez Becerril
Harshith Kumar Prasanna

Department of Electrical Engineering
*Division of System and Control*
Chalmers University of Technology
Gothenburg, Sweden 2021

Migration of 3D Simulation software in a Real Digital Twin framework
Moving from ABB RobotStudio to Process Simulate and an exploratory study for possible AI integration
José Alberto Valdez Becerril
Harshith Kumar Prasanna

Supervisor:
Atle Zvantesson, AFRY
Anton Albo, Chalmers

Examiner:
Petter Falkman, Department of Electrical Engineering

Master's Thesis 2021:EENX-30
Department of Electrical Engineering
Division of System and Control
Chalmers University of Technology
SE-412 96 Gothenburg
Telephone +46 31772 1000

Cover: Picture showing the 3D manufacturing model built-in Siemens Process Simulate.

Typeset in LaTeX
Gothenburg, Sweden 2021

Migration of 3D Simulation software in a Real Digital Twin framework
Moving from ABB RobotStudio to Process Simulate with an exploratory study for possible AI integration
José Alberto Valdez Becerril
Harshith Kumar Prasanna
Department of Electrical Engineering
Chalmers University of Technology

# Abstract

New technological developments, digital tools, and their high integration capacity have made it possible to develop complex virtual models with high fidelity of real processes. Above is how the concept of the Digital Twin (DT) is shaping up more comprehensively. A DT is a virtual replica of a physical system mimicking all its behavior with the help of simulation and emulation systems. This concept is becoming more and more applicable and implemented across the many areas of the business but especially in automation manufacturing systems where companies realize the importance and beneficial outcomes of its capabilities.

This thesis will present a methodology for the migration and integration of a Virtual Manufacturing Cell from ABB RobotStudio (RS) platform to Siemens Process Simulate (PS) into an existing Real Digital Twin (RDT) model developed under the AFRY´s framework. The software's SIMIT, PLCSIM Advanced, and RS are the key enablers for emulated processes, and TIA Portal and WinCC for control and automation purposes. All these are then integrated with PS to form functional RDT. Another essential step is the integration of Tecnomatix Virtual Robot Controller (VRC) Server to connect with RS and mirror the controller behavior in the newly built PS cell. High fidelity and a real-time digital model are achieved at the end of the project. The test of the model functionality is carried by continuous interactions with the given Human Machine Interface (HMI) to perform a complete run of the process.

A comparison between both platforms based on the functionalities and actions aligned with the migration method is presented. In addition, the feasibility of implementing Artificial Intelligence (AI) into this RDT framework is also put forward based on an exploratory study.

Keywords: Process Simulate, Real Digital Twin, RobotStudio, Emulation, Virtual Commissioning, Industry 4.0, Migration, Artificial Intelligence, Virtual Robot Controller.

# Acknowledgments

With genuine pleasure and gratitude, we would like to thank all of those who were supportive and responsible for accomplishing this project, from the academia and industry side. Respectively, special thanks to all the following members:

**Andreas Buhlin** at AFRY for providing a great topic and facility to work.
**Atle Zvantesson** at AFRY for leading, helping, and supporting us to achieve our goals in this project.
**Sanna Älegård** at AFRY for her technical support and advice.
**Anton Albo** at Chalmers University for supervising and guiding us throughout the journey of the thesis, including providing the licenses.
**Per Nyqvist** at Chalmers University for providing us the simulation platforms with their licenses.
**Petter Falkman** at Chalmers University for being the examiner for this project.

José Alberto Valdez Becerril and Harshith Kumar Prasanna, Gothenburg, 2021

# List of Figures

# List of Tables

# Nomenclature

| | |
|---|---|
| *AI* | Artificial Intelligence |
| *API* | Application Programming Interface |
| *CAD* | Computer Aided Design |
| *DES* | Discrete Event Simulation |
| *DT* | Digital Twin |
| *HiL* | Hardware-in-the-Loop |
| *HMI* | Human Machine Interface |
| *I/O* | Input/Output |
| *IoT* | Internet of Things |
| *LB* | Logic Block |
| *LOD* | Level of Detail |
| *MiL* | Model-in-the-Loop |
| *ML* | Machine Learning |
| *OLP* | Off-line Programming |
| *OPC* | Open Platform Communication |
| *PiL* | Processor-in-the-Loop |
| *PLC* | Programmable Logic Controller |
| *PS* | Siemens Process Simulate |
| *RCS* | Robot Control Simulation |
| *RDT* | Real Digital Twin |
| *RFID* | Radio Frequency Identification |
| *RRS* | Realistic Robot Simulation |
| *RS* | ABB RobotStudio |
| *SCADA* | Supervisory Control and Data Acquisition |
| *SiL* | Software-in-the-Loop |
| *VC* | Virtual Commissioning |
| *VRC* | Virtual Robot Controller |
| I4.0 | Industry 4.0 |

# 1

# Introduction

Industry 4.0 (I4.0) is opening new horizons and creating significant challenges and a competitive environment for companies[1]. It demands them to become more innovative, socially responsible, technologically capable, and efficient to accomplish the market needs. The increase in the digitization of processes has made essential changes in the business model of companies, exploring new opportunities that benefit them in the initial stages like planning, construction, and commissioning[2]; but also in later stages as optimization, change in product and redesign. Digitalization, as well as applications and innovations that merge the real world with the virtual world, create competitive benefits that include speed, flexibility, quality, efficiency, all above keeping affordable cost[3].

I4.0 comprehends many supportive pillars like Simulation, Systems Integration, Internet of Things (IoT), Advanced Robotics, Augmented Reality, among others. The technologies facilitate the companies to gain customer attention and satisfaction in a short lead time[4], but they should be used in an integrated way. The isolated use of technologies within companies is not enough to remain competitive in the market; they must change and adapt as techniques evolve, but also develop innovation capabilities in the organizational and work context[5].

Services and consulting companies should also move at the same rhythm as technology, adopting and integrating innovative solutions to their actual processes. Artificial Intelligence (AI) and Digital Twin (DT) are one among the solutions which are forcing digital enterprise into Process Industries for achieving smart manufacturing[6]. There is a big opportunity to synergize the processes with all these technologies to accomplish efficient requirements of the industry and master the future challenges.

## 1.1 Background

AFRY, one of the largest consulting companies in Scandinavia, has recently created a new unit (AFRY-X) focused to innovate, develop and scale up digital solutions. AFRY has implemented Virtual Commissioning (VC) and Real Digital Twin (RDT) framework to develop many case scenarios using different platforms and software. They work together, among many software solutions, with Siemens´Digital Enterprise and Automation software portfolio.

Now, they are exploring a new portfolio of solutions and techniques to implement on its working methods and expand its capabilities. In this context, they want to foresee the possibility of moving to a different 3D Simulation platform and identify pros and cons in a multidimensional model of functionalities. For this thesis, the company provided an existing running cell made in the frame of RDT using ABB RobotStudio simulation platform, SIMIT, and TIA Portal.

Based on a white paper issued by Boerjesson et al.[7], AFRY´S RDT framework implements several Siemens digital tools (SIMIT, TIA Portal, PLCSIM Advanced, and NX MCD) to create emulated and simulated environments in the field of discrete manufacturing process. The architecture uses SIMIT as a central connection for hardware, both physically and virtually represented. This framework aims to emulate the physical process of a cell or station, so the TIA Portal project and setup approved for the DT can also be carried out for the physical implementation. But also, allow to work with a common DT during different stages of a production cell and allow early collaboration on the whole life cycle to detect issues, save time and money.

The main components of this framework are:

- Digital model is the 3D representation of the physical system with its properties and dynamical behavior.
- Simulation tool propagates the physical process digitally according to given inputs.
- Emulation tool places the model in a virtual environment by having a realistic behavior with physics rules, interfaces, and interaction.
- Agile Software for Automation to implement PLC code, test-based development, and for Continuous Integration & Continuous Deployment.
- AI application to identify dead ends or optimize OEE by proposing parameters efficiently.

## 1.2 Objectives

For the intention of achieving required outcomes for this thesis, the objectives are described as follows:

- Transfer a given digital production cell from ABB RobotStudio (RS) to Siemens Process Simulate (PS) maintaining the RDT essentials.
- Integrate PS into an existing RDT platform to verify and validate its capabilities.

- Compare, within the framework of the activities to be carried, PS versus RS
- Explore how Machine Learning (ML) methods can be adopted in RDT to improve process efficiency.

## 1.3 Research Question

The core focus of this project and its goals is based on different levels of accomplishments shown in Figure 1.1. The first one involves moving the cell from RS to PS, the second consists of software integration, and the third aims to explore how AI techniques could be implemented in the built model. This complete bottom-up approach is summarised with the main research question as follows:

**How to achieve RDT essentials by using PS as simulation platform in the context of AFRY's developed architecture for future ML integration?**

The follow-up sub-questions are formulated to get better supporting needs for this work.

- 1. What should be considered when moving RS platform to PS platform to achieve the existing RDT architecture.
- 2. What kind of issues should be anticipated when integrating RDT software?
- 3. What are the strengths and weaknesses of different simulation software (RS and PS) from the framework of a RDT development?
- 4. In which way can AI elements be integrated with RDT model to optimize or accelerate a production process?

## 1.4 Scope

This thesis will implement PS in a RDT of an existing functional cell created on RS. The given model contains four main components: (1) The PLC project in TIA Portal, including HMI developed in WinCC, (2) the emulation of different components in SIMIT, (3) the robot controllers and programs executed in RS, and (4) the virtual manufacturing cell in RS. The main work of this thesis focuses on the latter part. The idea is to analyze the behavior, restrictions, connections, variables, programming in the current platform and migrate it into PS for later integration with the rest of the programs.

Given the complexity of the actual RDT model, the automatic execution is left for future work. Manual execution of the whole process of cell operated by the HMI will be performed to validate the functionality of the resources, the integration of the different software, and signal communication. Minor modifications of current programs in SIMIT

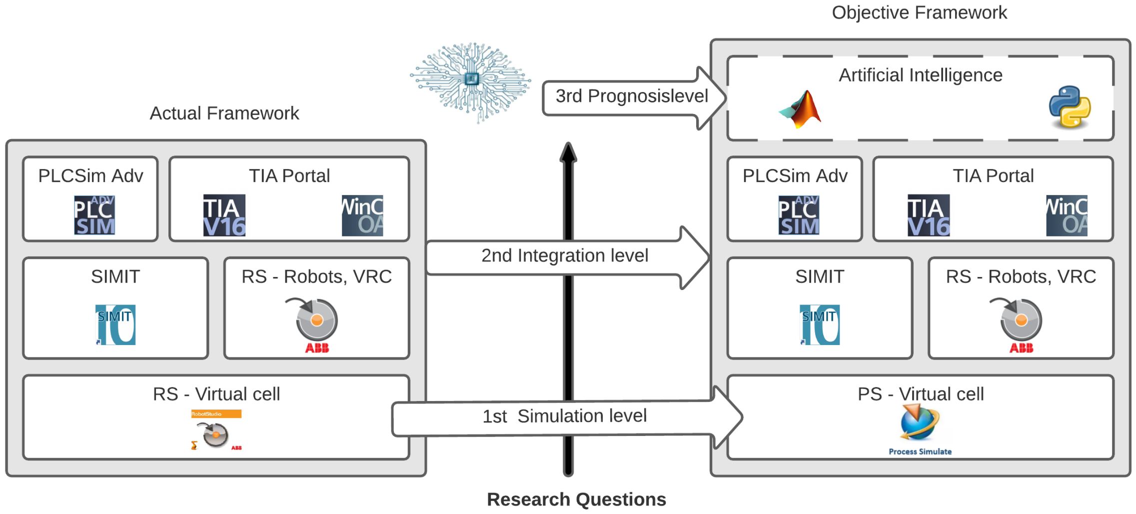


**Figure 1.1:** Research question model illustrating the level of accomplishment to be achieved within the thesis.

and RS will be made to fit and validate the new model features. There will be no alterations or changes in the PLC project and developed robot rapid codes.

PS Standalone version will be used, then the operation of other versions like PS on Teamcenter will not be addressed. As the digital cell will be tested in manual mode, the implementation of ML techniques for process optimization in this RDT framework will be restricted just to exploratory study.

## 1.5 Resources

The resources provided to develop the project are:

- Software licenses to develop the RDT
- Software installers
- Files of the existing RDT developed by AFRY: RS, TIA Portal, and SIMIT
- Computer with powerful technical specifications to run the actual RDT properly
- Developers and Engineers from AFRY

## 1.6 Confidentiality

The files, information, and data obtained from AFRY are confidential and undisclosed for any other sources. However, the results and development here created do not represent a risk to the company. The descriptions of processes, operations, and components will be depicted in a general way for illustrative purposes only.

## 1.7 Disposition

The structure of the thesis is presented as follows:

- Chapter 1. Introduction: It gives a general overview of the master thesis, presents the main topics to be addressed, and a brief background to better understand the context and field in which the project is developed. It shows the objectives, scope, and purposes intended to achieve.

- Chapter 2. Pre-Study: It presents a bench-marking of the other relevant developments and methods focused in the field of DT and AI driven DT models, including advantages and benefits in the different engineering perspectives.

- Chapter 3. Theory: It provides the theoretical information required to understand the framework of the report better. It presents topics from a general perspective to a specific one. Here is also included a brief description of the technological software tools implemented.

- Chapter 4. Methods: Describes the methodology to transfer the high fidelity simulation model from existing to a new platform, retaining the RDT essentials. It includes a more detailed description of the provided project.

- Chapter 5. Application Study: Presents the holistic view of the virtual model building based on the pre-defined methodology established in chapter 4. It also recapitulates issues and fixes considered to achieve the RDT framework.

- Chapter 6. Results: Summarizes the implemented PS model functionalities with RDT framework. It proposes the AI implementation in RDT based upon the exploratory study.

- Chapter 7. Discussion: Reflections on built model with its thought process and behavioral outcomes are stated here. It includes challenges that we came across are also put forward with suitable fixes. Finally, it suggests considerations for proposed AI implementation.

- Chapter 8. Conclusion and Future work: The contributions and conclusions are depicted based on our findings. This chapter is linked to the objectives and answer the different questions stated in Chapter 1. It presents some recommendations on what can be improved, developed, and implemented to better advantage this work.

# 2

# Pre-Study

*Different kinds of resource materials were studied relating our architecture framework (software's and their usage) to gain more knowledge about DT concerning simulation, emulation, and AI applications. The study is also extended with state-of-the-art techniques focusing the field mentioned above on knowing recent and current developments in research and industries.*

## 2.1 Literature study

The literature study was performed based on the related articles, books, research papers, conference papers, and web pages concerning DT in manufacturing and AI elements for prognosis of model behavior published on Chalmers University of Technology, research institutes, and scholarly articles. State-of-the-art highlights DT and AI-driven DT implementation work emerging in manufacturing automation sectors. Keywords considered are as follows: DT, manufacturing, simulation, emulation, robotics, big data, AI, and VC. To simplify and stipulate what information is needed in the literature study following questions were made up:

- What are the advantages of using DT in the context of manufacturing?
- What are the key enablers to form a DT?
- What is the purpose and importance of simulation in DT?
- How can DT benefit from AI applications?
- What kind of simulation software is present for DT architecture?
- What is the role of emulation software in the DT framework?

Literature study is devoted to outline and analyze different types of sources, mainly relating to this thesis work. Table 2.1 present the appraised set of papers, comprising all the main concepts involving in RDT framework.

Acknowledging the purpose of DT in different disciplines in the industry with its application and challenges is confirmed in [8]. A more detailed review of DT in the context of

**Table 2.1:** Studied material with description.

| **Title of article** | **Authors** | **Outcomes** | **Ref** |
|---|---|---|---|
| Digital Twin in Industry: State-of-the-Art. | Tao et al. (2019) | Review of DT with key enabling technologies and applications are presented. | [8] |
| Digital Twin in manufacturing: A categorical literature review and classification. | Kritzinger et al. (2018) | Described the DT concept with different level of integration based on the reviewed publications. | [9] |
| A virtual commissioning based methodology to integrate digital twins into manufacturing systems. | Barbieri et al. (2021) | A stepwise approach is demonstrated and validated for designed DT (Simulink) architecture, based on the virtual commissioning simulation methodology. | [10] |
| Digital Twin and Big Data Towards Smart Manufacturing and Industry 4.0: 360 Degree Comparison. | Qi & Tao (2018) | The combination of big data and DT are reviewed in the point of smart manufacturing. Thereby importance of responsive and predictive functionality and applications were illustrated. | [11] |
| Supporting the Design, Commissioning and Supervision of Smart Factory Components through their Digital Twin. | Martins et al. (2020) | Authors have built and tested a virtual model with ABB RobotStudio software and OPC UA feature. | [12] |
| A review on simulation in Digital Twin for aerospace, manufacturing, and robotics. | Phanden et al. (2021) | Role of simulation based DT in manufacturing, aerospace and robotics are presented with well know software platforms. | [13] |
| A Digital Twin to train deep reinforcement learning agent for smart manufacturing plants: Environment, interfaces, and intelligence | Kaishu et al. (2021) | DT model was implemented with Process simulate for which Deep Reinforcement Learning was constructed and trained with process simulate Application Programming Interface (API) feature. | [14] |
| Digital Twin: Applying emulation for machine reconditioning | Ayani et al. (2018) | Importance and benefits of emulation software (Simumatik 3D) for building DT is presented. | [15] |
| History and perspective of simulation in manufacturing | McGinnis & Rose (2017) | Discuss the issues, needs, and requirements in order to attain complex simulation models in manufacturing. | [16] |
| Digital Twin—The Simulation Aspect | Boschert & Roland (2016) | The role of simulation aspect of the DT over different lifecycle phases are discussed and highlighted. | [17] |

manufacturing is showed up in [9]. The importance of simulation and its aspects in DT are presented in [13][16][17]. The implementation of simulation-based model (DES) into DT architecture is demonstrated in [10][11]. Beneficial outcomes of the emulation model in DT are illustrated through the application study in [15]. The combination of Big data and DT in manufacturing including their application is discussed in [11]. AI-driven DT training implementation work is well exemplified with a case study in [14]. In summary, all the presented and reviewed articles are aligned with this thesis and will be used as the basis for proposed ideas (AI implementation).

## 2.2 State of the art

This section focuses on the cutting edge of technologies in the field of DT in manufacturing and application of AI. The study may comprise of different case scenarios of implementation of DT/VC and DT with AI applications both in terms of research and industrial side which are aligned to this project (eg. software). Table 2.2 summarizes the study work first three sources wholly highlighting DT implementation methodology and the rest five of them describe AI based DT integration.

**Table 2.2:** Related studies in State of the Art.

| No | Type of source | Title | Author | Ref |
|---|---|---|---|---|
| 1 | Journal article | A standardization approach to Virtual Commissioning strategies in complex production environments | Albo & Falkman | [18] |
| 2 | Journal article | A virtual commissioning based methodology to integrate digital twins into manufacturing systems | Barbieri et al. | [10] |
| 3 | Journal article | Proposed Virtual Commissioning of Robotic Cells based on the context of Industry 4.0 | Vitalli | [19] |
| 4 | Academic research | A Digital-Twin Assisted Fault Diagnosis Using Deep Transfer Learning | Xu et al. | [20] |
| 5 | Industry white papers | Value creation with plan modelling and simulation | Boerjesson et al. | [7] |
| 6 | Industry webpage report | Real-time AI powered by edge-deployed Digital Twins | Acharya & Mousavi | [21] |
| 7 | Industry webpage report | Commissioning a vision-based Supervised Learning solution | Mann et al. | [22] |
| 8 | Industry webpage report | Training a robotic task with Reinforcement Learning | Mann et al. | [23] |

**1)** The article presents a structured framework developed for implementation and integration of VC project, categorized based on the different levels of details with additional classes of functionality for each level. First level focuses on automation systems, the emulation of controllers, which is standard practice for line builders. The second level focuses on signal and communication protocols, defining signal properties and communication standard for performing SiL and HiL techniques. The third level focus on sensor, device, and actuator, to simulate the behaviour of different components or equipments. The fourth level focuses on resources modeling of systems, defining kinematics for different components in the system. Finally, the level five focuses on adding several connected systems to form a fully DT model.

**2)** The VC based methodology to integrate DT into an existing manufacturing system is proposed in this article for the development of "Digital Model" and "Pre-Digital Twin". The proposed methodology is validated by an implemented case study, where DT for production planning and control are developed into a flow shop line. The methodology includes a linear sequence of operation: framework, technology, DT, Intelligence layer, physical cyber interface, system modification, and VC. The same plan of sequence was also performed in the case study but more explicitly emphasizing each phase of execution practice. Finally, the author mentions the main benefits of this methodology as it provides virtual environment to simulate DT architecture and to detect viable issues that would happen in physical implementation. The drawbacks being only selected actors in DT architecture can be interfaced and simulated.

**3)**This article presents a methodology for virtual commissioning technology applied to a robotic cell, which is based on the different findings relating to calibrations of DT, VC system integration, and implementation work in the shop floor. This is mainly comprised of four steps; Step 1 mentioning about the collection of information based on the related work. Step 2 concentrating on technical development of works. Step 3 focusing on signal security protocols. Step 4 defining "absolute Zero" concept for calibrating the DT and VC and testing another type of validation trials.

**4)** In this article authors presented a two phase of DT assisted fault diagnosis method with the help of deep transfer learning to detect and make fault diagnosis from design phase to maintenance phase.

In the first phase, the intelligent development phase, the virtual model is constructed based on the manufacturing features, resources, and standards present in the physical space. This model was tested, validated, and optimized by changing the different parameters in simulation, which has not yet been occurred in the real world. In addition, Autoencoder (fault diagnosis model) embedded in the DT was trained with simulation data produced by virtual model. When the the virtual model achieved a satisfactory level, it was moved to a second phase.

In the second phase, virtual model was connected with a constructed physical entity, Digital Twin-assisted Fault Diagnosis using Deep transfer learning (DFDD) was implemented to transfer all the trained knowledge from the previous phase. This model was able to adapt rapidly to new working conditions and achieved protection and diagnosis of future issues.

Implementation work is presented as a case study in the car-body side production line as a physical entity, While the virtual model was achieved with Process Designer and PS. Process Visibility System was used for experimental data collection from PLC. The final accuracy results of DFDD were achieved to be 97 percent, and authors highlight the outcomes of the implemented model as it is sustainable, reliable, and efficient. On top of it, which also has reduced accidental breakdown issues greatly.

**5)** This article presents AFRY's RDT framework, done with Siemens digital enterprise software portfolio to achieve all the levels of integration in manufacturing set-up (factory, line, machine). This is enabled by the combination of SIMIT, S7 PLCSIM Advanced, NX MCD used based on the level complexity and domain specification values needed to be achieved.
RDT creates the emulated environment to test and validate the working of the model based on 100% true emulation signal coming from sensor and actor. This kind of test-based development with DT offers the ability to stress test the code, improving the functionality and quality, eliminating start up delays and ramp up issues, thereby faster time to market.
The beneficial outcomes were as follows:

- Faster time to market
- Reduces cycle time
- Reduced commissioning time
- Higher quality
- Limited downtime

**6)** This article emphasizes the cloud based DT's with different case scenario's along with its business advantages. Also, it answers the question of what capabilities can be achieved with the deployment of cloud based DT at the edge. While answering the question, the novel architecture of real time AI application was highlighted. Moving to edge make, the proposed architecture effectuates online-machine learning on streaming data making system self optimized and self tuned, attained by lower analytics latency and closed loop integration of analytics and local control. The presented business advantages are:

- Increased resilience
- Reduced cloud hosting costs
- Data preprocessing reduces the volume of data to be transmitted to the cloud

All this presented work was further illustrated by three use cases which mainly used python libraries and Google Tensorflow for training the various data sets. Raspberry pi as an edge computing device. They reinforce by saying the deployment of all these techniques are cost effective with meeting the optimum level of accuracy rate.

**7)** It presents a Supervised Learning case about how a DT of an equipment and product can ease actual industrial robotic methods inside the Process Simulate software. Nowadays industrial robots can make real time decisions to perform activities such as part detection, random part grasping, and assembly, all above by using cameras as inputs.

It shows an automated process where a robot, a vision system, and peripheral are integrated to detect and pick parts from a bin. RGBD (3D) camera, a new capability of PS software, is used to train, detect positions and dimensions of parts to machine learning algorithms training. An activity that in the traditional method requires manual assistance, too much time, stoppages in production lines, enough training examples, and may damage industrial equipment.

**8)** Endowing robots with human-like abilities is a frequently sought target in industrial robotics, but this implies a large amount of data, training, repetitive task, time, etc. Here is presented another case implementation in PS but now using Reinforcement Learning (RL). This method is based on learning from trial and error, and this often requires millions of attempts; translated into practical terms, it means damage of robots, equipment, and product, time, etc. Here is explained a case in which the goal is to make a virtual robot to insert a part through the help of 3D (RGBD) camera through RL training using Python.

## 2.3 Take away messages

By going through all of the above stated materials, some of the conclusions can be drawn and is mentioned as follows:

- DT is one of the key enablers for smart manufacturing. Definition of it remains in the way of approach and establishment. However, DT is still in its infancy.
- Before DT implementation, proper knowledge and scope of the system should be defined.
- Building high fidelity DT requires multi-dimensional level of competence and time consuming.
- Simulation and Emulation have a significant role in DT.
- Data is fundamental and plays an important role in the deployment of AI or ML.
- With successful integration of DT with AI, organizations can benefit from achieving favorable profit business margins.

# 3

# Theory

*This chapter presents essential information and insights about the main topics relevant for the understanding, implementation, and development phase in this project. It covers topics related to the RDT frameworks, including the software integrated into the project as well as AI elements.*

## 3.1 Industry 4.0 - Shaping Digital Twin technology

I4.0 has been the turning point for the creation and development of many technological concepts today; some of them have been implemented and redefined as technology advances. Among those, DT has been widely coined in recent years both in academic and industry side, providing synergy for different techniques and applications.

### 3.1.1 Concept of Industry 4.0

The fourth industrial revolution, also called Industry 4.0, is a term originated from the current advanced techniques and intelligent technologies that give shape to new concepts and developments. Formed primarily by the Industrial Internet of Things (IIoT), where closed-loop data modeling is used to enable autonomous workflow, decision-making, real-time monitoring, and much more to meet the customer demands of products or services throughout the life cycle chain. The main key paradigms that comprise it are[24]:

- Internet of Things (IoT)
- Cyber-Physical System (CPS)
- Simulation
- Autonomous Robots
- System Integration
- Augmented Reality
- Big Data and analytics
- Cyber-security
- Additive Manufacturing

The synergy of these technologies paved the way for the development of new techniques and methods, some such as digital technologies for the easy interconnection of intelligent components and the real-time monitor for the synchronization of activities in the real world with virtual space[25]. All of the above contributes, in different ways, to give a comprehensive and increasingly advanced shape to the concept of DT, which is also playing every time a more important role in I4.0.

### 3.1.2 Digital Twin - Current prospective

Digital Twin nowadays is a very complex topic as the term is applied every time in different areas and at different levels. A lack of consensus exists when defining the concept of DT based on the different standards, technologies, or procedures[6]. The concept and scope have been changing and evolving dramatically in recent years as capabilities and technological advances increase; likewise, both industry and academia define it differently[26]. However, its essence remains in the different approaches and definitions. According to Deloitte University Press it is defined *"as an evolving digital profile of the historical and current behavior of a physical object or process that helps optimize business performance. The DT is based on massive, cumulative, real-time, real-world data measurements across an array of dimensions"*[26].

Different categories of DT exist, some classify them as Product DT, Production DT, and Performance DT[27]. Table 3.1 shows different perspectives of DT based on definition, viewpoint, fidelity, and temporal integration[6]. In this thesis, the concept is narrowed according the application to implement: A DT of a discrete manufacturing production cell. In the frame of a production system, a DT is a virtual representation of the process to run different simulation disciplines that are distinguished by the synchronization between real and virtual systems thanks to the connected smart devices[25]. DT is more than a traditional modeling method or a PLC program; it is one step beyond simulation. It is digital reproduction of a product, process, machine or even a factory where you can transfer life real problems like mechanical and control issues to digital world so efficiency and productivity can be increased when the development process is rethought[28].

Production DT validates how well a manufacturing production cell will perform before its physical commissioning. Usually, the common resources implemented are robots, sensors, actuators, conveyors, tools, PLC programs, robot programs, and logic. Another DT definition with a more ambitious purpose is stated by Armstrong as *"a virtual representation of a physical object or system across its life-cycle. It uses real-time data and other sources to enable learning, reasoning, and dynamically re-calibrating for improved decision making"* [29]. According to a recent IoT implementation survey done by Gartner Inc, about 75% of companies implementing IoT already (13%) use DT or plan (62%) to do it within a year[30].

**Table 3.1:** Different perspectives of DT as presented in [6].

| | |
|---|---|
| **Definition** | Software representation<br>Virtual representation<br>Integrated Model<br>Reference mode<br>Digital Model<br>Cyber-physical system<br>Computerized counterpart<br>Digital Representation<br>Data & model |
| **Viewpoint** | Product(Machine)<br>Process(Line)<br>System(Factory) |
| **Fidelity** | Complete<br>Incomplete |
| **Temporal Integration** | Real-time<br>Offline |

### 3.1.3 Importance of Digital Twin: Advantages and business value

DT allows the companies to have a digital print of its product or process and allows them to detect and understand behaviors and problems in key stages of their life, thus enabling significant value to improve in its commercialization, operation, decision making, reduction of defects, incomes, among others[26]. The topic itself has a very positive impact in the sustainability field and will be favorable towards sustainability impact to minimize environmental issues. Further, these outcomes are addressed with the list of advantages of using a DT in a production system[6][31][32]:

**Faster**

- Real-time visualization, monitoring, and control of the products in working state even if they are far away
- Lower time to market and installation
- Reduce ramp up time in the production phase
- Time saving in automation engineering development work

**Better**

- Help the start-up of a real plant in an efficient and smooth way
- Used for predictive analytics to foresee the performance of the system
- Improves team synergy and collaboration
- Ordered transfer of engineering data

- Validation of the system model, automation functions before launch phase
- Managing the different level of interconnected systems
- Shortening or even negating future incurring problems
- Ensure higher engineering quality data

**Safer**

- Improves safety and efficiency
- Help to reduce compliance risk
- Reduced risk when used for training personnel
- No costly physical devices can be damaged

**Cheaper**

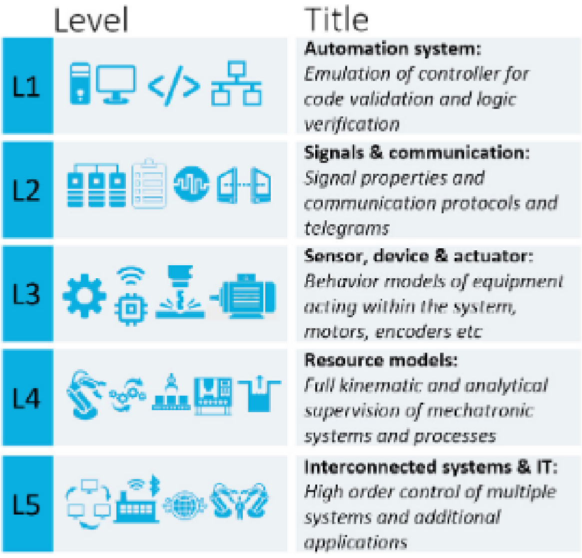
- Costly rework avoidance
- Avoid revenues loss produced by downtime
- Increases margin of profitability by a high extent

### 3.1.4 Virtual Commissioning versus Digital Twin

VC is the process of verifying and validating a system against a virtual model during or before the build of the entire system, providing advantages like reduction of commissioning time, development time, cost, and the stress of the real on-site commissioning [15]. As mentioned before, article [18] presents a structured framework developed for deploying the VC project based on the five different LOD. The elements of accomplishment will be considered in the present project work as a reference model in establishing a methodology to built PS model. Figure 3.1 showing the adopted framework. VC is used in the automation field to reveal and rectify faults; some of the domains where it is used are in automation of the process, manufacturing, building, energy, traffic, and intralogistics; the main goal activities are[33]:

- Identify control and sequence code errors
- Test user interface
- Test and validate the dynamic behavior of a plant
- Consider special scenarios
- Validate parameters
- Simulate critical states

One of the main differences is that DT provides a comprehensive models with more realistic and holistic measurements of unpredictability. Thanks to actual computer capabilities, it can be analyzed with advanced algorithms for real-time and offline[26]. According to Angarano (2020), DT is the synergy between simulation and emulation.



| Level | Title |
|---|---|
| L1 | **Automation system:**<br>*Emulation of controller for code validation and logic verification* |
| L2 | **Signals & communication:**<br>*Signal properties and communication protocols and telegrams* |
| L3 | **Sensor, device & actuator:**<br>*Behavior models of equipment acting within the system, motors, encoders etc* |
| L4 | **Resource models:**<br>*Full kinematic and analytical supervision of mechatronic systems and processes* |
| L5 | **Interconnected systems & IT:**<br>*High order control of multiple systems and additional applications* |

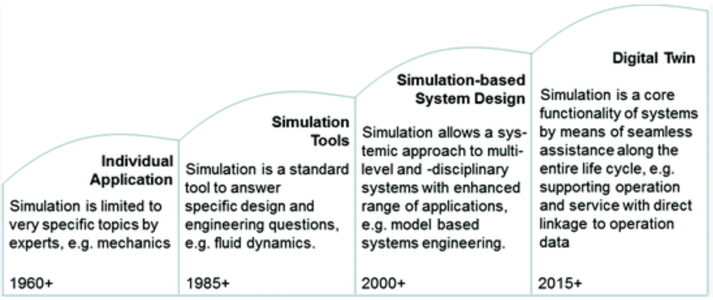
**Figure 3.1:** Hierarchical framework categorized on level of detail while deploying VC project adopted as in [18]

On one side, simulation allows performing several iterations where variables can be changed and respond faster than in real-time. On the other side, emulation allows the possibility to thoroughly test controls having a high detail level to get more accurate and precise runs in real-time. Emulation through DT is being embedded (timely manner) as a tool to investigate, test, validate machines and production lines in a 3D environment that uses operational logic connected to PLC thereby, saving the resources to be used[28]. Step wise approach by [10] as a part of DT implementation methodology with the defined case study, will be of our interest while outlining thesis methodology.

## 3.2 Simulation

Simulation is used in different domains of sectors (industrial) in order to mimic the real-world scenarios, to analyze and optimize the design, process, operation, among others[34][35]. The definition of simulation is expressed in numerous ways in the digital factory context. Shannon [36] defines it as *"the process of designing a model of a real system and conducting experiments with this model for the purpose of understanding the behavior of the system and/or evaluating various strategies for the operation of the system"*. Simulation is applicable when there is difficulty to work in physical space conditions, analyzing the problems in variable time scale and very complex to solve with mathematical model[37]. Reasons to adopt simulation are no need for the physical resource, to identify the bottlenecks and flaws in material flow and operation, diagnose problems, answers the "what if" scenarios and surpass the saving amount[36][38].

Simulation plays a significant role in the DT framework. Shao et al. acknowledge the importance of the relationship between the simulation and DT through different case instances[34]. In [17] the DT is demonstrated in the simulation point of view; further, the author emphasizes the waves of simulation in system development as shown in Figure 3.2.



Simulation is limited to very specific topics by experts, e.g. mechanics

Simulation is a standard tool to answer specific design and engineering questions, e.g. fluid dynamics.

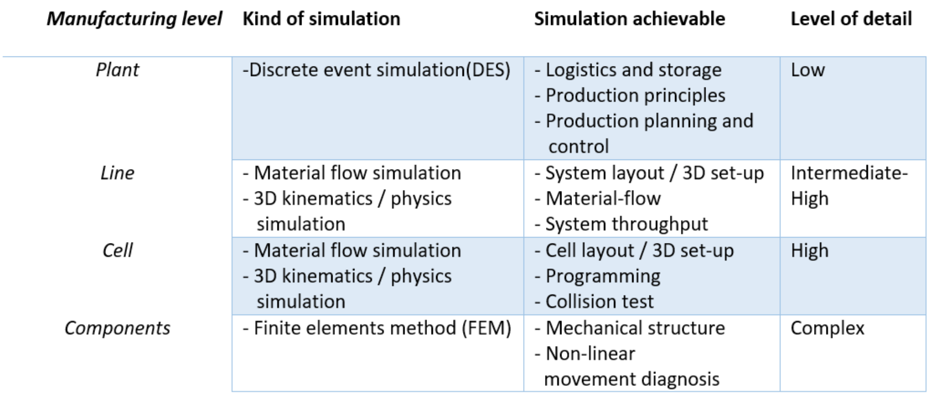
Simulation allows a systemic approach to multi-level and -disciplinary systems with enhanced range of applications, e.g. model based systems engineering.

Simulation is a core functionality of systems by means of seamless assistance along the entire life cycle, e.g. supporting operation and service with direct linkage to operation data

**Figure 3.2:** Waves of simulation technologies from 1960 to the present as in [17].

At the factory level, different kinds of simulation can be classified by the accomplishing LOD, which is presented in Table 3.2[35].

**Table 3.2:** Different kinds of simulation at different manufacturing levels modified as in [35].



| *Manufacturing level* | Kind of simulation | Simulation achievable | Level of detail |
|---|---|---|---|
| *Plant* | -Discrete event simulation(DES) | - Logistics and storage<br>- Production principles<br>- Production planning and control | Low |
| *Line* | - Material flow simulation<br>- 3D kinematics / physics simulation | - System layout / 3D set-up<br>- Material-flow<br>- System throughput | Intermediate-High |
| *Cell* | - Material flow simulation<br>- 3D kinematics / physics simulation | - Cell layout / 3D set-up<br>- Programming<br>- Collision test | High |
| *Components* | - Finite elements method (FEM) | - Mechanical structure<br>- Non-linear movement diagnosis | Complex |

### 3.2.1 Discrete Event Simulation

Discrete Event Simulation (DES) models sequential logical events based on the defined period. Events in the system occur at discrete points in time when the suitable event\signal is triggered in the system. Unless any changes in the triggering event\signal, the system will remain to be in the set instance. The system states are continuously updated based on the defined set of events and not on time between the events[39].

## 3.3 Emulation

Emulation software plays a key role in carrying out VC and DT of manufacturing systems by reducing time, adding flexibility, minimizing risk, and improving final quality in the results[15]. Emulation minimizes and bridges the gap between the real systems and the virtual model; it plays a crucial role in mimicking the behavior of the real control system. KB describes emulation as the imitation of a specific computer program or another platform in order to reproduce the same results as a real system; this is enabled by creating an extra layer between an existing computer and the platform to be mimicked[40]. Thus, an emulator creates models virtually in a realistic way, with physics rules, interactions, and interfaces that allow trustworthy feedback[7]. An emulated and simulated model differs in that the first one needs real control software to work, while the other does not[15].

According to many approaches, DT implies the need to have an interconnected physical environment to be considered a DT in its fullest extent of the word. In this situation, emulation plays an essential role in making this replacement of that physical environment.

## 3.4 Verification and validation of Digital Twin

Verification and validation play a significant role in checking whether the virtual model behaves and performs close to the real case scenario and accomplishes customer expectations. Different authors state the verification and validation context in various ways in their perspective. In [41] verification is defined as *"the model is akin to debugging—confirming that the model functions as the modeler intends"*, and validation as *"model confirms that the model is an accurate representation of the current or proposed system relative to all performance metrics to be assessed by management"*. graphically represented in Figure 3.3. The smaller the gap between the simulation model and the behavior system, the more accurate the model's performance can be considered. Hence, meeting the customer's satisfaction with a high-trust relationship.

Authors in [42] summarise the whole process of verification and validation as *"doing the right things right"*, where validation confirming *"doing the right things"* so that it fits for the end-users and verification confirming *"doing the things right"* followed by testing, analyzing and inspecting. IEEE accentuates verification and validation for the whole spectrum of systems, and its interfaces include software and hardware elements, determining the developed model bounds to the user-defined needs and intend[43].

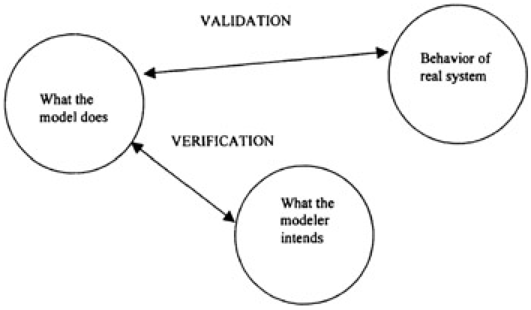


**Figure 3.3:** Relationship between model verification and validation as presented in [41].

### 3.4.1 Model-Based development

Among the many approaches for verification and validation, model-based systems engineering, also known as X in the loop, will be most likely applied in automotive industries in the field of electronic control units as it involves data-intensive signal-processing applications[35]. X in the loop includes closed-loop approaches at different abstraction level (as the SiL will be used in this thesis, emphasis on it here will be border):

- Model in the loop (MiL): System simulation is performed using the models of the system.
- Software in the loop (SiL): Performed by running different types of software on the required PC, it might work with both types of simulation scenarios: real plant and virtual controllers or virtual plant and virtual controllers. In the thesis work [35], SiL framework was considered and implemented with virtual controllers, emulated by S7-PLCSIM, SIMIT, IBH safety, S7 - Simulation PLC, and RS logic emulate 5000, and simulated plant. The author claims the controller code can be obtained from an automatic code generator or by regular programming to test and validate SiL framework.

  In [44], the authors have demonstrated the implementation of SiL architecture on the transportation system both in the low and high level of modeling of the complete system. They claim that more than 3000 Input/Output (I/O) signals were simulated with the help of COSIMIR Transport software. SiL architecture includes COSIMIR and S7 connected with the help of Open Platform Communication (OPC).

  The [45] presents the test case scenario of SiL for reconfiguration of a manufacturing system. The conclusion state that SiL techniques have a potential benefit for the manufacturer to deal with frequent changes in the manufacturing industry.

- Processor in the loop (PiL): It is used to test generated code (controller) on the target processor or simulator of a cell or plant, running in the offline simulation mode.
- Hardware in the loop (HiL): It is performed with real hardware control system carried in real-time. This kind of validation technique is not widely used in manufacturing automation systems[46]. Furthermore, they describe some of the shortcomings. It requires complete and accurate simulation models integrated with real components to accomplish high accuracy in simulation\validation, limited to small applications or system modeling, and requires customized setup for each project. On the other hand, the applicability of this technique in automation control are listed as follows:
  - Modular component dynamic testing
  - Real-time data transfer and handling
  - Efficient control algorithm verification

## 3.5 Industrial simulation and automation software solutions

To carry out a DT, software must be implemented according to the purpose and simulation level to achieve. As this project is based on Siemens platform, Figure 3.4 presents the different levels and approaches of simulations and the Siemens solution for each of them and highlights (yellow outline) those to use in this project.

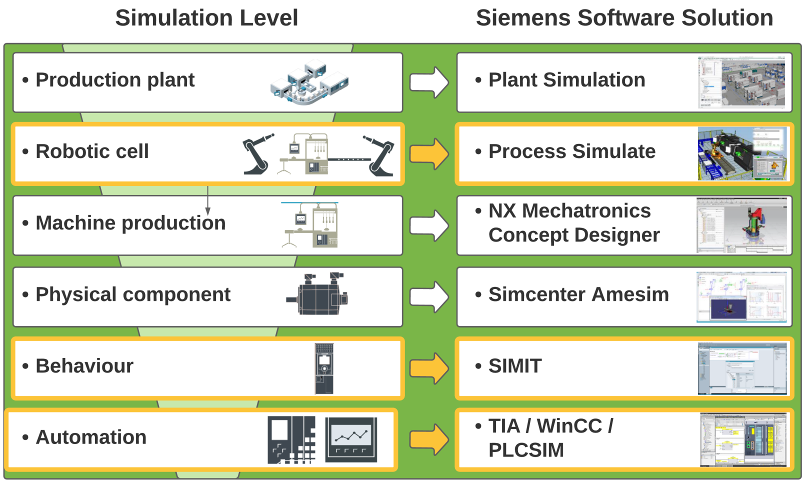


**Figure 3.4:** Siemens Software solution for each simulation level as adapted in [47].

### 3.5.1 Software connectivity

The connectivity between software is fundamental in the stages of integration in a DT project. To do so, 3D simulations and automation software provide several standard connections to third-party software and hardware like OPC DA, OPC UA, and others specific to the brand.

OPC DA is the most basic protocol of the OPC stack. It utilizes a group of client-server standards that allow real-time data (not historical data or events) communication between data acquisition devices such as PLC to interface devices like HMI, SCADA systems, or ERP/MES systems[48].

OPC UA is a standard platform-independent service-oriented, and is the successor of OPC Classic; it integrates all its functionalities but also accomplishes different design specification like mapping of all COM OPC Classic to UA, firewall-friendly while addressing security concerns, independent platform, extensible with the ability to add features without affecting existing applications, and comprehensive information modeling[49].

Shared Memory is an efficient way to communicate via the main memory, which is shared by all processors allowing to access any part of memory in parallel; it has two main advantages: simplicity and load balancing; but three problems: cost, limited extensibility, and low availability[50].

### 3.5.2 3D Simulation software

For the DT implementation of a robotic manufacturing cell process, Robotic 3D simulation software must be implemented. Nowadays, there are many with different capabilities like Process Simulate, RobotStudio, CIROS Studio, Virtuos, DELMIA 3D experience, and KukaSim. Based on the project context, not all the solutions will be addressed but only the first two.

#### 3.5.2.1 ABB RobotStudio

RS is one of the most commonly used offline programming tools, specifically for ABB robots. Inbuilt, ABB virtual controllers (emulates the real controller) is a replica of the software that runs the robots physically; it provides a more realistic and efficient simulation platform for performing different automation techniques such as DT, VC, Virtual Meetings, Augmented Reality, and much more[51]. The above is achieved thanks to its advanced modeling and simulation tools such as visualization of multi-robot control, safety features, 3D vision, and physics[51]. On the side of VC, it allows the connection to PLC and other external devices to fully validate and verify the logic and safety of the cell before physical commissioning[51]. Furthermore, this software also provides the virtual floor to train the operators and optimize the process flow without interruption in the production.

*ABB controllers* RobotWare is the controller software designed to achieve cost-efficient process performance with the unique advanced feature included in it[52]. RS offers a wide range of possibilities to install respective RobotWare based on the operational requirements needed to achieve.

#### 3.5.2.2 Process Simulate

PS is a specialized digital manufacturing solution from Siemens Tecnomatix portfolio focused on the design, creation, analysis, and optimization of 3D simulation and virtual environments. Among the main features of this software are resource modeling, workstations design, human task simulation, discrete/continuous process simulation, robotics process simulation, and VC[53]. It is a suitable tool to perform DT and VC projects. Among the capabilities to this area are: test/simulate PLC code, model control resources, create different manufacturing applications, plan material flow, develop Off-line Programming (OLP) for different robot´s brand, simulate internal logic resources, connection to VRC servers, and integrated simulation through external connections.

PS has two basic approaches to simulate the sequence of operations:

- Time-Based simulation: this mode simulates a single production cycle which means it is confined to one pre-defined operation within a start and a finish.
- Event-Based simulation: this mode simulates the production stations, which includes distinct type parts, resources, robots, etc., achieved through a dynamic sequence-based events and triggers as defined in Cyclic Event Evaluator (CEE) or PLC. However, this type of approach will accurately reflect the real manufacturing cell and provides space to fine-tune the mechanical, electrical properties of the manufacturing cell[54]. The unique level of capabilities in this mode enables the users to test, validate and synchronize the cell operation in real-time and foresee future automation problems[54].

### 3.5.3 Virtual Robot Interfaces

Another essential topic to introduce in order to achieve the desired integration is Virtual Robot Interfaces. Robot Controller Simulation (RCS) is an interface that allows a standard integration of robot controllers so the motion software of any robot can be integrated into any simulation system[55]. Virtual Robot Controller (VRC) Interface is a flexible coupling that enables the integration of robot's controller software, including the operating system, into simulation software, through asynchronous calls for the exchange of commands, responses, and events[56]. Tecnomatix Process Simulate has many solutions today to connect through VRC to several robot brands like ABB, Fanuc, and Kuka. When performing VC to ABB robots, PS sends and receives through VRC data from and to RS; information that includes signals values, joint values[54]. Software requirements are presented in section 5.9.

### 3.5.4 Agile software for automation

This section discusses the capabilities of automation software's which will be used in this thesis work.

#### 3.5.4.1 SIMIT SP

SIMIT SP allows real-time simulation on a single platform to perform comprehensive automation tests, VC (either systems, machines, or processes), and even realistic training environments. Some of the characteristics and benefits in engineering and commissioning are: provides good integration, openness, flexibility, improved operability, simulation, and testing of the automation functions.

SIMIT is a test-bed for the DT; among the different capabilities related to this purpose are emulation of controllers, simulation of technological behavior and equipment, signals, drives, and sensors[57]. It also can connect to third-party simulators through generic data couplings like OPC DA, OPC UA, Shared Memory, Software-in-the-Loop Couplings with SIMIT Virtual Controllers, PLCSIM Advanced, and PLCSIM; or Co-Simulations Coupling with gPROMS and MCD. From V10.2, a direct connection to PS is possible.

#### 3.5.4.2 TIA Portal

Totally Integrated Automation Portal from Siemens, available as software and hardware, provides a complete level of automation solution with seamless information flow both horizontally and vertically. It provides a way to optimize the engineering process and makes the integration of future technologies much easier[58]. TIA Portal includes, among many others, the following integrated software:

- SIMATIC STEP 7: used to configure and program the SIMATIC controllers with provided programming languages: Ladder Diagram (LAD), Function Block Diagram (FBD), Statement List (STL).
- WinCC: used to design HMI with the help of supervisory control and data acquisition (SCADA), enabling monitoring and controlling process data.

#### 3.5.4.3 PLCSIM Advanced

A PLC is a central component of a production cell. When working in a digital environment, it is necessary to replace it with a software solution; here is where PLCSIM Advanced plays an important role. PLCSIM Advanced is a simulation system (virtual controller) where CPU program code can be loaded and simulated, so there is no need for a real PLC, and the code changes can be tested and verified before code is loaded in a real one[59]. Some of the application areas are SiL for VC and combination with third-party software to simulate a production machine, plants, and simulation of automation and mechanics.

## 3.6 Elements of Artificial Intelligence

The availability of data from the IoT devices built from smart components (sensors, actuators, motors, etc.) in the factory opened a new field of study called big data. In order to analyze and act accordingly, Artificial Intelligence (AI) is at the top priority, which works similar to human cognitive skills: learning, reasoning, and self-correction for the big data processing[60]. The automation coupled with elements of AI referred to as Intelligent process automation is a powerful tool to achieve efficiency, increased speed, and time-savings when applied to robotic processes[61]. Among the different approaches in AI article [62] describes the relation between AI, Machine learning (ML), and Deep Learning (DL) as one of the widely used AI approaches. Figure 3.5 shows the AI encapsulation diagram.

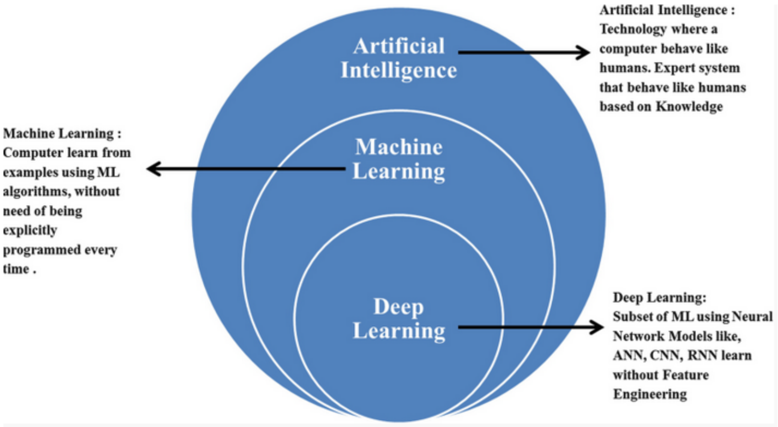


**Figure 3.5:** Relations between AI, ML, and DL as presented in [63].

Further, different ML can be carried with [64]:

- Supervised learning: training the model based on labeled output train data sets for classification or predictions. Example models are decision trees, support vector machines, naive Bayes classifiers, regression, and random forests.
- Unsupervised learning: training the model with unlabeled data sets for clustering or grouping. Example models are K-means and hierarchical clustering.
- Reinforced learning: training the model with data records, no labels on it, provid-

ing feedback to the AI system (reward basis). Example models are Monte Carlo learning, Q learning, and Deep Q learning.

### 3.6.1 AI in controller

A controller is a device used to manage all the input and output signals of manufacturing equipment based on the logic coded (hardcoded) behind it. There are three types of controller used in industrial automation: Programmable control logic, Distributed controlled systems, Programmable automation controller. In order to reduce the programming and engineering efforts needed to build automation solutions with much agile logic and precision, AI enables controllers will meet future challenges[65]. TIA Portal has introduced the AI in SIMATIC S7-1500 TM NPU module, a combination of AI algorithm and PLC logic, which can optimize the process in multi-dimension throughout the product flow[65].

### 3.6.2 Role of ML in robotics

The role of robots is increasing in modern industry with much flexibility, safety, human and robot collaborative workspace to achieve the predefined tasks in an intelligent way[63]. In context of Industry 4.0, the field of Robotics is advancing every rapidly to meet changing customer demands in the market[63]. The authors in [66] demonstrated the implementation of a Hybrid machine learning framework that includes supervised and reinforcement learning with a physics-based simulation model. They have successfully illustrated the transfer of simulation to real-world techniques with the modular structure for easy implementation in everyday production tasks.

### 3.6.3 Fault prognosis

Unplanned machine downtime in manufacturing can disturb the complete workflow of the factory; this shifts time to meet the expected throughput. Based on the collected historical data (health status) from the resources, the organization can benefit by building an ML-based data-driven model on specific resources to predict and diagnose before failure. However, changing the different scenarios based on changing customer demands becomes costly and takes more time and resources. DT with high fidelity provides the optimum possible outcome to benefit reliable data to form a data-driven DT model. The authors in [67] state the importance of DT for the prediction of aircraft structural life. [68] strengthens DT for prognosis by stating DT is capable of anomaly detection, giving prior warnings, prediction, and optimization.

Recent technological advances are helping to transfer skills learned in simulated environments to real ones; DT is an emerging approach to train robots virtually and accelerates time to market and reducing cost, manual intervention, and risk of equipment damage[23].

# 4

# Method

*This Chapter describes the given manufacturing cell in order to understand the initial conditions and the starting point of the project. Then is presented the methodology and steps proposed to migrate the PS model in the RDT architecture.*

## 4.1 Structure of the original RDT model

One of the main goals of this project consists of lifting the pre-build RDT from RS to PS platform using the same base of SIMIT and TIA Portal. Before moving on it is necessary to understand all the features of the existing platform.

Figure 4.1 provides a general overview of the initial status represented in blue while the main activities to develop are in green; this implies several activities and challenges to solve the objectives set.

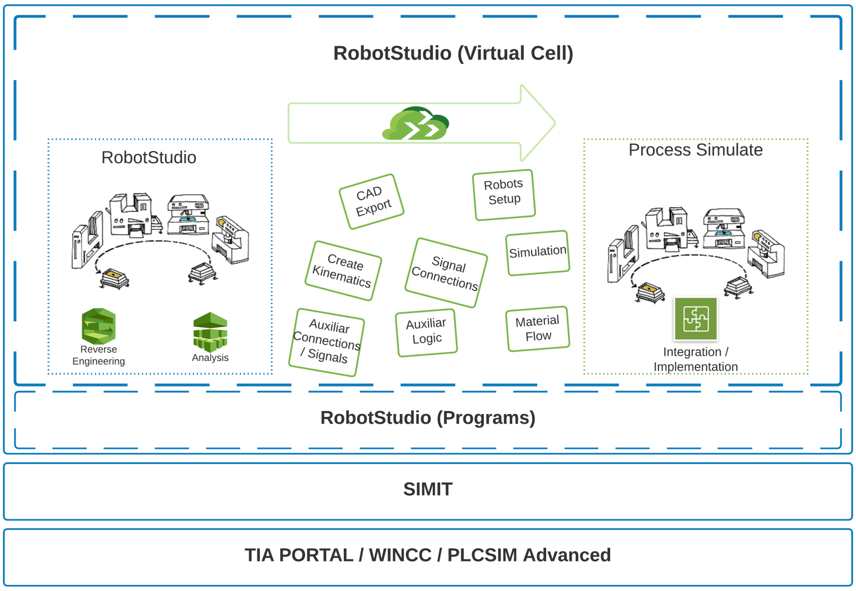


**Figure 4.1:** Main activities to perform within the RDT framework.

### 4.1.1 Cell control

The given RDT model contains integration among RS, SIMIT, TIA Portal, and PLCSIM Advanced. Here is presented a general description of each program and its role.

SIMIT

The project at SIMIT has direct communication to RS through the coupling with the four robots of the cell and another one for the I/O of the virtual cell. It includes another communication for sending information to the program loaded in PLCSIM Advanced. It has charts for different purposes: control and connection to different stations, connection with peripheral systems, control of safety systems, HMI, and emulation of RFID systems. Figure 4.2 show a general picture of the project developed in SIMIT

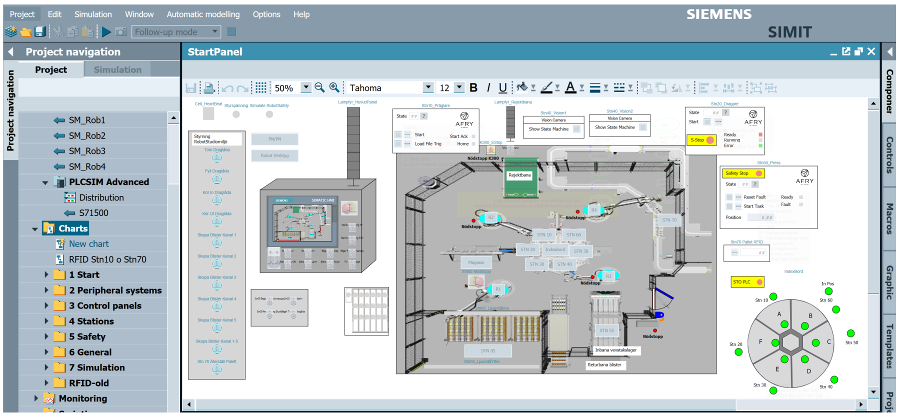


**Figure 4.2:** Original project developed in SIMIT.

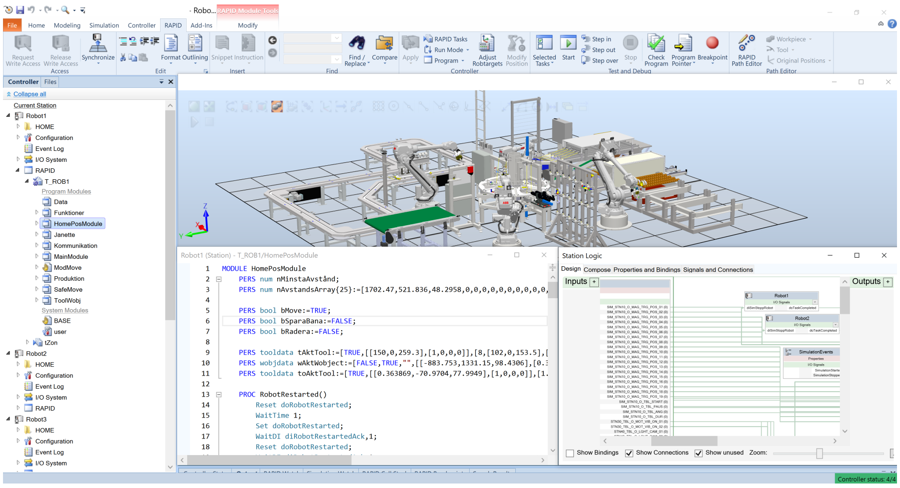
TIA Portal - WinCC

The file contains all the settings and programming ready to be executed. The project is developed in a security PLC SIMATIC S7 1500; it contains the configuration of components, program blocks, an HMI for a Comfort panel, as well as another mobile SIMATIC HMI (not explored in this project).

ABB RobotStudio

The given project is developed in RS where the robot programs and cell simulations take place. All types of equipment were logically modeled using smart component blocks. All the Inputs and Outputs signals form the basic need to handle the different kinds of tasks in the cell when connected with SIMIT. The robot programs are done with many I/O signals coming from the SIMIT as well as PLC. All the robot functionalities are programmed with safe move capability and monitored with ABB virtual controllers.

Figure 4.3 shows the project implemented in RS platform.



Robot1 (Station) - T_ROB1/HomePosModule

```
MODULE HomePosModule
    PERS num nMinstaAvstånd;
    PERS num nAvstandsArray{25}:=[1702.47,521.836,48.2958,0,0,0,0,0,0,0,0,0,0,

    PERS bool bMove:=TRUE;
    PERS bool bSparaBana:=FALSE;
    PERS bool bRadera:=FALSE;

    PERS tooldata tAktTool:=[TRUE,[[150,0,259.3],[1,0,0,0]],[8,[102,0,153.5],[
    PERS wobjdata wAktWobject:=[FALSE,TRUE,"",[[-883.753,1331.15,98.4306],[0.3
    PERS tooldata toAktTool:=[TRUE,[[0.363869,-70.9704,77.9949],[1,0,0,0]],[1.

    PROC RobotRestarted()
        Reset doRobotRestarted;
        WaitTime 1;
        Set doRobotRestarted;
        WaitDI diRobotRestartedAck,1;
        Reset doRobotRestarted;
```

**Figure 4.3:** Original project developed in ABB RobotStudio.

Connections

The connection between the RS and SIMIT is through shared memory coupling. A PLCSIM coupling in SIMIT is set to achieve the communication to PLCSIM Advanced. Also, a proper ethernet connection should be established between TIA Portal and PLC-SIM Advance to monitor the code and run the HMI. Figure 5.3 shows in blue lines the connections of the given project.

### 4.1.2 Virtual manufacturing cell description and layout

The main elements of the digital cell include:

- ABB robots IRB4600 and IRB2600
- Custom robotic tools fixed to each robot
- Process stations with tools, clamps, fixtures, and devices
- Conveyors for material input, output, and rejection
- Index table with six positions
- Magazine for parts
- Assembly parts
- Photoelectric sensors

The position of the index table, magazine, robots, conveyors, and other equipment was well defined in RS. All the positioning were made with respect to the world coordinate system. The layout of the virtual manufacturing cell is represented in Figure 4.4. This guides the blueprint for positioning each resource accordingly when building the PS model.

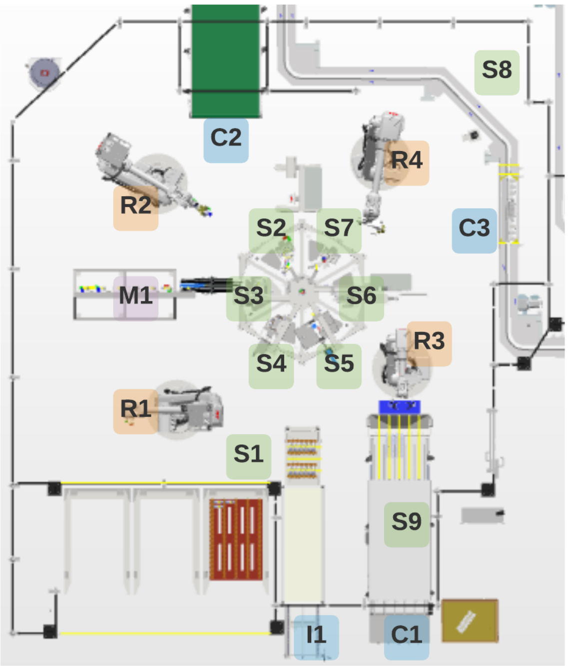


**Figure 4.4:** Layout of the manufacturing cell - top view.

There are nine stations (S1-S9) along with the whole-cell; here is a brief description of each of the and other main components:

- S1 contains an in-feeder conveyor (I1) and three buffers to store incoming products. I1 feeds the cell with pre-assembled products, a robot (R1) moves the product to a magazine (M1).
- Robot 2 (R2) moves the product into the index table or Conveyor 2 (C2) depending on the status of the product. The parts which are found to be fault are moved out through the rejection conveyor C2 to debug it or scrap, and the good ones are moved into S2.
- Index table constitutes a complex part of the manufacturing cell. It consists of six stations from S2 to S7. These stations consist of equipment to monitor the parts handling and quality in the process flow, complex equipment comprising of sensors to identify the parts, RFID for positioning of the index table, actuators, tools, fixtures, servo motors, image sensors, etc.

- S9 contains a linear conveyor (C1) that has five lines to transport pallets with products; Robot 3 (R3) places those parts into S6 on the index table. C1 has mounted many intermediate and final stoppers with sensors to provide materials when there is a need and to eliminate piling up situations along the flow line.
- S8 has a flex conveyor (C3) that moves the finished parts in a special pallet to pass the next cell.

Conveyors are integrated with sensors along the flow line to detect the parts and the different positioning of instances. All the robot tools are different and custom-designed based on the purpose of the task to be handled. Three tools fall into the gripper category with two jaws, three jaws and slider-crank properties to hold and manipulate the parts based on the required station operation. Another robot tool is equipped with two vacuum systems to hold small components.

## 4.2 Method for the migration and building of the model

The methodology presented in Figure 4.5 was followed to approach this project; it is inspired by those ones presented in the literature study. It is important to remark that this method is structured based on the platform already given, and it addresses only one section of the whole process.

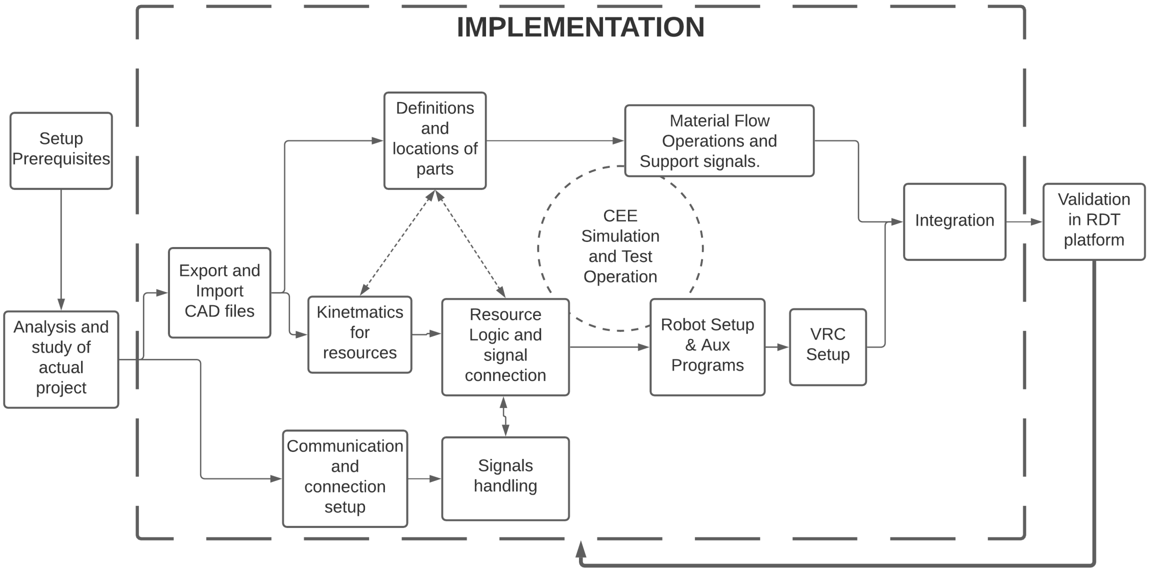


**Figure 4.5:** Methodology for the 3D simulation platform migration.

### 4.2.1 Setup pre-requisites

Before starting, it is necessary to perform several steps to set up the technological resources to run the given RDT. This implies to install:

- Installation of software with compatible versions
    - SIMATIC STEP 7 Professional in TIA Portal, V 16
    - SIMATIC WinCC Runtime Advanced in TIA Portal, V 16
    - SIMATIC S7-PLCSIM Advanced, V 3.0
    - SIMIT SP, V 10.2
    - RS, V 20
    - PS, V 16.1
    - Tecnomatix VRC Server ABB Real Time V 1.0
- Installation of updates and drives in TIA Portal
- Installation of Computer Aided Design (CAD) translators and redistributables packages for PS
- Adequacy of files (migration to new software versions and files retrieve)
- Connection and execution of all the software concurrently

### 4.2.2 Analysis and study of the actual model

In the way to attain the required Level Of Detail (LOD) on the PS, comprehensive analysis of the behavioral model in RS platform but also in the understanding and communications between the several software is required. This implies different activities and interpretations:

**RobotStudio**

- Understanding cell operation
- Kinematic analysis of tools, conveyors, actuators, and fixtures
- Sensor identification and functionality
- Robots configuration and functioning
- Analysis of Logic and expression of smart components
- Material flow understanding
- Signals mapping analysis between RS and SIMIT

**SIMIT**

- General identification and recognition of charts
- I/O mappings on different charts

- Cross-connection (coupling) to RS, TIA Portal
- General understanding of buttons and signals
- Resource behavior logic

**TIA Portal**

- General Understanding of Hardware Configuration
- General understanding of PLC code
- Communication and download to PLCSim Advanced

**WinCC Runtime Advanced**

- HMI execution
- HMI manipulation

**PLCSim Advanced**

- Setup a connections
- Make successful downloads

### 4.2.3 Export-Import CAD files

CAD models are the essential requirement to build the simulation model. Different features should be considered when exporting the CAD models: Rendering, file size, meshing quality, and also wireframe data. PS has some basic and limited CAD tools, then it is often hard to select and create key points in complex geometries (to define joint movements, TCPs, targets for gripping positions) or complex lines for conveyor or robot paths. Hence, it is desirable to define those in the original software before export.

The supported files for exporting CAD models in RS are VRML, ACIS, IGES, STL, COLLADA, DXF, SVG, among others. It is highly recommended to export the full cell in a single file, so locations of resources keep the same as in the original one. Later it is easier to split the file into the necessary resources. On the other hand, PS provides different file extensions to import CAD files; these are JT, NX, CATIA, ProE, STEP, IGES, and DXF.

Resource Definition

Once the complete cell is imported, it is necessary to create the different resources according to the simulation purpose. In RS exists only four types of resource definition (robot, tool, external axis, and devices), while in PS there are more options (clamp, container, conveyor, device, gripper, gun, robot, among others). It is important to give the proper selection, so it works as desired. Some CAD models do not play an important role in the simulation (they never take part in process flow and have not defined in the SIMIT); those can be grouped in a single file; thus, the model can be simple. Figure 4.6 depicts the general procedure of this stage. Blue rectangles represent the original file(s) imported with its own subassembly structure; these can be deleted at a later moment once the new structure (green rectangles) and files (green squares) are defined in the object tree in PS. So far, defining the resources in their appropriate category is sufficient; the logic and behavior will be described in section 4.2.9.

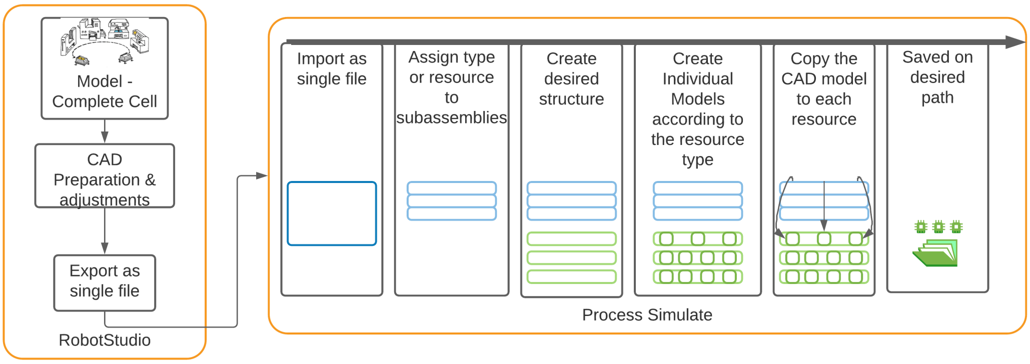


**Figure 4.6:** Export-Import CAD process between ABB RobotStudio and Process Simulate.

### 4.2.4 Definitions and location of parts

Parts are the pieces that conform to the manufactured product; they are manipulated by the tools, fixtures, devices, containers, conveyors, among others. They play an important role in the process and they should be set up adequately. They should be created and grouped according to the process, operations, and interactions with the tools and fixtures. The individual creation of parts is important; Although they are part of an assembly, their interaction in the processes is individual since some sensors and actuators must detect and manipulate only specific parts and not the assembly as a whole.

### 4.2.5 Kinematics for resources

Mechanism and kinematics are the fundamentals to define movements and properties to different tools and resources in the cell. The kinematics is established by maintaining the same structure of linkage between the links, joints configuration (prismatic and revolute), displacement, and dependencies as in RS. However, to attain the required functionalities as RS model, some resources needs an extra setup to perform the desired function like pose edition, tool definition, and set of gripped object list.

### 4.2.6 Communication and connection setup

When a project in RS is already done, it has the coupling with all the signals connected in the charts for different purposes. The most efficient way is just to replace the coupling in SIMIT - RS Cell with a new one where the signal will interact from PS to SIMIT. The communication methods supported by the external connection between these two software are OPC DA and OPC UA. Shared memory, available in SIMIT and RS, is not supported in PS. From V 16.0 of PS and SIMIT 10.2 is possible to have a SIMIT external coupling. In order to do so "ProcessSimulateCoupling" needs to be attached in SIMIT installation folder.

### 4.2.7 Material flow operations and support signals

The operations are actions to accomplish the manufacturing process. In simulation mode, they play an important role however, in the current framework of RDT, they play a complementary role to emulate activities such as the appearance of the material, manual manipulation of material but also support operations to validate process and resources. Material flow is a process that should be done for each part according to the requirement of the process and the way to be controlled. Non-simulation operations and flow operations are common in this stage.

When the RDT is running, many of these operations need to be executed either directly from PS or through buttons in any HM; to do this, the support signals must also be created and connected. It is important to add and link each of the operations in the material flow viewer so the parts can endure during the time they are required. It is recommended to create operations to appear parts in key positions so the validation of other processes can be done easily.

### 4.2.8 Signal handling

A key stage in the proper operation of DT is the right way of handling the signals, especially in this case where the connections and mapping area have already been done. The task is to replace manufacturing cell signals from SIMIT-RS to SIMIT-PS; since these signals are already connected in a different block inside SIMIT, no changes are allowed. To do so, the following procedure is proposed: Firstly, the cell signals from the original coupling (Shared memory) should be exported to an excel file and disconnected since it will be replaced by a new one. Secondly, the file needs to be modified based on the accepted standard of PS. Thirdly, signals must be imported on PS. Finally, the signals need to be exported from PS and imported again in SIMIT but now with a new external coupling (with the same name as the original one) as configured in section 4.2.6. That way, the path of signals in SIMIT is set by default. Figure 4.7 shows all the described words put forward graphically to get a better idea.

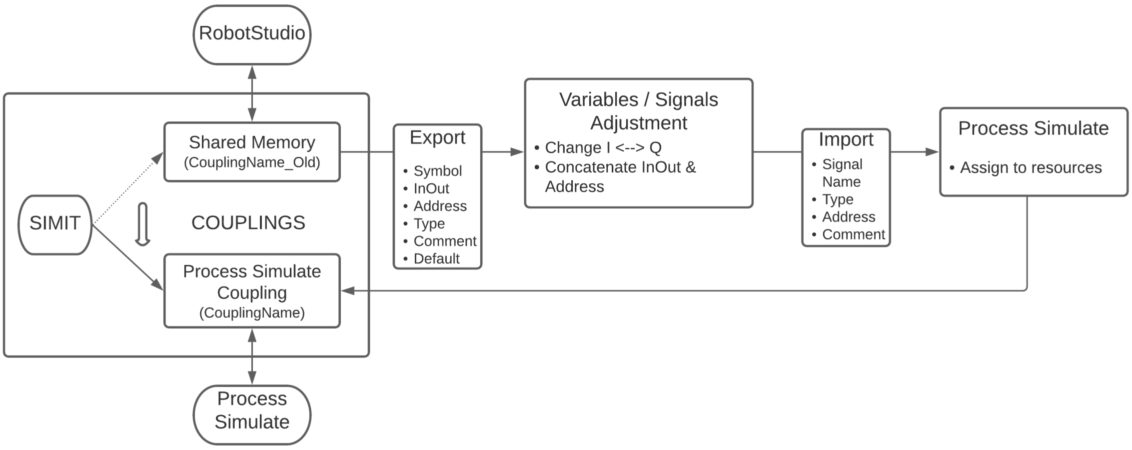


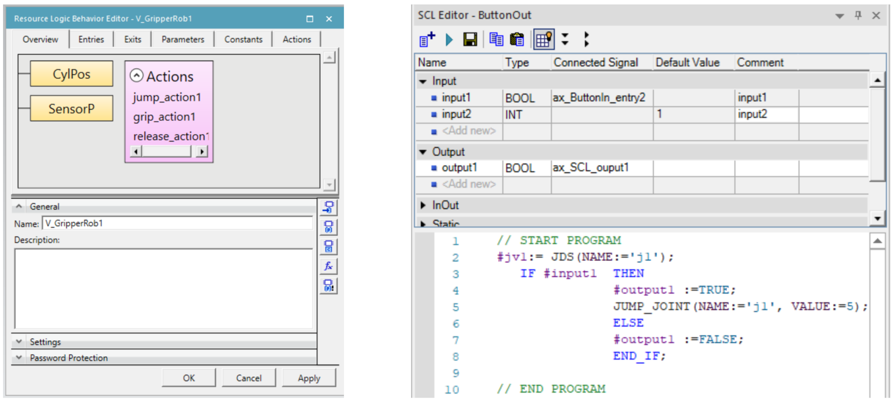
**Figure 4.7:** The process to set and export variables from SIMIT to Process Simulate based on existing couplings.

### 4.2.9 Resource logic and signal connection

With the kinematics created in the different resources, the characterization of the sensors, and the signals imported into PS, it is now possible to create logical behavior for each of the resources and connect them with I/O signals coming from SIMIT. For this project, two different ways are proposed, Logic Block and SCL Editor, see Figure 4.8. Input signals control the behaviour of resources; these signals must be connected into the logic each. Output signals are normally sensor signals or resource statuses that are generated automatically inside PS when the resources are created; since the signals are already defined from section 4.2.8, it is necessary to rename and replace them with the corresponding signal.

Sensors

To sense and govern the specific parts or resources according to the required process flow in the manufacturing cell, sensors and logic are fundamentals and are used significantly all around a cell. Based on sensors input, other operations are defined (with logic blocks) in a way when and what should happen. When defining a sensor, several parameters should be defined like detection length, parts/resources to detect, and name. The last one is very important since the name should be the same imported in section 4.2.8.



Name: V_GripperRob1

Description:

| Name | Type | Connected Signal | Default Value | Comment |
|---|---|---|---|---|
| Input | | | | |
| input1 | BOOL | ax_ButtonIn_entry2 | | input1 |
| input2 | INT | | 1 | input2 |
| <Add new> | | | | |
| Output | | | | |
| output1 | BOOL | ax_SCL_ouput1 | | |
| <Add new> | | | | |
| InOut | | | | |
| Static | | | | |

```
// START PROGRAM
#jv1:= JDS(NAME:='j1');
   IF #input1  THEN
            #output1 :=TRUE;
            JUMP_JOINT(NAME:='j1', VALUE:=5);
            ELSE
            #output1 :=FALSE;
            END_IF;

// END PROGRAM
```

**Figure 4.8:** Development of logic in resources within PS.

### 4.2.10 Robot setups and auxiliary programs

Regardless if the exported files include Robot CAD files with kinematics or not, they can be downloaded from the ABB website separately and imported into PS. Due to the advanced functionality and precise behavior of the RS virtual controllers, all the robot programs were complied and executed in RS running parallel with PS through VRC server. The robots will play just a shadow behavior in PS; however, it is recommended to configure them properly as they will use to create auxiliary programs to validate operations and tool's functionalities. It is possible to download complete robot file modules from RS in the zip format and then uploaded them into PS, which makes it easier to define the robot properties based on the existing data. Furthermore, it facilitates the programming of the robots with respective work objects, tool data, targets, and safe move information as in RS.

#### 4.2.10.1 Virtual Robot Controller setup

PS V16 and above allows the users to connect to RS (or other robot vendors) using a virtual robot controller (VRC) connection in order to simulate and emulate complete robot systems (controllers) maintaining the high accurate robot functionality. PS V16.1 additionally provides setting the pointer for a specific instance in the robot program running in RS. During this setup in our RDT framework, the PS exchanges the I/O signals, joint values, and much more coming from the RS and communicates with SIMIT and TIA Portal via a defined pathway and bridges the signal flow between all these software. For a better understanding of the signal exchange of VRC connection set up, it is presented in Figure 4.9.

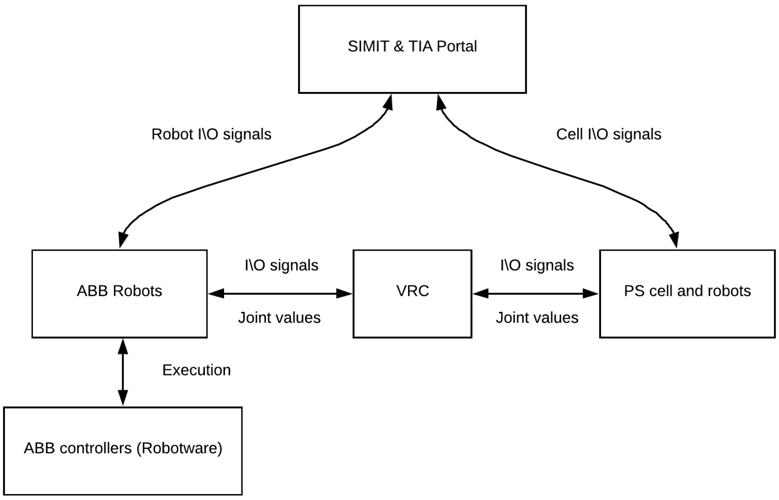
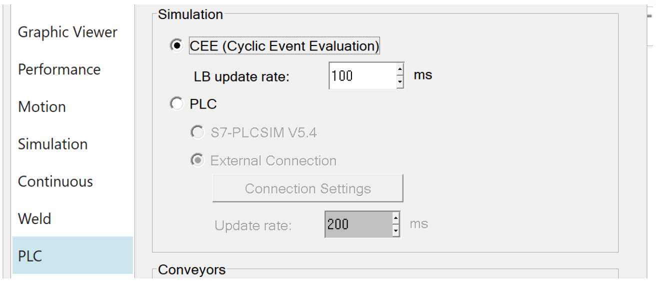


**Figure 4.9:** VRC connection and data transfer.

### 4.2.11 Cyclic Event Evaluator simulation - Test

PS has two different ways to run a simulation: by connecting to the internal one called Cyclic Event Evaluation (CEE) and by connecting to PLC (S7-PLCSIM or external connection). As shown in the Figure 4.10. So, CEE work inside PS and no other connection is needed. It is the core for the event base simulation engine, every cycle it collects the values of the PLC signal and determines the course of the simulation[54]. CEE will be implemented just for internal tests to validate signals and run robot programs, resource, and material flow operations. PLC connection will be used when running the complete cell integrated with SIMIT and TIA Portal.



Simulation

● CEE (Cyclic Event Evaluation)

LB update rate: 100 ms

○ PLC

○ S7-PLCSIM V5.4

● External Connection

Update rate: 200 ms

**Figure 4.10:** Options available to run a simulation: Cyclic Event Evaluation and PLC.

# 5

# Application study

*The application study presents essential steps adopted to develop cell in PS based on the defined methodology. Furthermore, it describes the system properties and the final process flow (manually executed) of the developed cell in the RDT framework.*

## 5.1 Experimental setup

The software installed was introduced in section 4.2.1. The system used for the implementation has the following key properties.

- Intel(R) Core (TM) i7-8700K @3.70 GHz
- 32.0 GB RAM
- Windows 10 Enterprise
- NVIDIA GeForce GTX 1080 Ti

## 5.2 CAD files

The CAD models were exported directly from RS as there was no access to the original one. On comparing with rendering, file size, and quality, the STEP file format was chosen to export to PS. To retain the same hierarchical structure and position, the cell was exported in a single file and imported into PS. Here, it was divided to create and define the resources required in the simulation process accordingly to its functionality.

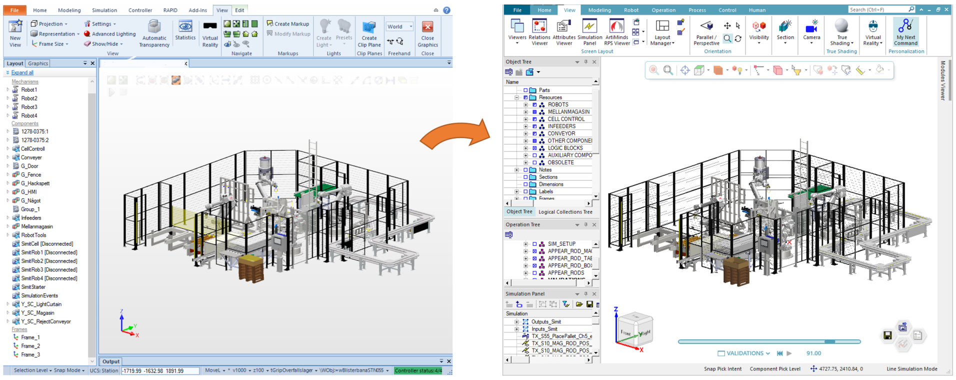


**Figure 5.1:** Final result after CAD migration from RobotStudio to Process Simulate.

## 5.3 Definitions of parts and material flow

Compared with the given scenario of RS where parts appear/disappear, or they are attached/detached, here it is proposed to move them all around the process so their logic and complexity can be decreased. Here the products are divided into individual components and grouped into assemblies. The parts are assigned to flow operations and linked with resources, especially containers that moved along conveyors transporting parts. The creation of object flow operation in material flow viewer is presented in Figure 5.2.

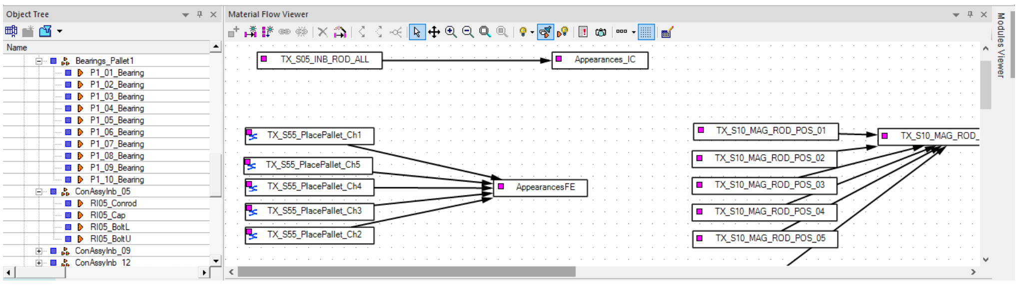


**Figure 5.2:** Parts and operations for material flow.

## 5.4 Communication and connection setup

Based on the analysis done in section 4.2.2 and software implemented in section 4.2.1, the diagram in Figure 5.3 was created to better understand how it was in the beginning (blue outline including dotted line) and later when connected to PS (green and blue outline excluding dotted line). The original I/O signals from the model were done through shared memory between SIMIT and RS(dotted line). Now in this project, the signals will come from PS through an external connection but also there is a communication between the robot's in PS to RS through a VRC server (see section 4.2.10.1).

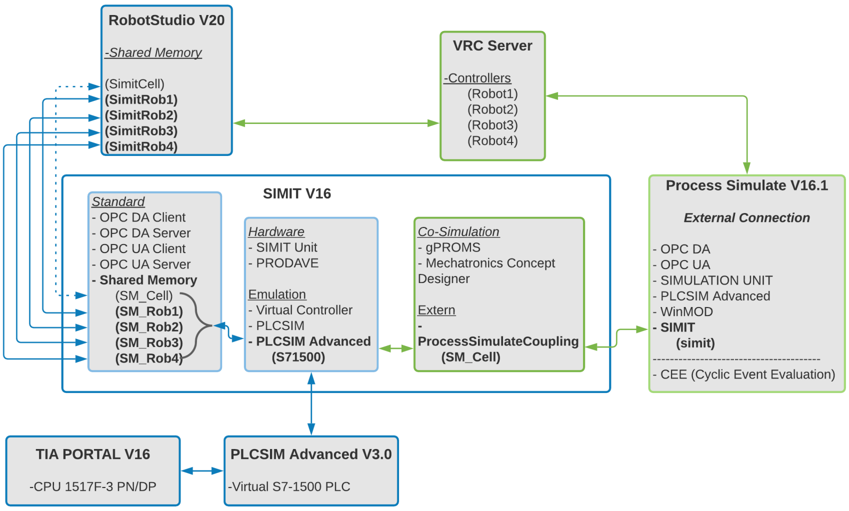


**Figure 5.3:** Software connections of both platforms.

## 5.5 Validation of connections

As described in Figure 5.4, several connections between the software tools must be created and validated, these connections are: a) PS - SIMIT, b) PS - VRC - RS, c) TIA Portal - PLCSIM Advanced, d) SIMIT - PLCSIM Advanced, e) PLCSIM Advanced - WinCC Runtime and f) SIMIT - RS

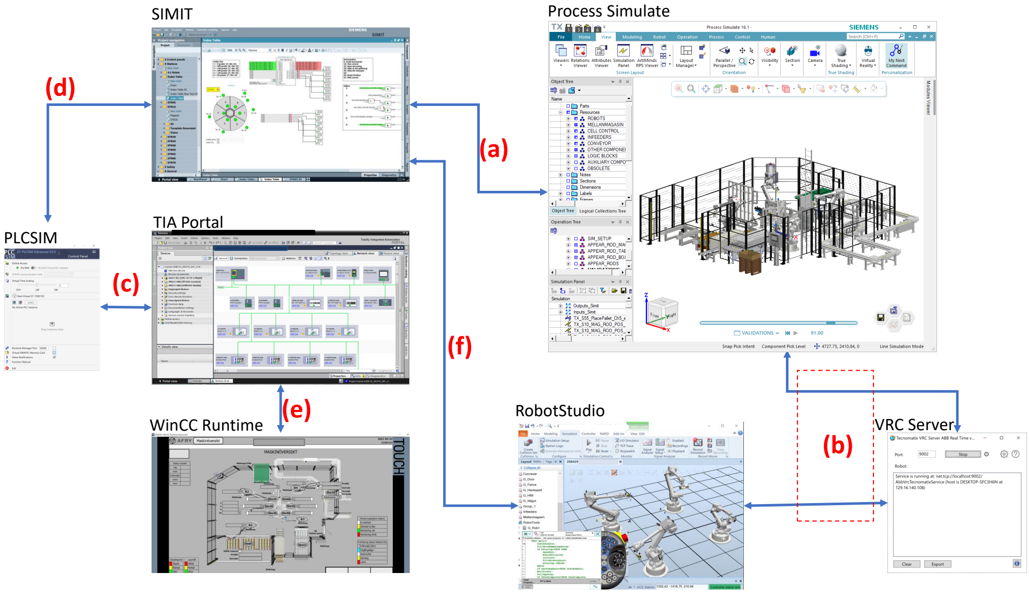


**Figure 5.4:** Connections between software tools in the new platform.

## 5.6 Kinematics

In some cases, the "create crank" feature from PS was used to build complex devices as shown in Figure 5.5. The kinematics of the necessary resources were developed to carry out the flow of the parts through the cell, but not for those does not take part in the simulation.

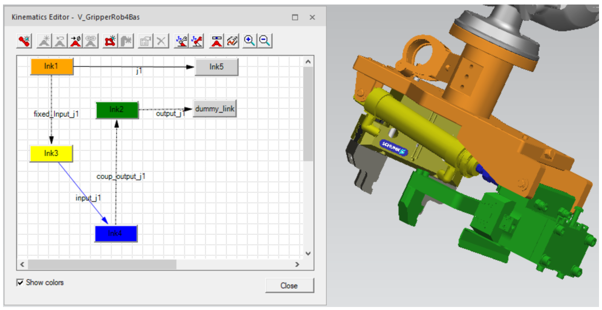
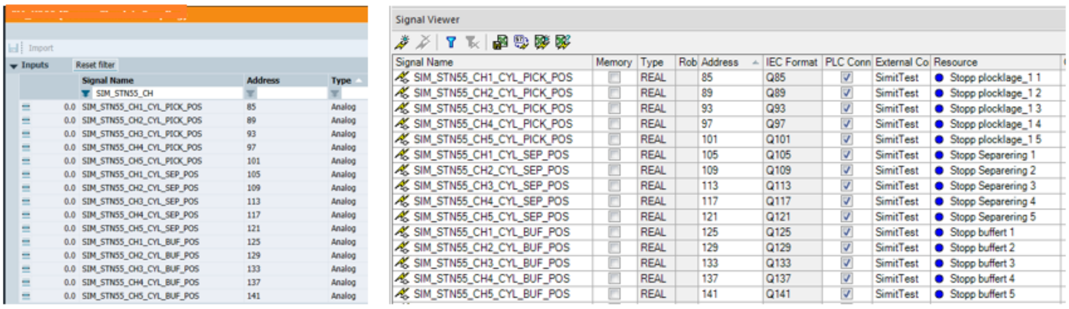


Figure 5.5: Kinematics definition in PS.

## 5.7 Mapping and validating signals

Around 280 I/O signals exist in the original project. These signals are composed of signals required for the code in the PLC but also for the simulation of movements or actions that are carried out manually by operators, such as the placement of material on the conveyors or other types of signals to have a more realistic operation of the process. In Figure 5.6, the left one shows the imported signals in SIMIT and the right one shows the imported signal in PS (nomenclature kept the same).



| | Signal Name | Address | Type |
|---|---|---|---|
| | SIM_STN55_CH | | |
| 0.0 | SIM_STN55_CH1_CYL_PICK_POS | 85 | Analog |
| 0.0 | SIM_STN55_CH2_CYL_PICK_POS | 89 | Analog |
| 0.0 | SIM_STN55_CH3_CYL_PICK_POS | 93 | Analog |
| 0.0 | SIM_STN55_CH4_CYL_PICK_POS | 97 | Analog |
| 0.0 | SIM_STN55_CH5_CYL_PICK_POS | 101 | Analog |
| 0.0 | SIM_STN55_CH1_CYL_SEP_POS | 105 | Analog |
| 0.0 | SIM_STN55_CH2_CYL_SEP_POS | 109 | Analog |
| 0.0 | SIM_STN55_CH3_CYL_SEP_POS | 113 | Analog |
| 0.0 | SIM_STN55_CH4_CYL_SEP_POS | 117 | Analog |
| 0.0 | SIM_STN55_CH5_CYL_SEP_POS | 121 | Analog |
| 0.0 | SIM_STN55_CH1_CYL_BUF_POS | 125 | Analog |
| 0.0 | SIM_STN55_CH2_CYL_BUF_POS | 129 | Analog |
| 0.0 | SIM_STN55_CH3_CYL_BUF_POS | 133 | Analog |
| 0.0 | SIM_STN55_CH4_CYL_BUF_POS | 137 | Analog |
| 0.0 | SIM_STN55_CH5_CYL_BUF_POS | 141 | Analog |

| Signal Name | Memory | Type | Rob | Address | IEC Format | PLC Conn | External Co | Resource |
|---|---|---|---|---|---|---|---|---|
| SIM_STN55_CH1_CYL_PICK_POS | ☐ | REAL | | 85 | Q85 | ☑ | SimitTest | ● Stopp plocklage_1 1 |
| SIM_STN55_CH2_CYL_PICK_POS | ☐ | REAL | | 89 | Q89 | ☑ | SimitTest | ● Stopp plocklage_1 2 |
| SIM_STN55_CH3_CYL_PICK_POS | ☐ | REAL | | 93 | Q93 | ☑ | SimitTest | ● Stopp plocklage_1 3 |
| SIM_STN55_CH4_CYL_PICK_POS | ☐ | REAL | | 97 | Q97 | ☑ | SimitTest | ● Stopp plocklage_1 4 |
| SIM_STN55_CH5_CYL_PICK_POS | ☐ | REAL | | 101 | Q101 | ☑ | SimitTest | ● Stopp plocklage_1 5 |
| SIM_STN55_CH1_CYL_SEP_POS | ☐ | REAL | | 105 | Q105 | ☑ | SimitTest | ● Stopp Separering 1 |
| SIM_STN55_CH2_CYL_SEP_POS | ☐ | REAL | | 109 | Q109 | ☑ | SimitTest | ● Stopp Separering 2 |
| SIM_STN55_CH3_CYL_SEP_POS | ☐ | REAL | | 113 | Q113 | ☑ | SimitTest | ● Stopp Separering 3 |
| SIM_STN55_CH4_CYL_SEP_POS | ☐ | REAL | | 117 | Q117 | ☑ | SimitTest | ● Stopp Separering 4 |
| SIM_STN55_CH5_CYL_SEP_POS | ☐ | REAL | | 121 | Q121 | ☑ | SimitTest | ● Stopp Separering 5 |
| SIM_STN55_CH1_CYL_BUF_POS | ☐ | REAL | | 125 | Q125 | ☑ | SimitTest | ● Stopp buffert 1 |
| SIM_STN55_CH2_CYL_BUF_POS | ☐ | REAL | | 129 | Q129 | ☑ | SimitTest | ● Stopp buffert 2 |
| SIM_STN55_CH3_CYL_BUF_POS | ☐ | REAL | | 133 | Q133 | ☑ | SimitTest | ● Stopp buffert 3 |
| SIM_STN55_CH4_CYL_BUF_POS | ☐ | REAL | | 137 | Q137 | ☑ | SimitTest | ● Stopp buffert 4 |
| SIM_STN55_CH5_CYL_BUF_POS | ☐ | REAL | | 141 | Q141 | ☑ | SimitTest | ● Stopp buffert 5 |

Figure 5.6: Signals couplings and connections SIMIT - PS.

All the I/O signals were tested; Table 6.2 shows the validation of different signals coming from SIMIT that control the resources in PS. The sensors were verified directly in the SIMIT interface.

## 5.8 Robots

To facilitate the validation of the process with the robots, OLP methods were done inside PS to create a program for each robot. Later these programs were downloaded into each robot virtual controller inside RS where the programs are compiled and executed. So, all the robots in the PS are manipulated based on the RS virtual controller commands defined with different program modules for each robot in the RS communicating with VRC servers.

## 5.9 Setting up the VRC connection

In order to set up the VRC connection, it is required to meet the following criterion on the software side:

- RS, V2020 or higher
- PS, V16 or higher
- ABB-Rapid OLP controller license in PS
- Tecnomatix VRC Server ABB Real Time execution file

Moving further, some of the features were modified on the PS to set ABB robots for VRC interface with ABB virtual controllers.

- In the controller settings, controllers were set to ABB rapid with motion planner as VRC, which must be done for all the robots individually.
- Controller name in PS and RS are kept the same (if not connection will be unsuccessful).
- Four VRC servers were created with different port instances and specified pointer as the main module in the VRC connection.

## 5.10 Description of the cell process flow

Here is depicted the general material flow of the cell, which was performed in the manual model: The process starts in Station 1 where several part assemblies enter the cell through Conveyor 1 (C1). Robot 1 (R1) grabs every part and places it on the Magazine (M), which holds up to 19 parts, each detected by proximity sensors. When the parts are available in the magazine, Robot 2 (R2) fetches a part and moves it to the scanning and marking process to finally releases it on the Index table in Station 2 (S2). On S2, the placed part is just clamped and then goes to the next station. On Station 3 (S3), the part is unscrewed with the help of clamps and automatic screwdrivers. Station 4 (S4) separates the two parts so new components can be inserted inside them. A vision system inspects and checks quality on Station 5. Station 9 contains the Conveyor 2 (C2), which moves the parts to be inserted in the initial part assembly, the Robot 3 (R3) moves these parts into a tool located in Station 6 (S6). Further, in S6, two new parts are inserted in the upper and lower part of the assembly. On station 7 (S7), the two sub-assemblies are fetched separately by Robot 4 (R4), which contains two gripper tools. Finally, each sub-assembly is placed on Station 8 (S8), which comprises a conveyor 2 (C2) with a special pallet that holds these two sub-assemblies separately. The graphical representation of the cell was shown in section 4.1.2.

Figure 5.7 shows the final integration of the PS model in the RDT; it depicts all the programs running and connected to each other. The functionality meeting the objectives established in the beginning through the established methodology.

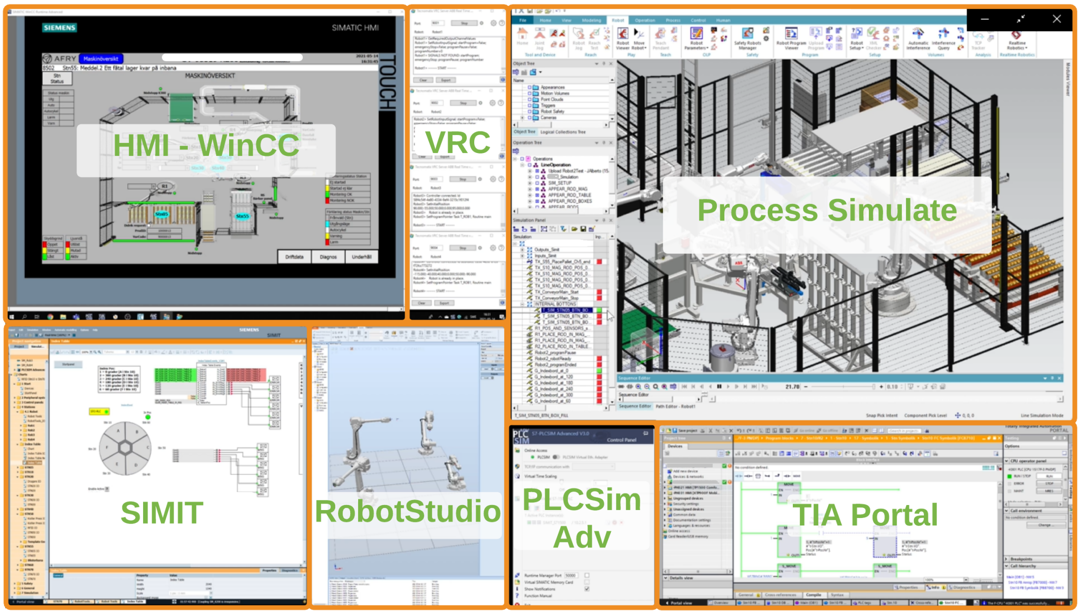


**Figure 5.7:** RDT framework with software in the loop architecture.

# 6

# Results

A systematic process was developed to create and migrate an RDT between platforms from RS to PS, all above under AFRY´s framework. For this purpose, the complete cell of the original scenario was migrated and characterized to operate in a complete, integrated, and communicated manner.

Based on the objectives defined initially, here it is described how they were addressed in this work.

**Objective 1**. The integration and communication between the different software tools were carried out correctly. The communication between PS and RS through the 4 VRC server was successfully validated by executing one of the developed rapid code functioning within the automatic mode.

Note: The developed emulated SIMIT and PLC logic architecture was considered to be very close to the real system. All the verification and validation were carried to meet these pre-defined functionalities (open, close, rotate, grip, release, etc.) as described in Table 6.2 & 6.3.

**Objective 2**. The virtual behavior of the cell was very close to the defined behavior (SIMIT and PLC logic) and even improved from the one designed in the initial RDT, which was modeled in RS. For example, it shows a real behavior of a) flow and b) material handling.

The automatic execution of cell operation is expected for future work. Comparing the virtual behavior of the cell in PS with RS model was not carried based on any determined entities (response time, delay, resources, and consumption). Behavioral verification (PS model) was done based on achieving required outcomes (precision in placing of parts).

a) Unique part entity is created when the process is started and is carried upon through different stations for assembly operation. Unlike the original twin, where animations appear and disappearing the same part as in the different stations and no material or entities ever moved in process flow of the cell.

b) The tools and fixtures grab the parts and are not appeared and disappeared as in the original model (RS).

- The process was systematically developed to be able to export and migrate a project developed in RS to the PS platform.

**Objective 3**. Comparison based on the activities carried (side by side) is presented in section 6.5.

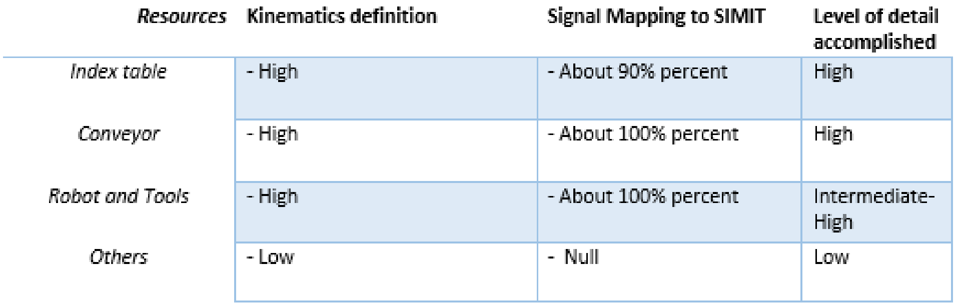
**Objective 4**. It was carried out through the review of the literature and the capacities, features, and developments of the RDT software tools in the area of AI.

## 6.1 Model functionality with RDT framework

The RDT framework running with the built PS model is outlined in Figure 5.4. For the successful execution of the RDT model, initialization of all the software with proper and valid connection was the fundamental measure to be achieved. As SIMIT is the main role in the RDT framework in connection mapping and emulating hardware behavior of the different devices in the cell, it was started first, then followed with TIA portal, PLC SIM Advance, PS, RS, VRC server, and WinCC. On top of it, port number, IP address, couplings, pointers should be correctly set. On successful integration of all the software, the handshake between PLC via HMI and devices in the PS model was done in the order (station, robot wise) to verify and validate everything works as intended. Table 6.2 & 6.3 lists the main components which were validated accordingly. The outcome of this holds promising based on the defined PLC and SIMIT logic. However, this complete validation was done manually by interacting continuously with the HMI panel and SIMIT commands. Figure 6.1 shows the level of accomplishment.

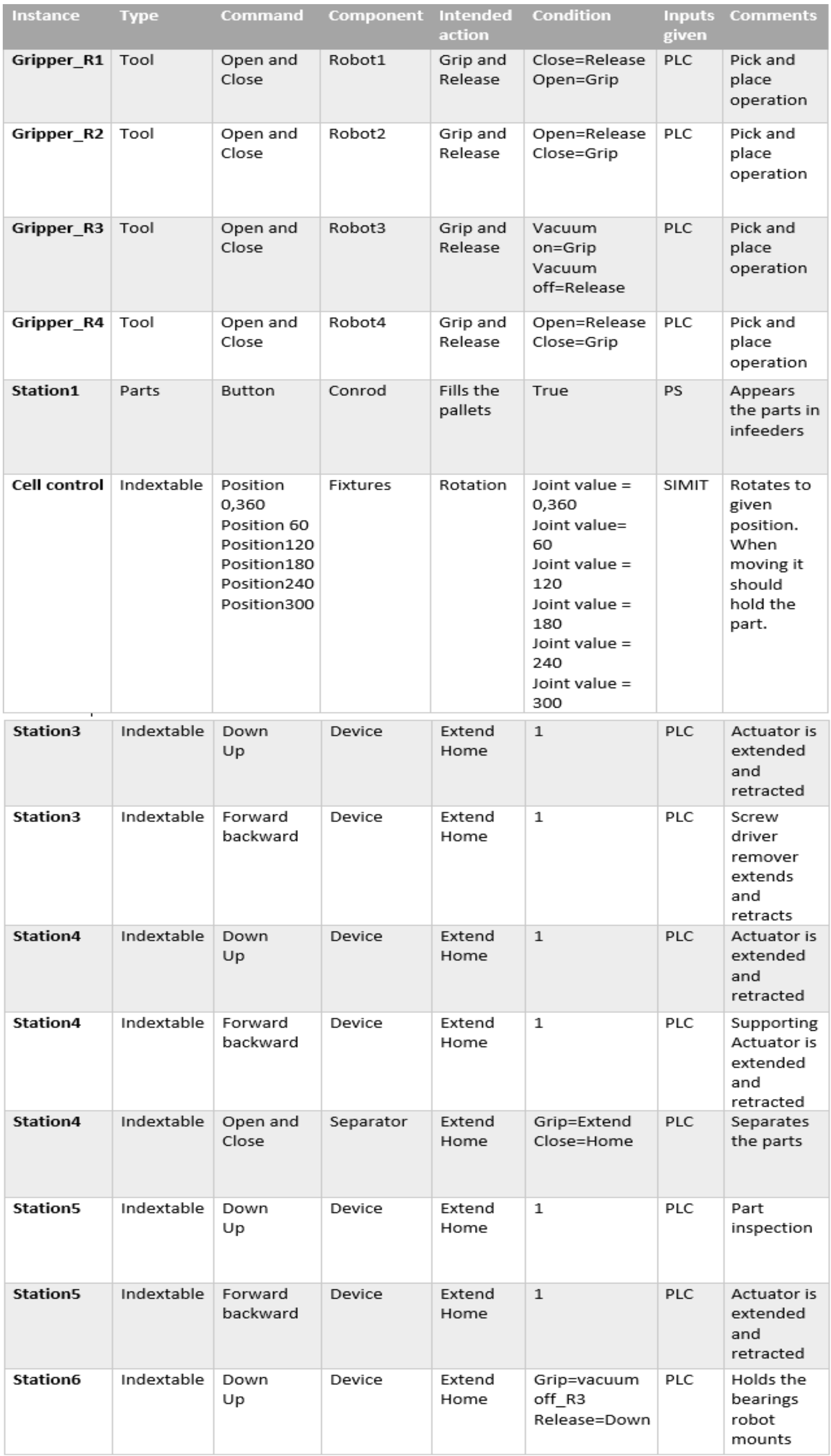
As the robots could not work with the developed rapid code (which includes many interdependences to be met like safety, I/O signals, tasks, PLC signals), a rapid code to find out the connection and communication works with RS, PS, SIMIT, and PLC was made. With the above, the correct connection and communication were confirmed. Finally, to emphasize that the robot shall work with developed rapid code, one of the programs was tested by executing it in automatic mode, which validates to be functional with all defined setup.

**Table 6.1:** Level of accomplishment of modeling and signal connections achieved in PS model.



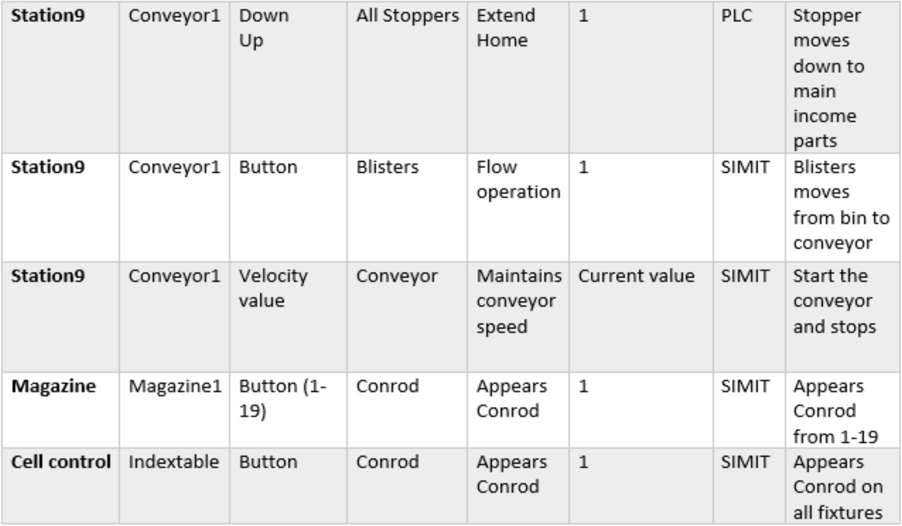
| *Resources* | **Kinematics definition** | **Signal Mapping to SIMIT** | **Level of detail accomplished** |
|---|---|---|---|
| *Index table* | - High | - About 90% percent | High |
| *Conveyor* | - High | - About 100% percent | High |
| *Robot and Tools* | - High | - About 100% percent | Intermediate-High |
| *Others* | - Low | - Null | Low |

**Table 6.2:** Verification & validation protocol of model functionality with SIMIT and HMI.



| Instance | Type | Command | Component | Intended action | Condition | Inputs given | Comments |
|---|---|---|---|---|---|---|---|
| **Gripper_R1** | Tool | Open and Close | Robot1 | Grip and Release | Close=Release<br>Open=Grip | PLC | Pick and place operation |
| **Gripper_R2** | Tool | Open and Close | Robot2 | Grip and Release | Open=Release<br>Close=Grip | PLC | Pick and place operation |
| **Gripper_R3** | Tool | Open and Close | Robot3 | Grip and Release | Vacuum on=Grip<br>Vacuum off=Release | PLC | Pick and place operation |
| **Gripper_R4** | Tool | Open and Close | Robot4 | Grip and Release | Open=Release<br>Close=Grip | PLC | Pick and place operation |
| **Station1** | Parts | Button | Conrod | Fills the pallets | True | PS | Appears the parts in infeeders |
| **Cell control** | Indextable | Position 0,360<br>Position 60<br>Position120<br>Position180<br>Position240<br>Position300 | Fixtures | Rotation | Joint value = 0,360<br>Joint value= 60<br>Joint value = 120<br>Joint value = 180<br>Joint value = 240<br>Joint value = 300 | SIMIT | Rotates to given position. When moving it should hold the part. |
| **Station3** | Indextable | Down<br>Up | Device | Extend<br>Home | 1 | PLC | Actuator is extended and retracted |
| **Station3** | Indextable | Forward backward | Device | Extend<br>Home | 1 | PLC | Screw driver remover extends and retracts |
| **Station4** | Indextable | Down<br>Up | Device | Extend<br>Home | 1 | PLC | Actuator is extended and retracted |
| **Station4** | Indextable | Forward backward | Device | Extend<br>Home | 1 | PLC | Supporting Actuator is extended and retracted |
| **Station4** | Indextable | Open and Close | Separator | Extend<br>Home | Grip=Extend<br>Close=Home | PLC | Separates the parts |
| **Station5** | Indextable | Down<br>Up | Device | Extend<br>Home | 1 | PLC | Part inspection |
| **Station5** | Indextable | Forward backward | Device | Extend<br>Home | 1 | PLC | Actuator is extended and retracted |
| **Station6** | Indextable | Down<br>Up | Device | Extend<br>Home | Grip=vacuum off_R3<br>Release=Down | PLC | Holds the bearings robot mounts |

**Table 6.3:** Verification & validation protocol of model functionality with SIMIT and HMI continued...



| | | | | | | | |
|---|---|---|---|---|---|---|---|
| **Station9** | Conveyor1 | Down<br>Up | All Stoppers | Extend<br>Home | 1 | PLC | Stopper moves down to main income parts |
| **Station9** | Conveyor1 | Button | Blisters | Flow operation | 1 | SIMIT | Blisters moves from bin to conveyor |
| **Station9** | Conveyor1 | Velocity value | Conveyor | Maintains conveyor speed | Current value | SIMIT | Start the conveyor and stops |
| **Magazine** | Magazine1 | Button (1-19) | Conrod | Appears Conrod | 1 | SIMIT | Appears Conrod from 1-19 |
| **Cell control** | Indextable | Button | Conrod | Appears Conrod | 1 | SIMIT | Appears Conrod on all fixtures |

## 6.2 Verification and validation

To examine PS model behavior meet all requirements based on the defined architecture of SIMIT and PLC logic, verification and validation are carried with authors developed protocol. The protocol is developed based on a different slice of cell operations. Each slice tested one by one, depending on the sequence of operation flow. The protocol is represented in Table 6.2 & 6.3. The protocol embodies the intended action to be achieved when the condition becomes true, while testing with SIMIT commands, more explicit results can be comprehended (position of the piston). On the other hand, testing with PLC will determine both behavioral conditions (simulation model) and precise results (SIMIT). However, this protocol was tested repeatedly to prove that all the operations meet the desired outcome when the cell is executed in automatic mode.

The validation of the robot functionalities was not performed through the developed protocol because of the existing safety code logic. Instead, changes in the tag value (RS) were considered when the respective inputs are given through HMI. This way, complete verification and validation were accomplished.

## 6.3 Model limitations

Some shortcomings were encountered during the project realization, which should be considered and embraced in the future, especially when executing in automatic mode.

### 6.3.1 Absence of physics engine in the simulation model

3D CAD models in PS do not interact with any of the other parts in the simulation, making some of the behaviors static and unrealistic. PS V16.1 has some new physics functionalities, but they are limited to material and property definition. However, the need for collision physics is still anticipated. Collisions are helpful in one or more instances in PS model (gripping, conveyor, containers, and path robotics), which is compensated by adding secondary features for the specified components.

### 6.3.2 Robots communication through VRC server

PS inbuilt virtual controller lacks the emulation behavior and the precise functionality of ABB robot's virtual controllers. All the robots in RS were connected with the VRC server platform to PS robots(mimic's movements and signals). During this setup, sometimes PS tries to overwrite the position of robots controlled by ABB virtual controllers, which seems deceptive. Care should be taken before running the cell in automatic mode.

### 6.3.3 Standalone model

All the modeling work was carried with PS Standalone platform means it is operated independently without any database accompanied with it. So, when whatever changes are made in this platform will not be updated in any other systems running parallel to it. All the files should be copied entirely from the original "sysroot" path and pasted on the desired one. Note: Care should be taken that all files name does not include special letters like ä, å, or ö. Sometimes this might cause issues while transferring.

## 6.4 AI application in RDT framework

The present manufacturing cell is completely automated, where all the parts are handled by the four robots. The implementation of the ML techniques on the robot's path optimization can be more beneficial to attain better operation cycle time. Among the different ML techniques, Reinforcement learning can be adopted as it enables the robots to discover an optimal solution (path trajectory) through self-learning from trial and error interaction with its bounded system[69].

As the developed robot codes were not able to execute completely within the time frame, the cell was operated in manual mode. However, it was found that the robots were defined to move to the preferred location based on the defined target values. The

paths between the targets were interpolated by the controller based on this, the robot's moves are desired. This path trajectory can be monitored continuously and trained with reinforcement learning algorithms relating to the kinematics data accessible from the robot viewer information panel (graphical interface) in PS, for achieving a shorter path with maintained accuracy.
On top of it, Tecnomatix.NET API can be used to develop the custom-made API and bridge the connection between AI-driven models (Python, MatLab, etc.) and PS, which can be scripted in C# using visual studio. This also provides the testbed to achieve the aforementioned architecture; the data from PS can be automatically fed into an AI-driven model, which makes the process still efficient. Finally, transferring the trained model from virtual platform to physical space can be assisted with transfer learning techniques such as DFDD proposed in [20]. Nevertheless, computing RDT at edge may open-up new ways to meetup increased resilience, as highlighted in [21].

The proposed scenario can be validated as in a similar case presented in [14], where PS was used as a virtual model that includes a Motoman robot, a Siemens SIMATIC S7-1500 PLC, the TIA Portal, and the WinCC counterpart being the same on physical cell. The training of the AI-driven model (deep reinforcement learning) was carried in three phases. First, a fast-iterative low-resolution simulation which was done iteratively only in the PS platform (using custom-built API in Tecnomatix.NET API) carried offline. Second, high-resolution training is done with PS and physical platforms. Third, physical verification training is done with a physical platform. For SiL and HiL techniques, physical PLC was used. The real-time live signal communication was achieved with the OPC server in phases two and three, which was managed to execute near synchronously under the option 1:1 real-time simulation speed. Within their work, they accentuate PS functionality as a powerful tool for achieving controlled simulation (robot motion and paths) needed for training phases, provided simulation model fidelity being high and accurate.

## 6.5 Comparison between ABB RobotStudio and Siemens Process Simulate

The following section has the intention to compare both software from the author's perspectives, framed and limited to the activities here performed to carry out the DT.

**Table 6.4:** PS and RS Comparison Part 1.

| **Concern** | **PS** | **RS** |
|---|---|---|
| CAD | - Easy to export a single CAD file to later divide it into the required components.<br>- Offers the capability to friendly organize resources/parts in compounds so that they can be easily manipulated, hidden, unhidden, even relocated together with other resources.<br>- Has many types of resources type like clamp, container, conveyor, device, dock system, equipment prototype, fixture, flange, gripper, gun, human, plc resource, robot, security window, tool prototype, turntable, work table- Some of them with specific features.<br>- Frames are a friendly and functional capability that allows saving key positions that can ease the manipulation, location, simulation, flow, paths of different resources. They can be easily managed through Frame folders in the object tree or inside each of the resources and parts. | - Provides a wide range of possibilities for importing and exporting project files. In accordance with this project, CAD translators (needs additional license) made things easier to export the required file format into the PS, with a varied range of supportive file formats.<br>- Provides the option to import the CAD data as empty part, component group (includes several parts), smart component (includes logic models connected with different smart components or resources in the model, with or without 3D graphical representation). However, all of them are grouped under components in the layout tree based on the cell definition. The visible option is feasible to hide and show the 3D geometries in different levels of the layout tree.<br>- The definition of resources is restricted to robot, tool, external axis, and device. While defining the kinematics and mechanism for the CAD model. The definition of links, joints, and dependencies are made easier without writing any expression or logic. |
| CONNECTION | - Ability to connect to several third parties software through different connections like OPC UA, OPC DA, Simulation Unit, PLC-SIM Advanced, WinMOD, SIMIT, and S7-PLCSIM V5.4 | - The connection to third parties software is achieved with shared memory coupling (SHM). However, the IRC5 controller can be used to connect with different networks in the factory devices. |

**Table 6.5:** PS and RS Comparison Part 2.

| **Concern** | **PS** | **RS** |
|---|---|---|
| PLC, LOGIC | - Capability to program with SCL as in a PLC in the smart component, which makes it easier and in a standard way.<br>- The conveyors can add control points and other logic to appear, disappear, stop pallets, containers. It also has the "conveyor drive" option to simulate positions without start the simulation. | - All the smart components are programmed and controlled with logic blocks; each of them is modified based on functionality to achieve the process of the cell. The complete model can be manipulated in the station logic panel. |
| SIMULATION | - It can easily verify and validate process through simulation using sequence editor.<br>- Pose editor can be saved in resources and robots, this help to validate path or sequences quickly.<br>- Robot tool positioning is easily accomplished with frame placement.<br>- Gripping objects is possible without the need to attach, detach, appear, disappear components, or create logic | - Based on the defined station logic, required simulation flow can be achieved. This simulation can be recorded in .exe format and can be tested and used for future commissioning work.<br>-Gripping can be achieved through collision physics functionality. Another way to achieve this is by using attach and detach feature. |
| PHYSICS | -Collision detection is a common tool in PS. From V16 it has a beta version of physics where material can be assigned to parts, resources and have a static or dynamic behavior. | -Includes some of the physics properties like material characteristics given for the robot tool materials, weight, center of gravity. It also has collision physics property in conveyors and robots. |
| ROBOTS | - Can interact with other brands of robot (Cloos, Comau, Denso, Fanuc, Kawasaki, Kuka, Mitsubishi, etc.) through VRC or RCS. | -OLP with realistic virtual controllers (features) provides an advanced level of development to build high sophisticated solutions maintaining a higher level of accuracy, reliability, and efficiency in the robot operation. Safe move functionality enables unsurpassed safety with reduced total cost investment in the factory (enabling close human-robot workspace). |

# 7

# Discussion

## 7.1 Thought process in modeling

The PS model was created in order to understand its functionalities and capabilities when introduced into RDT framework. While dealing with this case scenario following set of questionnaires were appraised: what are the needs to be considered while lifting simulation model from one platform to another, what steps should be followed, how it should be done, what should be achieved, do the defined things behave as intended and so on.

In our case, as the complete simulation model was defined and built in RS, some of these tasks to achieve became comfortable while building simulation model in PS. All the 3D CAD models were available, kinematics definition with their limits value, constraints, dependencies were accessible and signals connections made with SIMIT were interpreted. However, managing logic, material handling, defining, and connecting signals were completely different and were established based on the process requirements of the stations.

## 7.2 Model behaviour

As our aim was to retrofit PS model into the existing RDT framework fulfilling all the needs to achieve a functional RDT model as desired. The mechanism of all the defined components was precise as intended as SIMIT and PLC logic (verify and validated individually). However, when handling the parts in the different stations, based on the defined logic's at the component level, PS model behaves more realistically as happens in the actual case (gripping and releasing the part). This made many of other things work better than RS model when simulating material flow operation. Along with this PS have the functionality to define which parts to be gripped and sensed at different instances to achieve the required operation in the respective station. All the operations carried were very close to real time simulation clock. A more smooth material flow sequence (picking and placing of parts precisely) was observed in comparison to RS model. In a nutshell, our model built in PS platform established all the essential needs to achieve a functional RDT framework. In some instances, PS platform has better capabilities than the existing simulation platform in RDT framework.

## 7.3 Challenges to achieving RDT framework

As our thesis was more concerned about moving simulation platform RS to PS ample amount of time was spent to accomplish a high fidelity model which should fit into the previously developed RDT framework. At the beginning of this journey, the major demanding and stimulating job was to know and understand how the defined framework put together and individually communicate and functions. However, moving further, our learning was more intensified with studying RS and SIMIT (signal connection and mechanisms) model.

As we had worked with RS and PS before, it was comfortable to manage certain levels in model building. But, while dealing with complex situation, we had the opportunity to access Siemens learning advantage, which assisted in reaching a suitable solution. After defining all the kinematics and sensor signals to the devices in the cell, assigning the created and exported signals (about 300 I/O signals) from the SIMIT with suitable logic combined to it was found to be time-consuming, work-intensive and sensitive part of the RDT framework.

As we were asked to run emulated virtual controllers from RS platform. Getting robots up in RS and sharing the I/O signals from the cell running in PS was one of the important steps to procure in RDT framework. Thanks to VRC server platform from Siemens, which made it possible and worked as required.

Within our thesis time frame, although we have achieved high fidelity model which works as intended, executed in the manual mode. It requires still more effort, time, and resources to know operability and viability of PLC and rapid code to bring up the complete cell working in automatic mode.

## 7.4 Proposed AI application

The proposed AI implementation architecture was made based on the exploration work carried with this RDT framework and literature studies concerning PS and AI-driven models. However, for implementing the proposed architecture, many aspects are to be considered and analyzed before and after. Through the studies, the proposed architecture seems to be achievable. However, this may still require more time and effort, which was scarce in this thesis time frame, and it is out of our scope in this thesis work.

# 8

# Conclusion and Future work

*This chapter aims to answer the research questions and objectives formulated at the beginning of the thesis. Besides, future work and ethics & sustainability are also discussed.*

## 8.1 Acknowledging the Research Question

Answering the formulated RQ was one of the outcomes of this thesis project; Here, it is explicitly addressed along with the sub-questions.

**How to achieve the RS functionalities in virtual manufacturing cell by using PS platform in the context of AFRY's RDT framework for future ML integration?**

The methodology was presented with defined steps and considerations in different areas to migrate and achieve the same functionalities in PS. The methodology was implemented in an application where a complete run of the 3D Virtual process. In this way, the correct integration of the different programs, data communication, the functionality of all the components of the virtual cell, and the correct interaction of the robots between PS and RS were validated. Likewise, an exploratory study of the ways to apply AI concepts in this framework was presented.

**RQ1. What kind of issues should be considered when integrating RDT software?**

As the architecture involves more than seven applications running on the same framework, maintaining the real and synchronous data communication and data transfer among all the applications is a must. Meeting computational requirements on the system side (processor speed, RAM speed, and size, GPU speed, and size) is required for successful integration. Compatibility between the software is essential, some couplings are only available from some specific versions, and extra installation must be done (SIMIT - "ProcessSimulateCoupling"), and VCR Server implemented only runs on PS V16 and above.

**RQ2. What should be considered when moving RS platform to PS platform to achieve existing functionality of RDT?**

Concerned with this project, quite amount of time was spent on defining the kinematics for all the devices in the PS model, same as in the RS model. It was found that mapping and establishing signals between PS and SIMIT with defining proper logic's for all the devices to be most sensitive and intensive part to achieve high fidelity model. Nevertheless, dealing with robots, rapid code, and virtual controllers was also challenging. A modular approach should be considered while moving the high fidelity model from one platform to another. A virtual library that contains all the common model files to built DT could be beneficial when dealing with projects with different expectations.

**RQ3. What are the strengths and weaknesses of different simulation software (ABB RobotStudio and Process Simulate) from the framework of a RDT development?**

On the side of OLP, ABB has notorious advantages since the virtual controllers are inbuilt in the software since it is a software dedicated to ABB robots, so a more realistic behavior and emulated interfaces are provided. However, this gap has bridged thanks to VRC connections to PS, having the capability to run and connect multiple instances of the virtual controller software. RS is limited only to ABB robots, while PS is a software that can interact with other brands (Cloos, Comau, Denso, Fanuc, Kawasaki, Kuka, Mitsubishi, Staubil, Universal Robots, Yaskawa, among others.) through VRC or RCS. On the side of virtual simulations, different strengths and weaknesses were found; These are better described and compared in section 6.5.

**RQ4. In which way elements of AI can be integrated with RDT model to optimize or accelerate a production process?**

In the results under section 6.4, we have proposed the AI application for RDT framework mainly focusing on optimizing material flow at the process simulation level. However, this work was kept more on the theoretical side rather than practical implementation due to the constrained time frame in this project. Moreover, it was found that the application of AI can be applied in many other areas within RDT framework; still, more knowledge and effort should be needed to endeavor better outcomes.

## 8.2 Final statements

While working with the RDT framework following regards are put together in this section.

- Before building a RDT model, it should be scoped and narrowed according to the purpose; otherwise, it will be ambiguous.

- DT with AI can have lots of benefits in the manufacturing-based sector. However, most of these techniques are still on the theoretical side.
- For achieving the successful RDT framework, a multi-level of competence is required. Continuously improving RDT model may require an adaptive platform and knowledge to maintain it for a longer lifetime.
- Using standard dynamic libraries that share common and frequently used data on different projects may cut down the time for building RDT models.

## 8.3 Future work

Tecnomatix integration: Integrate and validate the operation of the current model by eliminating ABB VRC Server and farseeing the possibility to connect directly with ABB virtual controllers when the technology is available; this will no longer require RS application to run additionally including all the resources in the cell. When running all the software, this will enable better computing power by minimizing the load on CPU and GPU's in the system.

Mode of operation: Executing the presented PS model in automatic mode with RDT framework can further be validated based on the PLC coded functionality, improved, and adjusted. So then, it can be used for future analysis, information gathering, and optimization.

Computing power: As running all the software might be more resource-intensive, an investigation can be made by comparing both platforms from the computational perspective and obtaining information on the performance and computational resources needed.

Embedding physics: Dealing with physics machines inside the simulation model will make the RDT framework more inclined towards the realistic behavioral cell. Both software have the capability to add physics (each one to a different extend), but also Mechatronics Concept Designer (MCD) from NX can be integrated into this project; this module has physics-based capabilities that can help to elevate the fidelity of the RDT on a higher scale.

Integrating AI elements: Moreover, the different levels of AI implementation were explored and commented on (mentioned theoretically) in this thesis. Deploying the proposed AI technique in the RDT framework may open up new ways to achieve process optimization in the field of robotics.

Virtual reality: Stepping into the virtual cell makes the one step closer to interacting with all the models present in the simulation. It can be used for further verification of the built simulation model in PS environment. This can help the organization foresee the occurring problems in the cell and improve the process planning before making big decisions, hence saving time and money. PS has an inbuilt virtual reality (VR) functionality that can be integrated with different VR system providers (Oculus).

## 8.4 Ethics and sustainability

The built DT in this thesis with the previously developed PLC program, HMI, and SIMIT logics are all based on AFRY's framework concerning their customer's requirements. Data connected around the DT system is managed and stored ethically between their clients, from the stage of development to its life cycle. Using this model with emulated signals can be beneficial for the companies in decision-making and judging different case scenarios like reducing the machine downtime, optimizing the cycle time, and minimizing energy consumption.

On the sustainability side, virtual twin models positively impact the environment as companies will not require any physical resources to be tested repeatedly to reach better results of the process or operation. Using this kind of model, companies can foresee the occurring faults, errors, or outbreak downs well before installing the factory. Hence, reducing the risks with no stress and money wastage. Along with this company can benefit from the improved manufacturing quality of the product with a shorter time to market. Nevertheless, all these virtually built models can be reused in different projects, thus moving towards a circular economy. The detailed profitable outcomes are mentioned in section 3.1.3.

# Bibliography

[1] J. Antunes, A. Pinto, P. Nogueira Reis, and C. Henriques, "Industry 4.0: A challenge of competition," *Millenium - Journal of Education, Technologies, and Health*, pp. 89–97, May 30, 2018. DOI: `10.29352/mill0206.08.00159`.

[2] M. Rachinger, R. Rauter, C. Müller, W. Vorraber, and E. Schirgi, "Digitalization and its influence on business model innovation," *Journal of Manufacturing Technology Management*, vol. 30, no. 8, pp. 1143–1160, Jan. 1, 2019, Publisher: Emerald Publishing Limited, ISSN: 1741-038X. DOI: `10.1108/JMTM-01-2018-0020`. [Online]. Available: `https://doi.org/10.1108/JMTM-01-2018-0020` (visited on 02/15/2021).

[3] Siemens AG, "Siemens extends digital enterprise for greater efficiency and flexibility in industry," Karlsruhe, Germany, Press release PR2017030199COEN, Mar. 14, 2017. [Online]. Available: `https://press.siemens.com/global/en/pressrelease/siemens-extends-digital-enterprise-greater-efficiency-and-flexibility-industry` (visited on 02/15/2021).

[4] Essentra Components, *How industry 4.0 is changing business models | global manufacturer & distributor of component solutions*, Global Manufacturer & Distributor of Component Solutions — Essentra Components, Sep. 10, 2019. [Online]. Available: `https://www.essentracomponents.com/en-gb/news/news-articles/how-industry-40-is-changing-business-models` (visited on 04/20/2021).

[5] C. F. Rocha, D. F. Mamédio, and C. O. Quandt, "Startups and the innovation ecosystem in industry 4.0," *Technology Analysis & Strategic Management*, vol. 31, no. 12, pp. 1474–1487, Dec. 2, 2019, Publisher: Routledge _eprint: https://doi.org/10.1080/09537325.2019.1628938, ISSN: 0953-7325. DOI: `10.1080/09537325.2019.1628938`. [Online]. Available: `https://doi.org/10.1080/09537325.2019.1628938` (visited on 05/25/2021).

[6] G. Shao and M. Helu, "Framework for a digital twin in manufacturing: Scope and requirements," *Manufacturing Letters*, vol. 24, pp. 105–107, 2020, ISSN: 2213-8463. DOI: `https://doi.org/10.1016/j.mfglet.2020.04.004`. [Online]. Available: `https://www.sciencedirect.com/science/article/pii/S2213846319301312`.

[7] M. Boerjesson, P. Hayes, E. Mirstam, E. Nordmark, M. Oppelt, M. Pohl, J. Shwaab, P. K. Andersson, A. Buhlin, and G. Haider, "Value creation with plant modelling and simulation - AFRY and Siemens showcase real digital twins," Siemens and AFRY, White paper 1, Oct. 2020, p. 18. [Online]. Available: `https://afry.com/sites/default/files/2020-11/siemens_afry_-`

`_value_creation_with_plant_modelling_and_simulation.pdf` (visited on 12/19/2020).

[8] F. Tao, H. Zhang, A. Liu, and A. Y. C. Nee, "Digital twin in industry: State-of-the-art," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 4, pp. 2405–2415, Apr. 2019, Conference Name: IEEE Transactions on Industrial Informatics, ISSN: 1941-0050. DOI: `10.1109/TII.2018.2873186`.

[9] W. Kritzinger, M. Karner, G. Traar, J. Henjes, and W. Sihn, "Digital twin in manufacturing: A categorical literature review and classification," *IFAC-PapersOnLine*, 16th IFAC Symposium on Information Control Problems in Manufacturing INCOM 2018, vol. 51, no. 11, pp. 1016–1022, Jan. 1, 2018, ISSN: 2405-8963. DOI: `10.1016/j.ifacol.2018.08.474`. [Online]. Available: `https://www.sciencedirect.com/science/article/pii/S2405896318316021` (visited on 05/22/2021).

[10] G. Barbieri, A. Bertuzzi, A. Capriotti, L. Ragazzini, D. Gutierrez, E. Negri, and L. Fumagalli, "A virtual commissioning based methodology to integrate digital twins into manufacturing systems," *Production Engineering*, vol. 15, no. 3, pp. 397–412, Mar. 1, 2021, ISSN: 1863-7353. DOI: `10.1007/s11740-021-01037-3`. [Online]. Available: `https://doi.org/10.1007/s11740-021-01037-3` (visited on 04/25/2021).

[11] Q. Qi and F. Tao, "Digital twin and big data towards smart manufacturing and industry 4.0: 360 degree comparison," *IEEE Access*, vol. 6, pp. 3585–3593, 2018, Conference Name: IEEE Access, ISSN: 2169-3536. DOI: `10.1109/ACCESS.2018.2793265`.

[12] A. Martins, H. Costelha, and C. Neves, "Supporting the design, commissioning and supervision of smart factory components through their digital twin," in *2020 IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC)*, Apr. 2020, pp. 114–119. DOI: `10.1109/ICARSC49921.2020.9096072`.

[13] R. K. Phanden, P. Sharma, and A. Dubey, "A review on simulation in digital twin for aerospace, manufacturing and robotics," *Materials Today: Proceedings*, 2nd International Conference on Future Learning Aspects of Mechanical Engineering, vol. 38, pp. 174–178, Jan. 1, 2021, ISSN: 2214-7853. DOI: `10.1016/j.matpr.2020.06.446`. [Online]. Available: `https://www.sciencedirect.com/science/article/pii/S2214785320349245` (visited on 05/22/2021).

[14] K. Xia, C. Sacco, M. Kirkpatrick, C. Saidy, L. Nguyen, A. Kircaliali, and R. Harik, "A digital twin to train deep reinforcement learning agent for smart manufacturing plants: Environment, interfaces and intelligence," *Journal of Manufacturing Systems*, vol. 58, pp. 210–230, 2021, Digital Twin towards Smart Manufacturing and Industry 4.0, ISSN: 0278-6125. DOI: `https://doi.org/10.1016/j.jmsy.2020.06.012`. [Online]. Available: `https://www.sciencedirect.com/science/article/pii/S0278612520301059`.

[15] M. Ayani, M. Ganebäck, and A. H. C. Ng, "Digital twin: Applying emulation for machine reconditioning," *Procedia CIRP*, 51st CIRP Conference on Manufacturing Systems, vol. 72, pp. 243–248, Jan. 1, 2018, ISSN: 2212-8271. DOI: `10.1016/j.`

procir.2018.03.139. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S2212827118302968 (visited on 12/19/2020).

[16] L. F. McGinnis and O. Rose, "History and perspective of simulation in manufacturing," in *2017 Winter Simulation Conference (WSC)*, ISSN: 1558-4305, Dec. 2017, pp. 385–397. DOI: 10.1109/WSC.2017.8247801.

[17] S. Boschert and R. Rosen, "Digital twin—the simulation aspect," in *Mechatronic Futures: Challenges and Solutions for Mechatronic Systems and their Designers*, P. Hehenberger and D. Bradley, Eds. Cham: Springer International Publishing, 2016, pp. 59–74, ISBN: 978-3-319-32156-1. DOI: 10.1007/978-3-319-32156-1_5. [Online]. Available: https://doi.org/10.1007/978-3-319-32156-1_5.

[18] A. Albo and P. Falkman, "A standardization approach to virtual commissioning strategies in complex production environments," *Procedia Manufacturing*, vol. 51, pp. 1251–1258, 2020, 30th International Conference on Flexible Automation and Intelligent Manufacturing (FAIM2021), ISSN: 2351-9789. DOI: https://doi.org/10.1016/j.promfg.2020.10.175. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2351978920320321.

[19] R. A. P. Vitalli, *Proposed virtual commissioning of robotic cells based on the context of industry 4.0*, Jul. 2020. [Online]. Available: https://zenodo.org/record/3940139#.YMKopUyxWUk.

[20] Y. Xu, Y. Sun, X. Liu, and Y. Wei, "A digital-twin-assisted fault diagnosis using deep transfer learning," *IEEE Access*, vol. 7, pp. 19 990–19 999, 2019, Conference Name: IEEE Access, ISSN: 2169-3536. DOI: 10.1109/ACCESS.2018.2890566.

[21] M. P. Acharya and M. J. Mousavi. (Feb. 2019). "Real-time AI powered by edge-deployed digital twins | ABB," News, [Online]. Available: https://new.abb.com/news/detail/24412/real-time-ai-powered-by-edge-deployed-digital-twins (visited on 04/22/2021).

[22] Z. Mann, A. Harounian, and S. Zuler. (Jun. 12, 2020). "Part 1: Commissioning a vision-based supervised learning solution," Tecnomatix - How Digital Twins Help Scale Up Industrial Robotics AI, [Online]. Available: https://blogs.sw.siemens.com/tecnomatix/how-digital-twins-help-scale-up-industrial-robotics-ai/ (visited on 01/22/2021).

[23] Z. Mann, A. Harounian, and S. Zuler. (Jun. 15, 2020). "Part 2: Training a robotic task with reinforcement learning," Tecnomatix - How Digital Twins Help Scale Up Industrial Robotics AI, [Online]. Available: https://blogs.sw.siemens.com/tecnomatix/part-2-how-digital-twins-help-scale-up-industrial-robotics-ai/ (visited on 01/22/2021).

[24] S. Tay, L. Te Chuan, A. Aziati, and A. N. A. Ahmad, "An overview of industry 4.0: Definition, components, and government initiatives," *Journal of Advanced Research in Dynamical and Control Systems*, vol. 10, p. 14, Dec. 2018.

[25] E. Negri, L. Fumagalli, and M. Macchi, "A review of the roles of digital twin in CPS-based production systems," *Procedia Manufacturing*, 27th International Conference on Flexible Automation and Intelligent Manufacturing, FAIM2017, 27-30 June 2017, Modena, Italy, vol. 11, pp. 939–948, Jan. 1, 2017, ISSN: 2351-9789. DOI: 10.1016/j.promfg.2017.07.198. [Online]. Available: https://

www.sciencedirect.com/science/article/pii/S2351978917304067 (visited on 04/25/2021).

[26] Deloitte University Press, “Industry 4.0 and the digital twin,” *Deloitte Insights*, [Online]. Available: https://www2.deloitte.com/us/en/insights/focus/industry-4-0/digital-twin-technology-smart-factory.html (visited on 03/22/2021).

[27] Siemens. (). “Digital twin | siemens,” Siemens Digital Industries Software, [Online]. Available: https://www.plm.automation.siemens.com/global/en/our-story/glossary/digital-twin/24465 (visited on 03/28/2021).

[28] G. Angarano, “Gemelos digitales: Un paso más allá de la simulación,” *Automática e Instrumentación*, vol. 517, pp. 64–67, Feb. 2020. [Online]. Available: http://www.automaticaeinstrumentacion.com/es/notices/2020/04/emulate3d-gemelos-digitales-un-paso-mas-alla-de-la-simulacion-46390.php (visited on 12/20/2020).

[29] M. M. Armstrong, *Cheat sheet: What is digital twin? internet of things blog*, Aug. 2020. [Online]. Available: https://www.ibm.com/blogs/internet-of-things/iot-cheat-sheet-digital-twin/.

[30] *Gartner survey reveals digital twins are entering mainstream use*, Feb. 2019. [Online]. Available: https://www.gartner.com/en/newsroom/press-releases/2019-02-20-gartner-survey-reveals-digital-twins-are-entering-mai.

[31] A. Rasheed, O. San, and T. Kvamsdal, “Digital twin: Values, challenges and enablers from a modeling perspective,” *IEEE Access*, vol. 8, pp. 21 980–22 012, 2020. DOI: 10.1109/ACCESS.2020.2970143.

[32] Siemens AG. (). “Digitalization in industry: Twins with potential,” The Digital Twin - Industry - Siemens Global, [Online]. Available: https://new.siemens.com/global/en/company/stories/industry/the-digital-twin.html (visited on 01/04/2021).

[33] V. D. Ingenieure, *Virtual commissioning - Model types and glosary*, ser. VDI/VDE 3693 Part 1. Aug. 2016. [Online]. Available: https://www.vdi.de/richtlinien/details/vdivde-3693-blatt-1-virtuelle-inbetriebnahme-modellarten-und-glossar (visited on 05/21/2021).

[34] G. Shao, S. Jain, C. Laroque, L. H. Lee, P. Lendermann, and O. Rose, “Digital twin for smart manufacturing: The simulation aspect,” in *2019 Winter Simulation Conference (WSC)*, 2019, pp. 2085–2098. DOI: 10.1109/WSC40007.2019.9004659.

[35] P. Hoffmann, “On virtual commissioning of manufacturing systems - proposals for a systematic vc simulation study methodology and a new simulation model building approach,” Ph.D. dissertation, Aug. 2016.

[36] R. Shannon, “Introduction to the art and science of simulation,” in *1998 Winter Simulation Conference. Proceedings (Cat. No.98CH36274)*, vol. 1, 1998, 7–14 vol.1. DOI: 10.1109/WSC.1998.744892.

[37] R. E. Shannon, “Simulation modeling and methodology,” *SIGSIM Simul. Dig.*, vol. 8, no. 3, pp. 33–38, Apr. 1977, ISSN: 0163-6103. DOI: 10.1145/1102766.1102770. [Online]. Available: https://doi.org/10.1145/1102766.1102770.

[38] J. Banks, “Introduction to simulation,” in *2000 Winter Simulation Conference Proceedings (Cat. No.00CH37165)*, vol. 1, 2000, 9–16 vol.1. DOI: `10.1109/WSC.2000.899690`.

[39] G. S. Fishman, *Discrete-event simulation.* [Online]. Available: `https://books.google.se/books?hl=en&amp;lr=&amp;id=y5fuBwAAQBAJ&amp;oi=fnd&amp;pg=PR7&amp;dq=discrete%5C%2Bevent%5C%2Bsimulation&amp;ots=xvuMfJCKz3&amp;sig=IpkgLP1ywfhWJB4TbUNc8Mc8wOs&amp;redir_esc=y#v=onepage&amp;q=discrete%5C%20event%5C%20simulation&amp;f=false`.

[40] KB, *What is emulation.* [Online]. Available: `https://www.kb.nl/en/organisation/research-expertise/research-on-digitisation-and-digital-preservation/emulation/what-is-emulation`.

[41] O. M. Ülgen and E. J. Williams, *Simulation methodology, tools, and applications*, Jan. 1970. [Online]. Available: `https://www.accessengineeringlibrary.com/content/book/9780070411029/chapter/chapter90`.

[42] H. Shokry and M. Hinchey, “Model-based verification of embedded software,” *Computer*, vol. 42, no. 4, pp. 53–59, 2009. DOI: `10.1109/MC.2009.125`.

[43] IEEE, “IEEE standard for system and software verification and validation,” *IEEE Std 1012-2012 (Revision of IEEE Std 1012-2004)*, pp. 1–223, 2012. DOI: `10.1109/IEEESTD.2012.6204026`.

[44] R. Wischnewski and E. Freund, “Cosimir transport: Modeling, simulation and emulation of modular carrier based transport systems,” in *IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA ’04. 2004*, vol. 5, 2004, pp. 5171–5176. DOI: `10.1109/ROBOT.2004.1302538`.

[45] A. A. Abdul Rahman and N. Mohamad, “Software-in-the-loop technique: An approach to support reconfiguration of manufacturing system,” vol. 11, pp. 9789–9795, Aug. 2016. [Online]. Available: `https://www.researchgate.net/publication/309119269_Software-in-the-Loop_technique_An_approach_to_support_reconfiguration_of_manufacturing_system`.

[46] F. Gu, W. S. Harrison, D. M. Tilbury, and C. Yuan, “Hardware-in-the-loop for manufacturing automation control: Current status and identified needs,” in *2007 IEEE International Conference on Automation Science and Engineering*, 2007, pp. 1105–1110. DOI: `10.1109/COASE.2007.4341787`.

[47] ProgrammerSought. (). “PLCSIM, SIMIT, amesim, NX MCD, process simulate, plant simulation,” Programmer Sought, the best programmer technical posts sharing site. [Online]. Available: `https://www.programmersought.com/article/11744192131/` (visited on 03/25/2021).

[48] Germanedge. (2021). “What is OPC DA (OPC data access)?” Germanedge, [Online]. Available: `https://www.germanedge.com/en/glossary/opc-da-opc-data-access/` (visited on 04/23/2021).

[49] The OPC Foundation. (). “OPC technologies,” OPC Foundation - The Industrial Interoperability Standard, [Online]. Available: `https://opcfoundation.org/about/opc-technologies/opc-ua/` (visited on 04/23/2021).

[50] P. Valduriez, “Shared-memory architecture,” in *Encyclopedia of Database Systems*, L. LIU and M. T. ÖZSU, Eds., Boston, MA: Springer US, 2009, pp. 2638–2638,

ISBN: 978-0-387-39940-9. DOI: `10.1007/978-0-387-39940-9_1082`. [Online]. Available: `https://doi.org/10.1007/978-0-387-39940-9_1082` (visited on 04/23/2021).

[51] ABB, *Robotstudio: Abb robotics.* [Online]. Available: `https://new.abb.com/products/robotics/robotstudio`.

[52] ABB AB Robotics, *Robotware*, 2014. [Online]. Available: `https://library.e.abb.com/public/70ed3b221a577355c1257db000533446/ROB0296ENrobotware6.pdf`.

[53] Siemens PLM. (). "Manufacturing process verification in powerful 3d environment," Process Simulate - Siemens PLM, [Online]. Available: `https://www.plm.automation.siemens.com/en_gb/Images/7457_tcm642-80351.pdf` (visited on 04/08/2021).

[54] [Online]. Available: `https://docs.sw.siemens.com/en-US/release/288782031/Tecnomatix%5C%2016.1?releasesku=PL20201130102000506`.

[55] F. IPK. (2019). "Realistic robot simulation (RRS)," [Online]. Available: `http://www.realistic-robot-simulation.org/` (visited on 04/26/2021).

[56] R. Bernhardt and C. Willnow, "COST REDUCTION VIA ACCURATE CONTROLLER SIMULATION," *IFAC Proceedings Volumes*, 15th IFAC World Congress, vol. 35, no. 1, pp. 379–384, Jan. 1, 2002, ISSN: 1474-6670. DOI: `10.3182/20020721-6-ES-1901.00065`. [Online]. Available: `https://www.sciencedirect.com/science/article/pii/S147466701538486X` (visited on 05/26/2021).

[57] Siemens, "Simit simulation platform, real-time simulation – a foundation for your digitalization strategy," 2019. [Online]. Available: `https://assets.new.siemens.com/siemens/assets/api/uuid:4ce72796-3757-493c-8afe-e1fefb988025/simitsimulationplatformdatasheet2019-1en-144.pdf`.

[58] Siemens AG, *Totally integrated automation from siemens integration.* [Online]. Available: `https://new.siemens.com/global/en/products/automation/topic-areas/tia.html`.

[59] Siemens. (Sep. 2016). "S7-PLCSIM advanced - function manual," S7-PLCSIM Advanced - Siemens Industry Online Support, [Online]. Available: `https://cache.industry.siemens.com/dl/files/153/109739153/att_895955/v1/s7-plcsim_advanced_function_manual_en-US_en-US.pdf` (visited on 04/08/2021).

[60] T. Durand and M. Hattingh, "Data mining and artificial intelligence techniques used to extract big data patterns," in *2020 2nd International Multidisciplinary Information Technology and Engineering Conference (IMITEC)*, 2020, pp. 1–8. DOI: `10.1109/IMITEC50163.2020.9334069`.

[61] O. Shani, *Automation and ai: What you need to know.* [Online]. Available: `https://www.marketingaiinstitute.com/blog/automation-and-ai-what-you-need-to-know`.

[62] Y. Hu, W. Li, D. Wright, O. Aydin, D. Wilson, O. Maher, and M. Raad, "Artificial intelligence approaches," *Geographic Information Science Technology Body of Knowledge*, vol. 2019, Jul. 2019. DOI: `10.22224/gistbok/2019.3.4`.

[63] R. Goel and P. Gupta, "Robotics and industry 4.0," in *A Roadmap to Industry 4.0: Smart Production, Sharp Business and Sustainable Development*, A. Nayyar and A. Kumar, Eds. Cham: Springer International Publishing, 2020, pp. 157–169, ISBN: 978-3-030-14544-6. DOI: `10.1007/978-3-030-14544-6_9`. [Online]. Available: `https://doi.org/10.1007/978-3-030-14544-6_9`.

[64] M. M. Rathore, S. A. Shah, D. Shukla, E. Bentafat, and S. Bakiras, "The role of ai, machine learning, and big data in digital twinning: A systematic literature review, challenges, and opportunities," *IEEE Access*, vol. 9, pp. 32 030–32 052, 2021. DOI: `10.1109/ACCESS.2021.3060863`.

[65] Siemens, *Intelligent automation redefined.* [Online]. Available: `https://new.siemens.com/in/en/products/automation/systems/industrial/io-systems/artificial-intelligence.html`.

[66] M. El-Shamouty, K. Kleeberger, A. Lämmle, and M. Huber, "Simulation-driven machine learning for robotics and automation," *tm - Technisches Messen*, vol. 86, no. 11, pp. 673–684, 2019. DOI: `doi:10.1515/teme-2019-0072`. [Online]. Available: `https://doi.org/10.1515/teme-2019-0072`.

[67] E. J. Tuegel, A. R. Ingraffea, T. G. Eason, and S. M. Spottswood, *Reengineering aircraft structural life prediction using a digital twin*, Oct. 2011. [Online]. Available: `https://www.hindawi.com/journals/ijae/2011/154798/`.

[68] O. Shani, *Automation and ai: What you need to know.* [Online]. Available: `https://www.marketingaiinstitute.com/blog/automation-and-ai-what-you-need-to-know`.

[69] J. Kober, J. Bagnell, and J. Peters, "Reinforcement learning in robotics: A survey," *The International Journal of Robotics Research*, vol. 32, pp. 1238–1274, Sep. 2013. DOI: `10.1177/0278364913495721`.

**Department of Electrical Engineering**
**CHALMERS UNIVERSITY OF TECHNOLOGY**
Gothenburg, Sweden
www.chalmers.se

CHALMERS
UNIVERSITY OF TECHNOLOGY