Tampere University

Syed Manzar Abbas Kazmi

METHODOLOGY FOR VALIDATING MECHATRONIC DIGITAL TWIN

Automation Engineering
Master's Thesis
May 2019

ABSTRACT

Syed Manzar Abbas Kazmi: Methodology for Validating Mechatronic Digital Twin
Master's Thesis
Tampere University
Automation Engineering
May 2019

The market place is changing rapidly along with the increasing requirements of the clients. To meet up these challenges, there is need for new and efficient methods for the identification and visualization of the problem. Additionally, these methods should allow users in designing and finding alternative solutions to achieve the desired working. In this thesis, studies related to mechatronic model was performed and an approach was presented for the validation of mechatronic digital twin. The concept of digital twin is one of the core concepts of modern industrial revolution. Digital twin can be defined as the digital representation of a physical system, which behaves exactly as actual hardware.

A mechatronic digital twin of Festo MPS 500 system was modeled using Siemens NX Mechatronics Concept Designer. The methodology was implemented which involved storing the process parameters of the operation of actual hardware. The stored information was passed to the mechatronic model for verification and validation purpose. Several engineering tools were used for the implementation of the system. These tools were integrated with each other to provide the proof of concept of the methodology. The developed approach can be used with the mechatronic models of existing systems.

This enables the user to test and observe different scenarios and alternative solutions in the mechatronic model before implementing it to actual hardware. The proposed methodology can be used for the troubleshooting purpose by re-playing the stored data of the operation of actual hardware in the mechatronic model. By this way, user can visualize the whole operation and identify the problem easily.

Keywords: Digital Simulation, Mechatronics, Programmable Logic Controller

The originality of this thesis has been checked using the Turnitin Originality Check service

PREFACE

بِسْمِ اللهِ الرَّحْمٰنِ الرَّحِيْمِ

اللهم صل على محمد وآل محمد وعجل فرجهم

علی علیہ السلام امام من است و منم غلام علی علیہ السلام

I would like to take this opportunity to thank those people who have been motivating me since I started this master’s program. First, I am deeply thankful to my family who have been there for all the difficult times. The prayers of my father Syed Mazloom Hussain Kazmi and motivational words of my brother Syed Danyal Abbas Kazmi has always encouraged me to push my limits.

I am greatly thankful to Professor Jose L. Martinez Lastra for giving me an opportunity to work under his supervision. I would also like to thank Mr. Wael Mohammed for the supervision, despite his busy schedule. I am also grateful to Dr. Borja Ramis for the continuous support in this journey.

I would also like to extend my gratitude to all my friends especially Ali Hussnain, Adnan Ali, Shah, Jawad, Jazib, Hamza, Daniyal, Waqar, Mohsin, Aqib and Usman Ahmed. They have always helped me and supported me to achieve my goals.

In the end, I would like to thank the Tampere University for the master’s program. This program gave me the necessary exposure and skills needed in life.

Thank you everyone for the kind support.

Tampere, 17 May 2019

Syed Manzar Abbas Kazmi

CONTENTS

LIST OF FIGURES

LIST OF SYMBOLS AND ABBREVIATIONS

VR	Virtual Reality
HIL	Hardware in loop
SIL	Software in loop
V & V	Verification and Validation
MCD	Mechatronics Concept Designer
TIA	Totally Integrated Automation
OPC	Open Platform Communication

1. INTRODUCTION

This chapter starts with the background of the problem, which needs to be explored. It is about defining goals and some research questions that need to be answered. Few limitations and challenges that can be faced while solving the problem are also defined. In last, document structure has been stated for the understanding of the reader.

1.1 Background

As the market place is continuously changing, introducing more and more competition for the manufacturing companies than ever. In this era of technological advancement, companies are spending on research and development more than ever so they could stay in the competition. As the money spent in research and development can bring economic prosperity [1]. By staying in the competition means providing customers with best services in the shortest time, with a limited amount of budget.

If we analyze the case of manufacturing industries, they cannot afford the mistakes especially in the design phase of product. It can cost them a lot of money, in terms of delaying to deliver the product, re-designing and damage to the reputation of a firm [2]. The concept of digital twin can be used by the manufacturing industries to cope up with these problems. The digital representation of the physical product can be used to test different solutions of client's problem without using hardware. Moreover, these models can be integrated with the programming software for the validation, which can reduce the risk of errors. As this is an era of industry 4.0, many are already adopting the concept of digital twin and virtual commissioning to provide their clients with best possible solutions like Volvo, GE, Siemens, Alfa Laval [3, 4].

1.2 Problem Statement

Traditionally, when a problem is faced in a manufacturing process, conventional methods are used for observing and troubleshooting purpose e.g. checking control logic of the process and trends of control signals. With the technological evolution, there is need for more efficient method for the identification of the problem so that troubleshooting could be done easily.

Secondly, manufacturing industries usually build and assemble the same type of products. This type of arrangement is feasible if there are no changes in the design. When there are some variations required, then it’s a challenge for manufacturers. The changes should integrate with existing design along with better efficiency and standards. The mistakes in the design phase can cost the manufacturers a lot of money and time, which cannot be wasted while the competition is so fierce. This highlights the need for a method to verify the designed model.

As the industry 4.0 revolution is going on, manufacturers tend to invest less and make more profit, by utilizing the latest technologies present in the market. They do not have to assemble hardware for their initial design check and validation. Instead, this can be achieved by creating the digital twin of the required product, which enable the manufacturers to test alternative solutions for a problem. By this, flaws in the design can be detected in the early stages, which leads to a higher degree of accuracy, product of better quality and ultimately customer satisfaction [5]. This gives manufacturers freedom of testing a plethora of ideas, which cannot be implemented or probably risky in hardware scenario [6].

1.3 Objective

The objective of this thesis is to create mechatronic model of distribution and testing stations of FESTO MPS 500 system, which are being controlled by Siemens ET200SP and proposing a methodology for the verification of the created mechatronic model. It will help to clarify the concept of digital twin. The goals of the thesis work include:

- Designing of mechatronic model of the Festo MPS 500 stations
- Development of the control program for distribution and testing stations in TIA Portal
- Storing the data of actual hardware process signals in database
- Establishing communication between control program and mechatronic model
- Passing the logged data of actual hardware to mechatronic model for verification and validation purpose

1.4 Limitations and Challenges

There are few challenges that can be faced in the implementation of thesis including the selection of the software to successfully create the mechatronic model of stations of Festo MPS 500 system. Second problem that can be faced is mechanical modelling and designing, which requires a lot of time and effort to make it accurate as of the actual hardware. The next challenge could be to establish a communication link between control program and mechatronic model along with logging and retrieving data from the database.

1.5 Thesis Structure

The documentation of this thesis has been distributed in six chapters. Second chapter is to discuss the research and concepts for better understanding of the topic. The research methodology adopted for achieving the defined goals and implementation of the system is discussed in third chapter. Fourth chapter is to briefly describe the system architecture, engineering tools and their configuration for the integration purpose. Fifth chapter states the discussion and results which were gathered while the implementation of the whole system. Chapter six provides a summary of whole thesis and future work.

2. THEORETICAL BACKGROUND

In this Chapter, digital twin and general concepts related to the thesis topic are discussed. Afterward, simulation and its methods for verification and validation are also briefly introduced along with the latest tools for 3D simulation. Further, methodologies for testing control programs are also described. An overview of open platform communication (OPC) is also provided in the end.

With the passage of time, there were many technological advances which paved the way for the industries to be more efficient and productive. These technological advancements lead to the four industrial evolutions. These industrial evolutions and developments, which started at the end of the 18th century are shown in figure 1. First evolution is related to mechanical manufacturing facilities which were powered by steam engines. With the advent of electricity in the twentieth century, steam-powered industries were revolutionized. Second evolution resulted in giving new vision to investors because it enabled them to do the mass production. Electromechanical systems dramatically enhanced the capabilities of existing systems, which lead to shape the industries for a better and profitable future.

In 1970, innovations in the field of Information technology enabled automation in the industries, which started the third evolution. It permitted the machines to sense the environment around and taking decisions based on readings and results [7]. With all these industrial evolutions and breakthroughs, still there was need of more innovations as limited resources and constantly changing demand of the customers made a challenging situation for the industries. All these problems lead toward a new industrial revolution, which can enable the industries to satisfy the customer needs with modest means. This need for advancement from the third evolution was named as Industry 4.0.

According to Darth and Horch [8], Industry 4.0 is one of the most famous topics that were being discussed among inventors and academics in Germany. As the German federal government announced it as part of their technology strategy in 2011 [9]. The term ”Industry 4.0” became famous when the initiative ” *Industrie 4.0 – an association of representatives from business, politics, and academia*” came into effect so the German manufacturing industry could be enabled to compete effectively in the market. Industry 4.0 is the only industrial revolution which is apriori, not observed ex-post. It means it provides the opportunity for the researchers and industrialists to have a plethora of ideas, which can shape the world [10].

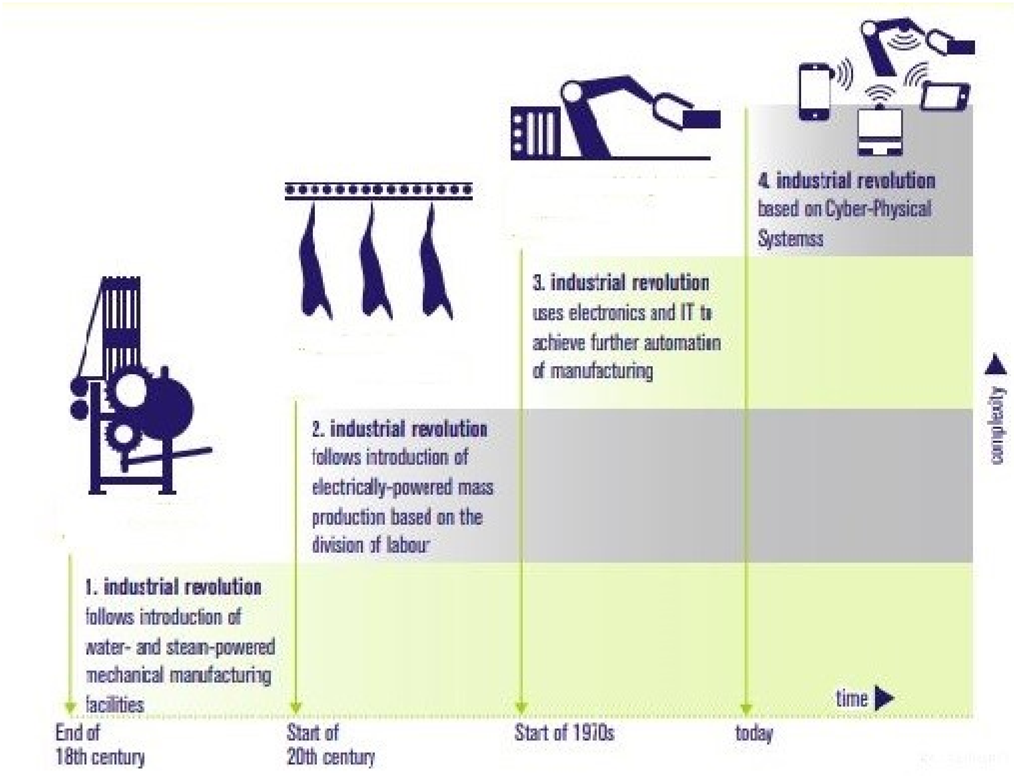


Figure 1 *Industrial revolution [10]*

Industry 4.0 has a huge impact on the economy as it can help develop new business models, services and products which could substantially increase the efficiency in operations. According to Lorenz and Gerbert, this industrial revolution has numerous benefits which could dramatically affect not only production capabilities but also revenue growth and employment opportunities. It has shown results in Germany, by soaring the production of industries by 90 to 250 billion Euros [7].

In the last few years, a lot of research has been done to clarify the Industry 4.0 revolution, its boundaries, benefits and how we can transform our present industry according to this revolution. It was still difficult to get familiar with the concepts of Industry 4.0 as the research lacks a clear definition. To get a better understanding and vision of this revolution, a report was published by the German *Industry 4.0 Working Group* in 2013, which are also the key promoters of this idea. In this report, they emphasized some key points, which could help and make the industrial 4.0 beneficial for us, to its true potential [9]. In those key points, they highlighted some specific areas in which action is needed. Some of those are:

1. There is a need for a set of standards which are common so a collaborative partnership could be possible between different companies through value network. Moreover, for the implementation of these standards, there is also a need for a reference architecture.
2. With technology advancement, we are designing more complex systems, which are capable of flexible manufacturing. To handle these sophisticated and complex systems, there is a need for proper planning and explanatory models. Professionals should be equipped with suitable tools and methods so these models can be developed.
3. One of the key requirement for Industry 4.0 is communication networks. These networks should have the ability to exchange information with a very high speed, great reliability and accuracy.
4. According to authors, Security and safety of the smart manufacturing systems is also a very critical issue. It is very important to ensure that the information and all the data related to these systems and products is well protected against unauthorized access.
5. With the fourth revolution in the industry, there will be significant changes in working practice of employees as the control will be real-time and accurate. This will impact the work content and will bring certain changes in the environment of the workplace. In this scenario, employees can get the opportunity of personal development and working on higher levels with great responsibility if socio-technical systems (STS) approach is implemented. This approach can help in recognizing the interaction between technology and employees in a working environment.
6. Authors also emphasized on the implementation of training strategies for the employees. As the Industry 4.0 will impact their jobs and will require effort to understand. There should be digital learning programs and training to cope out with these problems.
7. As there will be many new innovations with the advent of Industry 4.0, which needs to be defined and adjusted in the existing legislation. In the meantime, smart manufacturing industries and corporate networks should also comply with the law so that the problems and challenges like data protection, liability issues and trade policies can be solved.
8. Industry 4.0 enable the industries to be efficient and more productive, which means consumption of raw materials will also increase, causing a threat to the

environment. It is mandatory that tradeoff between the resources that are being used and potential savings generated are being calculated so there can be a balanced state.

The aim of this report was to divert some attention towards the adoption of new standards and requirements so the existing industries and technologies could be transformed. According to authors, Industry 4.0 is an evolutionary process and will provide opportunities for new products, markets and innovations [9]. These requirements seem important for benefiting from the ongoing industrial revolution to its full extent. Industry 4.0 is a fusion of existing technologies with modern information and communication technologies, which can help in improving the reliability of current methods.

2.1 Digital Twin and Validation Approaches

Industry 4.0 revolution can enable the industries to produce the products with great efficiency, by utilization of resources smartly using sensor technologies, integration of different systems for monitoring and controlling purpose and miniaturization. According to Haag and Anderl, all the above-mentioned factors and price decline has helped to create any product virtually. Industry 4.0 evolution has made possible for the products to sense their condition and the environment also. These systems can communicate with each other and also have the ability to process the given information. All this information exchange and communication has enabled us to create virtual copies of the whole physical systems, which is also known as digital twin [6]. There are many definitions by the researchers as stated below, in which they defined the digital twin as a virtual copy of the physical assets or systems, which can be utilized in many ways for various purposes.

The term ”Digital twin” was first used during the course of Dr. Michael Grieves, on product lifecycle management (PLM), while he was working in the University of Michigan in 2003. According to him, a virtual copy of the physical products (as shown in figure 2) was a relatively new idea at that time. The information available regarding any physical product was paper or manual based and it was quite limited. According to the author, digital twin is the virtual representation of the physical products which are already manufactured. He used it to compare the manufactured product with the conceptual design [11].

According to Rosen and his group, the concept of the digital twin was introduced first in NASA’s Apollo Program. During this program, two space vehicles were built having identical properties. The aim was to copy the exact movements of the space vehicle, which was sent in space for a mission. The space vehicle which was on earth was termed as twin and it was used to simulate the alternatives for the real one. It was also significantly used for the training of flight operations [12].

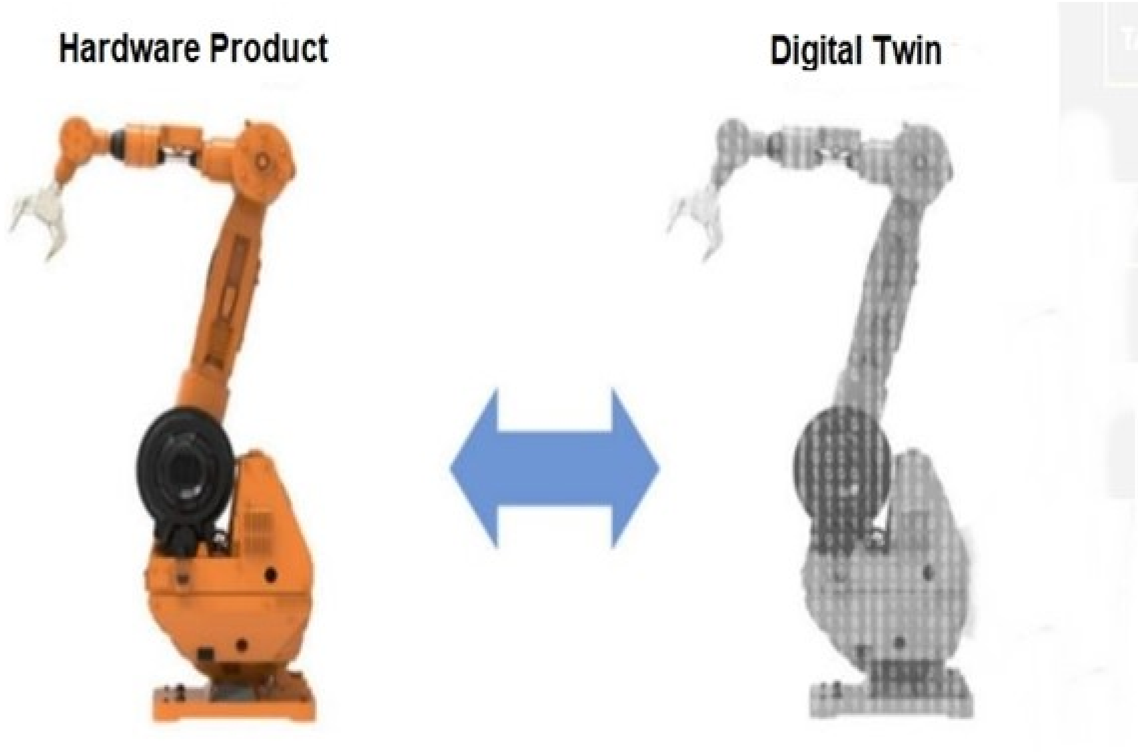


Figure 2 *Digital Twin [65]*

According to Reid and Rhodes of Massachusetts Institute of Technology, Digital twin or digital thread is defined as a model of real product having characteristics like life cycle, sensors data, physics and simulations so it could help in the designing phase and point out the defects before hardware implementation. It should have a simulation with the integration of multi-scale and multi-physics of the real product so it can successfully mimic the activities of the actual hardware By this way, It can also help in managing the product safely and effectively [13].

According to Christopher Ganz (R & D Manager, ABB), Digital twin provides the virtual image of the hardware equipment. The collected information physical models can be used with the virtual image to test the hardware in different scenarios, which can be used for better understanding. As per him, there can be many applications of digital twin [14]. Some of those applications are highlighted below:

1. Digital twin can help in the early stages of development as it enables the 3D view of the model and designers can verify if all the parts fit together. It can help them to detect the flaws in design before the actual implementation.
2. If there are simulations of the whole systems, it can help to check some constraints in advance e.g. physical connections. Digital twins of the different system can be integrated and the flow of information can be checked. It can help to compare the requirements with the expected outcomes. If the difference appears not acceptable then certain changes can be applied, which can save a lot of capital and the customer downtime will automatically reduce.
3. 3D simulations can also be used for troubleshooting purposes if used with virtual reality (VR) glasses. Simulations can be used to provide the information of parts which are difficult to access, for the technicians. It can help them to get the overall idea of the problem.
4. The data of sensors and other hardware can be logged, which can be passed through the algorithms to predict the condition of the hardware and possible failures. It can help to plan preventive maintenance, which can reduce the unplanned shutdown time.
5. While the installation of systems, if IOT devices and advanced algorithms settings are configured in the digital twin then the customer can subscribe to these services and in the ideal case there will be no need of engineering anymore.

According to the author, digital twin is like a complete digital representation of the complete hardware system or subsystem. It can greatly help in designing phase as alternative ways can be adopted and tested without any risk. The collected data can be used for future improvements as faults can be predicted, by passing the information from an algorithm [14].

Digital twin involves the modeling of the system which is under consideration for the simulation of a process. Simulation is one of the resources of the digital twin along with artificial intelligence and analytics [66]. Simulation, as defined by Banks, is the imitation of the manufacturing process over a period of time. It can be very useful to check different scenarios before actually implementing it on the factory floor thus, very helpful for the designers [27]. Shannon defines simulation as model-based experimentation and extracting information about the physical systems as a result of these experiments. The same author also explained simulation as the development and designing a model, which can be closely related to the actual system. It enables us to understand the behavior of the system in a better way [25]. Simulation works as a powerful tool for the designers and programmers of the complex manufacturing systems.

Meinert and co-authors define simulation as the best tool for examining the behavior of the system in a constantly changing environment. However, they also considered this approach as a time-consuming and iterative task. They presented a modular approach specifically for the systems used for material handling [26].

With technological advancement, competition is very fierce between the organizations. The demand of the time is flexible manufacturing systems with efficiency and quality for the complex processes. In this scenario, banks suggested to work using simulations and it can have a great impact on the whole process [27]. Now, simulation is being used in a lot of industries especially for automotive manufacturing, ship manufacturing, food industry, etc. It is making the life of the designers easy and making the whole process more reliable. Along with many advantages, there are also some disadvantages which are addressed in the next section.

2.1.1 Benefits and Drawbacks of Model Simulation

According to Thapa and co-authors, the market is very competitive. Now, It is very difficult to satisfy the customer with only textual verification. They want to see the 3D models of their requirements or usage of virtual commissioning technique because these tools make the whole manufacturing process very reliable [28]. Simulation can provide the non-technical persons, a good understanding of the whole process.

As said by Shannon, simulations can increase the reliability of the whole manufacturing process as validation of the code can be done before the commissioning of the actual system. It also enables the designers to check the unusual scenarios and to find alternative ways for accomplishing a task [29].

Banks also defined this as a useful concept for the manufacturing industries to analyze the behavior of the real systems in a number of situations without the disruption of any on-going operations. We can also get the idea of the most important aspects of the processes by using this approach [27].

One drawback that is pointed by both Shannon and Banks is the model building for the simulation as it is a very complicated task and consumes a lot of time. It requires special training for the designers to grasp the important concepts related to model building [27, 29]. Meinert also highlighted the point of designing models and making changes in the existing models using the modular approach. According to the author, it can greatly affect the model building process [26].

2.1.2 Verification and Validation

Verification and validation is an important practice for analyzing the system in the design phase or implementation phase. It provides the proof if the designers are working in the right direction. According to Andersson and Runeson, It is very important to test the system, by changing the environmental factors so the customer could know that design is as per their requirements [30].

As per authors, companies spent a large share of the budget for verifying and validating the implemented system. They conducted a survey of 11 Swedish companies to observe and study the validation methods adopted in those companies. According to authors, there is a substantial difference in methods of verification when we consider small and big firms. Big firms use commercial software's or tools. In contrast, small firms usually adopt make-in-house tools for validation. Authors suggested that there is still a need for improvement in these methods [30].

IEEE define verification and validation as a standard and all the life cycle models like hardware, software, system are compatible with it. As per IEEE, there are various useful processes that are helpful in determining if the designed system or developed product matches the customer requirements and the product itself satisfies the intended use [31].

As explained by Maropoulos and Ceglarek, verification and validation methods hold great importance in designing and implementation as they have a direct impact on product production, functionality and customer's perception about the whole system. In the process of model building of simulation, these methods verify that if the model is accurately representing the actual system and it can be used as required or as defined by the clients [32].

According to Rabe, Spieckermann and Wenzel, model or product suitability as per the requirements can be compared by using the verification and validation approaches. It can also greatly help in accurate modeling of the considered system as these methods make it possible for the designers to check their designs after every change. This helps to find alternative ways to achieve a common task hence providing options for the designers and customers to choose according to their situation. As per authors, it also helps to understand the behavior of the system better as changes can be implemented and the impact of those actions can be observed in the system [33].

William and Ülgen also explained about the verification and validation by using a similar definition as provided by other authors. They also discussed that the simulation model should have the ability to represent the actual systems effectively, which also follows the requirements provided by the end user. Authors also tried to explain their understanding using figure 3 and discussed the difference between the definition of verification and validation. The comparison between the requirements of the model or which designers intends to make and the actual designed model can be done using the verification. Authors defined the validation as an evaluation of the designed model and the actual physical system. If the difference is not big and the model is working accurately, it can be said validated [34].

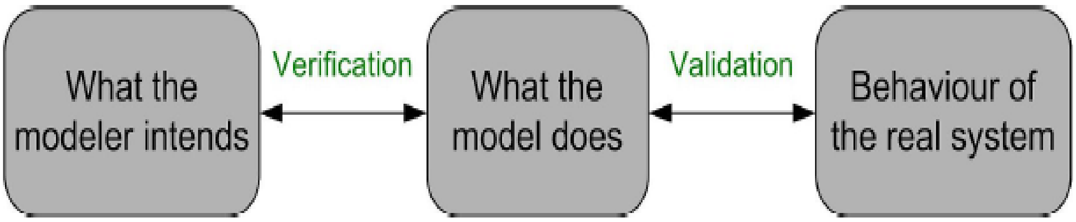


Figure 3 *Verification and Validation of Simulation Model [34]*

Verification and validation methods are not only used in the software side but they are also applicable and being used in mechanical and electrical domains as well. It can produce the results, which provide the surety that the system will work fine in the real environment also, which is the priority of both designers and customers.

There are many approaches of verification and validation, which we see mostly applicable in the field of modern model-based embedded systems. These methods which are also known as X-in-the-loop methods are ordinarily used with the electronic control units usually for software development and validation purpose. In the following subsections, some of these methods are briefly discussed.

1. Model in the loop
2. Software in the loop
3. Processor in the loop
4. Hardware in the loop

1. *Model in the loop:* This simulation approach is considered as the most commonly used method for model-based developments. It is usually implemented using graphics oriented tools like Matlab (Simulink) and It is capable of testing the control code, which is implemented using the model language (e.g. finite state machines). However, according to standard IEC 61131-3, it cannot validate the code, which is in the standardized PLC language format [35].

According to Vepsäläinen and Kuikka, there is lack of work done for the integration of simulations to the model-driven development methods which currently exists. Authors used the Unified Modeling Language (UML) Automation profile tool for testing the control system of a crane with its simulation [35]. It is one of the many applications in which Model in the Loop simulation approach was used.

2. *Software in the loop:* This simulation methodology can be defined as the use of the virtual controller, also called ”Emulator” connected with the virtual model of the actual physical system, through a communication channel (e.g. Open Platform Communication (OPC), Transmission Control Protocol (TCP/ IP), etc). It can also be explained as the integration of the virtual controller with the simulation of an actual plant. Figure 4 provided below can be very helpful for a better understanding of the concepts, related to this approach.

Software-in-the-Loop has been used for various manufacturing systems. Phillips and Montalvo used the emulation for consumer product packaging line. As per authors, emulated testing caused the decline in the startup time of packaging line by up to 50 percent, by using the virtual 3D model of the production line. By this way, control engineers were able to work with it in the early stages. Authors used the RSLogix Emulate 5000 as a virtual controller [36].

Dzinic and Yao used software-in-the-loop approach for a production cell, placed in a lab. They used a tool called ”Xcelgo Experior” for the simulation of production cell and established TCP/ IP communication with the emulator using a third party tool ”Net-to-PLCSIM” [37].

The same approach was used by the Krause, for the Liquified Natural Gas (LNG) plant in 2007. The author used the process simulator ”Sandvika” published by Kongsberg Process Simulation and emulator of AC 870P/ Melody System, known as an active part of the ABB 800xA Distributed Control System (DCS). He successfully managed to connect the simulation with more than 10,000 inputs and outputs of the LNG plant, situated in Norway [38].

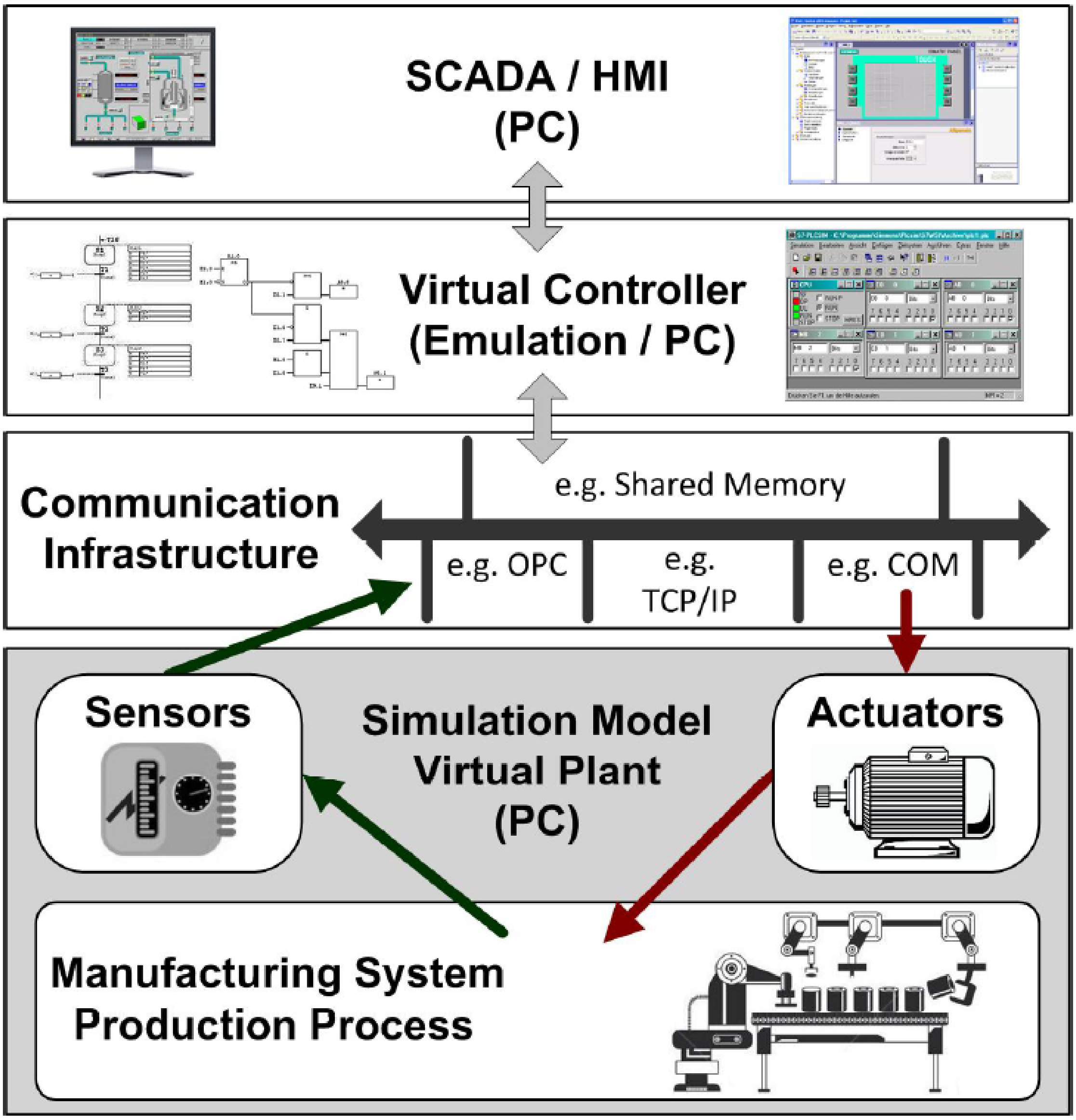


Figure 4 *Software in the loop [21]*

3. *Processor in the Loop:* According to Hoffmann, It is the approach mostly used for testing the programs of Electronic Control Units (ECU) in the automotive industry but very rarely used industrial control applications [21]. Processor-in-the-loop, also known as PIL is used to test the actual control code on the processors but not in real time. It is also capable of testing the different scenarios automatically and detect the bugs which are present in the control code.

4. *Hardware in the Loop:* In hardware in the loop simulation approach, a real controller is connected with the simulation and actual hardware, unlike software in the loop approach where the emulator is present to mimic the working of the actual controller. As shown in figure 5, some of the components of the system are simulated and some are

real hardware. This approach is very familiar in the aircraft and automotive industries as a lot of projects has been implemented using this method.

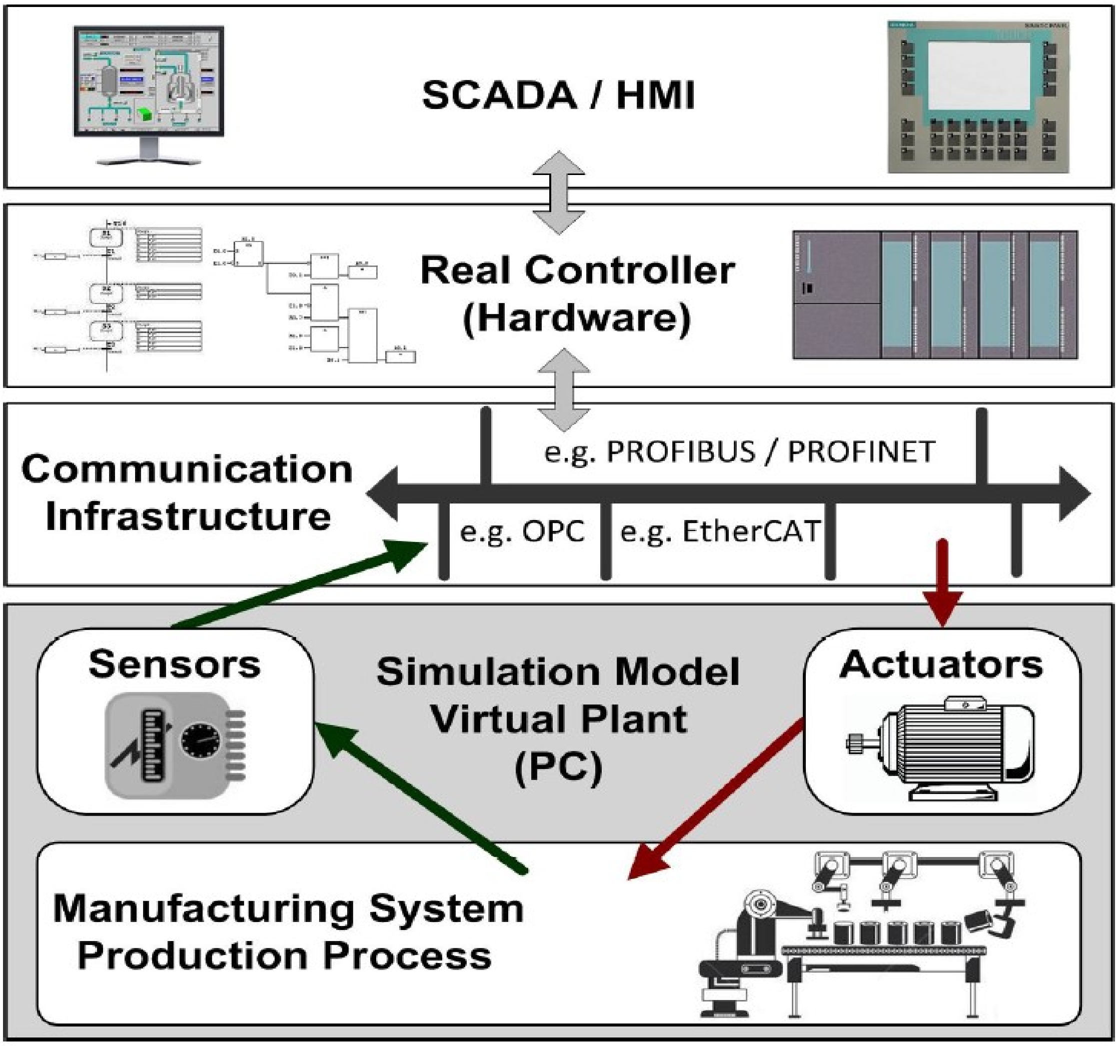


Figure 5 *Hardware in the loop [21]*

According to Isermann, Schaffnit and Sinsel, simulation approaches like hardware in the loop are adopted when some of the system components are missing or it is very expensive to experiment with the real system. There is also a problem of time. Authors used this approach for the detailed analysis of the diesel engines and for the development of new control algorithms [39].

As per M.Bacic, for designing and evaluating a control system hardware in the loop is a verified technique. It enables us to integrate real hardware (controller) with the simulation, which helps us to understand the behavior of the system in real-time. This gives important information on the designed system competence if it has the capability of delivering, in the required time. The author also highlighted that this approach has been used by the designers for more than four decades for major projects like missile guidance system, highly maneuverable aircraft technology (HiMAT), the design of anti-lock braking system (ABS) but it still lacks the formalization [40].

Figure 5 provides information for a better understanding of this approach and how it has been used for a number of projects. Dzinic and Yao tested their simulation of production cell (which was made using Experior) using both hardware in the loop and software in the loop simulation approaches. As a result of their verification, they observed similar results for their case study [37].

Hardware in the loop approach is also useful in the aspect that control programs are made and validated using the same controller, which later should be used for controlling purpose on the factory floor. It provides the basis for the integration test of the whole automation system.

2.1.3 Softwares for Modelling Systems

With the evolution of Industry 4.0, there exist a number of 3D simulation tools that can be used for modeling and replicating the actual physical system. These tools are very advanced, each having their own benefits and downsides. The name of some of these engineering tools are:

1. Kuka Sim
2. Automation Builder
3. Factory I/O
4. Siemens NX
5. Visual Components

Kuka Sim was developed on top of Visual Components platform, for the creation of 3D layouts integrated with Kuka robots as shown in figure 6. Kuka also offers Kuka work visual, which can be used for configuring, programming and diagnosis [46].

It offers the functionalities like the integration of input and output signals, conversion of geometries into kinematic systems, signal mapping for the components that need to be controlled [47]. Data can be exchanged between PLC’s from different vendors and Kuka Sim Pro Package. The downside of Kuka Sim is that it only supports the simulation of Kuka robots [21].

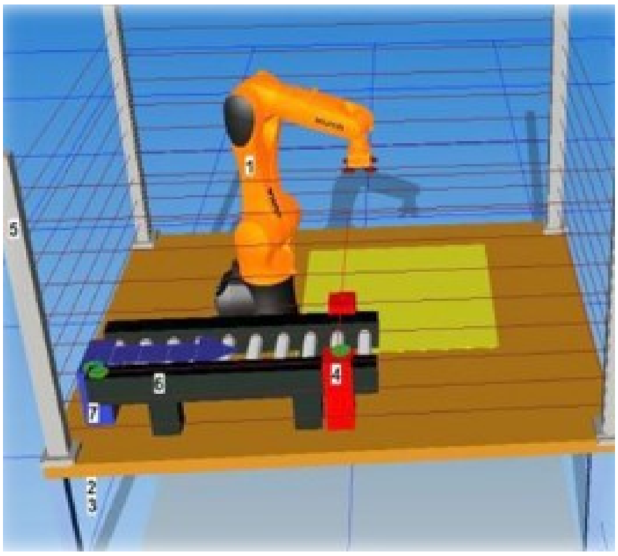


Figure 6 *Kuka Robot Simulation [51]*

Automation builder is offered by ABB, which enables the user to validate the PLC and robot programs. It can also be used for offline programming of robots. Robot movements can be simulated by using Rapid programming. Codesys has been integrated inside the software for the PLC programming. 3D models can be imported into this software and kinematics can be added to sort out the rigid bodies from static bodies [48].

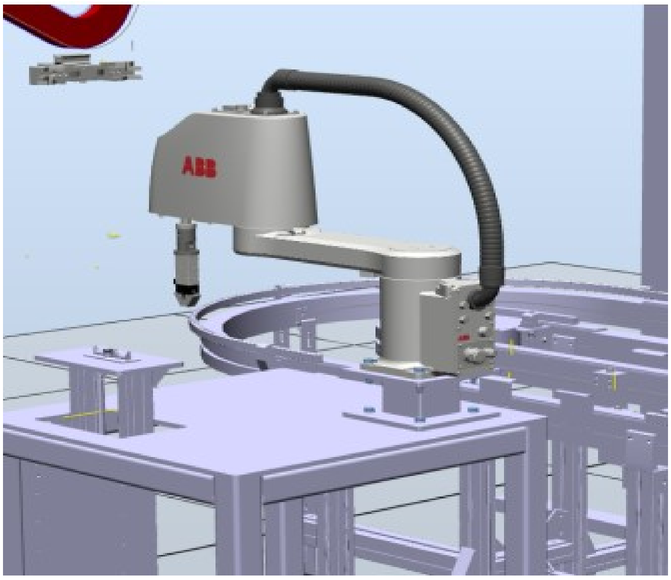


Figure 7 *ABB SCARA Robot Simulation [52]*

Factory I/O software is provided by real games, for the 3D simulation of automation technologies as demonstrated in figure 8. It contains the library of industrial components like conveyors, sensors, encoders, robots based processing stations, etc. It enables the user to connect PLC of different vendors like Siemens, Allen Bradley, Mitsubishi etc with the industrial components and validate the code.

Additionally, microcontrollers, Modbus and SoftPLC can also be used for controlling the components. The main drawback of this software is new models cannot be imported and only the components present in the library can be used for code verification and validation [49].

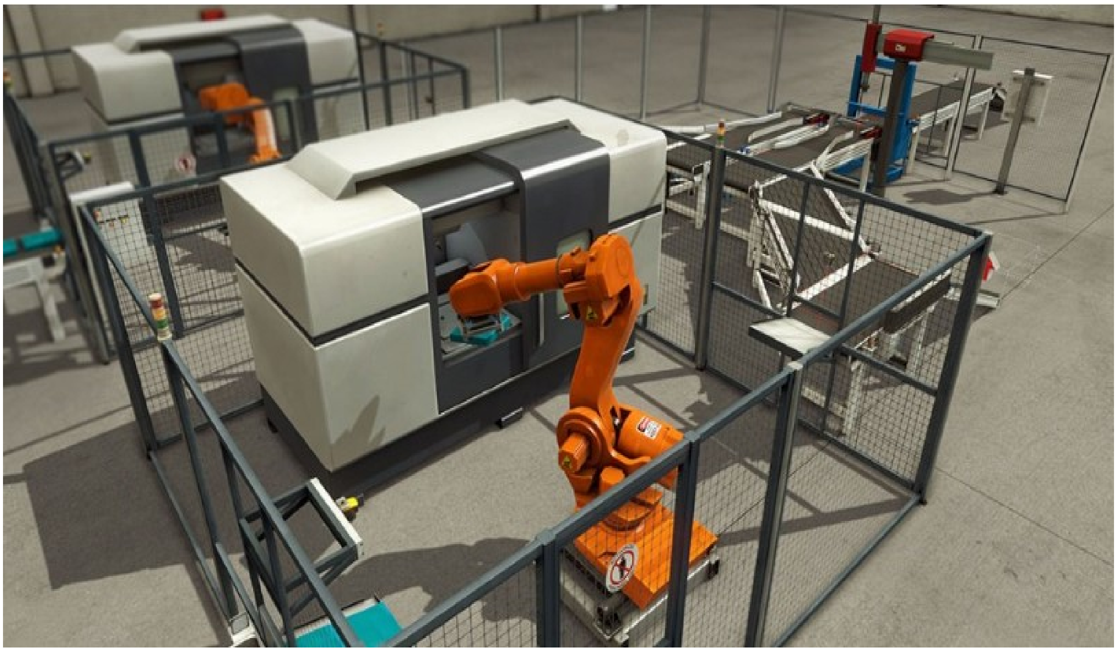


Figure 8 *Robot based Processing Station Factory I/O [49]*

Siemens NX enables the user to design and simulate the required designs. Siemens has incorporated the modeling, design and manufacturing features in one software, which is easy for the user rather than installing all the softwares individually. For the simulation purposes, Siemens offers Mechatronics Concept Designer (MCD). It has the capabilities of creating a virtual environment and offline programming using ”Operations” prior to PLC programming for verifying the accurate working of the model under consideration. This tool can be used by the designers to check the conceptual designs and checking the flaws in those designs. Similarly, alternative ways can be tested for a single goal, according to the customer requirement.

3D models made in other software’s can be imported in Siemens NX and further Kinematics behavior can be added for different parts of the imported or designed model as shown in figure 9. PLC programs can also be tested and validated with the mechatronic model using industrial communication protocols including OPC. Siemens MCD is often used for testing ideas and hence saves the cost of the hardware. There is also NVIDIA PhysX engine integrated into this software for testing the behavior of the model when certain physical forces are acting on it like gravitational force, friction etc. As per Siemens, this software can be used by mechanical, electrical and automation engineers for designing and manufacturing of a complete system [50].

The drawback is there is not much information or guidelines for using MCD present online for free. It requires effort and time by the user for fetching the concepts so an accurate model could be made.

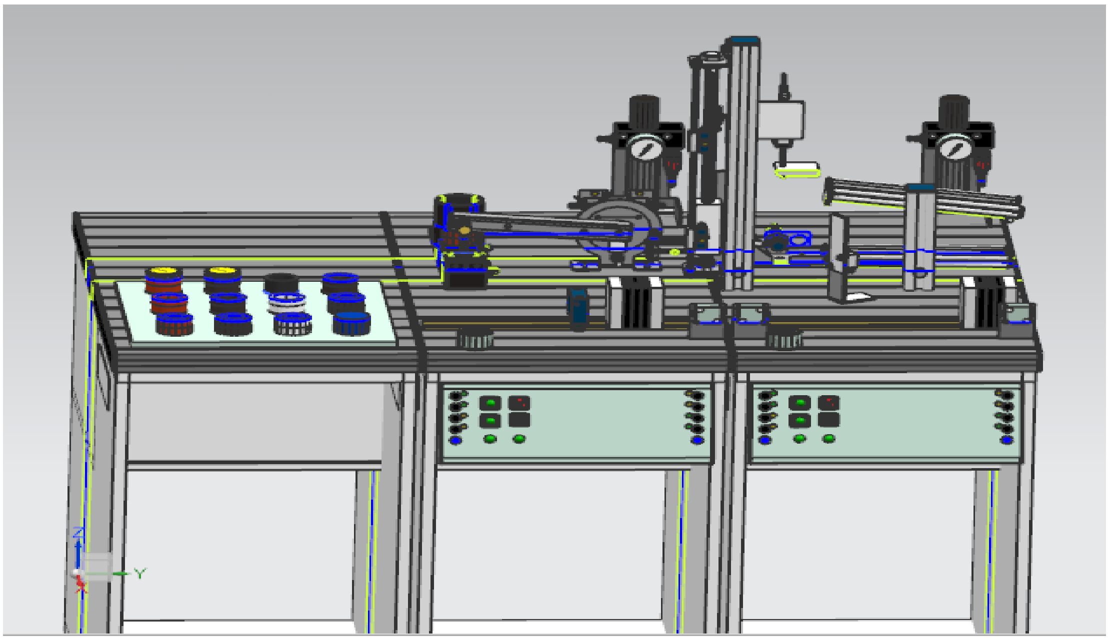


Figure 9 *FESTO Distribution and Testing Stations in Siemens NX MCD*

Visual Components is a commercial 3D simulator, co-founded by JOT Automation, in 1999. It is capable of showing the simulation of material flow and robotics using discrete events [53]. This software also has NVIDIA PhysX engine as in Siemens NX, which makes the possibility of visualizing the impact of physical forces affecting the model like gravity, collisions and properties according to the material type [54]. The software also has the feature of offline programming.

Visual components has the library of different industrial components already defined, including robots of different brands as demonstrated in figure 10. According to the website, there is also a CAD converter to quickly import geometries from other tools to visual components. Visual components was developed using the .NET framework, which is commonly used by developers and it is also integrated with Python API [54]. Visual components also have a capability of connecting the designed model with Programmable logic controller (PLC). It can help the designers a lot as they can easily validate their design and control code before implementing it with the hardware. Similarly, as Siemens NX, this tool can also be used for testing alternative ways of achieving a customer goal.

Additionally, Visual Components platform is used by the other companies for their simulation software's like Kuka Sim developed by Kuka [46] and Octopuz simulation software developed by In-House solutions [55].

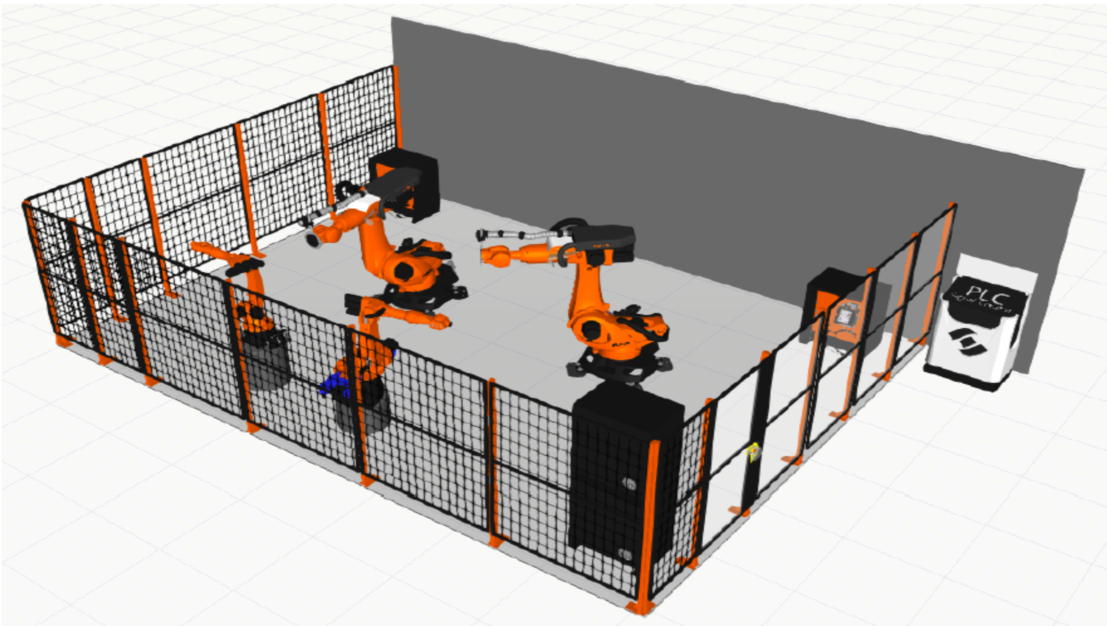


Figure 10 *Visual Components Robotic Simulation [56]*

2.2 Methodologies for Testing Control Programs

There are different techniques and methodologies which can be used for the validation and verification of the control logic. Danielsson and his group defined 18 methods which were further classified into four categories, along with with their advantages and disadvantages [23]. Those four categories are:

1. Hardware methods
2. CPU response methods
3. Logic analysis method
4. Simulation method

2.2.1 Hardware Methods

The first method gives detail about conventional commissioning and testing. It is about connecting all the required hardware, uploading untested software's and then starting the system. It is called "*Online test on real equipment*" by the authors. The second one is ”*Online Monitoring*” which is about testing the hardware using oscilloscopes, lamps and other tools before commissioning. The third method proposed by Danielsson and research group is about hardwired test panels. It is related to a manual approximation of actual working hardware. It also involves connecting the hardware (e.g coils, switches, sensors, relays, potentiometers, etc) to the signal modules of Programmable logic controller (PLC) to get the idea of the working of the system. According to Dougall, this method is not very efficient as it is a very complicated task to estimate the behavior of the system using manual switching and also the validation of fast switching tasks is very difficult [24].

2.2.2 CPU Response Methods

As one of the main methodologies of testing the control program, this method was also used by Danielsson, to validate the system. It involved the execution of the program for controlling purpose, along with a response program. With this methodology, the behavior of the process or system can be successfully implemented. One of the advantages of using this approach is, unlike hardwired testing technique, process behavior can be easily implemented. According to the authors, the con for using this approach is, the testing response program should be removed before using the plc with actual hardware. It gives the challenge of modifying the control program for physical systems. This modification still leaves the system with the untested program, loaded on the controllers.

2.2.3 Logic Analysis Methods

The first analysis method is ”*State space search*” in which analysis of the output is done based on the chart generated by all the possible combinations of inputs, needed for that specific output. It is rather a complicated task to implement but it has the capability of considering a large state space. The author also highlighted the need of rules for the automatic detection of errors in the states. Second analysis technique ”*Lexical analyser*” as described by the author is to change the language of the control program or transform it into another form before analysis.

2.2.4 Simulation Methods

In simulation models, authors replaced the response program (which was being used for testing the code) with a PC based simulation model. There were six simulation methodologies defined by Danielsson and his group.

1. First one is ”*logic simulator method*”, which was also named as ”*model in the loop (MIL)*” by Hoffmann [21]. In this approach, the author tested the logic with the model of a real controller. It was better than previous approaches but still, there was room for improvement.
2. The second method is termed as ”*logic emulator*”. The main advantage of using this approach is that the same programming language can be used for testing and real controller. There is no need for modification as required in the CPU response method. Thus saving a lot of effort and time of programmers.
3. The third method which is named as ”*control system simulators*”, deals with the simulation of the physical device. Simulation modules of actual hardware can be integrated with simulation systems for that specific physical system i.e. robots for better understanding and code validation.
4. According to Danialsson and group, The fourth term is ”*real control systems*” also known as ”*hardware in the loop (HIL)*”. In this approach, simulation is connected with the actual controller. This approach can provide a good understanding of the whole process with real values. The main drawback using this approach at that time, was the possible lags in the data flow between controller and PC, causing a disturbance in the simulation.

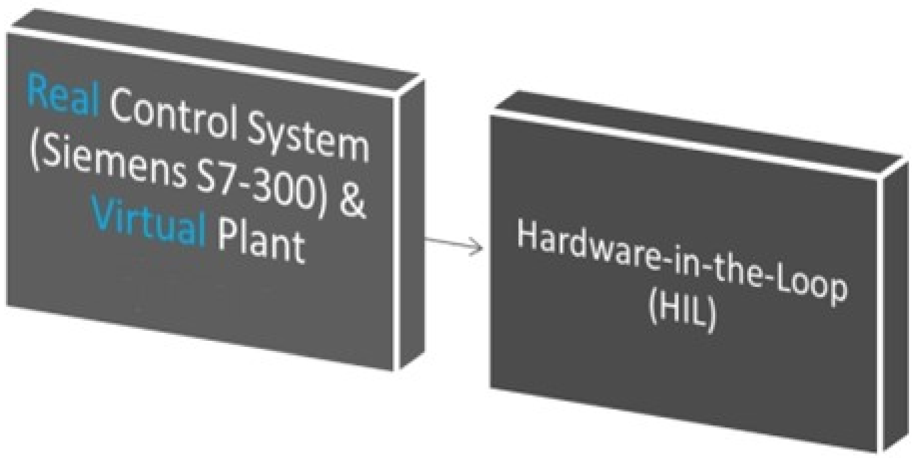


Figure 11 *Real Control System Method [4]*

5. The fifth method which is called as ”*online control*” involves the connection of both simulation and physical systems with an actual controller. This methodology resembles the real control systems method but the only difference is the connection between controller and hardware.

6. According to the author, the sixth methodology for the simulation is ”*control system emulator*”, which is also known as ”software in the loop (SIL)”. It involves the replication of the real controller. This method can be used for the validation of the code and the same program can be used for the controlling purpose, without any modifications. By using this approach, simulation models and emulators can be synchronized for testing the number of scenarios, without using the actual hardware as shown in figure 12. Thus, it allows to test and validate a plethora of ideas without any risk.

   There are two challenges that could be faced in this approach as represented by Danielsson and group. First is the time synchronization between emulation and simulation models. It represents a major challenge for all researchers. If we take the example of Siemens PLCSIM, it can be effectively used for code validation, without using any actual controller. According to Hoffmann, there is a problem of time synchronization while using PLCSIM with simulation models, which effects the whole idea of the testing. This problem was addressed by Siemens while the development of PLCSIM Advanced emulator [21].

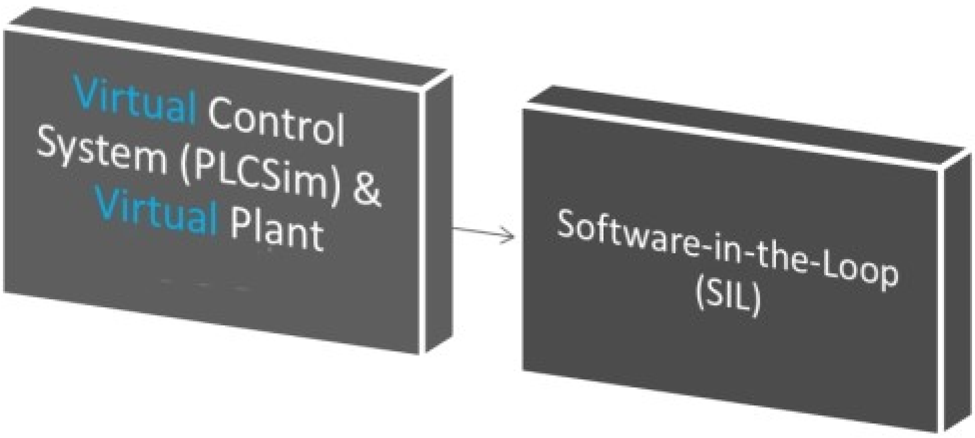


Figure 12 *Control System Emulator Method [4]*

The second problem which was highlighted by Danielsson and group was the designing of simulation models of actual processes. It requires a lot of effort and time to create a model, which replicates the actual hardware. Hence, representing a challenge for this approach.

2.3 Open Platform Communications (OPC)

In today's world, automation is used by the processing industries to produce products with more efficiency and reliability. Communication is one of the important factors for the whole process to operate properly and also for supervision purpose of the system. This communication can be enabled between devices with the help of various industrial communication protocols. Industrial communication protocol can be selected depending on the type of application under consideration and user requirement.

Open platform communication (OPC) consists of a series of specifications for the communication between various industrial devices. It was defined in 1995 with the help of a number of players of automation along with Microsoft. Initially, it was known as object linking and embedding (OLE) especially for process control, which was only restricted to Windows operating systems. Over time, there has been significant progress and it became the popular way of communication between automation layers of various industries [41].

OPC is a server-client based technology as illustrated in figure 13. OPC server converts the hardware communication protocol of PLC into OPC protocol. The client uses this OPC server for the data exchange between the devices. There can be one or more servers which respond to the requests of clients. These servers are able to perform the requested actions of clients because of their connection with the controllers and devices. While the implementation of software in the loop or hardware in the loop simulation approaches, the communication channel needs to be established between the programmable logic controller and simulation of the system. OPC can be used for the information exchange between virtual systems and controllers.

OPC has simplified the communication establishment process between the devices of different vendors. By using this standard, data exchange between different controllers, Human Machine Interface (HMI), Supervisory control and data acquisition (SCADA) and simulation tools (Siemens MCD, Process Simulate, Factory I/O, etc) is possible. According to Hoffmann, commonly OPC server is provided by the vendor of PLC's or by third parties (Kepware, Softing, etc). This server can be connected with more than one OPC clients and data can be exchanged [21].

Heidari and Salamon used the OPC as a bridge for communication between DELMIA V6 and GX IEC Developer. DELMIA V6 contained the simulation part and was acting as an OPC client. GX IEC Developer was used for the PLC programming purpose. They successfully established the communication between 110 Inputs/ outputs and simulation using the Beijer OPC server [4].

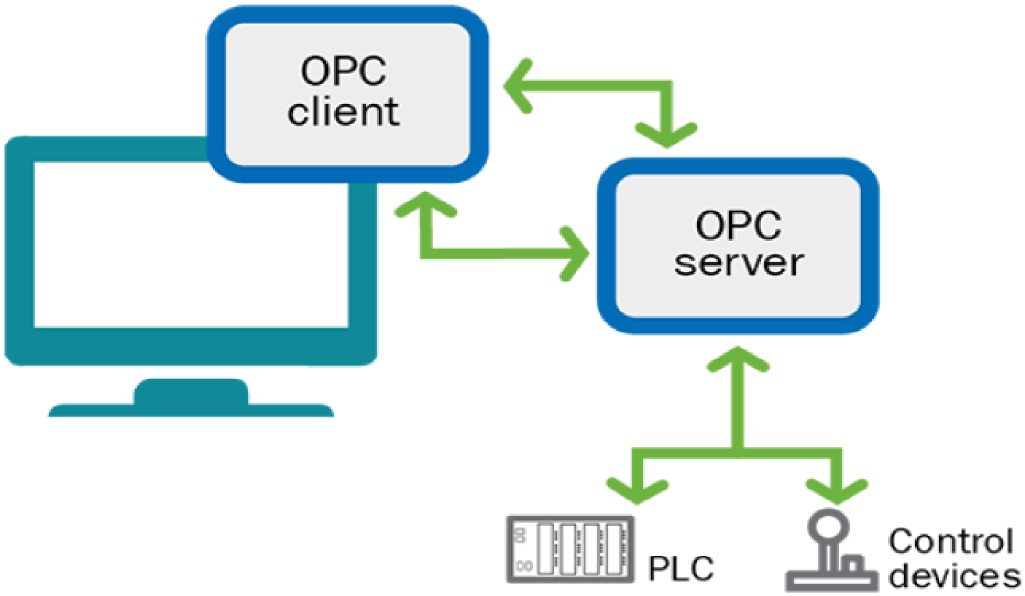


Figure 13 *Open Platform Communications [41]*

Hincapié and co-authors used DELMIA automation as a PLM tool, for the creation of simulation. They tested the PLM tool in different scenario and for the data exchange between PLC and PLM tool, OPC was used [42]. Pick and place station of AMATROL mechatronic system was used by Guerrero and co-researchers for the study of virtual environment and control code validation. Authors used Process Simulate tool for the simulation purpose. Siemens S7-300 was used as a controller. For the communication between S7-300 and Process Simulate, OPC was used [43].

Wischnewski and Freund presented the work related to modeling the transport systems along with the interaction with other machines and robots. Authors encountered some problems when using the OPC for the data exchange between the controller and the virtual system. They further explained that when the process is fast, then it is not possible for the OPC to transfer data between simulation and controller in real-time hence disturbing the whole process. As a solution to the highlighted problem, they suggested decreasing the speed of simulation and PLC so there is more time for OPC to process information [44].

Kim and co-authors discussed virtual commissioning of steel manufacturing plants. In their implementation, they connected the HMI and emulator of the system using OPC for the data communication between these devices. Authors validated their simulation model and control logic successfully [45].

Krause created the simulation of Liquified Natural Gas (LNG) plant using Sandvika and for the control purpose emulator of AC 870P/ Melody System, known as an active part of ABB 800xA Distributed Control System was used. He successfully established the communication between the simulation model and the controller using OPC with the update time of 500ms, same as in the emulator [38].

3. RESEARCH METHODOLOGY

The aim of the thesis is to develop a methodology for validating the mechatronic digital twin. The proposed methodology can be used with the mechatronic models of existing systems to assure the proper operation of the actual system. It enables the user to observe and detect the problems, by replaying the recorded process data of actual hardware operation and movements in the mechatronic model, which can assist the user to point out the problem. A number of steps were adopted to implement the desired system as described in figure 14. The research approach is briefly described in the text below.

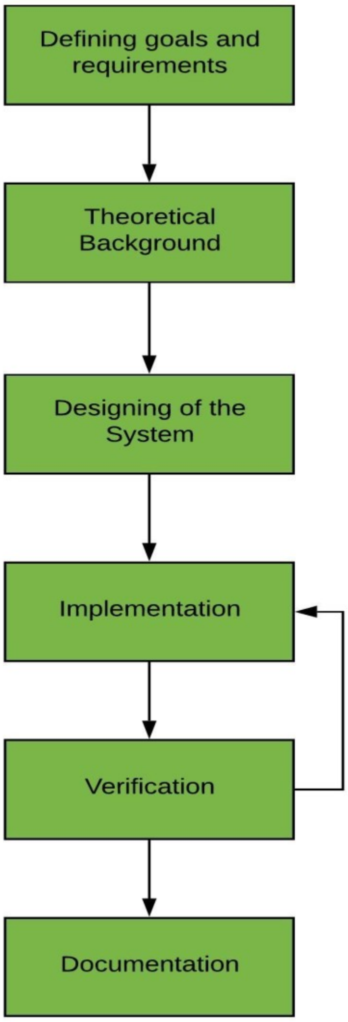


***Figure 14** Research Methodology*

Defining requirements and goals: Identifying the problem and setting the goals is the most crucial part of any research. A thorough analysis and effort is required to study the root cause of the problem. After the problem definition, there is need for some objectives which are stated in the introduction section of this thesis. On the basis of the goal defined at this stage, a work plan was made in order to achieve an efficient solution. In this step, the aim of the thesis was decided, which suggested that a methodology should be developed so a mechatronic digital twin could be validated and verified.

Theoretical Background: An efficient and reliable solution requires the literature review based on the problems defined in the introduction section. For getting the overall idea and knowledge of the existing technologies, literature review was done with the help of articles, books, conference proceedings and other reliable online resources. It gave the understanding of the work previously done by other researchers and their results so the same mistakes can be avoided while the designing and implementation phase of the desired system.

Keeping the aim in mind which was defined in the requirements of the thesis, research was conducted using the above-mentioned sources. In this phase, different technological decisions were made based on the outlined goals. First decision was the selection between Inico S1000 controller and Siemens ET200SP Controller. ET200SP controller was preferred as it is one of the latest technologies by Siemens for process control and also because Siemens NX was selected for the designing of mechatronic model of the said system and initially the assumption was that it will be comparatively easy to integrate these tools as both are offered by Siemens.

One example of compatibility between these tools is the possibility of detection of the PLCSIM Advanced in the Siemens NX, which is an extra channel for communication with mechatronic model offered by Siemens. However, third-party software was used as data logging of the process parameters of actual operation is required, which is necessary for the validation of the mechatronic model.

Designing: Literature review helped in understanding the existing systems and provided a necessary understanding of the concepts related to it. Tools were selected in this phase based on the information gathered in the theoretical background. Software and hardware components were also finalized in this phase to proceed further. As Festo MPS 500 system is used in the implementation so both stations of the Festo system were studied thoroughly to understand the process. Initial testing was performed considering a simple system with the selected tools in the literature review phase and communication was established for the purpose of exchanging the information between these softwares.

Implementation: After the selection of the tools and initial testing in the designing phase, the next step was to move towards the implementation of the actual system instead of working with the example system. In this phase, mechatronic model of the testing and distribution stations are made in the selected software. Communication established in the designing phase is used for data exchange between the control program and mechatronic model. After the integration of database, it was successful completion of the implementation phase.

Verify: As a next step after completing the implementation of the system under consideration, verification was done. Various test were performed for the purpose of analyzing the system and its capabilities. In the test sessions, the data of the actual operation which was initially stored in the database was sent to the mechatronic model of the distribution and testing stations. At this stage, the outcome was compared with the goals defined in the first phase. If there were problems which are not expected then changes were made in the implementation section to achieve the desired results.

Documentation: When a working solution was developed as defined in the initial stages of the thesis, documentation was done. In the documentation, detail about development stages, engineering tools and integration technologies were included that can help the researchers in the future. It also contains the outcome of the thesis work for future reference.

4. IMPLEMENTATION

This chapter provides the detail about the hardware and software used for the implementation purpose of a mechatronic digital twin of distribution and testing stations of Festo MPS 500 System. It gives a brief idea about the link that was established for the data exchange between various engineering tools, using OPC. Different programming languages for the control logic are also discussed along with establishing communication between two controlling modules.

4.1 System Architecture

This section provides detail about the whole system architecture. It is to introduce engineering tools and technologies that are used in the implementation of the whole setup. The thesis was divided into two phases to get a clear picture of the work being done. The first phase was to study about selected stations of Festo MPS 500 system. Making a control program and verifying it was also a part of it. Additionally, the link for storing the process parameters in the database was also established, which was needed in the later stages of the implementation. Figure 15 shows the architecture of the first phase of the implemented system.

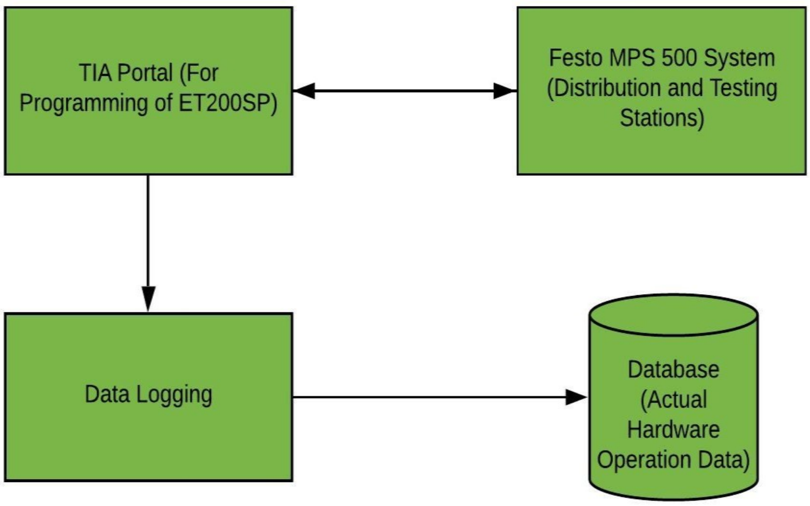


Figure 15 *Process Data Storing from Actual Hardware to Database*

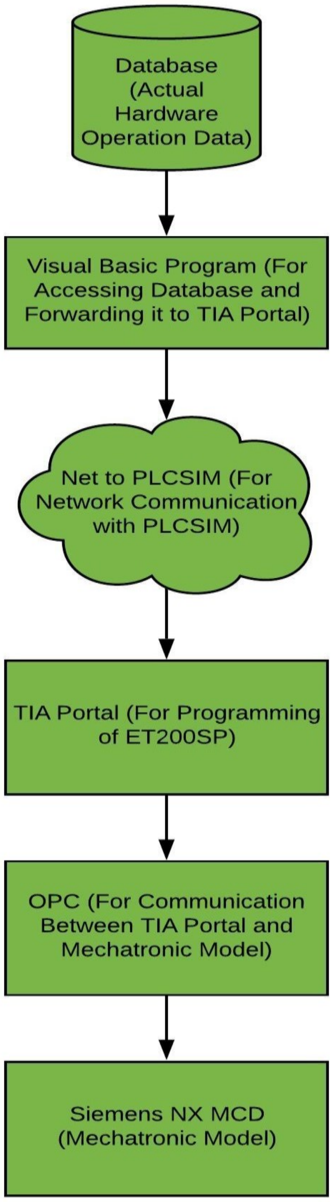


Figure 16 *Data Retrieval from Database to Mechatronic Model*

The second phase of the thesis was to fetch the process parameters data stored in the database and passing that data to the controllers as shown in figure 16. Microsoft Access database is used in this thesis. Data of the process signals were stored with timestamps so that the exact movements of the system could be seen in Siemens NX. In visual basic, S7.net library was used for the connection with the ET200SP's of distribution and testing stations. After the successful connection with the TIA Portal, OPC was used for data exchange between control program of both stations and mechatronic model. The communication link between all these tools was established successfully and system was implemented as defined in the initial stages.

4.2 Hardware

In Fast lab of the Tampere University of Technology, Festo MPS 500 System has been installed as shown in figure 17. It consists of various stations, which have different sensors, actuators, light barriers, valves and switches. All of these stations are connected with their respective ET200SP, for the control purpose. More detailed review about hardware is provided in the subsections below that was used for the implementation purpose of this thesis.

4.2.1 Festo MPS 500 System

Each station in the Festo line is the whole world in itself. Festo is one of the leading companies in the automation world, especially in the mechatronics field. Festo MPS 500 is widely used all over the world for industrial vocational training by which programmers can get the experience of working with the whole system and synchronizing it.

Figure 17 *FESTO MPS 500 System- FAST Lab*

Distribution station is mainly for separating the workpieces from the stack magazine module. Magazine tube of the stack has the space of storing up to 8 pieces. The light barrier is also present in the stack magazine for the detection of the workpiece. A double acting cylinder, which is right beneath the stack module, can push out the workpieces individually.

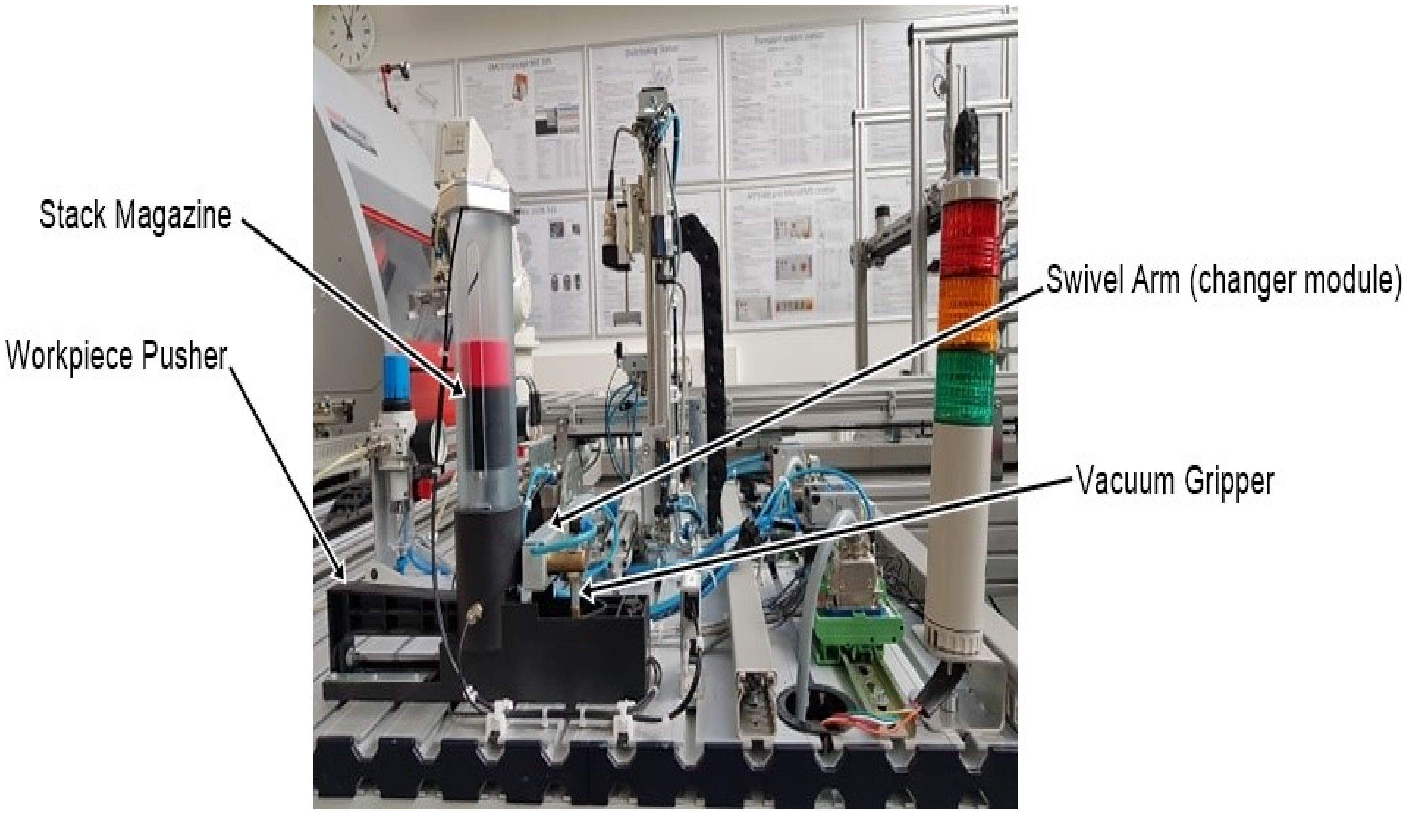


Figure 18 *Distribution Station*

There is also pneumatic swivel arm installed on the distribution station for transferring the workpiece from distribution station to the next station as shown in figure 18. Swivel arm is further attached with vacuum gripper, which can hold the workpieces during the transfer from one station to another. A vacuum switch is also integrated with the vacuum gripper so the user could know if the workpiece has been picked up.

There are limit switches installed to track the movement of the cylinder, installed beneath the stack module. It helps to know that if the cylinder is in a retracted position or extracted position. Same is the case with swivel arm, to get the information that if the swivel arm is in the distribution station or the next station. These limit switches are very important for the safety of the swivel arm.

In the lab, adjacent to the distribution station, the testing station is installed. Testing station is capable of differentiating between the workpieces, inserted by the swivel arm of distribution station. A capacitive sensor is present in the station for sensing the availability of the workpiece. Next to the capacitive sensor, the diffuse sensor is present for sensing the color of the inserted workpiece. It is calibrated in such a way that it gives false bit value for red workpieces and true bit value for silver workpieces.

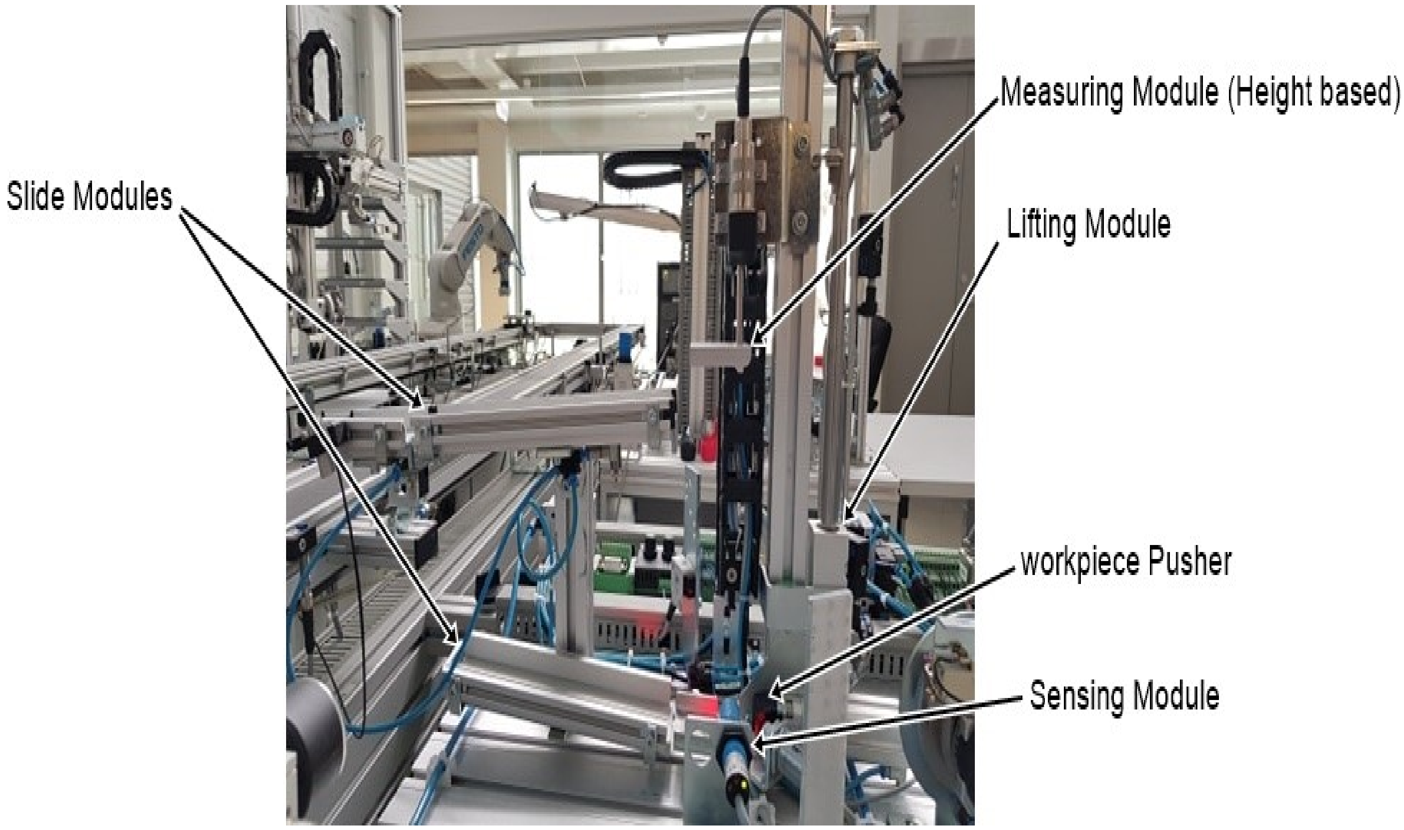


Figure 19 *Testing Station*

Moreover, as shown in figure 19, a lifting module is also present in the testing station. It carries the workpiece towards the upper position. A sensor of a measuring module is installed on the top position of the lifting module. It can sort out the pieces on the basis of their heights as in the lab there are workpieces present with heights different from each other. When workpieces touch this height sensing part, a bit turns true otherwise it stays false.

There are limit switches installed with the lifting module to know that if it is in the upper position or on its normal (downward) position. A through-beam sensor monitors whether the workspace above the lifter module is empty or not as the swivel arm of the distribution station can reach this space and can cause damage to itself and the lifter module.

There are two air cushion slides also integrated with the system. According to the control program and the sorting method used for the workpieces, upper air cushion slide and lower air cushion slide can be used.

4.2.2 ET200SP Control Unit

ET200SP is a distributed Input-output control system, which is highly flexible and scalable. It is capable of connecting the integrated process signals to central control station using Profinet. According to Siemens, it is very easy to operate for the user as it has the room for the connection of Profinet technology by using a bus adapter. There is push-in technology used with modules, which enables the user to do wiring without using traditional tools. It is also integrated with channel-specific diagnostic functions.

Figure 20 *ET200SP Module- Distribution Station*

It is very compact in design and can be fitted in standard control boxes of dimension 80mm. It means a lot of space can be saved for other modules and less complexity. It also has the option of expandability up to 64 modules, which is really useful for setting up a process with a lot of process and control signals.

Fast data exchange is also possible using ET200SP when used with Profinet. Additionally, its configuration can be changed using the software e.g. TIA Portal, Step 7. There is also a concept of ”Hot-Swapping”, which is applicable in this control unit. It means that signal modules and terminal boxes can be changed even if the system is running. It is very beneficial for troubleshooting scenarios [57]. Further ET200SP also has various connection technologies for Profinet, by using bus adapters. In figure 20, The first module is BA 2xRJ45 for the connection with RJ45 plugs. The second module is the Central Processing Unit (CPU) of ET200SP, capable of controlling the signal modules. With Profinet connection, it behaves the same as CPU 1511/ CPU 1513 of the S7-1500 controller [58]. It is commonly used as Remote terminal Unit (RTU) with S7-1500 in the industries. However, it can be used as an independent module as installed with the stations of Festo MPS 500 system in Fast lab.

Next to the CPU module of ET200SP, there are signal modules attached to the rail. In this case, there are three digital input signal modules and two digital output modules. One CPU module of ET200SP can support up to 64 input and output modules. One signal module has seven digital inputs or digital outputs so it is capable of reading real-time values of seven sensors or giving output to seven motors or actuators. The fail-safe function is also integrated with these signal modules for the safety of the user and the hardware. The same setup as shown in figure 20 is used for controlling purpose of the testing station, with the same number of signal modules and bus adapter.

4.3 Software

This section briefly describes the software modules used for the implementation purpose of this thesis. The implementation is not possible without the establishment of communication between these engineering tools (e.g. Siemens TIA Portal, Siemens NX MCD). These tools are discussed in the subsections below.

4.3.1 Mechatronics Concept Designer (MCD)

Mechatronics concept designer is an add-on offered by Siemens in Siemens NX software. In the MCD, it is possible to make the 3D model of any machine or station under consideration, following the desired requirements. Along with 3D design, It enables the user to make the kinematic simulation of the desired model. For the kinematic simulation, 3D model is needed to be converted to a mechatronic model by connecting the appropriate sensors, actuators or cylinders. These automation tools can help the design to sense the environment around. MCD is used in the early stages of development of a prototype because it enables the user to test alternative paths for achieving the desired actions. With more than one options, it is comparatively easy to select an efficient and reliable design.

In this thesis, 3D models of the distribution and testing stations have been converted to mechatronic model using mechatronics concept designer. 3D models of both stations are imported from the Ciros Studio. Siemens NX can import the models from a number of formats like CGM, Part, AutoCAD DXF/ DWG, Parasolid, VRML etc. From Ciros Studio VRML file type was imported to Siemens NX MCD.

Next step is the conversion of the 3D model to mechatronic model, which starts with defining the kinematics of it. Mechatronics Concept Designer enables the user to select the parts of the model which are moveable and assign them as rigid body. The NVIDIA PhysX engine is integrated into Siemens NX, so by defining the rigid bodies forces like gravity start acting on the assigned parts. There is also need for assigning collision bodies, which will touch with each other. If collision bodies are not assigned then upon collision those parts will pass from each other and it refrains us to see the actual impacts on collision. Hence, collision bodies were assigned in the 3D model wherever needed like workpieces, pickup place of distribution and testing stations, swivel arm, air cushion slides etc. If the collision body property is not assigned or appropriate joint not used with rigid parts used then these parts will fall down while playing the simulation.

After defining the rigid bodies and collision bodies in the model, there is need for joints so parts can move together. It can help the designers to simulate their desired movements according to the design requirements. There are variety of joints available in Mechatronics Concept Designer like fixed joint, sliding joint, cylindrical joint, hinge joint, screw joint, planar joint etc so the model can be very accurate and close to designers expectations. Some of these joints were used during the conversion of imported 3D model to mechatronic model e.g. Sliding joint with cylinders, pushers, lifter module, Cylindrical joint with swivel arm, Fixed joint with the bases so those can bear the effect of gravity and do not fall down while simulation.

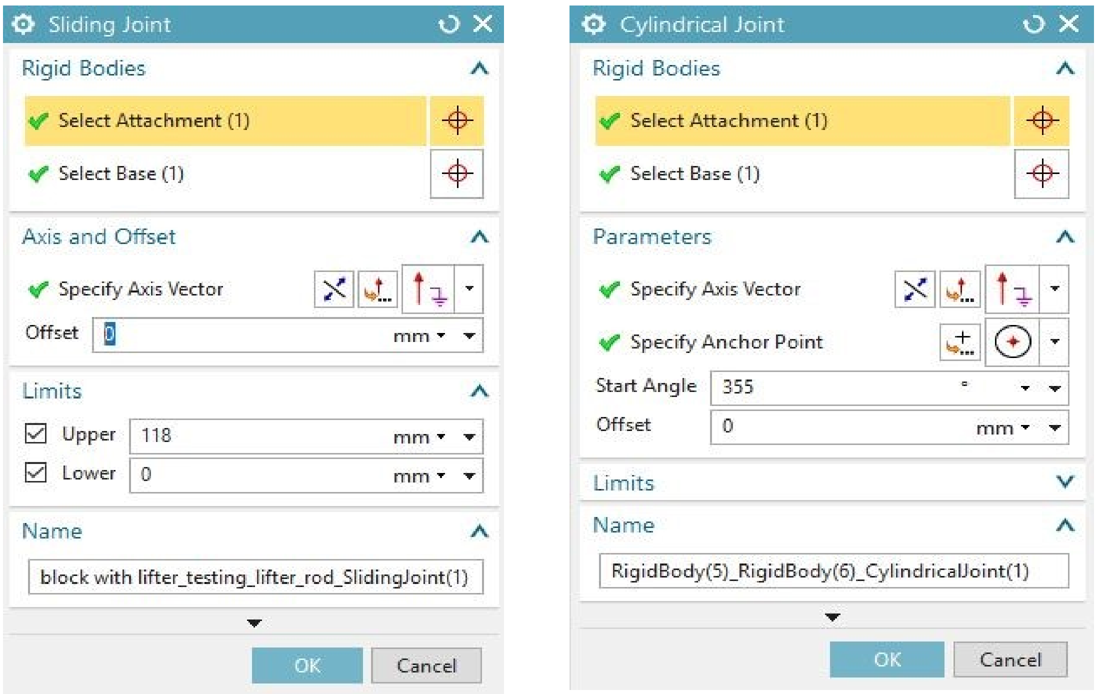
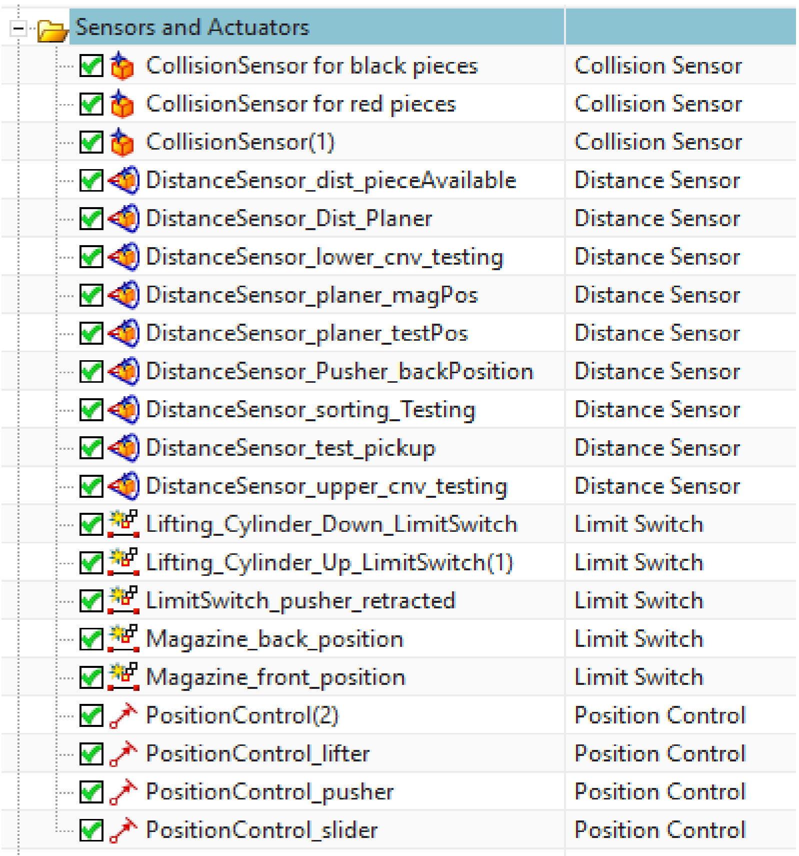


Figure 21 *Parameters for Sliding and Cylindrical Joint in MCD*

There are many parameters that should be assigned while making a joint in MCD as in Figure 21. For example, in the case of a fixed joint, it is necessary to select one attachment and one base. In case of a sliding joint, along with attachment and base, there is also need of axis vector. Axis vector provides the axis along which the attachment will move. There is also an option of restricting the movement of the attachment by defining the upper and lower limits so that specific part cannot move further even if the conditions are true. These limits represent the freedom of movement of actual hardware parts.

Mechatronics Concept Designer also allows the user to add the sensors in the model. It enables the model to sense the environment around as in the case of real-time physical systems. There are many sensors available in the library of MCD, which can be integrated into the model under consideration. It allows the user to check different sensors for fulfilling the requirements defined by the customers. It can greatly help to choose an efficient and reliable model. The sensors which are available in the MCD library are collision sensor, distance sensor, position sensor, inclinometer, velocity sensor etc. Additionally, there are also limit switches for the limitation of a part’s movement between a certain range.



Sensors and Actuators	
CollisionSensor for black pieces	Collision Sensor
CollisionSensor for red pieces	Collision Sensor
CollisionSensor(1)	Collision Sensor
DistanceSensor_dist_pieceAvailable	Distance Sensor
DistanceSensor_Dist_Planer	Distance Sensor
DistanceSensor_lower_cnv_testing	Distance Sensor
DistanceSensor_planer_magPos	Distance Sensor
DistanceSensor_planer_testPos	Distance Sensor
DistanceSensor_Pusher_backPosition	Distance Sensor
DistanceSensor_sorting_Testing	Distance Sensor
DistanceSensor_test_pickup	Distance Sensor
DistanceSensor_upper_cnv_testing	Distance Sensor
Lifting_Cylinder_Down_LimitSwitch	Limit Switch
Lifting_Cylinder_Up_LimitSwitch(1)	Limit Switch
LimitSwitch_pusher_retracted	Limit Switch
Magazine_back_position	Limit Switch
Magazine_front_position	Limit Switch
PositionControl(2)	Position Control
PositionControl_lifter	Position Control
PositionControl_pusher	Position Control
PositionControl_slider	Position Control

Figure 22 *Sensors for Distribution and Testing Stations- Siemens NX MCD*

In this implementation as demonstrated in figure 22, for the conversion of 3D model to mechatronic model collision sensor, distance sensor and limit switches were mostly used in the overall design. For example, for the detection of the workpiece, distance sensors were used. When a collision body comes in range of the distance sensor, it gives the ”True” signal and vice versa. The distance sensor is also used for the height based sorting in the testing station and it works well.

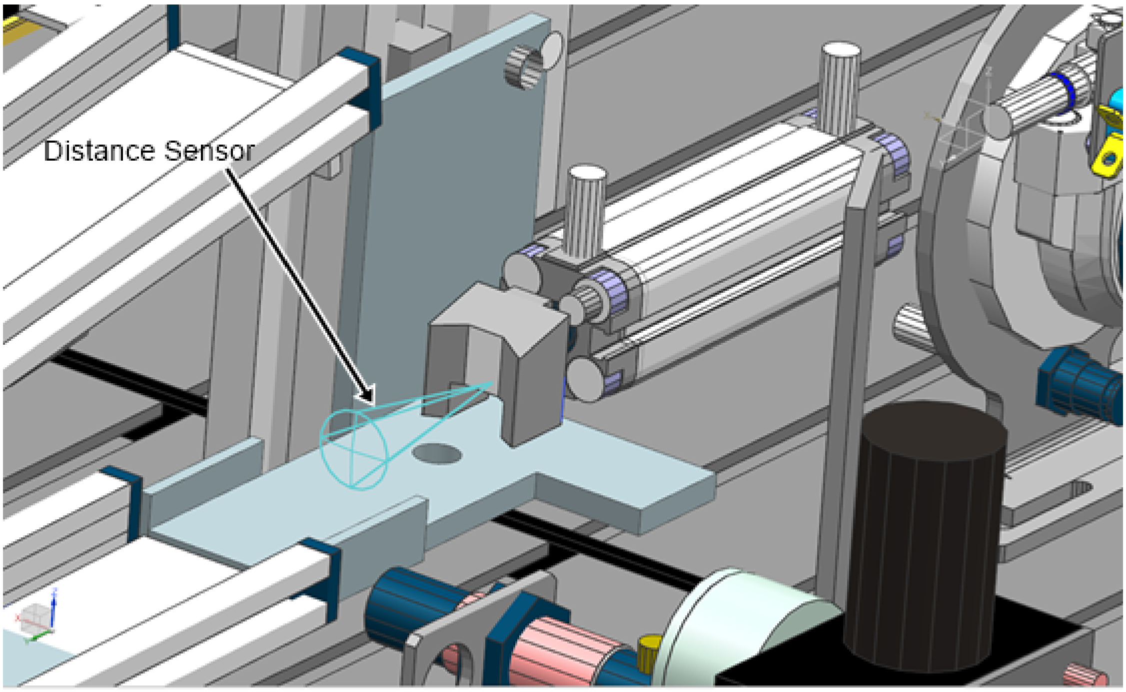


Figure 23 *Distance Sensor at Workpiece Pickup Point of Testing Station*

Distance sensor as shown in Figure 23, can only detect the parts which are assigned as collision bodies in the basic physics section of Siemens NX MCD. Collision body property is mandatory to be assigned to parts which need to be detected by the distance sensor. The part is not visible to distance sensor unless this property is activated.

There is also collision sensor shown in figure 24, which can also be used for the detection of some specific pieces. Collision sensor has various shapes like sphere, line and box which can be selected depending upon the application. When some specific collision body passes through it, it gives ”1” as a result otherwise it stays ”0”. There is also a feature of ”highlight on collision” with which color of the part gets changed. It is very useful for the user to get the visual indication of the sensor output, while the simulation is running.

In the properties of collision sensor, there is one named ”Category”. It is specifically useful when the designer wants to detect some specific part or workpiece of the model. It enables the user to design more complex processes and manufacturing systems by differentiating the parts based on material, color etc. Category property of the collision sensor can be used for differentiating the color of workpieces in the testing station of Festo MPS 500 system.

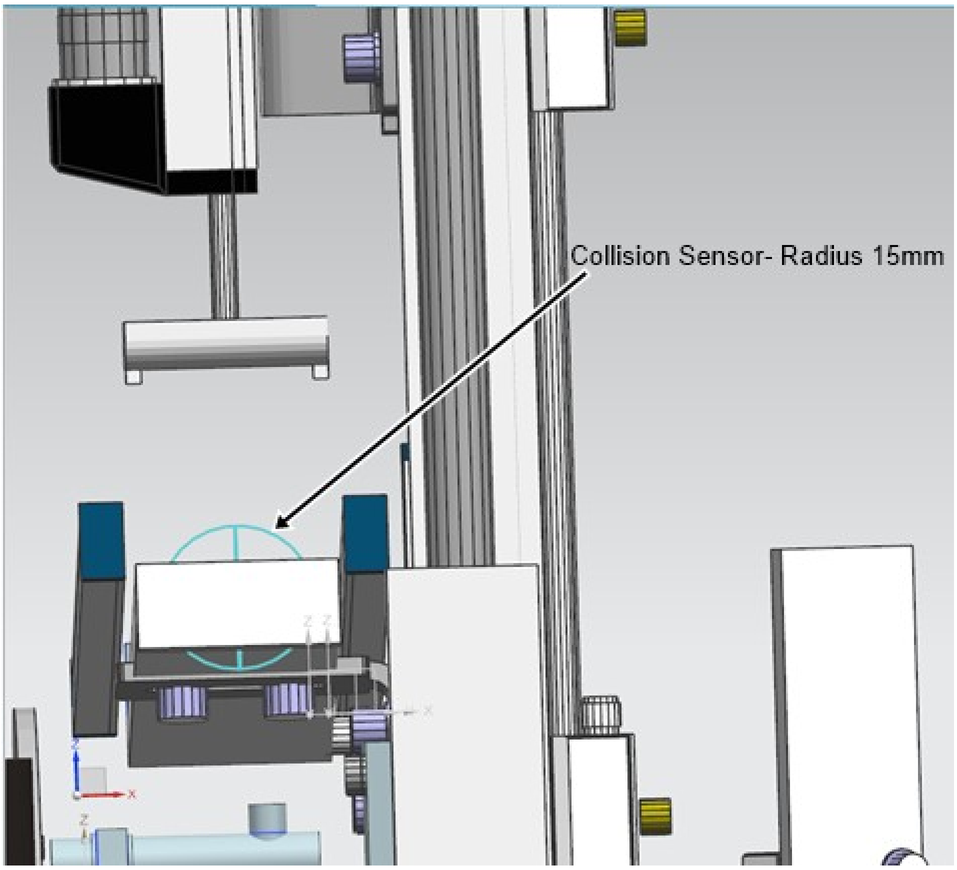


Figure 24 *Collision Sensor on air cushion slide of Testing Station*

Additionally, there are limit switches to restrict the motion of some specific parts. Two limit switches are used with the workpiece pusher of distribution station, two limit switches with lifting module of the testing station and one limit switch for the workpiece pusher of testing station. Modeling of the automation system is very difficult without restricting the motion of some parts so that desired actions can be implemented. Limit switches play a key role in this scenario and these are also very useful for the safety of the actual hardware.

There are many useful properties offered by Siemens in Mechatronics Concept Designer. Object Source and Sink is one of the most useful properties while creating the prototype of a mechanism or searching for the best alternative design. Object source can be used to create the replica of any specific object after a specific time period or on the happening of an event. The copy event can be selected in the property section of object source whether it is once per activation or time based as shown in Figure 25. By using this tool, the speed of the whole manufacturing process can be controlled and the model under consideration can be tested for a number of scenarios.

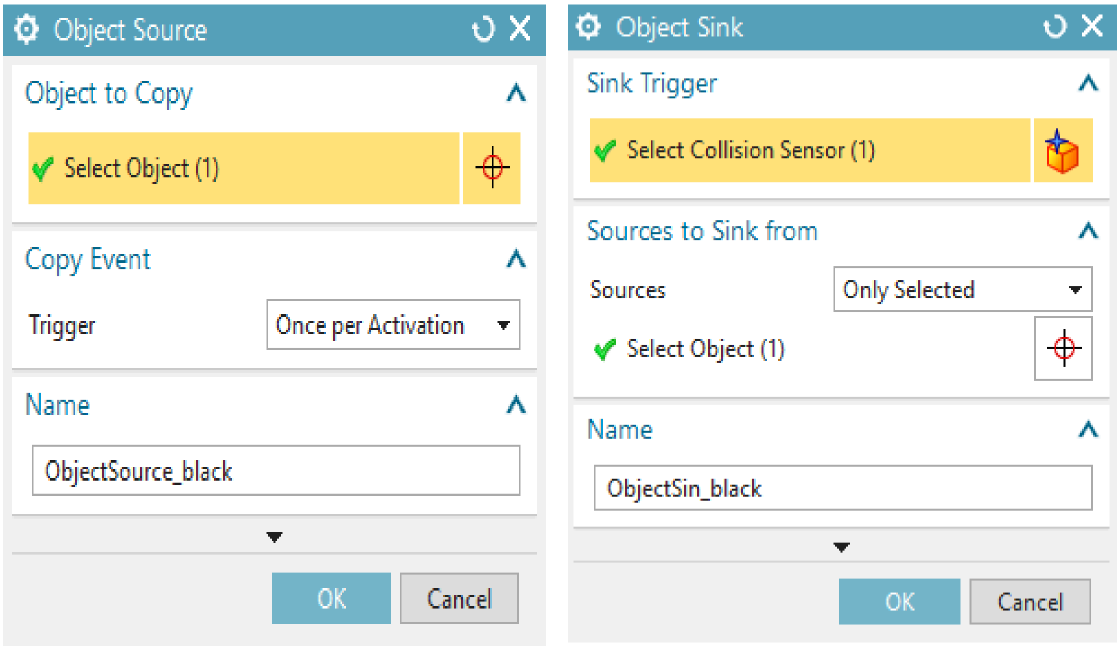


Figure 25 *Object Source and Sink for Black Pieces*

Object Sink is the tool in mechatronics concept designer for removing the parts or workpieces created by the object source at some stage of the process. The part is not removed unless it collides with the collision sensor. It can be seen from the figure above that the selection of the collision sensor is necessary to select so object sink could know the presence of a part in its range.

Additionally, this tool also offers a feature of removing the specific workpieces from the process by using the ”only selected” property in the dialog box. If ”All” is selected in the ”source to sink from” property of object sink then it has the capability of disappearing all the parts without differentiating between object sources. While the implementation of this thesis, both object source and sink were used for the generation and removal of the black and red workpieces, with the properties shown in the figure above. Two object sources were used on the workpiece stack module of distribution station. It is capable of creating a replica of red and black workpieces on activation of the object source.

For removing the workpieces generated by the object source, two object sinks were created on the slide modules of testing station. The testing station model has been designed to pass red pieces on the upper air cushion slide and black pieces on the lower air cushion slide, after sorting so two different collision sensors were created for both pieces on air cushion slides. When these collision sensors get activated by the object of their defined object source, the workpiece disappears from the slide module. With these useful features of mechatronics concept designer, it is possible to test the model with a large number of workpieces and with great speed. This testing can greatly help designer to reach an effective and reliable solution as per customer's requirements.

After setting up all this setup in the model of distribution and testing stations, next step to integrate actuators with the parts. There are various types of actuators available in the library so an appropriate one can be selected according to the designer's requirement. These actuators can be controlled by electric, pneumatic or hydraulic power. In the case of pneumatic cylinder control, pneumatic valve is needed to be integrated first in the model. These are many properties for setting up a pneumatic cylinder which needs to be set like the pressure of chamber A and B, piston rod type to show if its single rod or double rod, piston diameter, piston rod diameter, volume extension of inner chamber (A) and outer chamber (B), maximum piston stroke which defines the length of the piston, ratio of specific heat of the gas, gas constant along with gas temperature. These properties can be set by an expert designer according to the required functionality and application. For the electrical actuators, there is position control, motion control, force control and transport surface available in the library of Siemens NX.

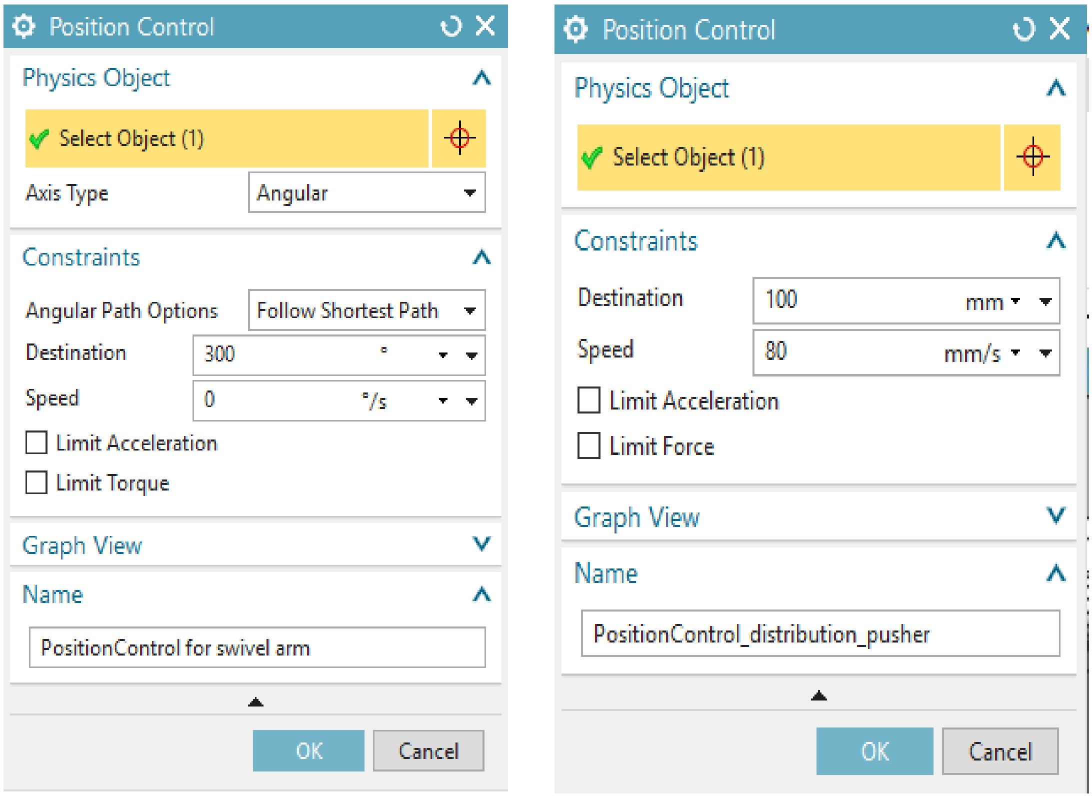


Figure 26 *Angular and Linear Position Control of Distribution Station*

Position control has been used in this implementation for the workpiece pushers of distribution and testing stations, swivel arm module of distribution station and lifting module of testing station. There are number of parameters that need to be defined as per requirement, for using the position control. Parameters defined for the position control of the swivel arm are shown in the figure above. First step is the selection of appropriate joint that needs to be controlled by the position control. Second, the axis type needs to be selected that defines if the joint will rotate or move in a linear axis. On selection of angular path, there are more options which define the needed rotation e.g. clockwise, anti-clockwise, follow the shortest path, track multiple turns etc. Further, destination angle and speed of the movement is also needed from the designer. For the linear movement, user only has to define the destination and speed of the joint movement as in the case of a pusher of distribution station (Figure 26).

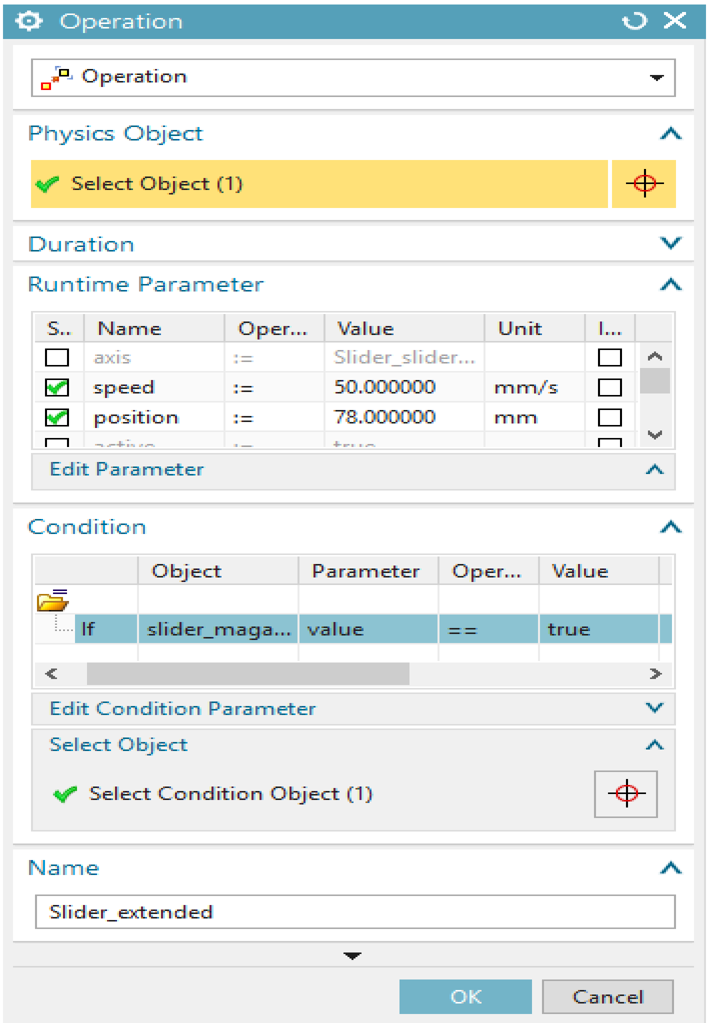


Figure 27 *Event-Based Operation of Distribution Station for Slider extension*

There is also a very useful feature named ”operation” in Siemens NX MCD. It helps the designer to test and control the movements of joints and constraints as the design will work in actual operation. With this feature, it is possible to access any object and their parameters of the model as shown in figure 27. The operation can be time-based or event-based depending upon the user requirements. The main difference between these two modes is the involvement of the control signal.

The execution of event-based operation happens when the pre-defined conditions (defined in operation) are met otherwise it will stay put. Time-based operation is totally time-dependent and after the pre-defined time has passed, the execution will start. In this way, designers are able to observe the simulation of the model and test alternative paths for the desired working. In the implementation of this thesis, initially time-based operations were used for the movements. It helped to visualize the working of the model and as per observation, adjustments were made so it could be more accurate. Later, control signals were introduced in the mechatronics concept designer which helped to make the model more reliable because of the sensors feedback. These control signals were integrated into event-based operations for all the moveable parts of the model and time-based operations were removed. It is also possible to export these operations to TIA Portal as a sequential function logic for further testing with emulators or real PLC's [64].

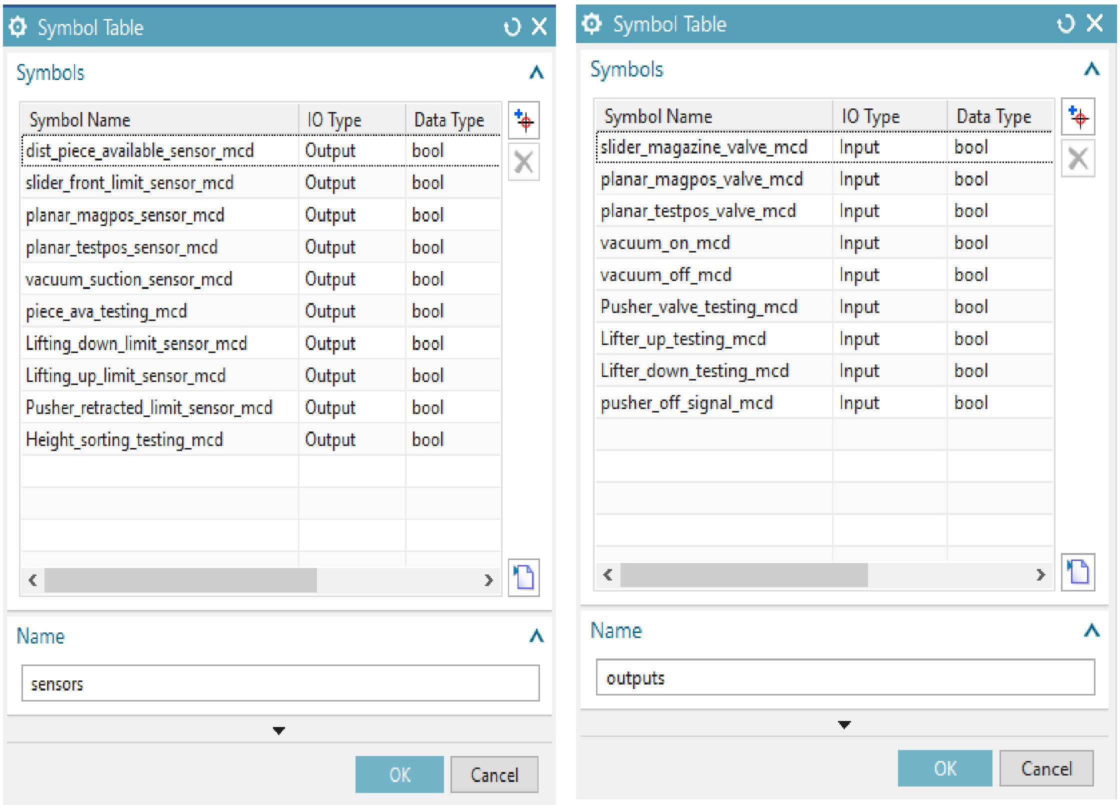


Figure 28 *MCD Signals for Integration with External Signals*

For the purpose of controlling the movements of different parts of the model, external signals are needed in the mechatronics concept designer. These external signals can be integrated with the MCD signals, by using signal mapping in MCD as shown in figure 28. OPC has been used for the data exchange between the TIA Portal and MCD in this implementation. The integration with the external signals is possible if the MCD signals are created for the joints of the model, in Siemens NX MCD as shown in the figure above. For this thesis, two symbol tables were created in MCD for inputs (sensors) and outputs. For data exchange, sensor signals are defined as output signals because these signals are meant to be communicated to external tools. Similarly, actuators are defined as input signals. For the external signals in the MCD, OPC server needs to be detected in the ”External Signal Configuration”. After the selection of the server, the external signals can be seen in the tags section.

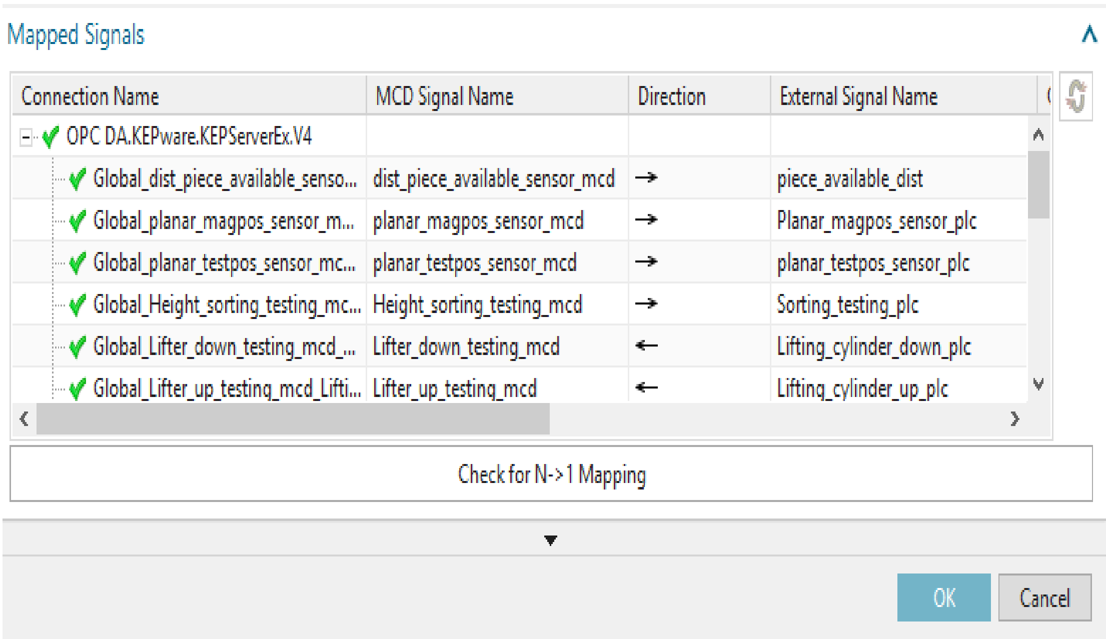


Figure 29 *Signal Mapping for MCD and External Signals*

After acquiring all the desired external signals, signal mapping is needed between the MCD signals and the acquired external signals. As seen in the figure above, signal mapping was done for both distribution and testing station signals (as external signals) with MCD signals. After the signal mapping, these signals can be used in the operations for event-based execution of any specific function or movement in the simulation. The data of these input and output signals of both stations were exchanged with TIA Portal using the Kepserverex OPC server. After receiving the data from mechatronics concept designer via Kepserverex server, control logic produces the output for the next cycle.

According to the instructions of control logic, changes and movements happen in the simulation as defined in the operations.

4.3.2 OPC and Data Logging

In this thesis, Kepserverx has been used for Open Platform Communication (OPC) and data logging. It is a client-server application designed for the data exchange between several industrial systems and client applications on the PC. Client-server applications are used in the market mainly for the sharing of process or production data with the various other applications and devices including Human Machine Interface (HMI), data historians, Enterprise Resource Planning (ERP) and Manufacturing Execution Systems (MES) also.

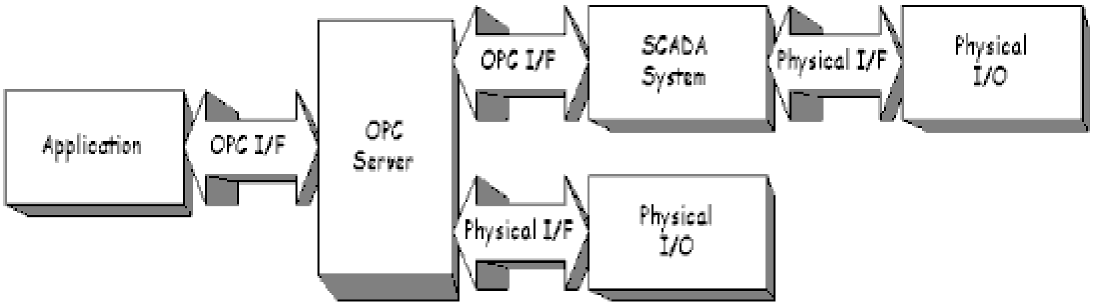


Figure 30 *OPC Connection using Client-Server Technology [60]*

A generic Client-Server technology-based application was introduced by Microsoft, in the early stages of Windows development. It was a basic layout for the information sharing between the applications of different vendors, being used on Microsoft Windows and this architecture was named as Dynamic Data Exchange (DDE), however it was not designed for the communication between industrial devices and systems. Later, a number of vendors from various disciplines of industries got together and founded the OPC Foundation in 1994. The main goal of this OPC Foundation was to develop a client-server application which could enable them to share data between their applications in a fast and reliable way. As figure 30 shows that now OPC client-server technology has enabled us to communicate between devices and system of different vendors. The communication can be done in a reliable manner with fast data exchange capability [60].

In this thesis, Kepserverex has been used for the data sharing between two ET200SP's and Mechatronic model designed in the Siemens NX software. The process starts by defining a communication channel and its properties. The channel can be referred as the drivers or protocols for devices so for every protocol, different channel is needed. Multiple channels containing devices from different vendors can be defined in the software for simultaneous data exchange. Channel acts as a basic building block for the devices having different drivers and properties according to the vendor-specific protocol.

As in this thesis, Siemens ET200SP is being used so a suitable driver "Siemens TCP/ IP Ethernet" was installed for the establishment of communication between various engineering tools, along with various properties like baud rate, communication port and parity of the channel. Next step is to define the device or devices in the channel, which need to communicate with the server. Programmable Logic Controller (PLC) or any other industrial hardware are the devices.

For the purpose of implementation in this thesis, two ET200SP were defined as devices, in the pre-defined channel. There is a requirement of defining various properties for each device as shown in the figure 31. These properties include defining the name of the device, model and ID. Model is the actual version of the hardware being used. As ET200SP is one of the technologies offered by Siemens and the exact model has not been defined in the KepServerEx for the OPC communication but after some research, it was observed that by selecting S7-1500 as the Model can work successfully for ET200SP too.

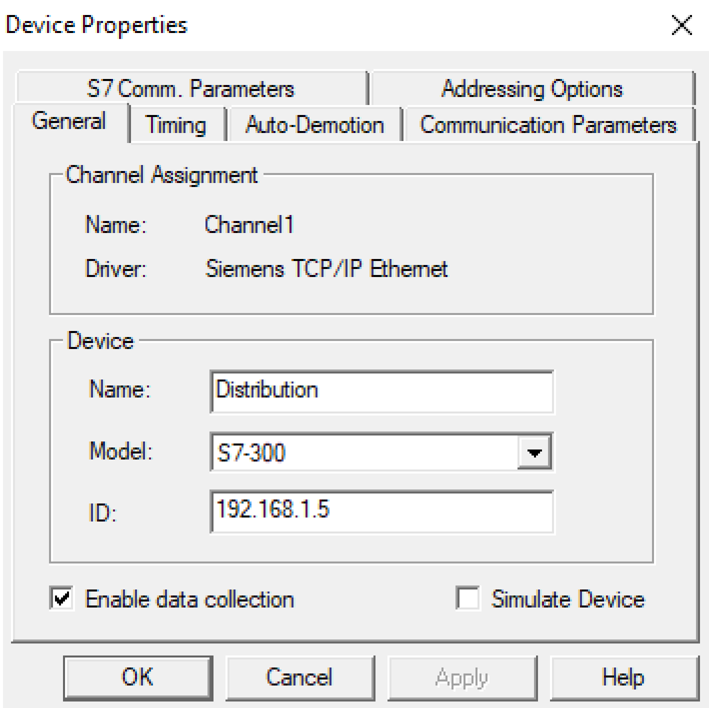


Figure 31 *Device Properties of ET200SP for OPC*

For defining the driver specific station for the selected model, Device ID parameter needs to be defined. The communication ID depends upon the driver that is in use. Normally, it is just a numeric value however for the cases of Ethernet-based driver or unconventional station, TCP/ IP ID is needed. TCP/ IP ID's contain four values with the break using "period" e.g. 192.168.1.1. In this thesis, Ethernet TCP/ IP driver is being used so four values were defined in ID as in case of TCP/ IP. The ID's being used in this thesis are the ones defined in Ethernet settings of the host computer. As we are using two ET200SP's so one additional IP was defined that can be seen in Figure 32.

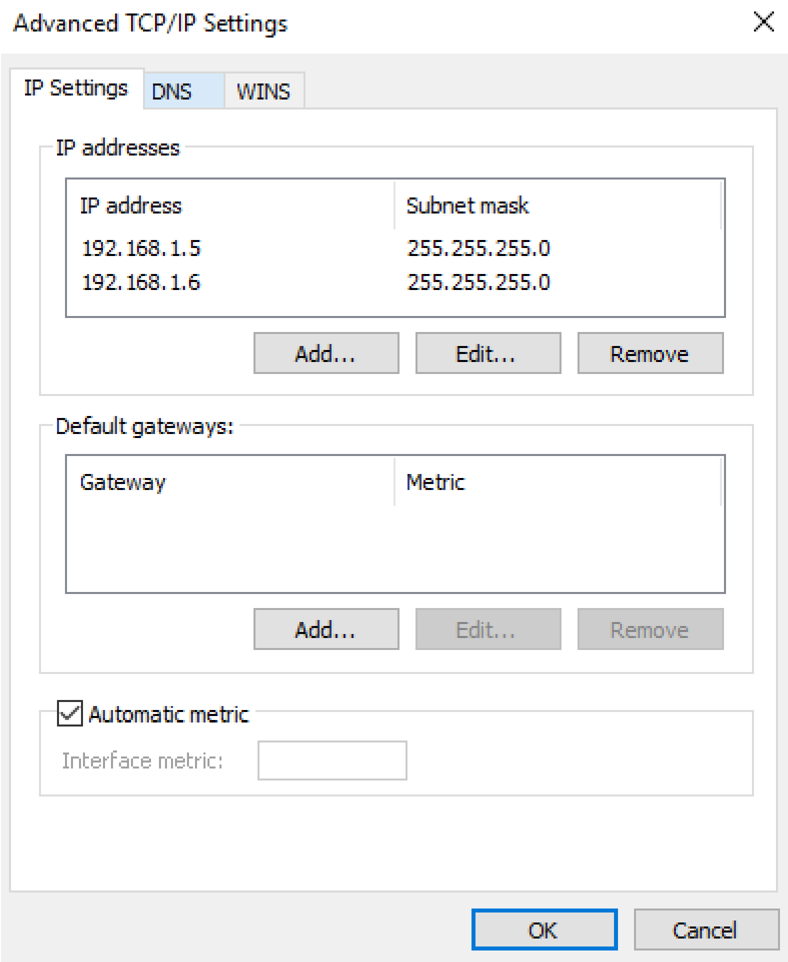


***Figure 32** Device ID- TCP/ IP Settings*

There are various other properties that need to be defined for a device on the channel. Some of those properties include Timing (connect timeout, request timeout, fail after), Communication parameters (Port, MPI ID), Addressing options (Byte order: Big Endian or Little Endian) and S7 communication parameters like Link type (PC, PG, OP), CPU settings e.g. Rack and CPU slot. All of these properties were similarly defined for the other ET200SP (used for Testing Station) except the different device ID.

When the channel is set and device properties have been defined then the next step is to define the tags inside the device. The tags are defined with the same addresses as defined on the hardware side. The address of the tags that were defined in the TIA portal for the control logic, were defined as it is in the device tags section for the establishment of OPC. There are different parameters for defining each input or output. Some of those include name for the tag, description, address as defined in the hardware, data type (boolean, integer, float etc), client access and scan rate.

For analyzing data or some specific parameters of a system, data logging is used as shown in figure 33. It is a process of gathering the information and storing it in the integrated database over a certain period of time. It enables the user to access the stored data and to track all the data exchanges that have been done in some specific time. This process is mostly used in manufacturing industries for the troubleshooting purpose. When a machine stops working then it is part of the troubleshooting process to see the trends of the input and output devices such as temperature sensors, pressure sensors, actuators etc to reach the cause of the problem and usually the problem can be detected from the trends. In the implementation of this thesis, data logging has been done using KepserverEx and data has been stored in Microsoft Access Database so it can be further analyzed.

Table1

ID	mazagie_front_limit_plc	T0	vaccum_suction_sensor_plc	T1	Planar_magpos_sensor_plc	T2	planar_testpos_
422	0	5/6/2019 5:38:01 PM	0	5/6/2019 5:38:01 PM	1	5/6/2019 5:38:01 PM	0
423	0	5/6/2019 5:38:01 PM	0	5/6/2019 5:38:01 PM	1	5/6/2019 5:38:01 PM	0
424	0	5/6/2019 5:38:02 PM	0	5/6/2019 5:38:02 PM	1	5/6/2019 5:38:02 PM	0
425	0	5/6/2019 5:38:02 PM	0	5/6/2019 5:38:02 PM	1	5/6/2019 5:38:02 PM	0
426	0	5/6/2019 5:38:03 PM	0	5/6/2019 5:38:03 PM	1	5/6/2019 5:38:03 PM	0
427	0	5/6/2019 5:38:03 PM	0	5/6/2019 5:38:03 PM	1	5/6/2019 5:38:03 PM	0
428	0	5/6/2019 5:38:04 PM	0	5/6/2019 5:38:04 PM	1	5/6/2019 5:38:04 PM	0
429	0	5/6/2019 5:38:04 PM	0	5/6/2019 5:38:04 PM	1	5/6/2019 5:38:04 PM	0
430	0	5/6/2019 5:38:01 PM	0	5/6/2019 5:38:01 PM	1	5/6/2019 5:38:01 PM	0
431	0	5/6/2019 5:38:05 PM	0	5/6/2019 5:38:05 PM	1	5/6/2019 5:38:05 PM	0
432	0	5/6/2019 5:38:01 PM	0	5/6/2019 5:38:01 PM	0	5/6/2019 5:38:05 PM	0
433	0	5/6/2019 5:38:05 PM	0	5/6/2019 5:38:05 PM	0	5/6/2019 5:38:05 PM	0
434	0	5/6/2019 5:38:06 PM	0	5/6/2019 5:38:06 PM	0	5/6/2019 5:38:06 PM	0
435	0	5/6/2019 5:38:06 PM	0	5/6/2019 5:38:06 PM	0	5/6/2019 5:38:06 PM	0
436	0	5/6/2019 5:38:01 PM	0	5/6/2019 5:38:01 PM	0	5/6/2019 5:38:05 PM	1
437	0	5/6/2019 5:38:07 PM	0	5/6/2019 5:38:07 PM	0	5/6/2019 5:38:07 PM	1
438	0	5/6/2019 5:38:01 PM	0	5/6/2019 5:38:01 PM	0	5/6/2019 5:38:05 PM	1
439	0	5/6/2019 5:38:07 PM	0	5/6/2019 5:38:07 PM	0	5/6/2019 5:38:07 PM	1
440	0	5/6/2019 5:38:08 PM	0	5/6/2019 5:38:08 PM	0	5/6/2019 5:38:08 PM	1
441	1	5/6/2019 5:38:08 PM	0	5/6/2019 5:38:01 PM	0	5/6/2019 5:38:05 PM	1
442	1	5/6/2019 5:38:08 PM	0	5/6/2019 5:38:08 PM	0	5/6/2019 5:38:08 PM	1
443	1	5/6/2019 5:38:09 PM	0	5/6/2019 5:38:09 PM	0	5/6/2019 5:38:09 PM	1
444	1	5/6/2019 5:38:09 PM	0	5/6/2019 5:38:09 PM	0	5/6/2019 5:38:09 PM	1
445	1	5/6/2019 5:38:10 PM	0	5/6/2019 5:38:10 PM	0	5/6/2019 5:38:10 PM	1

Figure 33 *Logged Data of Actual Operation*

4.3.3 Net to PLCSIM

Net to PLCSIM is a useful tool to test and validate the various systems e.g. SCADA, HMI with the Siemens emulator. It enables the user to do network communication with PLCSIM. When PLCSIM is working, this tool uses the network interface of the host PC for the establishment of network communication.

With the utilization of this tool, there is no need for using actual PLC for testing client applications. One of the most useful function is the possibility to run multiple instances of PLC's simultaneously, which can enable the communication between various PLC's. Additionally, it can successfully read and write data blocks also.

A library implemented using S7ProSim-Com-Object was utilized in the first version of Net to PLCSIM. According to website information, this interface was slow. Author also mentioned that S7-1200 and S7-1500 use S7online-interface because they do not have S7ProSim interface [59].

S7online-interface uses the OSI layers for the applications in the Simatic universe. This interface passes the data from the application layer to transport layer using Transmission Control Protocol (TCP)/ Internet Protocol (IP), Multipoint Interface (MPI) or Profibus. Net to PLCSIM tool works with the transport layer (IP/ IsoOnTCP) and data exchange is done with S7online-interface as shown in figure below [59].

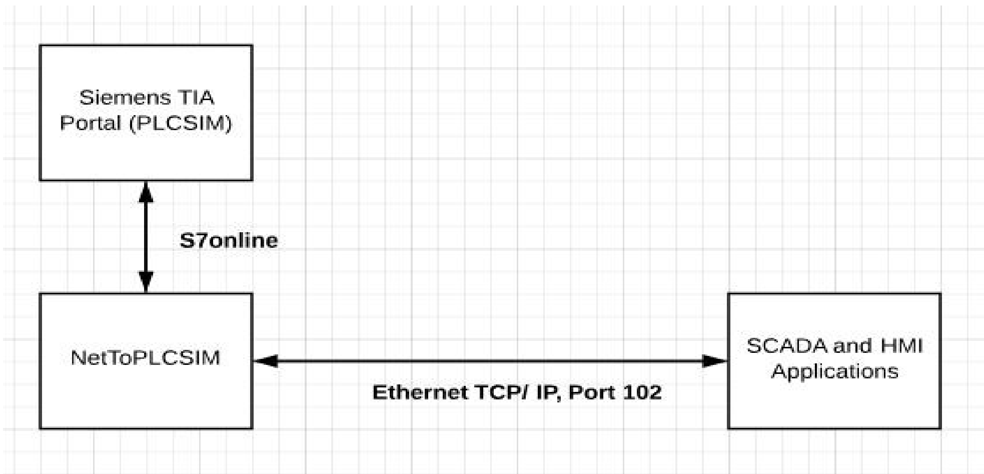


Figure 34 *Net to PLCSIM [59]*

In this thesis, Net to PLCSIM is used for the purpose of testing control code with the mechatronic model. It is also used for connection with visual basic. There was more than one instance of PLC's used and communication was established successfully.

4.3.4 PLC Programming in TIA Portal

A PLC can be defined as a computer that contains memory, Input/ Output components and a processor [61]. CPU permits the change of outputs depending upon the state of input signals and control logic. PLC can also be defined as the industrial computer which is capable of controlling lower-level automation equipment, in a harsh industrial environment. Some events are generated as a result of changes in the industrial environment or manufacturing process and PLC is capable of handling those events in real-time. According to Bolton, PLC is a special form of controller based on a microprocessor, which uses the program memory for storing the control program and implementing that control program using timers, counters, arithmetic and logic. This program is then used for controlling the whole manufacturing processes or part of it depending upon the program and type of PLC used [62].

For the communication with external devices or PC, a variety of communication interfaces are integrated within PLC such as Ethernet, Profibus etc. Generally, a modular structure is followed by PLC and it is beneficial in many terms. One of the benefits is that additional input and output modules can be integrated with the PLC (depending on the type) after the implementation, to cope up with the changes in the original requirement. PLC's are mostly used in manufacturing industries for the control purpose of the process. For handling more complex manufacturing processes, special purpose PLC's can be used which contain additional co-processors for processing the data in a reliable way with great speed. As PLC is a microprocessor-based control system so it can be used for a variety of purposes and applications by changing the programming of the control code.

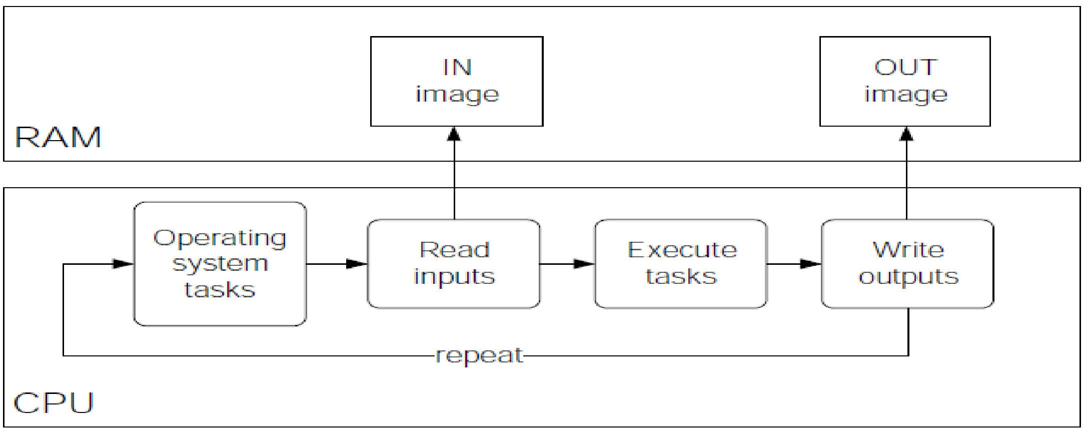


Figure 35 *Cycle of Control Code in PLC [63]*

PLC code which is stored in the program memory is executed in cycles as demonstrated in figure 35. It automatically reserves some computing time for the operating system tasks at the start of scan cycle so all the process can be executed in real-time. The input signals which are received via sensors or any other source are stored in the RAM and according to these received signals, PLC executes the control code. When the control code is executed, the output variables having different values are forwarded towards the out interface so a specific motor/ actuator or any other industrial equipment can be turned on or off.

A new scan cycle is started by the CPU upon the completion of the ongoing cycle, which is usually of few milliseconds. In this way, multiple tasks can be executed in a single cycle. There exist many languages that can be used for writing the control code, according to the requirement. Languages for the PLC programming are standardized and approved by the International Electrotechnical Commission (IEC). This commission published the IEC-61131-3 standard for providing users with enough guidelines and information. This standard also mentioned some methodologies for effectively writing the control code. There are five PLC programming languages recommended by the IEC 61131-3 standard, whose names are:

1. Instruction List
2. Structured Text
3. Function Block Diagram
4. Ladder Diagram
5. Sequential Function Chart

In this thesis, Ladder logic diagram has been used for the programming of both distribution and testing stations. This language originated when the electromechanical systems were used for controlling the manufacturing systems. A ladder program can contain contacts (Inputs) and coils (outputs) along with some elements of the functional block diagram which represent logical expressions. There are methodologies that can be followed while using ladder programming for a big setup. These methodologies include timing diagram, flowchart diagram, sequence bits and state diagram. A well-structured program can be made using these methodologies, which can further help in troubleshooting purpose.

Totally Integrated Automation (TIA) Portal is offered by Siemens for PLC programming. In this engineering tool, it is possible to work with Human Machine Interface (HMI), motion controllers and drives. The interface of TIA portal is easy to understand, which helps to do the programming efficiently. There is also a library of function blocks, already programmed and capable of performing some specific function. This interface is shown in figure 36 and it is easy to understand as compare to Siemens Step7 Programming. One more advantage is that different PLC's can be integrated and programmed in one file. Usually, a system is comprised of more than one controllers and it is easy to work for the programmers if both can be programmed in the same file.

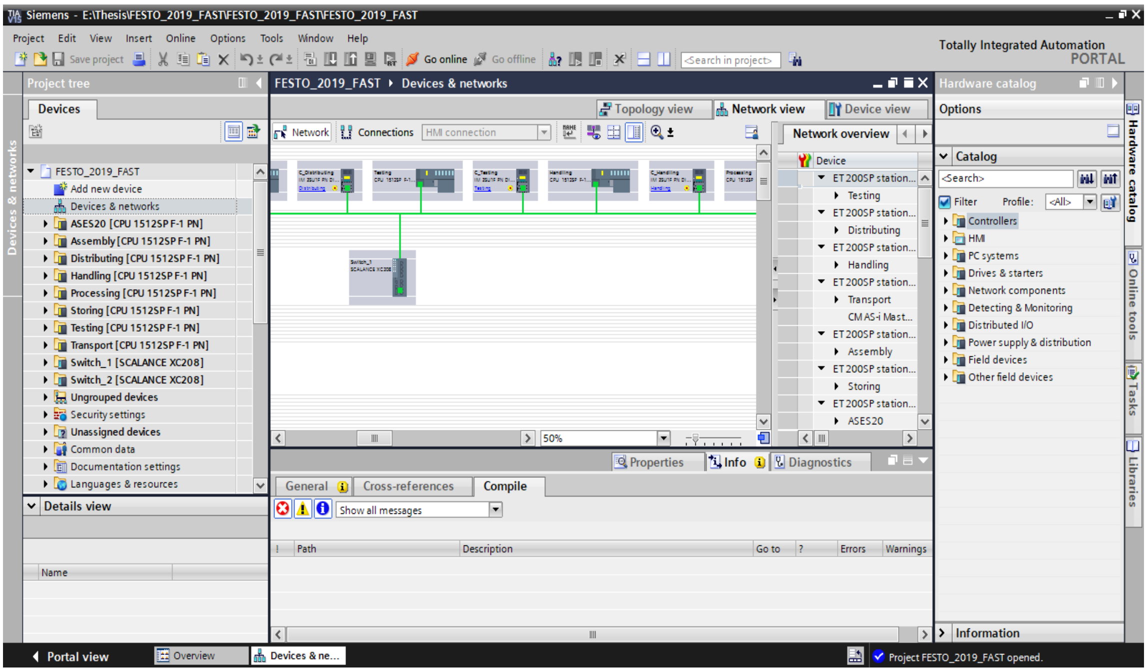


Figure 36 *Interface of TIA Portal*

For programming the Distribution and Testing Stations of Festo MPS 500 system, complete analysis was required. There were a few steps that were followed to completely understand the working of the mentioned stations. Ciros studio and manuals were used for this purpose. Second step was to get familiarize with the actual hardware of the station. Signals addressing was cross-checked with the ones provided in manuals. Simple logic was implemented in the TIA portal and it was downloaded to the actual ET200SP to see the working. After getting the overall idea of the system, the next step was to create a sequence diagram of the whole process, considering all the signals. After the sequence diagram, it is relatively easy to do ET200SP programming in TIA portal. Finally, the operation of both stations was verified by putting workpieces of different color and heights in the system.

Additionally, communication between controllers of distribution and testing station was established for the safety of the actual hardware as shown in figure 37. It was enabled by using the function blocks provided in the library of TIA portal and data blocks used in programming of both stations. ”PUT” function block is used for sending the data from one controller to the other controller. SD_x is for mentioning the bits that need to be sent to the other controller. ADDR_x is the address of the bits of another controller, where data will be sent. The clock of 10Hz has been used with these blocks to get or send the latest data in real-time.

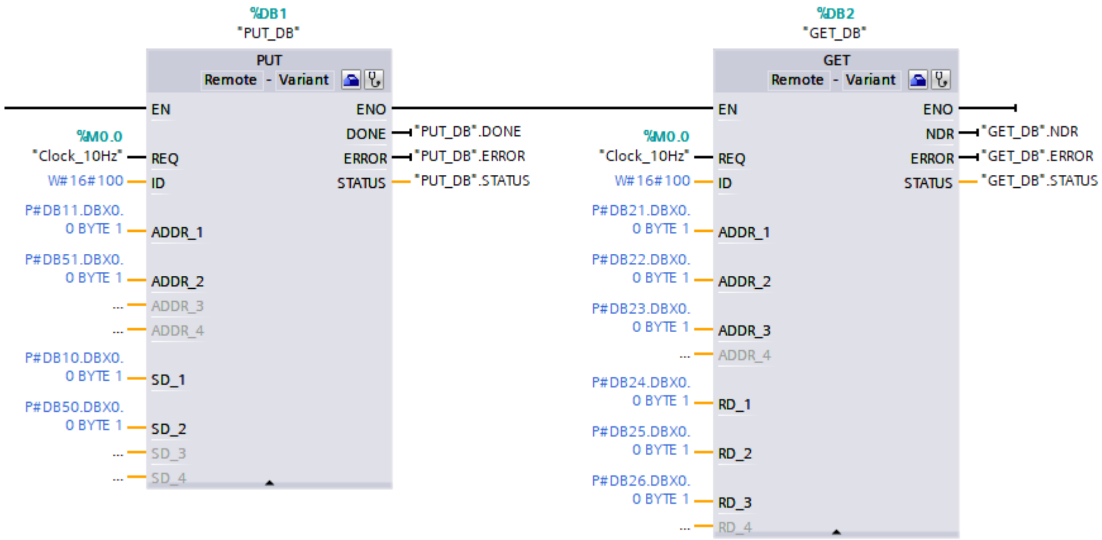


Figure 37 *Data Exchange between ET200SP's of Festo Stations*

Like ”PUT” block, there is also a ”GET” block for receiving the data from the other controllers. Here in ADDR_x, the address of the bits are mentioned from which data will be received. RD_x shows the bits where data of ADDR_x will be stored for further use in control logic. By this way communication between controllers of distribution and testing station was setup. The data exchange between the controllers is necessary for the safety purpose of the actual hardware and workers.

5. RESULTS AND DISCUSSION

This chapter provides detail about the results which were achieved after completion of system implementation as discussed in the previous chapter. Several engineering tools have been used for the completion of the goals defined in the initial phase of this thesis. The designed system was tested and verified with the actual system. Some notable results and discussions regarding the whole system are stated in the section below.

5.1 Colour Sensing in Mechatronics Concept Designer

There is no specific sensor present in the mechatronics concept designer for differentiating the workpieces based on colour. However, it is possible to detect difference of colour using the collision body properties. As shown in figure 38, category is one of the many properties of any collision body in the mechatronics concept designer.

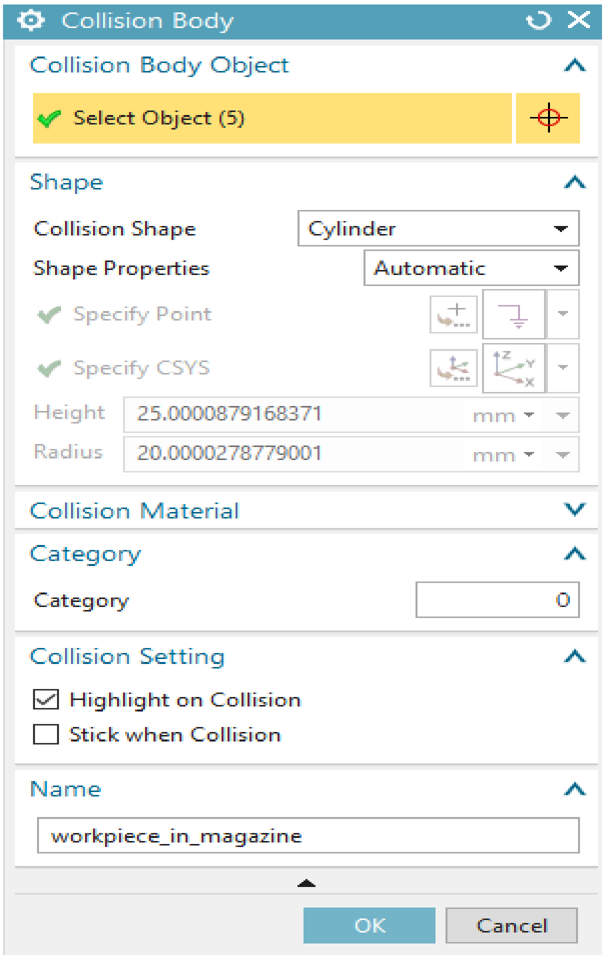


Figure 38 *Category Property of collision body for Color Sensor*

This property enables the system to distinguish workpieces on the basis of colour or height, depending upon the user requirement. The problem is that collision bodies of the same category have an effect on each other and it will interfere with the bodies of other categories. In other words, collision bodies of different categories do not collide with each other and as only the same category bodies have collision properties. It can be used in the mechatronic model if one piece is stacked at a time. This setting cannot be used if there is usage of workpieces having multiple colours as it causes the disturbance in the whole operation.

5.2 Imported 3D Model Joint Displacement in Siemens NX

Initially, 3D models of the Festo MPS 500 system was imported from the Ciros studio. After the connection between Siemens PLM and TIA portal was successfully established, it worked well for a few cycles then it stopped working. Problem is the imported 3D model, which does not necessarily fulfill all the model requirements of Siemens PLM software. After some cycles of operation, its joints (especially cylindrical joints) gets unattached from the original position. I have tried to demonstrate it in figure 39.

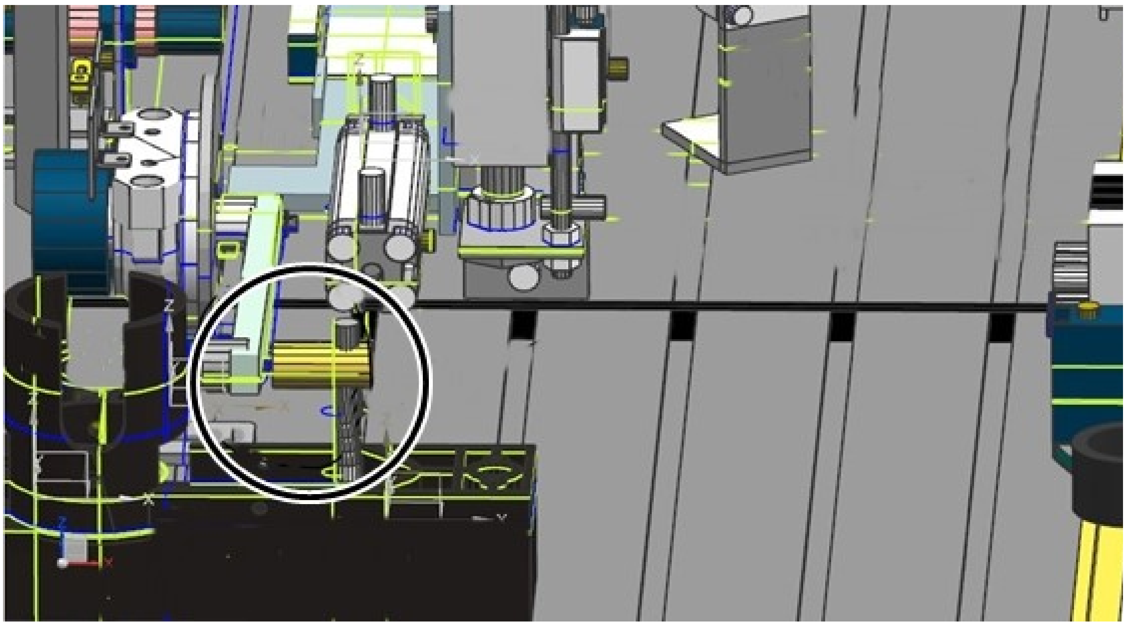


Figure 39 *Problems of Imported 3D Model from Ciros Studio*

The problem can be identified in the picture above as the cylindrical joint of swivel arm (distribution station) gets misplaced after a few cycles of operation. As a solution, I had to redesign most of the moving parts of this mechatronic model. After the imported parts are replaced with the redesigned parts in the software, it works fine.

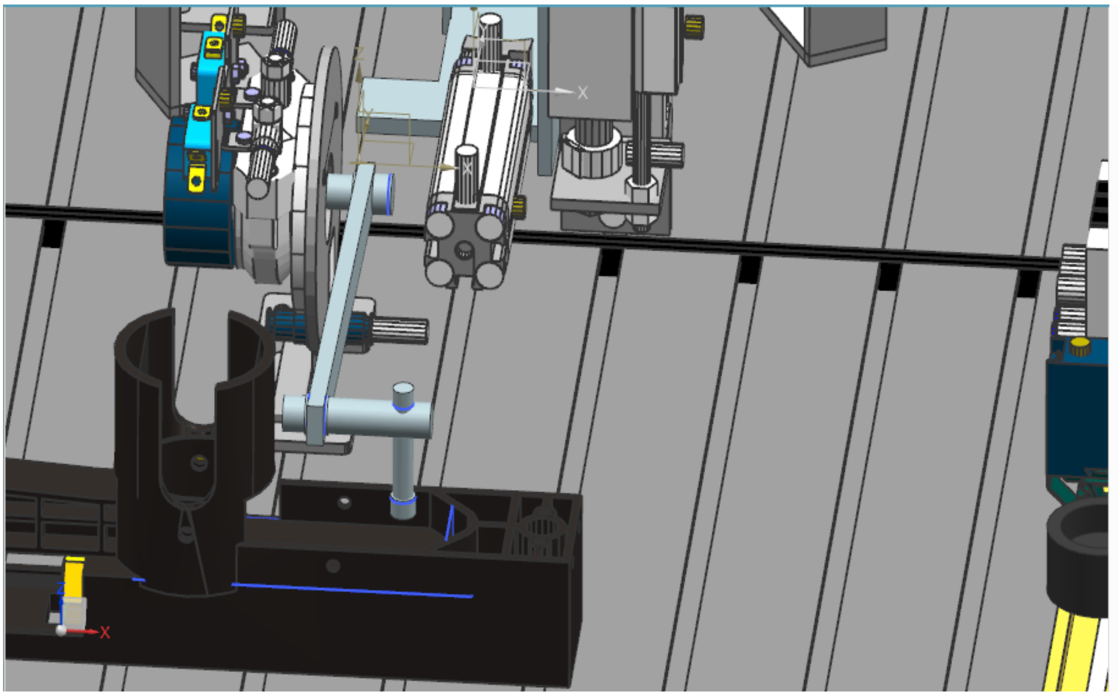


Figure 40 *Designing and Replacement of Swivel Arm*

Without the model which is designed in the Siemens NX, it is a very difficult task to select the edges of the body. The same problem was faced with the lifting module of the testing station. As a solution, it was again designed in the Siemens NX and the imported body was replaced with the newly designed module, as shown in figure 41.

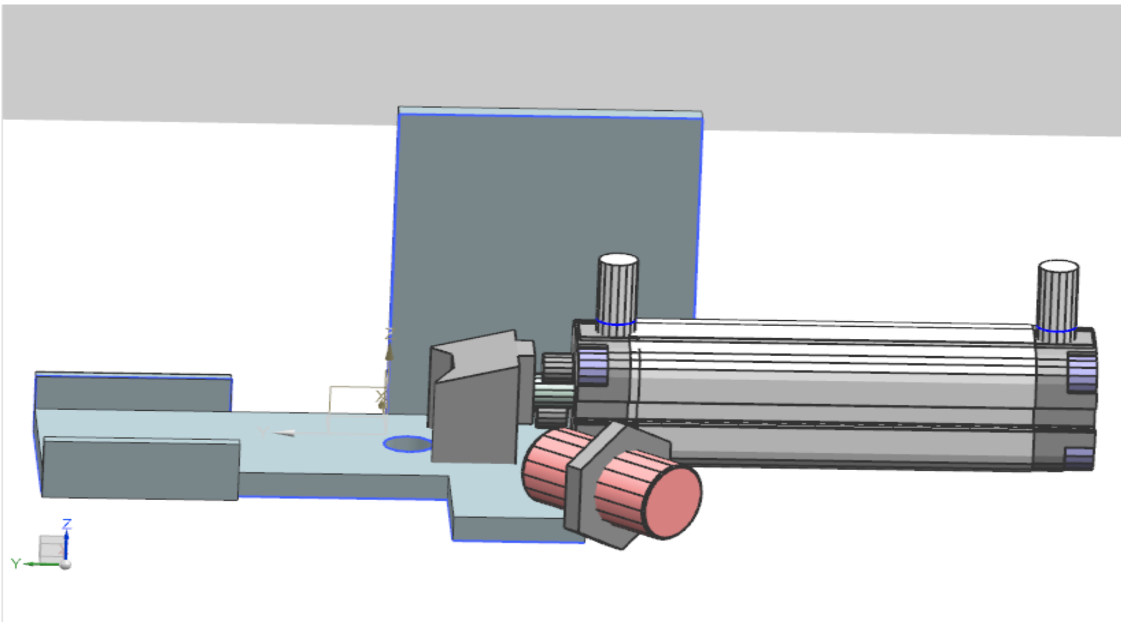


Figure 41 *Re-designed Lifting Module of Testing Station*

5.3 Control of Analog Components of MCD

In Siemens NX, integer values are required for the operation of cylinders, valves and motors. The problem was that in actual hardware only digital signals have been used for the control program. After some study and working with TIA Portal and Siemens NX, I figured out that "Operations" in Siemens NX software can be used for this purpose as shown in figure 42. I am passing the Boolean signal from the PLC and when PLM software receives this Boolean signal, it passes the required integer value to the target part resulting in the successful completion of the operation.

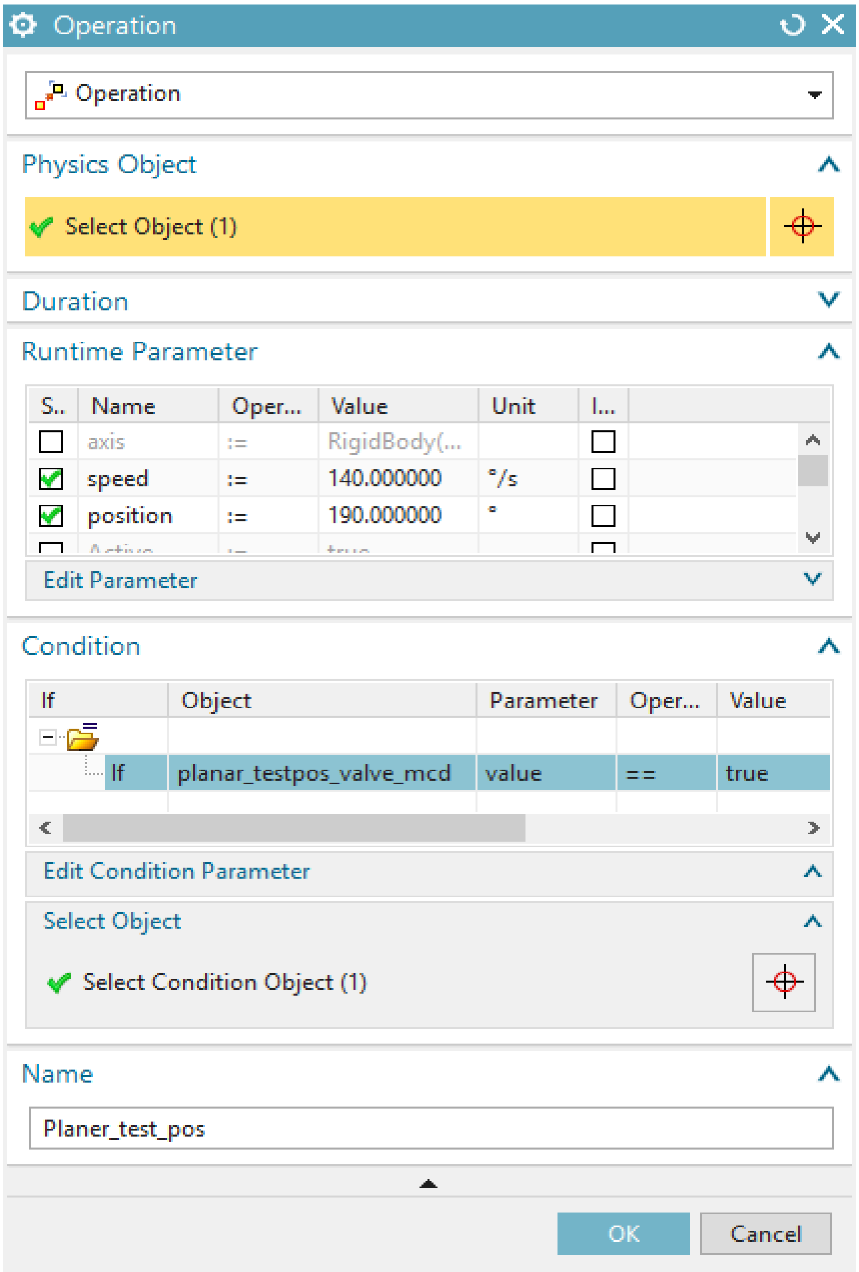


Figure 42 *Controlling Analog Components with Boolean Signals in MCD*

5.4 Verifying Mechatronic Model Operation

Initial testing of the mechatronic model of desired system can be done by using operations of the Siemens NX MCD. Operations enable the user to utilize the internal signals for checking the motion of selected bodies. Appropriate adjustments can be made to meet the requirements. This initial testing can help the designer to make the model more accurate and reliable.

5.5 OPC for ET200SP Controller

OPC has been used for the data exchange between TIA Portal and mechatronic model of distribution and testing station. Kepserverex has been used in this implementation for open platform communication. The problem is that devices should be defined while configuring the OPC for the other engineering tools. At the moment, ET200SP is not available as a device in selected software. After some research and testing, I was able to connect with ET200SP with the settings of S71500 Controller of Siemens as shown in figure 43. This device setting was tested while the integration of all the other engineering tools used in the implementation and communication was established successfully.

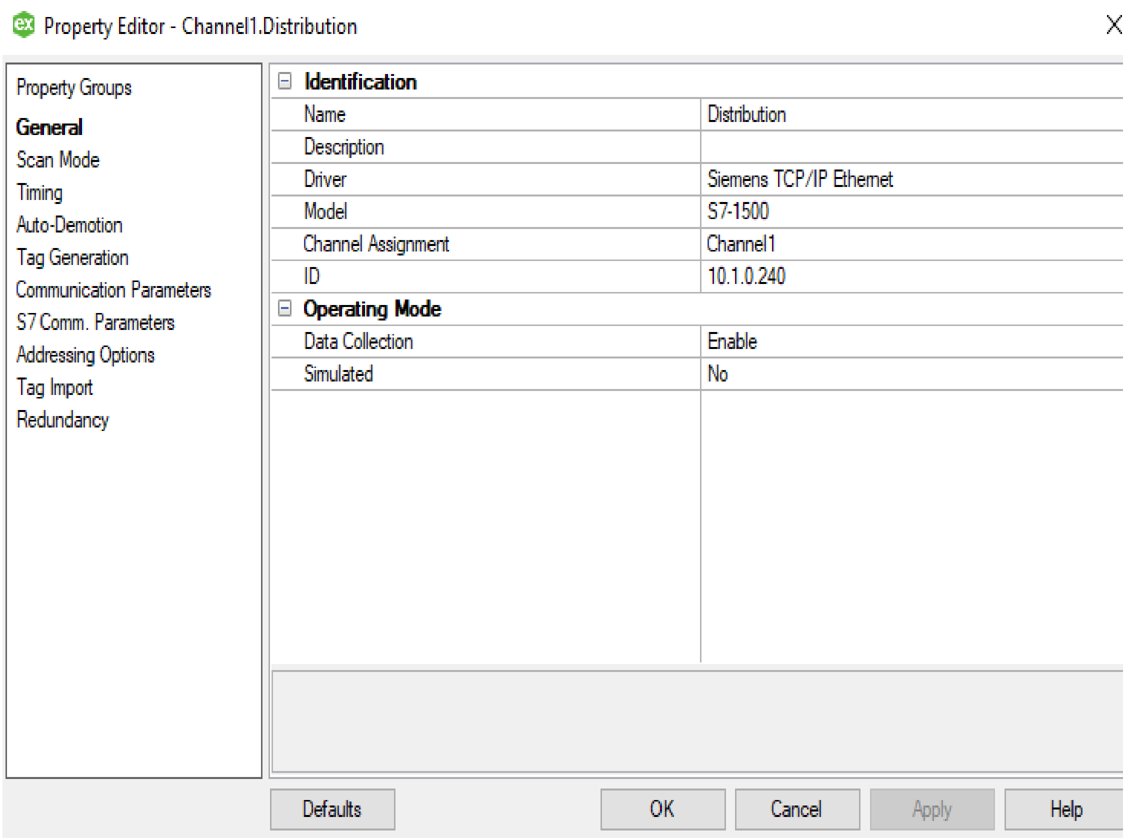


Figure 43 *OPC Device Selection for ET200SP*

5.6 Integrating Mechatronic Model of Desired System

The mechatronic model of distribution and testing stations are made in the Siemens NX. The designed model works very similarly to actual hardware. In this implementation, workpieces are sorted out based on height in the testing station. There is also the possibility of integrating the desired system with the existing one either by modelling the required model in Siemens NX or importing it from some other software. In this way, different approaches can be tested to meet the defined requirements. This enables the user to choose the most efficient and reliable solution for the required operation.

Similarly, it can be used for the training purpose of process operators and students for understanding the operation of the whole process better. They can test their control programs with the simulation without risking their and hardware safety. Different systems can be integrated within the model which can provide new challenging scenarios for the students to program and test.

6. CONCLUSION

This chapter provides a summary of the work that was implemented. Furthermore, the prospects and challenges that need to be solved are also highlighted in this section.

6.1 Summary

This thesis presents an approach for testing and validating the mechatronic model of existing systems. Mechatronic models are the vital part of the digital twin of a system and also for the designing and implementation of new system. The implemented methodology provides the proof of concept that how data of actual operation of a process can be stored and re-used for validating and verifying the mechatronic model of the system under consideration.

A system has been developed with the integration of various engineering tools. These tools enable the system to store the process data while the operation is being performed. The data logging can be performed using the third party software with the appropriate configuration. The presented system has the ability to work with more than one controllers simultaneously.

The connection between the visual basic program and programmable logic controller was established using a library, which uses Ethernet connection and works with all the Siemens controllers. The program is able to fetch data stored by the actual hardware operation and forward it using the network connection between the control program and visual basic program.

As Siemens controllers have been used for the controlling purpose of the distribution and testing stations so emulators of both stations were set up for the interaction with mechatronic model. The driver for the data exchange between control program and other tools uses TCP/ IP Ethernet. After the communication link establishment between the database and control program, OPC was utilized for the interaction with mechatronic model. OPC data access has been used for the sever detection from the Siemens NX side.

By utilizing the operations in Siemens NX MCD, it is possible to control the analog values of the desired components with the boolean signal of an external controller. The designed system is able to differentiate and sort out the workpieces based on height. Additionally, changes can be made in the presented model for adding new system components or removing the existing ones. With the proposed methodology, new integrated system can be tested and verified by making adequate changes in the mechatronic model.

6.2 Future Work

The implemented system provides the idea about validating and verifying the mechatronic model of the existing system. However, there are areas which can be explored for the development of more functionalities of this system. Following the proposed methodology, other hardware systems can be verified and tested. It can be used for comparing the designed system with the conceptual model.

The methodology is implemented using the Siemens NX engineering tool. The library of the software needs to be expanded. With the newly added tools in the library, it will be possible for the designers to implement different and effective solutions. There is also very limited information available about implementing a mechatronic system in Siemens mechatronics concept designer. It will be very helpful for future researchers if there are more tutorials and documentation for the development of a prototype or complete digital twin of any existing physical system.

The data of process parameters of an actual physical system, stored in the database can be used for automatically generating information which can be utilized by the management, leading to a more sophisticated system. It can help the organization in planning and producing the output as required.

REFERENCES

[1] Research, development, and firm growth. Empirical evidence from European top R&D spending firms, in: Research Policy, 2012, pp. 1084-1092

[2] P. Goossens Industry 4.0 and the Power of the Digital Twin, web page. Available: https://www.maplesoft.com/ns/manufacturing/industry-4-0-power-of-the-digital-twin.aspx

[3] J. Persson, J. Norrman, Virtual Production Line- Virtual Commissioning, Lund University, 2018

[4] Ali Heidari, Oliver Salamon, Virtual Commissioning of an Existing Manufacturing Cell at Volvo Car Corporation Using DELMIA V6, Chalmers University of Technology, 2012

[5] M. Grieves, Digital Twin: Manufacturing Excellence through Virtual Factory Replication, 2014

[6] Reiner Anderl, Sebastian Haag, Digital twin – Proof of concept, in: Manufacturing Letters, Elsevier, Technische Universität Darmstadt, Department of Computer Integrated Design (DiK), Darmstadt, Germany, 2018, pp. 64-66

[7] M. Rüßmann, M. Lorenz, P. Gerbert, M. Waldner, J. Justus, P. Engel, M. Harnisch, Industry 4.0: The future of productivity and growth in manufacturing industries, Boston Consulting Group, 2015

[8] R. Drath, A. Horch, "Industrie 4.0: Hit or Hype?", IEEE Industrial Electronics Magazine, vol. 8, no. 2, pp. 56-58, 2014

[9] H. Kagermann, W. Wahlster, J. Helbig, Recommendations for implementing the strategic initiative Industrie 4.0: Final report of the Industrie 4.0 Working Group, Berlin, 2013

[10] Mario Hermann, Tobias Pentek, and Boris Otto. Design principles for industrie 4.0 scenarios. In System Sciences (HICSS), 2016 49th Hawaii International Conference, pages 3928–3937. IEEE, 2016

[11] D.M. Grieves, Digital Twin: Manufacturing Excellence through Virtual Factory Replication, 2014, pp. 1-7

[12] Roland Rosen, Georg Von Wichert, George Lo, and Kurt D. Bettenhausen. About the importance of autonomy and digital twins for the future of manufacturing, IFAC PapersOnLine, 48(3):567–572, 2015.

[13] Jack B. Reid, Donna H. Rhodes, Digital System Models: An investigation of the non-technical challenges and research needs, Massachusetts Institute of Technology, 2016.

[14] Christopher Ganz, Digital twin – virtually identical? 2018, web page. Available: https://new.abb.com/news/detail/5080/digital-twin-virtually-identical

[15] Robert H. Bishop, The Mechatronics Handbook, CRC Press LLC, USA, 2002.

[16] Auslander, D. M. and Kempf, C. J., Mechatronics: Mechanical System Interfacing, Prentice-Hall, Upper Saddle River, NJ, 1996

[17] Kleanthis Thramboulidis, Challenges in the development of Mechatronic systems: The Mechatronic Component, IEEE, Hamburg, Germany, 15-18 Sept. 2008,

[18] Otto Mayr, The Origins of Feedback Control, MIT Press, Cambridge, 1970

[19] Richard C. Dorf, Robert H. Bishop, Modern Control Systems, 12th ed. Pearson Education, Inc., Upper Saddle River, New Jersey

[20] Vimela Project: An Innovative Concept for Teaching Students in Mechatronics using Virtual Reality, The 7th International Symposium on Applied Electromagnetics, Podčetrtek, Slovenia, 2018

[21] Peter Hoffmann, On Virtual Commissioning of Manufacturing Systems, University of South Wales, 2016

[22] Haq, I., Monfared, R., Harrison, R., Lee, L. and West, A new vision for the automation systems engineering for automotive powertrain assembly, International Journal of Computer Integrated Manufacturing, 23, 4, 2010, pp. 308-324.

[23] Fredrik Danielsson, Philip Moore, Patric Eriksson, Validation, off-line programming and optimisation of industrial control logic, Mechatronics 13, 2003, pp. 571–585.

[24] David J. Dougall, Applications and benefits of real-time I/0 simulation for PLC and PC control systems, ISA Transactions, Vol. 36, 1998

[25] Robert E. Shannon, Simulation Modeling and Methodology, 6-8 December 1976, Prentice-Hall, Inc., pp. 9-15

[26] Timothy S. Meinert, John R. English, G. Don Taylor, A modular simulation approach for automated material handling systems, Simulation Practice and Theory, Vol. 7, Issue. 1, 1999, pp. 15-30.

[27] Jerry Banks, Introduction to Simulation, AutoSimulations, Inc., 1999, pp. 7-13.

[28] Devinder Thapa, Chang Mok Park, Suraj Dangol, Gi-nam Wang, III-Phase Verification and Validation of IEC Standard Programmable Logic Controller, 28 Nov.-1 Dec. 2006, IEEE, Sydney, NSW, Australia

[29] Robert E. Shannon, Introduction to Art and Science of Simulation, 1998, pp. 7-14

[30] C. Andersson, P. Runeson, Verification and Validation in Industry-A Qualitative Survey on the State of Practice, Proceedings of the 2002 International Symposium on Empirical Software Engineering, October 03 - 04, 2002, IEEE Computer Society, Washington, DC, USA, pp. 37–47

[31] IEEE Standard for System and Software Verification and Validation, IEEE Std 1012-2012 (Revision of IEEE Std 1012-2004), 2012, pp. 1-223. https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=6204026

[32] P.G. Maropoulos, D. Ceglarek, Design verification and validation in product lifecycle, CIRP Annals, Vol. 59, Issue. 2, 2010, pp. 740-759. http://www.sciencedirect.com/science/article/pii/S0007850610001927

[33] Markus Rabe, Sven Spieckermann, Sigrid Wenzel, a new procedure model for verification and validation in production and logistics simulation, Proceedings of the 2008 Winter Simulation Conference, pp. 1717-1726

[34] Kjell B. Zandin, Statistics and Operations Research and Optimization - Simulation Methodology, Tools, and Applications, in: 5th (ed.), Maynard's Industrial Engineering Handbook, McGraw-Hill Education, 2001, pp. 101-119

[35] T. Vepsäläinen, S. Kuikka, Integrating model-in-the-loop simulations to model-driven development in industrial control: SIMULATION, Vol. 90, Issue. 12, 2014, pp.1295-1311

[36] Richard Phillips, Bonnie Montalvo, Using Emulation to Debug Control Logic Code, Proceedings of the 2010 Winter Simulation Conference, 5-8 Dec. 2010, IEEE, Baltimore, MD, USA

[37] Jasmin Dzinic, Charlie Yao, Simulation-based Verification of PLC Programs, Chalmers University of Technology, 2013, 1-33 p, Gothenburg, Sweden

[38] Herbert Krause, Virtual commissioning of a large LNG plant with the DCS" 800XA" by ABB, 2007, https://www.secolon.de/P172.pdf

[39] R. Isermann, J. Schaffnit, S. Sinsel, Hardware-in-the-loop simulation for the design and testing of engine-control systems, Control Engineering Practice, Vol. 7, Issue. 5, 1999, pp. 643-653

[40] M. Bacic, On hardware-in-the-loop simulation, Proceedings of the 44th IEEE Conference on Decision and Control, 15 Dec. 2005, IEEE, Seville, Spain, pp. 3194-3198

[41] Novotek OPC and OPC UA Explained, https://www.novotek.com/en/solutions/kepware-communication-platform/opc-and-opc-ua-explained. [Accessed: 09-04-2019]

[42] M. Hincapié, M.d.J. Ramírez, A. Valenzuela, J.A. Valdez, Mixing real and virtual components in automated manufacturing systems using PLM tools, International Journal on Interactive Design and Manufacturing (IJIDeM), Vol. 8, Issue. 3, 2014, pp. 209-230. https://link-springer-com.libproxy.tuni.fi/article/10.1007/s12008-014-0206-7

[43] L.V. Guerrero, V.V. López, J.E. Mejía, Virtual Commissioning with Process Simulation (Tecnomatix), Computer-Aided Design and Applications, Vol. 11, Issue. sup1, 2014, pp. 11-19. https://doi.org/10.1080/16864360.2014.914400

[44] R. Wischnewski, E. Freund, COSIMIR Transport: Modeling, Simulation and Emulation of Modular Carrier based Transport Systems, IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA '04. 2004, 26 April-1 May 2004, IEEE, New Orleans, LA, USA, pp. 5171-5176

[45] Yong S. Kim, Kee Y. Shin, Jin H. Lee, Sang S. Lee, Kwang S. Kim, Kyo C. Kang, Jin S. Yang, Application of Virtual Commissioning Technology in a Steel Making Industry, 2013 13th International Conference on Control, Automation and Systems (ICCAS 2013), 20-23 Oct. 2013, IEEE, Gwangju, South Korea, pp. 1718-1720

[46] Duško Lukač, Comparative Selection of Industrial Robot Simulation Systems for Educational Purposes, Infoteh-Jahorina, pp. 859-863

[47] Kuka Smart components – ready for Industrie 4.0, https://www.kuka.com/en-de/products/robot-systems/software/planning-project-engineering-service-safety/kuka_sim. [Accessed: 10-04-2019]

[48] Dennis Binnberg, Viktor Johansson, Virtual Commissioning, Emulation of a production cell, University of Skövde and Volvo Group Truck Operations, 2016, 1-71 p

[49] FACTORY I/O, https://factoryio.com/docs/ [Accessed: 11-04-2019]

[50] Mechatronic Concept Design (MCD), https://www.plm.automation.siemens.com/global/en/products/mechanical-design/mechatronic-concept-design.html [Accessed: 11-04-2019]

[51] Duško Lukač, Miljana Milić, Simulation of a Pick-and-Place Cube Robot by Means of the Simulation Software Kuka Sim Pro, Vol. 21, Issue. 2, 2017, pp. 95-99.

[52] Sanna Älegård, Stefan Knutsson, Virtual Commissioning of Smart Factory, Chalmers University of Technology, Gothenburg, Sweden, 2017, 1-35 p

[53] Visual Components – A world leader in 3D simulation, https://www.visualcomponents.com/about-us/ [Accessed: 11-04-2019]

[54] Visual Components 4.1 - The next generation of 3D manufacturing simulation, https://www.visualcomponents.com/products/visual-components-4-0/ [Accessed: 11-04-2019]

[55] In-House Solutions Introduces OCTOPUZ Robotic and Simulation Software, 2014

[56] Aksel Øvern, Industry 4.0 - Digital Twins and OPC UA, Norwegian University of Science and Technology, 2018, 1-111 p

[57] Siemens System Overview - I/O Systems - Siemens, http://w3.siemens.com/mcms/distributed-io/en/ip20-systems/et-200sp/system-overview/pages/default.aspx. [Accessed: 12-04-2019]

[58] Siemens Distributed Controller based on SIMATIC ET 200SP, https://w3.siemens.com/mcms/programmable-logic-controller/en/distributed-controller/et200sp-based/pages/default.aspx [Accessed: 12-04-2019]

[59] NetToPLCsim - Network extension for PLCSIM, http://nettoplcsim.sourceforge.net/ [Accessed: 13-04-2019]

[60] Kepware Technologies, OPC Server Help, http://www.opcturkey.com/uploads/manuals/kepserverex-v4-manual.pdf.

[61] E. Dummermuth, O. Chesterland, Programmable Logic Controller, 3,942,158, 473,149.

[62] W. Bolton, Programmable Logic Controllers, Fifth ed. Newnes, Amsterdam, 2009.

[63] Manuel Goster, Concept and Implementation of a Factory Simulation, Ulm University, Germany, 2017

[64] Siemens,Siemens Documentation: Operation, https://docs.plm.automation.siemens.com/tdoc/nx/12/nx_help/#uid:id1109424.

[65] NetObjex How Can Digital Twin Technology Benefit Your Organization? https://www.netobjex.com/how-can-digital-twin-technology-benefit-your-organization/ [Accessed: 12-04-2019]

[66] Modelling, Simulation and Optimization in a Data rich Environment, Foundation EU-MATHS-IN, European Service Network of Mathematics for Industry, Netherlands, 1-44 p.