



저작자표시-비영리-변경금지 2.0 대한민국

이용자는 아래의 조건을 따르는 경우에 한하여 자유롭게

- 이 저작물을 복제, 배포, 전송, 전시, 공연 및 방송할 수 있습니다.

다음과 같은 조건을 따라야 합니다:



저작자표시. 귀하는 원저작자를 표시하여야 합니다.



비영리. 귀하는 이 저작물을 영리 목적으로 이용할 수 없습니다.



변경금지. 귀하는 이 저작물을 개작, 변형 또는 가공할 수 없습니다.

- 귀하는, 이 저작물의 재이용이나 배포의 경우, 이 저작물에 적용된 이용허락조건을 명확하게 나타내어야 합니다.
- 저작권자로부터 별도의 허가를 받으면 이러한 조건들은 적용되지 않습니다.

저작권법에 따른 이용자의 권리는 위의 내용에 의하여 영향을 받지 않습니다.

이것은 [이용허락규약\(Legal Code\)](#)을 이해하기 쉽게 요약한 것입니다.

[Disclaimer](#)

Doctor of Philosophy

**Implement of digital twin for predicting and monitoring  
grinding process behavior**



The Graduate School

of the University of Ulsan

Department of Mechanical Engineering

BOWEN QI

**Implement of digital twin for predicting and monitoring  
grinding process behavior**

Supervisor: Professor Chang Myung Lee

A Dissertation

Submitted to

the Graduate School of the University of Ulsan

In partial Fulfillment of the Requirements

for the Degree of

Doctor of Philosophy in Mechanical Engineering

by

BOWEN QI

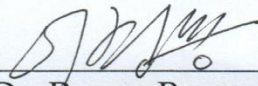
Department of Mechanical Engineering

University of Ulsan, Korea

[August 2022]

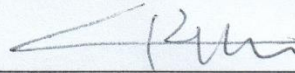
Implement of digital twin for predicting and monitoring  
grinding process behavior

This certifies that the dissertation of BOWEN QI is approved.



---

Committee Chair Dr. Byung Ryong Lee




---

Committee Member Dr. Doo Man Chun



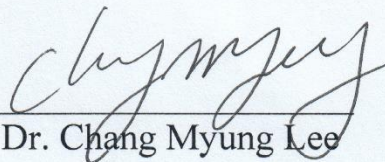
---

Committee Member Dr. Duck Young Kim



---

Committee Member Dr. Hyun Joon Chung



---

Committee Member Dr. Chang Myung Lee

Department of Mechanical Engineering

University of Ulsan, Korea

[August 2022]

*Dedicated to my family and friends who have supported me the whole way.*

## **Acknowledgment**

First of all, I would like to thank my advisor, Prof. Hong Seok Park, and Prof. Chang Myung Lee, especially Prof. Park, who gave me an excellent opportunity to work in the Laboratory of Production Engineering at the University of Ulsan. Apart from imparting immense knowledge of research, they showed kindness and patience, whose support, advice, and encouragement constantly guided me on the long road of this doctoral journey.

It has been my pleasure to get training from Prof. Park in terms of doing research work and the successful completion of any research project from the start with satisfactory research outcomes. During my research, I got the opportunity to work with industries to solve real manufacturing-related problems. I am incredibly grateful to the lab members, past and present, who have been friends, wonderful colleagues, and have been a support throughout the conduct of this piece of research as well as my stay at the University of Ulsan.

I would also take this opportunity to thank Prof. Byung Ryong Lee, Prof. Doo Man Chun, Prof. Chang Myung Lee, Prof. Duck Young Kim, and Prof. Hyun Joon Chung for their insightful comments, suggestions, and encouragement, which helped improve the quality of this dissertation.

Last but not least, I would like to thank my parents for their continuous and unconditional support throughout my study abroad. They have always been a source of motivation.

Ulsan, May 2022

**Bowen Qi**

## **Abstract**

The market is rapidly evolving in response to the growing demands of customers. To overcome these difficulties, innovative and efficient approaches for issue identification and visualization are required. Digital twin gives an approach to manufacturers to gain a clearer picture of real-world performance and operating conditions of a manufacturing asset via near real-time data captured from the asset and make proactive optimal operation decisions. It was discovered that the digital twin is a complicated notion currently in its early stages of growth, with differing viewpoints on how it should be defined across various professions and academia. The presented dissertation aims to develop and implement a process prediction system through a digital twin-based system for testing the process in virtual space before implement to shopfloor, as well as guidelines for implementing and creating a digital twin for companies within the production industry.

Grinding is one of the most common precision machining methods because of its high machining efficiency and good finish quality. In the traditional grinding process, in order to set the appropriate machining parameters, researcher use the trial error method to calculate the appropriate machining parameters, which takes a lot of time. The grinding force is the most important parameter in the grinding process which affects the product quality. Today's products are become more individualized and high uncertainty, it is necessary to reduce the product design cycle of products and improve production efficiency. In this dissertation, the digital twin technology was implemented to the grinding process for predicting process behavior. By testing the machining parameters in advance, predicting the machining behavior, optimizing the machining parameters through the test results. At the same time, in the actual on-site machining process, by connecting with industrial PLC and sensors, input real-time machining parameters and sensors data, and calculate the cutting state under ideal conditions. These values

are compared with the actual values as reference values, and if there is a deviation and exceeds the threshold, it means that there is a failure in the production process. On-site operators will inspect the machine and product.

The proposed methodology for grinding behavior prediction using a digital twin approach, with the vertical double side grinding machine performing the required work while connecting the PLC program. The proposed approach integrates the information obtained from the sensors, physic models, and operation of the system to establish the grinding machine model and carry out grinding force. Simulation results show the proper connection between models and communication. The digital model was established to exactly match the operation of the physical system. Comparison between predicted result obtained from the proposed digital twin model and experiment, revealed a good agreement between proposed model and practice, indicating therefore that the model may be suitable for industrial applications further.

The created system is installed on a SIEMENS-based platform in industrial PC and is compatible with a variety of additional programming tools. The technology makes a substantial contribution to Industry 4.0 and the autonomous operation of production facilities. It has increased total brake disc production productivity by lowering the number of quality failures and minimizing reliance on operators for process knowledge.

This dissertation will begin with an overview of related literature before moving on to the research strategies used in the current study. It assesses the notion of digital twins for smart factories as well as the technology underlying them. It examines the different definitions of a digital twin, how far technology has progressed, and what manufacturing organizations need to deploy and develop digital twins. The development of a digital twin-based system, as well as its mechatronic modeling, deployment, and manufacturing implementation, are covered in



the following sections. The final section informs the producers about the advantages of this technology.

Table of Contents	
<b>Acknowledgment</b> .....	<b>iv</b>
<b>Abstract</b> .....	<b>i</b>
<b>Table of Contents</b> .....	<b>iv</b>
<b>List of Figures</b> .....	<b>vi</b>
<b>List of Tables</b> .....	<b>viii</b>
<b>Chapter 1 . GENERAL INTRODUCTION</b> .....	<b>9</b>
1.1. Motivation .....	9
1.2. State of art .....	11
1.2.1. Concept and definition of a digital twin.....	11
1.2.2. Application fields of digital twin.....	15
1.2.3. Technologies of smart manufacturing for digital twin applications .....	19
1.2.4. Digital twin-based performance testing.....	21
<b>Chapter 2 . RESEARCH METHODOLOGY</b> .....	<b>23</b>
2.1. Research object.....	23
2.2. Research strategy followed .....	25
2.3. Verification and validation .....	26
2.4. Testing control programs .....	35
<b>Chapter 3 . MODELLING OF GRINDING PROCESS</b> .....	<b>38</b>
3.1. Modelling of grinding force .....	38
3.2. Modelling of power flow.....	41
3.3 Theoretical model for the total spindle power calculation .....	45
3.3.1. Mechanical Losses in spindle.....	46
3.3.2. Electrical Loss .....	53
3.3.3. Module for power flow.....	55
<b>Chapter 4 . DEVELOPMENT OF DIGITAL TWIN-BASED SYSTEM</b> .....	<b>57</b>
4.1. System architecture .....	57
4.2. Virtual process layer .....	59
4.2.1. Virtual model.....	59
4.3. Data process layer .....	67
4.4. Communication and data mapping layer .....	68

4.4.1 Open platform communications .....	68
4.4.2 Data mapping strategy .....	72
4.4.3 Net to PLCSIM.....	75
4.5 Virtual controller .....	76
4.6. Application .....	84
<b>Chapter 5 . RESULT AND DISCUSSION .....</b>	<b>86</b>
5.1.1. Implementation of developed system .....	86
5.1.2. Integrate to the control room .....	87
5.2. Functionality testing of developed system .....	88
5.2.1. Real factory environment testing.....	88
5.3. Experiment and result.....	89
<b>Chapter 6 . CONCLUSION AND FUTURE WORK.....</b>	<b>92</b>
<b>Publications .....</b>	<b>93</b>
<b>Reference .....</b>	<b>95</b>
<b>Appendix : Experiment data.....</b>	<b>107</b>

## List of Figures

<b>Figure 1.1.</b> Conceptual idea of digital twin.....	11
<b>Figure 1.2.</b> Elements of digital twin.....	12
<b>Figure 1.3.</b> RAMI 4.0.....	17
<b>Figure 1.4.</b> System architecture for digital twin test.....	22
<b>Figure 2.1.</b> Manufacturing process of brake disc.....	23
<b>Figure 2.2.</b> Digital twin system for evaluating grinding process behavior .....	24
<b>Figure 2.3.</b> Research strategy followed.....	26
<b>Figure 2.4.</b> Verification and validation of simulation Model.....	28
<b>Figure 2.5.</b> Software in the loop.....	31
<b>Figure 2.6.</b> Hardware in the loop.....	34
<b>Figure 2.7.</b> Real control system method.....	36
<b>Figure 2.8.</b> Control system emulator method.....	37
<b>Figure 3.1.</b> Grinding process of brake disc.....	38
<b>Figure 3.2.</b> effective rotation angle of the CBN grinding segment.....	41
<b>Figure 3.3.</b> Analysis of mechanical mechanism of components along force flow.....	42
<b>Figure 3.4.</b> Power flow in a motorized spindle.....	42
<b>Figure 3.5.</b> Free-body schematic diagram of the spindle.....	43
<b>Figure 3.6.</b> Ball angular speed vector in a non-zero ball-raceway contact.....	50
<b>Figure 3.7.</b> Power loss due to the spin relation friction in bearings vs. spindle speed.....	52
<b>Figure 3.8.</b> Flowchart for calculate effective grinding power.....	56
<b>Figure 4.1.</b> System architecture of digital twin system.....	58
<b>Figure 4.2.</b> 3D model of components.....	60
<b>Figure 4.3.</b> sliding mechanism.....	61
<b>Figure 4.4.</b> gripper assembly.....	61

<b>Figure 4.5.</b> Final assembly of the grinding production line.....	62
<b>Figure 4.6.</b> basic physics of the grinding production line.....	63
<b>Figure 4.7.</b> Joints and constraints of virtual model.....	64
<b>Figure 4.8.</b> Model coupling through gears.....	65
<b>Figure 4.9.</b> Actuators for twin movement.....	66
<b>Figure 4.10.</b> Open platform communications.....	69
<b>Figure 4.11.</b> Data acquisition and exchange module.....	71
<b>Figure 4.12.</b> Digital twin data mapping strategy.....	73
<b>Figure 4.13.</b> Signals adapter.....	74
<b>Figure 4.14.</b> Signal Mapping of digital twin.....	75
<b>Figure 4.15.</b> Net to PLCSIM.....	76
<b>Figure 4.16.</b> Cycle of control code in PLC.....	78
<b>Figure 4.17.</b> Virtual controller in TIA Portal.....	80
<b>Figure 4.18.</b> Data Exchange.....	81
<b>Figure 4.19.</b> Virtual PLC encoding.....	82
<b>Figure 4.20.</b> Motion Control.....	83
<b>Figure 4.21.</b> Application of implemented system.....	85
<b>Figure 5.1.</b> Implementation of the system at the site.....	86
<b>Figure 5.2.</b> Control room display for the plant manager.....	87
<b>Figure 5.3.</b> Real time experiment of digital twin system.....	90

## List of Tables

<b>Table 1.1.</b> Definitions of digital twin.....	14
<b>Table 1.2.</b> Application field of manufacturing digital twin.....	16
<b>Table 3.1.</b> Values of $f_v$ depending on ball bearing type and method of lubrication.....	53
<b>Table 5.1.</b> Process parameter for brake disc grinding.....	88
<b>Table 5.2.</b> Accuracy level of data sets.....	91

## Chapter 1 . GENERAL INTRODUCTION

### 1.1. Motivation

As the marketplace 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. 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. 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.

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. This gives manufacturers freedom of testing a plethora of ideas, which cannot be implemented or probably risky in hardware scenario.

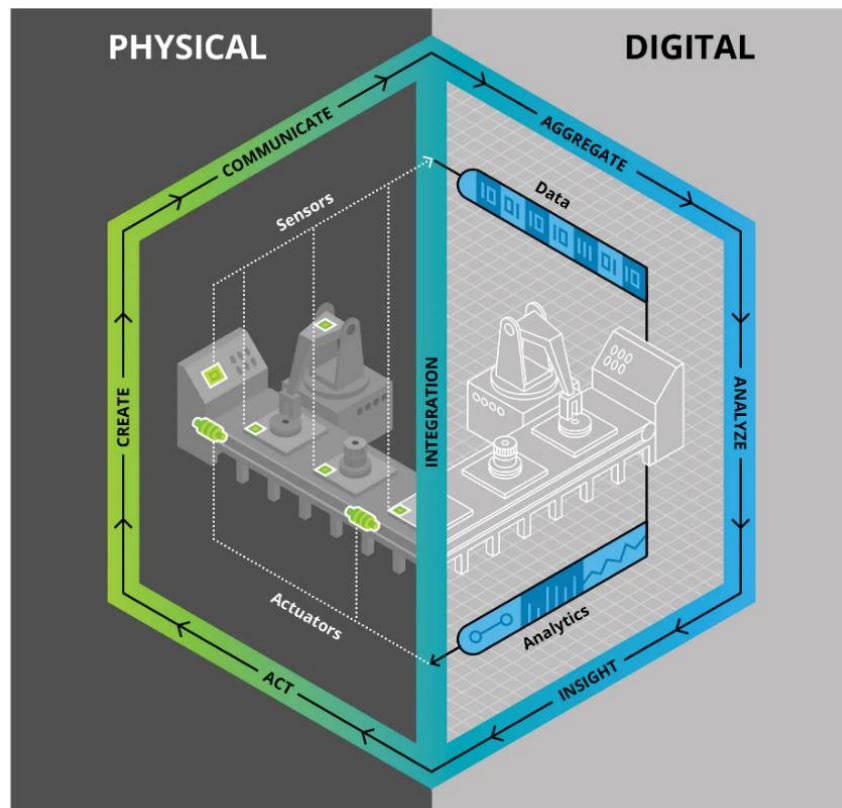
Therefore, a digital twin-based system is developed to predict the process behavior to evaluate and manage the machining process in the virtual space test phase, predict product defects in advance, reduce costs and improve competitiveness.



## 1.2. State of art

### 1.2.1. Concept and definition of a digital twin

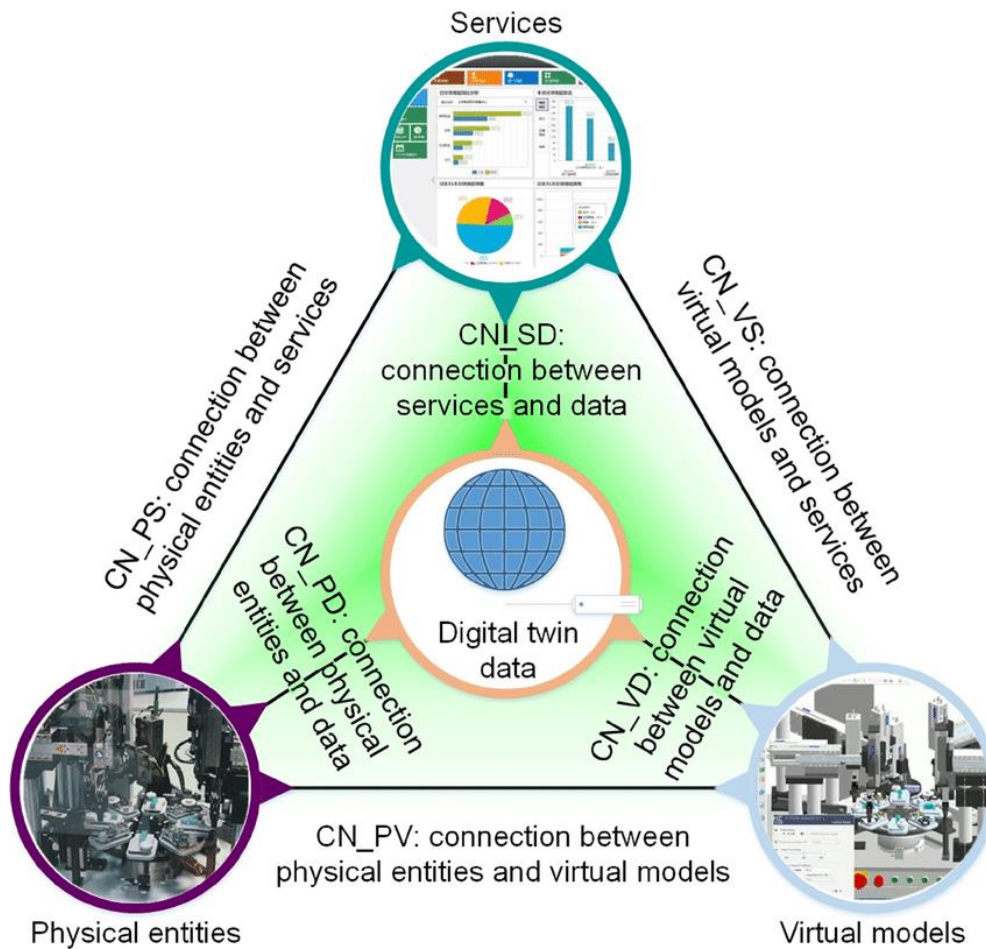
The concept of using a “twin” originated from the construction of two identical space vehicles in NASA’s Apollo program. The vehicle that remained on the Earth was configured to mirror the state of the space vehicle during the mission and called the “twin.” Models of space vehicles that remain on the Earth during flight missions have been used to simulate alternatives, reflecting data on flight conditions to support astronauts in critical situations.[1] This application demonstrates the potential of virtual simulation in linking physical and virtual spaces. A virtual simulation model reflects the constraints of physical assets without errors and direct training on physical assets to extract appropriate solutions through virtual simulation.



**Figure 1.1.** Conceptual idea of digital twin [1]

As such, it is an efficient tool for extracting powerful actions for physical assets, and it is more advantageous than performing trial-and-error method on physical assets in terms of time and cost because it is possible to solve problems with physical assets in advance through experiments [2-3].

Nevertheless, because virtual simulation does not involve linking data between the physical and cyber worlds, it cannot actually generate a “twin.” Therefore, virtual simulation cannot be regarded as DT [4].



**Figure 1.2.** Elements of digital twin [4]

As mentioned earlier, the concept of DT was first introduced by Michael Grieves in an executive course on PLM at the University of Michigan, USA [5]. According to Grieves [6,7], a DT model comprises the following three basic elements:

1. Physical space containing physical objects.
2. Cyber space containing cyber objects.
3. Link for data and information to flow from physical space to cyber space in order to synchronize the physical and cyber systems by exchanging data.

Despite the fact that digital representations were relatively new, the technology at the time was not advanced enough to portray physical products as digital representations. This is because there was little information gathered while physical things were being produced [5].

The advancement of cloud services, IoT, AI, big data, and other new-generation information technologies in the decade after the introduction of DT has made it possible to gather data on the production site, or the real world. This has made it possible to communicate with the online community and eventually realize DT [8–10]. Numerous research on DT in many industries led to its inclusion in Gartner's Top 10 Strategic Technology Trends from 2017 to 2019, demonstrating its significance [11–13]. A DT has been defined in a variety of ways by various academics, as indicated in Table 1.1.

**Table 1.1.** Definitions of digital twin [14]

No	Year	Author	Definition
1.	2014	Grieves	Virtual representation of what has been produced
2.	2015	Rosen et al.	Realistic models of the current state of the process and their own behavior in interaction with their environment in the real world
3.	2016	Canedo	Creating a digital representation of a real-world object with focus on the object itself
4.	2017	Weber et al.	With regard to the software, a DT is the digital representation of all the states and functions of a physical asset
5.	2018	Tao et al.	DT is an integrated multiphysics, multiscale, and probabilistic simulation of a complex product and uses the best available physical models, sensor updates, etc., to mirror the life of its corresponding twin
6.	2019	Talkhestani et al.	The DT is a virtual representation of a physical asset in a CPPS, capable of mirroring its static and dynamic characteristics
7.	2020	Rasheed et al.	DT can be defined as a virtual representation of a physical asset enabled through data and simulators for real-time prediction, optimization, monitoring, controlling, and improved decision making."

### **1.2.2. Application fields of digital twin**

In 1950, Dean [15] introduced the idea of a product lifecycle. The four stages of the product lifetime are introduction, growth, maturity or stabilization, and decline. The product lifecycle encompasses all procedures associated with a product, from market adoption through product disposal [16]. Product lifetime has been redefined to include the entire process, including product demand analysis, design, manufacture, sales, after-sales support, and recycling as a result of the emergence of concurrent engineering [17,18]. It therefore includes the full process of developing the product concept, designing, manufacturing, transporting, selling, using, providing after-sales support, and recycling or discarding. Each stage generates a significant amount of data, as seen in Table 1.2 [19–22].

To give systematic instructions for manufacturing processes, big data collected during the product lifetime may be examined [21]. However, Tao et al. [22] found the following issues with PLM's application and product data management:

1. Information islands can arise from data collected during the product lifecycle due to actions and purposes that are not mutually exclusive.
2. There is a significant amount of duplicate data throughout the product lifetime.
3. It is challenging to compare numerous operations and the analysis of massive data produced over the product lifecycle with actual production facilities since there is little to no recurrence and contact.
4. Analyzing real product data rather than model data from virtual products is ideal for big data applications.

Additionally, it may be used to anticipate how a product will operate in the future in addition to virtual representation. In order to give performance data, optimize design, and enhance production processes, DT may also generate predictions regarding actual goods [23,24].

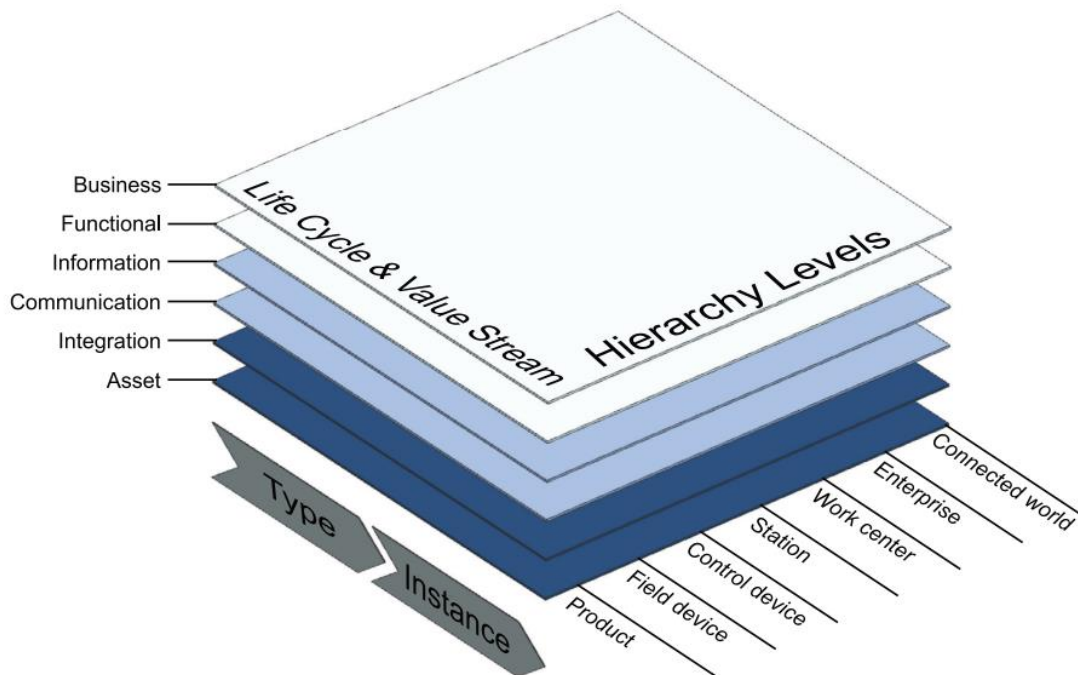
**Table 1.2.** Application field of manufacturing digital twin

Phase	Description
Concept generation	Define product design and main functions based on customer requirements and information on related products in the market.
Design	Establish necessary product definitions and designs before production. Divided into product design specification (collect product information and define in precise terms), conceptual design (find and define solutions that satisfy the design specification), and detailed design (finalize the size, shape, etc. of individual components and materials considering manufacturing functions).
Manufacturing (Production)	Manufacture or produce products by processing and assembling parts according to the specifications defined in the design phase.
Transportation	After completing manufacturing or production, transport the product to the point of sale according to market demand and orders.
Sales	Sell the product to customers (companies or individuals).
Utilization	Customer operates the product based on information provided in the manual.
After-sales service	Provide maintenance, service, and repair in the event of product malfunctions and the like during customer use.
Recycle/disposal	Discard or repurpose the product after the customer has finished using it.

By creating RAMI 4.0, BITKOM, VDMA, and ZVEI have created a new architecture model that satisfies Industry 4.0's requirements. A three-dimensional (3D) model was created, as seen in Fig. 1.3, to illustrate all the elements of the interrelated technical qualities [25–27]. To

describe the elements of Industry 4.0, the product lifecycle and value stream are merged and hierarchically organized, and thus offers a flexible description of the industry 4.0 environment [23].

The following is a summary of each axis of RAMI4.0 [28]:



**Figure 1.3. RAMI 4.0**

a) Levels axis: Consists of six layers that break down an object's features into an organized hierarchy for each layer, such as asset, integration, communication, information, functional, and business. It is a digital map of the thing as a result. The layers, lifetime and value stream, and hierarchy level axes may be used to map out key elements of Industry 4.0, and objects can be categorized using the models. Industry 4.0 principles may be freely explained and put into practice using RAMI 4.0.

b) Lifespan and value stream axis: Based on IEC 62890, this axis categorizes facilities or goods into kinds and instances to describe the lifecycle. The fundamental lifecycle management philosophy for systems and components used in industrial process measurement, control, and automation is specified in IEC 62890, and it supports the lifespan of product type and instance.

c) Hierarchy level axis: Based on IEC 62264 and IEC 61512, this axis has hierarchical levels that reflect different operations inside a facility or factory. The fundamental paradigm for enterprise-control system integration is defined by IEC 62264, while the particular model for batch control is provided by IEC 61512. These two standards aid in the comprehension of the manufacturing system since they describe the factory resources that need to be managed for production and operations. The hierarchy levels include product, field devices, control devices, station, work unit, enterprise, and linked world to incorporate as many manufacturing system domains as feasible.

Numerous studies have made use of some of the RAMI 4.0 elements discussed above in DT, and there are numerous instances where elements have been applied to the interior or outside of plants along the hierarchy levels axis. The concepts used to describe the hierarchy levels axis are broad in scope [23]. From a manufacturing standpoint, commonly used phrases like "product," "machine," "process," "factory (plant)," and "logistics" might be reinterpreted. These comprise not just the goods to be created and the manufacturing facilities, but also the portion for cooperation, much as the specific components of the hierarchy levels axis [23].



### **1.2.3. Technologies of smart manufacturing for digital twin applications**

Smart manufacturing is described as "a manufacturing system that is entirely integrated and collaborative to meet dynamic demands and requirements of customer needs, factories, and supply chain network in real time" by the National Institute of Standards and Technology (NIST) [29]. Contrarily, "a novel production pattern integrating manufacturing assets with sensors, computer platforms, information and communication technology (ICT), control, simulation, data-intensive modeling, and engineering" has been used to characterize smart manufacturing [30]. In other words, smart manufacturing is a revolution in the manufacturing industry that tries to address a number of challenging and irregular situations. The IoT, including cyber-physical systems (CPS), the Internet of Things (IoT), cloud computing, big data, and sensors, is needed for smart manufacturing, on the other hand [31]. By interacting and combining the physical and digital worlds in line with the requirements of the manufacturing business, these technologies are intended to generate smart manufacturing [32–34]. Additionally, since DT was introduced, these technologies have advanced greatly, and DT that links manufacturing sites and cyber models has developed at the same rate as element technologies. In other words, ICT has expanded the utility of DT, and DT has been recognized as a significant contributor to the revolution in smart manufacturing [35]. As a result, studies are being conducted to determine the role of DT based on how DT and the industrial Internet interact as well as enabling technologies that increase the viability of technology [36,37].

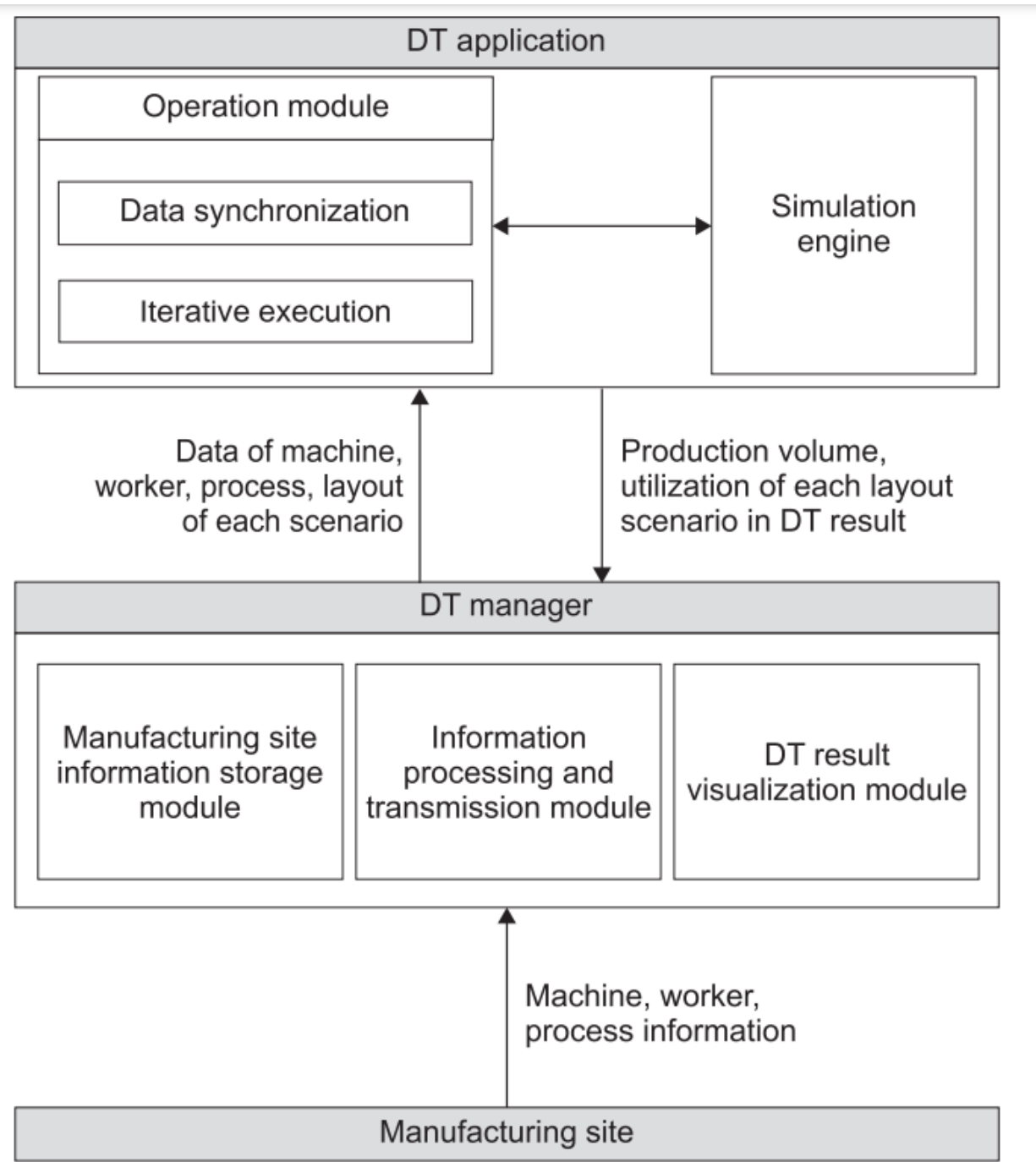
A cloud-based CPS is advised for high-level processes and production systems due to Industry 4.0's revolution of the manufacturing paradigm [38]. Cloud technologies facilitate data flow in production systems and manufacturing digitalization. It suggests that by digitizing real-time data from the floor and offering a range of services via cloud-based modules, a cloud-based

CPS may instantly assess the condition of a shop floor. In other words, real-time sensing, interoperability, and adaptability tasks may be performed more efficiently when DT or CPS are developed utilizing cloud technology [38,39]. The production paradigm has changed from conventional to cloud manufacturing as a result of the utilization of digitalized resources in the cloud [40]. Digital Twin as a Service, a new concept introduced by cloud technology, increases the level of smart manufacturing (DTaaS). Using cloud technology, everything may be provided as a service, including SaaS (Software as a Service), PaaS (Platform as a Service), and IaaS (Infrastructure as a Service), according to Pallis [41]. (Infrastructure as a Service). An expansion of DT as a service is DTaaS, which engages vertically with consumers using augmented interfaces deployed with technologies like augmented/mixed/virtual reality (AR/MR/VR) for physical, digital, and cyberspace as a service [42]. The development of several applications that improve human-machine interaction is also made possible by this interface technology [43–46]. In other words, DT is digitalized, similar to other cloud resources, and as one of the numerous modules, it offers useful services. However, because CPS and DT also need the maintenance of information systems on an external server in addition to the collection, processing, and exchange of large amounts of data, cloud technology raises concerns about privacy, integrity, and security [47]. Cloud technology for DT has to use encryption and security standards, as well as research into standards and communication protocols, to address these problems [48,49].

#### **1.2.4. Digital twin-based performance testing**

The studies listed below are examples of earlier research that employed DT to conduct pilot tests on the production system for equipment, processes, and plants.

When the task type of a robotic cell changed, Erdos et al. [50] sought to know if it was feasible to reconfigure the cell to operate on a different product. For this purpose, they employed DT to investigate if robot arrival or collision occurs as well as whether the robotic cell may be altered. The robot's route may be enhanced by continuously operating the DT while reflecting the robot's motions, path, and task order, as well as by changing the workspace's arrangement. The researchers found that using the conventional method takes roughly 5 days for revision and confirmation of cell reconfiguration, tool wear, and robot motions in a real operating cell, but employing DT only took a few hours to a day. Liu and co. To reduce costs and increase efficiency, [51] offered a DT-based advanced design approach for an automated flow-shop production system. According to this study, which compared and validated current AFMS with AFMS created and maintained by DT, AFMS utilizing DT reduced inventory by 28.6 percent, including WIP, and costs by 22.2 percent. Research has shown that DT may be utilized to dramatically boost AFMS effectiveness consequently. For the customized design of a hollow glass manufacturing line with various capacity, layout, and process route limitations, Zhang et al. [52] created a DT-based model. Based on the idea of "iterative optimization between static design and dynamic execution," this DT resolves an optimization issue in the design of a production line combining process constraints and the research of configuration variables. As a virtual twinning method based on distributed simulation, this study used simulation as a verification tool for the optimization solution.



**Figure 1.4. System architecture for digital twin test [52]**

# Chapter 2 . RESEARCH METHODOLOGY

## 2.1. Research object

The goal of this dissertation is to implement of digital twin for predicting and monitoring grinding process behavior. The research object chosen for this study is the manufacturing process of brake disc. There are five production steps of brake disc, including material input automation, balancing correction, double side grinding, measurement and stamping, and finished product loading automation. This study will focus on the double side grinding process of brake disc.

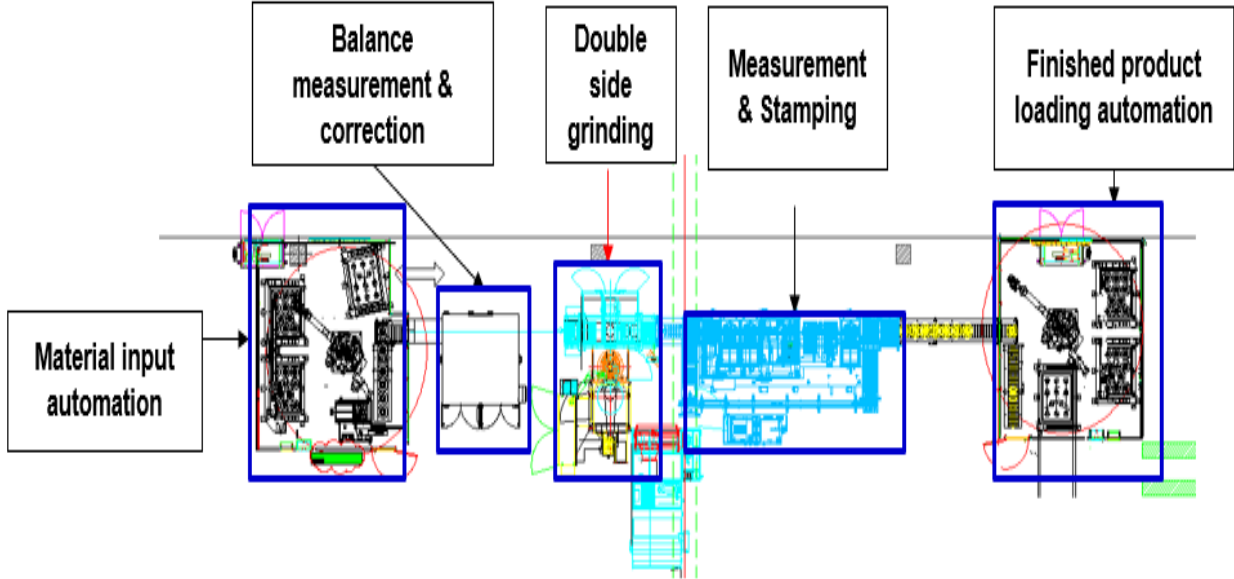
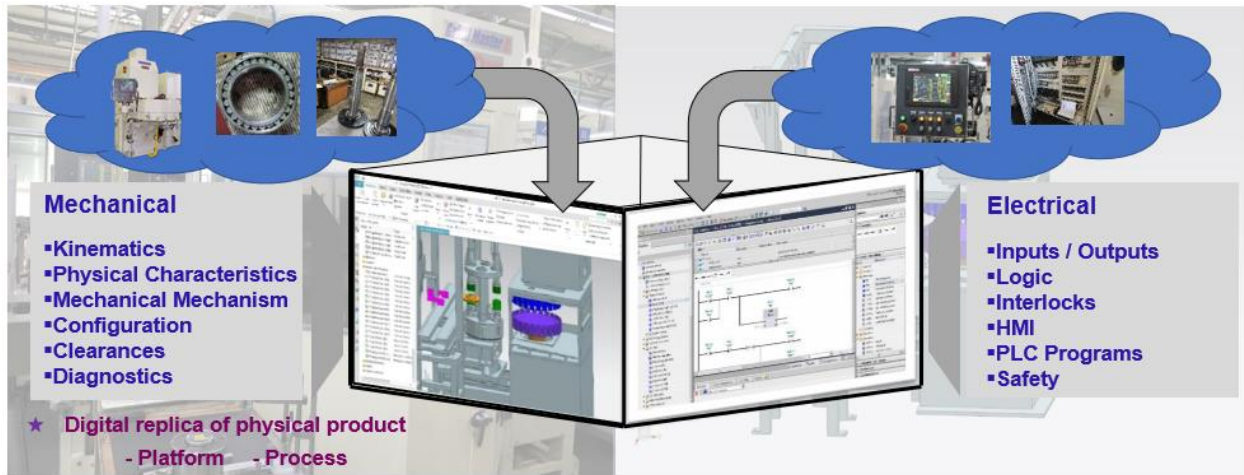


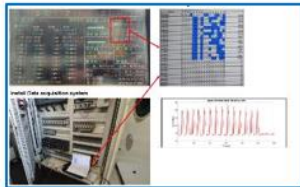
Figure 2.1. Manufacturing process of brake disc

The contribution of this paper is to apply DT techniques to the grinding process. The DT system is realized by the physical mechanical properties and electronic control properties of the grinding process. The developed system will be used to carry out the simulation of the machining

parameters to ensure the correctness of the machining parameters before they are applied to reality. At the same time, during on-site processing, real-time monitoring is performed by connecting PLC and sensors to the system, and dynamic parameters are imported for simulation. The simulated ideal current value is compared with the actual current value as a reference value to determine reliable operation.



### Process parameter testing and setting



### Process monitoring

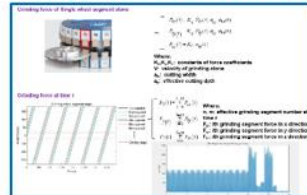


Figure 2.2. Digital twin system for evaluating grinding process behavior

## **2.2. Research strategy followed**

The comprehensive step by step methodology is given below figure. The current grinding process, as well as the machine behavior, is analyzed. Firstly, the 3D model with mechanism of each component was generated. Then, the physical behavior of grinding machine was described. A virtual controller was developed by analysis the real process controller behavior. A grinding process model was developed to calculate grinding process behavior. To prove the accuracy of the system, the simulated cutting force values should be compared with the actual values. Due to two grinding wheels and workpiece rotate simultaneously becomes a season that difficult to measure the grinding process behavior by using force sensors. Therefore, the system will describe the power flow, convert the simulated grinding force value into a current value along the power flow, and compare the current value with the actual measured current value to prove the accuracy of the system. The developed system is implemented to the actual factory. Since the system is developed based on Siemens platform, and the factory uses Mitsubishi PLC, data acquisition and data exchange modules will be developed at the same time to realize the interaction between the digital twin system and the physical machine. Finally, compare the experimental data to prove the accuracy of the system.

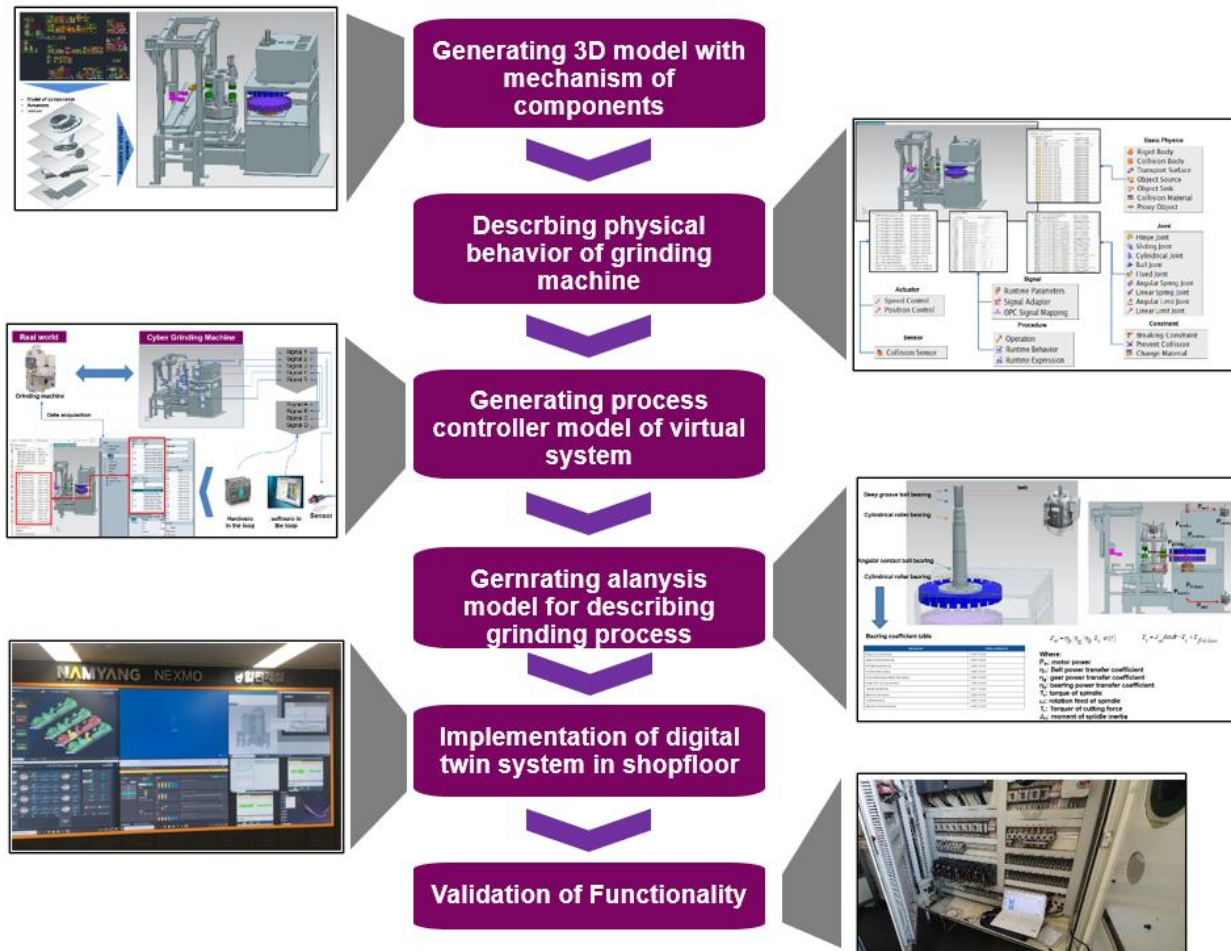


Figure 2.3. Research strategy followed

### 2.3. Verification and validation

Verification and validation are crucial procedures for system analysis during the design or implementation phases. It shows if the designers are moving in the correct path or not. In order for the consumer to be sure that the design meets their needs, it is crucial to test the system while altering the ambient conditions, according to Andersson and Runeson [53].

According to authors, businesses devoted a significant portion of their money to evaluating and confirming the established system. To examine and research the validation procedures used in those firms, they performed a survey of 11 Swedish companies. When comparing small and large



enterprises, the authors claim that the techniques of verification differ significantly. Large companies employ commercial tools or software. Small businesses, on the other hand, typically employ house-made equipment for validation. These approaches still need to be improved, according to authors [53].

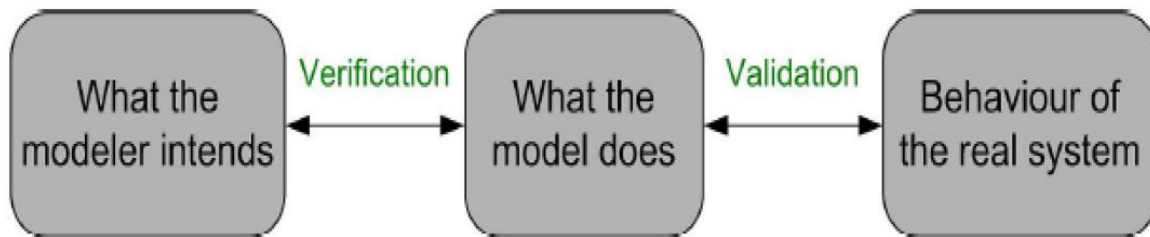
All life cycle models, including those for hardware, software, and systems, are consistent with IEEE's definition of verification and validation as a standard. According to IEEE, there are a number of practical procedures that are helpful in figuring out whether the created system or product fits the intended purpose and matches the customer's needs [54].

Verification and validation procedures are crucial for planning and implementing since they directly affect product creation, functioning, and the way customers view the entire system, according to Maropoulos and Ceglarek. These techniques check the accuracy of the model's representation of the real system and its suitability for application in accordance with customer specifications or requirements throughout the model-building stage of simulation [55].

According to Rabe, Spieckermann, and Wenzel, the verification and validation procedures may be used to compare the applicability of a model or product to the criteria. The ability of the designers to examine their designs after each modification provided by these technologies can also substantially aid in correct modeling of the system under consideration. By identifying several approaches to do a same goal, this aids in giving designers and clients choices to select from in accordance with their circumstances. The ability to make modifications and see how they affect the system, in the authors' opinion, also aids in a better understanding of how the system behaves [56].

Utilizing a terminology comparable to that offered by other writers, William and Ülgen also gave an explanation of the verification and validation process. They also highlighted how the simulation model should be able to accurately reflect the actual systems, which also complies with the end user's expectations. The distinction between the definitions of verification and validation was also examined, and authors attempted to clarify their knowledge using figure 2.4.

Verification may be used to compare the actual developed model to the requirements of the model or the model that the designers want to create. The validation process was described by the authors as an assessment of the proposed model and the actual physical system. The model can be regarded to be verified if the difference is small and it is operating correctly [57].



**Figure 2.4. Verification and validation of simulation Model**

Methods for verification and validation are relevant to, and in use in, the mechanical and electrical domains in addition to the software side. It can yield findings that give designers and clients the assurance that the system will function well in the real world as well, which is a top goal.

There are several methods for verification and validation, but we usually see them used in the context of contemporary model-based embedded systems. These techniques, often referred to as X-in-the-loop techniques, are typically utilized with electronic control units for the creation and validation of software. Some of these techniques are briefly covered in the subsections that follow.

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 strategy is said to be the one that is employed the most frequently for model-based advancements. It may test the control code, which is developed using the model language, and is often built using graphics-oriented tools like Matlab (Simulink) (e.g. finite state machines). The code, which is in the format of the standard PLC language, cannot be validated, nevertheless, in accordance with standard IEC 61131-3 [58].

Vepsäläinen and Kuikka claim that not enough has been done to include simulations into model-driven development methodologies. The control system of a crane was tested by authors using the simulation provided by the Unified Modeling Language (UML) Automation profile tool [58]. The Model in the Loop simulation technique was employed in this application, which is only one of many such.

2. Software in the loop: The use of a virtual controller, also known as a "Emulator," coupled with a virtual representation of the real-world system via a communication channel (such as Open Platform Communication (OPC), Transmission Control Protocol (TCP/ IP), etc.) is known as this simulation methodology. Another way to put it is that it combines the virtual controller with a simulation of a real plant. For a better grasp of the ideas associated with this method, refer to Figure 2.5, which is presented below.

Software-in-the- Numerous production methods have employed loop. Emulation was utilized in the packaging line for consumer goods by Phillips and Montalvo. According to the authors, by employing a virtual 3D model of the manufacturing line, mimicked testing reduced the starting time of the packing line by up to 50%. Control engineers were able to use it in its early phases in this fashion. The RSLogix Emulate 5000 was utilized by the authors as a virtual controller [59].

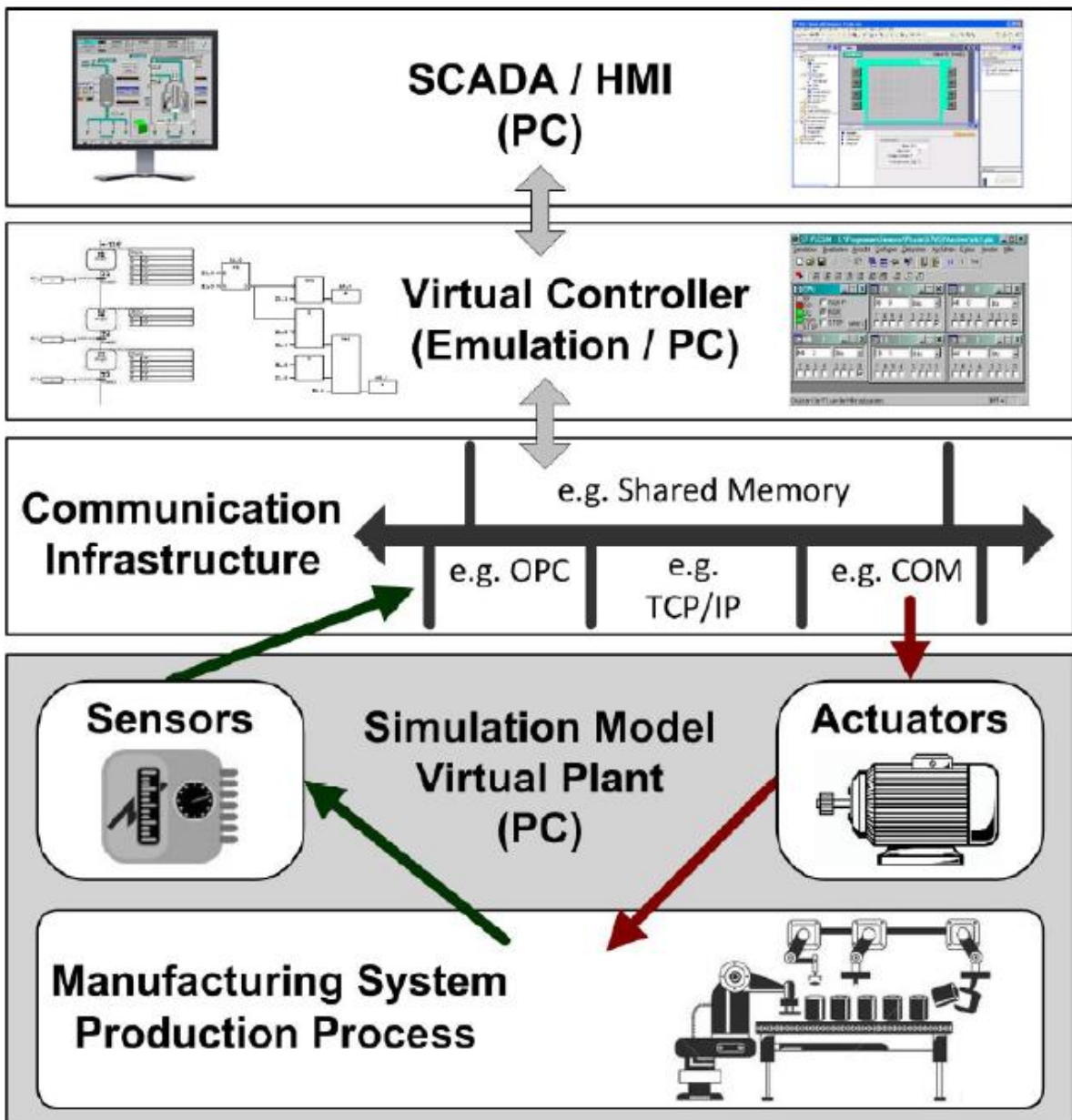


Figure 2.5. Software in the loop

Software-in-the-loop methodology was utilized by Dzinic and Yao for a manufacturing cell that was installed in a lab. For the simulation of the production cell, they utilized a program named "Xcelgo Experior," and they established TCP/IP connectivity with the emulator using a third-party application called "Net-to-PLCSIM" [60].

The Krause employed the similar strategy in 2007 for the liquefied natural gas (LNG) facility. The process simulator "Sandvika" from Kongsberg Process Simulation and the AC 870P/Melody System emulation, both of which are active components of the ABB 800xA Distributed Control System, were utilized by the author (DCS). He was able to effectively link the simulation with more than 10,000 inputs and outputs from the Norwegian LNG facility [61].

3. Processor in the Loop: Hoffmann claims that this method is mostly utilized in the automobile sector to evaluate the software of Electronic Control Units (ECU), but it is seldom ever applied to industrial control applications [62]. The actual control code on the processors is tested using a technique called processor-in-the-loop, or PIL, although not in real time. Additionally, it has the ability to automatically test various situations and find any errors in the control code.

4. Hardware in the Loop: Unlike software in the loop, where an emulator is used to simulate the operation of the real controller, hardware in the loop simulation uses a genuine controller that is connected to the simulation and actual hardware. As seen in figure 2.6, some of the system's components are simulated, while others are made of actual hardware. The aerospace and automobile sectors are highly familiar with this strategy since many projects have been carried out utilizing it.

Isermann, Schaffnit, and Sinsel claim that when some system components are absent or it is highly expensive to experiment with the real system, simulation techniques like hardware in the

loop are used. The issue of time also exists. This method was utilized by authors to design novel control algorithms and to analyze diesel engines in detail [63].

Hardware in the loop is a proven method for creating and assessing a control system, according to M. Basic. It lets us to combine actual hardware (a controller) with the simulation, which aids in our ability to comprehend how the system behaves in the present. If the intended system is capable of providing in the appropriate amount of time, this provides crucial information about its competency. The author also noted that although designers have been using this method for more than 40 years for significant projects including the design of anti-lock brake systems (ABS), highly maneuverable aircraft technology (HiMAT), and missile guidance systems, it has not yet been formalized [64].

For a deeper understanding of this strategy and how it has been applied to other projects, see Figure 2.6. Dzinic and Yao used both hardware in the loop and software in the loop simulation techniques to verify their Experior-created simulation of a manufacturing cell. They noticed identical outcomes for their case study as a result of their verification [60].

As control programs are created and verified using the same controller, which should then be utilized for controlling purposes on the factory floor, the hardware in the loop method is also beneficial in this regard. It serves as the framework for the overall automation system's integration test.

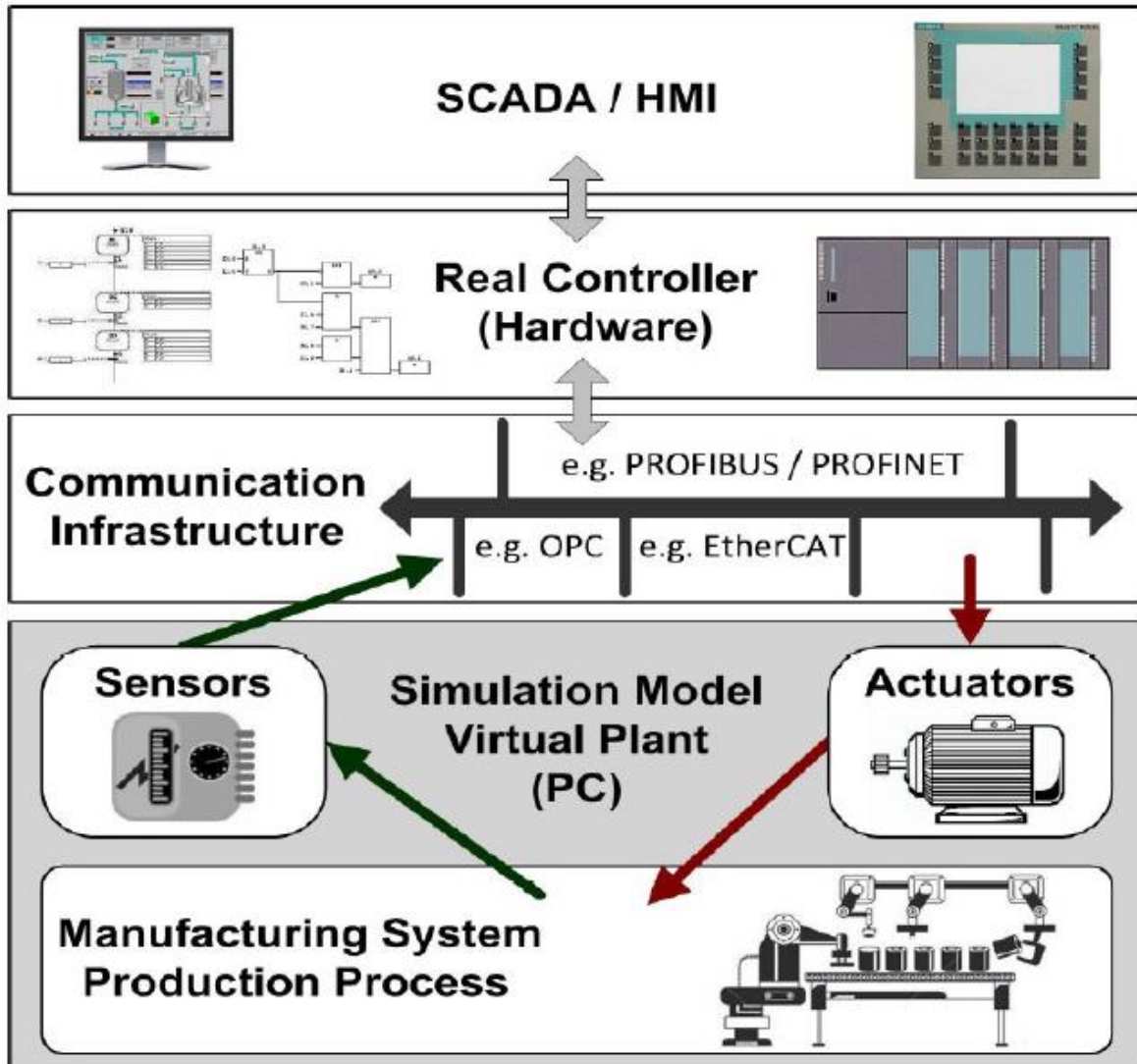


Figure 2.6. Hardware in the loop



## 2.4. Testing control programs

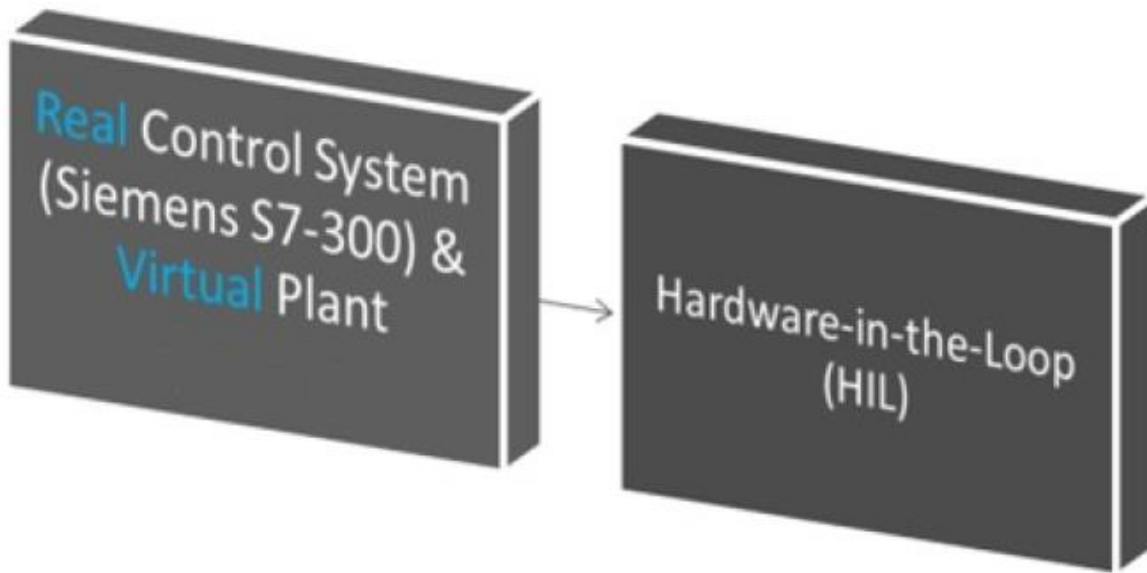
Verification The response program, which was being used to test the code, was swapped out with a PC-based simulation model in the authors' simulation models. By Danielsson and his team, six simulation approaches were established.

1. The first is the "logic simulator approach," which Hoffmann [61] sometimes referred to as "model in the loop (MIL)". In this method, the author used a model of a genuine controller to evaluate the logic. Although it was an advance over earlier methods, there was still space for growth.

2. "Logic emulator" is the name given to the second technique. The key benefit of this strategy is that both the testing and the actual controller may be written in the same programming language. The CPU response method's requirement for change is not necessary here. Thus, programmers may save a great deal of time and effort.

3. The third technique, referred known as "control system simulators," deals with simulating a real-world object. For a better knowledge and code validation of that particular physical system, such as robots, simulation systems for real hardware can be incorporated.

4. "Real control systems," often known as "hardware in the loop (HIL)," is the fourth phrase, according to Danielsson and group. This strategy links simulation to the actual controller. This strategy may offer a thorough comprehension of the entire process together with genuine values. The primary disadvantage of this method at the time was the potential for delays in the data transfer between controller and PC, which might disrupt the simulation.



**Figure 2.7. Real control system method**

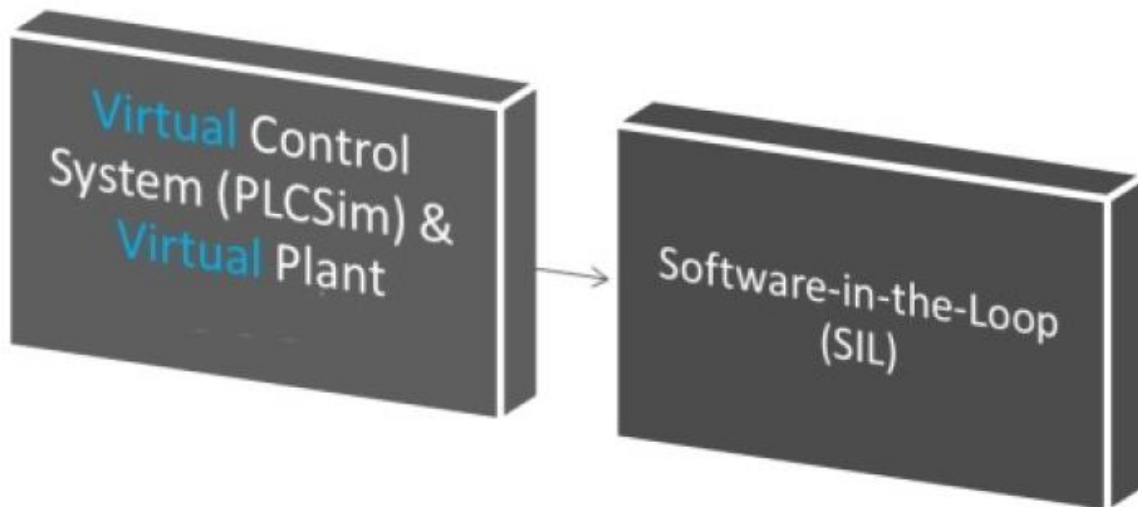
5. The fifth technique, referred to as "online control," is tying simulation and real-world systems together through a controller. The only distinction between this methodology and the real control systems approach is how the controller and hardware are connected.

6. The author claims that "control system emulator," commonly referred to as "software in the loop (SIL)," is the sixth way for the simulation. The actual controller is replicated in this process. This technique may be used to check the code, and the exact same program can be used for controlling without any changes. This method, as illustrated in figure 2.8, synchronizes simulation models with emulators to test a variety of situations without requiring real hardware. As a result, it enables the risk-free testing and validation of several concepts.

According to Danielsson and colleagues, there are two potential difficulties with this strategy. The synchronization of time between simulation and emulation models comes first. It

poses a significant challenge to all scholars. Siemens PLCSIM, for instance, may be successfully utilized for code validation without the need of a real controller. Hoffmann claims that there is a time synchronization issue when utilizing PLCSIM with simulation models, which has an impact on the entire purpose of the testing. Siemens solved this issue while creating the PLCSIM Advanced emulator [61].

Designing simulation models of real processes was the second issue mentioned by Danielsson and colleagues. The creation of a model that accurately represents the hardware takes a lot of time and work. Consequently, this approach's problem is represented.



**Figure 2.8.** Control system emulator method

## Chapter 3 . MODELLING OF GRINDING PROCESS

### 3.1. Modelling of grinding force

The double side grinding process are shown in figure 3.1. The grinding wheel connected with spindle directly. 24 CBN grinding segment stone evenly distributed in the circumferential direction. The upper and lower grinding wheel, which rotated in different direction, simultaneously process the surface of brake disc. Therefore, the mechanical effect of the upper and lower parts is different and need to be analyzed separately.

Firstly, the grinding force of single grinding segment stone will be analyzed. The upper grinding wheel rotated in clockwise and brake disc rotated in counterclockwise. The absolute speed of the grinding stone segment of the upper grinding wheel relative to the workpiece in the Cartesian coordinate system is shown in the following formula.

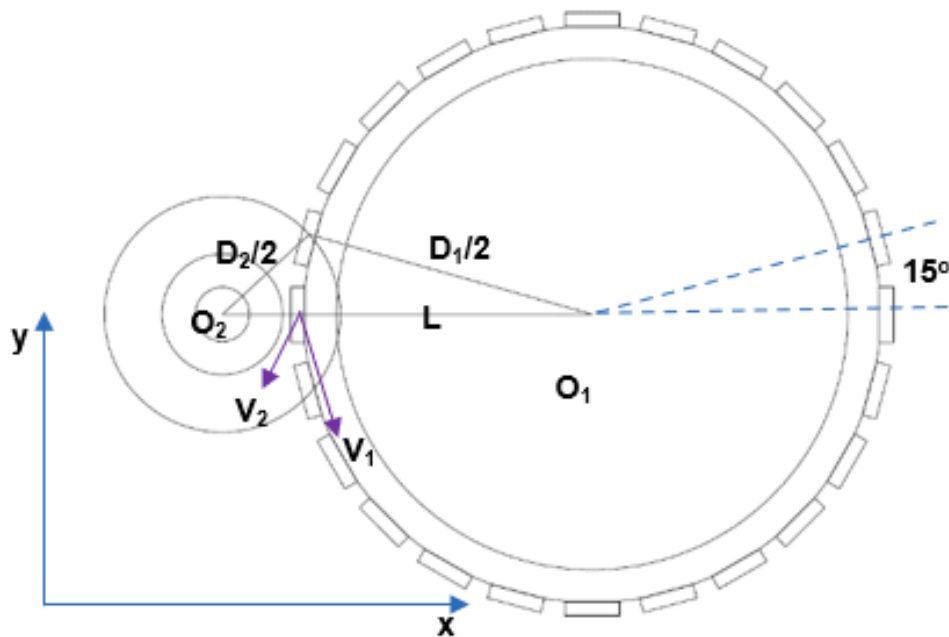


Figure 3.1. Grinding process of brake disc

Then calculate the tangential velocity to the grinding wheel.

$$\begin{aligned}
 V_x &= \frac{D_1}{2} \omega_1 \sin(\omega_1 t) - \sqrt{l^2 + \left(\frac{D_1}{2}\right)^2} - l D_1 \cos(\omega_1 t) \omega_2 \sin(\omega_2 t) \\
 V_y &= \frac{D_1}{2} \omega_1 \cos(\omega_1 t) - \sqrt{l^2 + \left(\frac{D_1}{2}\right)^2} - l D_1 \cos(\omega_1 t) \omega_2 \cos(\omega_2 t)
 \end{aligned}
 \tag{3.01}$$

where :

D1: Grinding wheel diameter

$\omega_1$ : rotation speed of grinding wheel

$\omega_2$ : rotation speed of brake disk

L: center distance between grinding wheel and brake disk

The force can be calculated as follow:

$$\begin{aligned}
 F_{ix}(t) &= K_x \cdot V_{ix}(t) \cdot a_w \cdot a_{ie}(t) \\
 F_{iy}(t) &= K_y \cdot V_{iy}(t) \cdot a_w \cdot a_{ie}(t)
 \end{aligned}
 \tag{3.02}$$

Where

$K_x, K_y, K_z$ : constants of force coefficients

V: velocity of grinding stone

$a_w$ : cutting width

$a_e$ : effective cutting depth

The effective rotation angle of the CBN grinding segment on the workpiece is -18 to +18. The figure 3.2 illustrate the trajectory of CBN segment relate to rotation angle. The area between the red line is effective area. Calculate the effective tangential force of single grinding stone segment in specific time, and accumulate to get total tangential force:

$$F_x(t) = \sum_{i=n}^{i=m} F_{ix}(t)$$

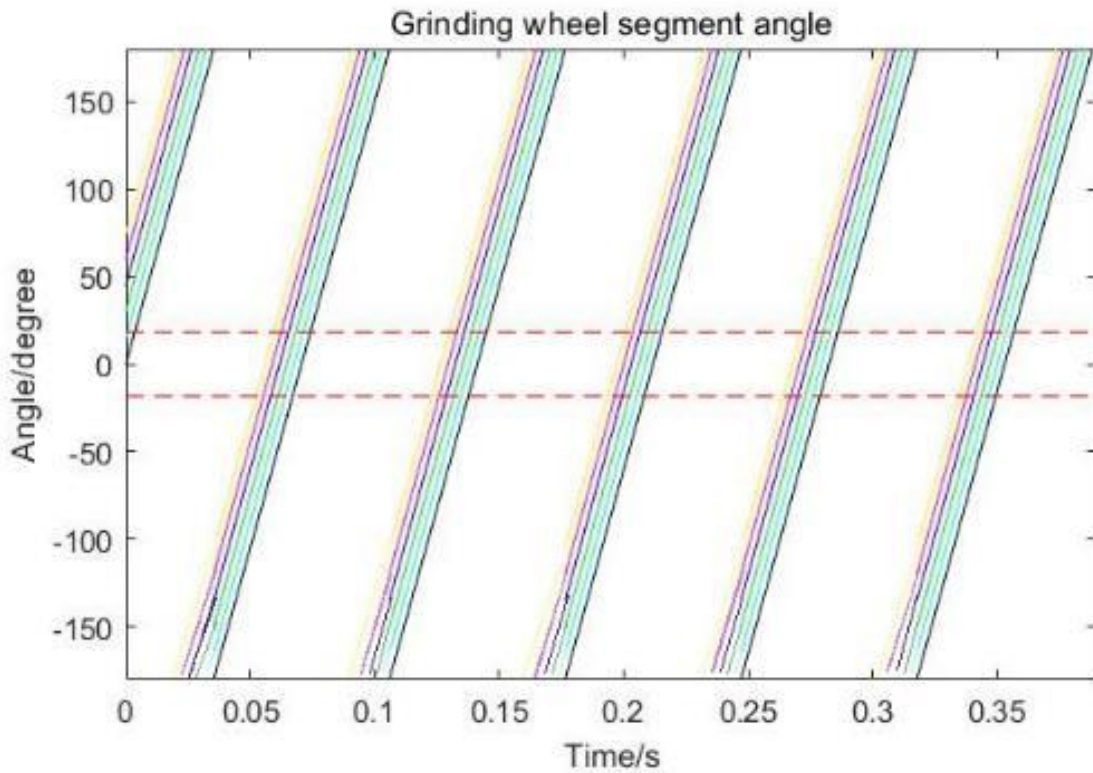
$$F_y(t) = \sum_{i=n}^{i=m} F_{iy}(t)$$
(3.03)

Where:

n, m: effective grinding segment number at time t

Fix: ith grinding segment force in x direction

Fiy: ith grinding segment force in y direction

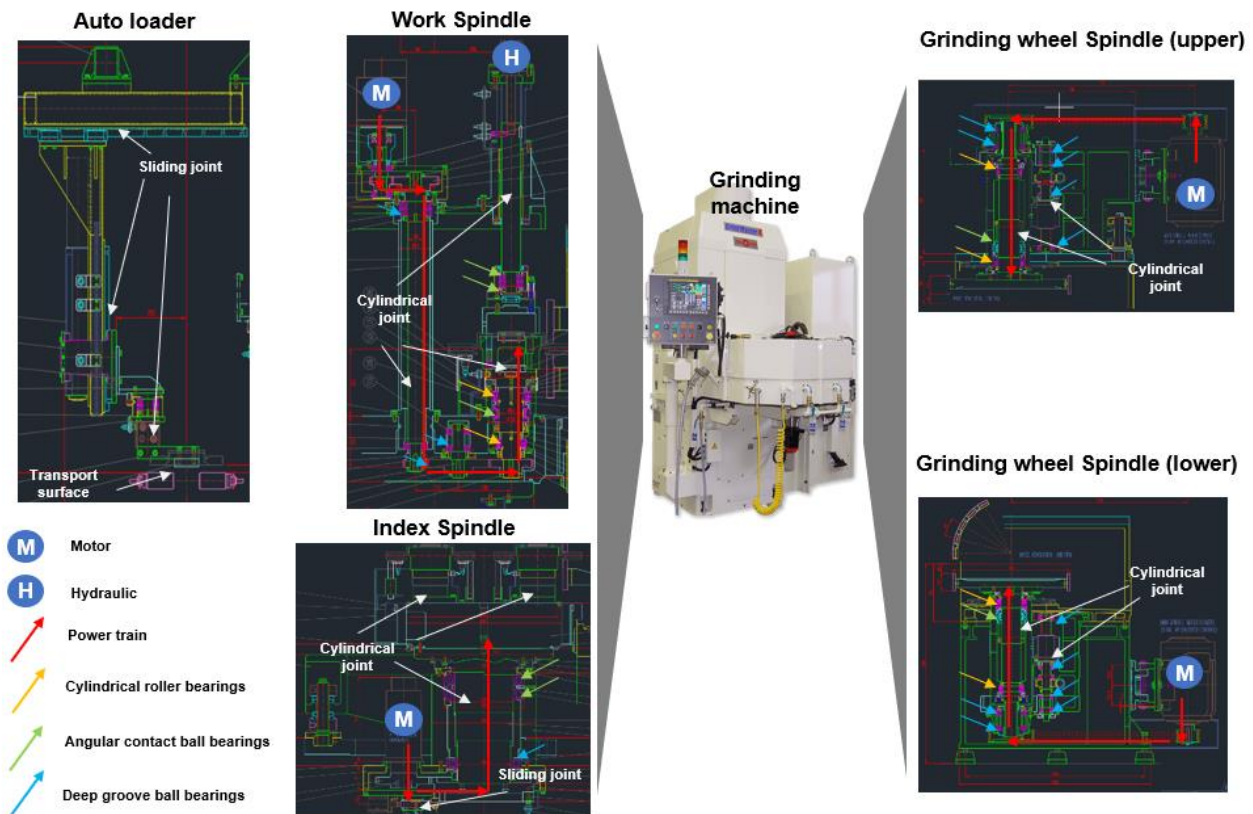


**Figure 3.2.** effective rotation angle of the CBN grinding segment

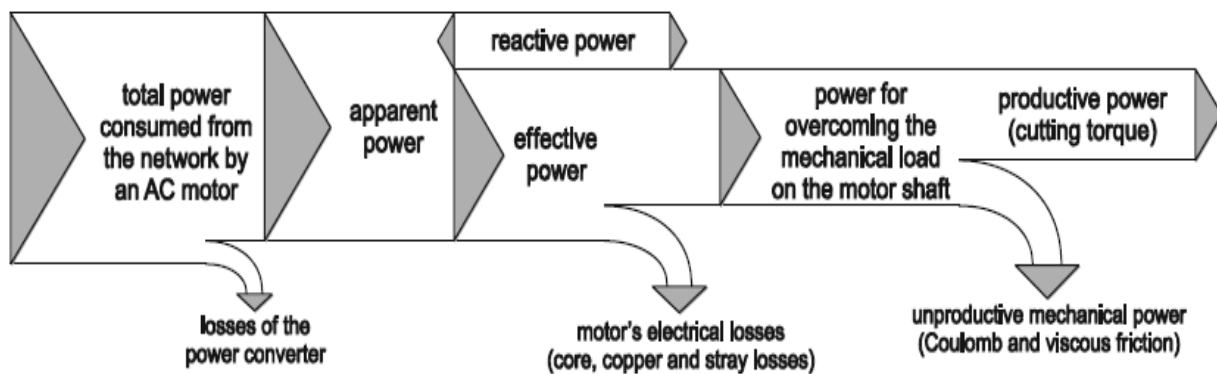
Through real-time grinding wheel and workpiece rotation speed monitoring, the effective grinding power is calculated.

### 3.2. Modelling of power flow

The research object is driven by several motors as shown in figure 3.3, this chapter mainly analyzes the upper grinding wheel spindle behavior. All mechanical and electromechanical losses need to analyze to calculate the effective grinding power. Figure 3.4 illustrate all kind losses along the power flow.



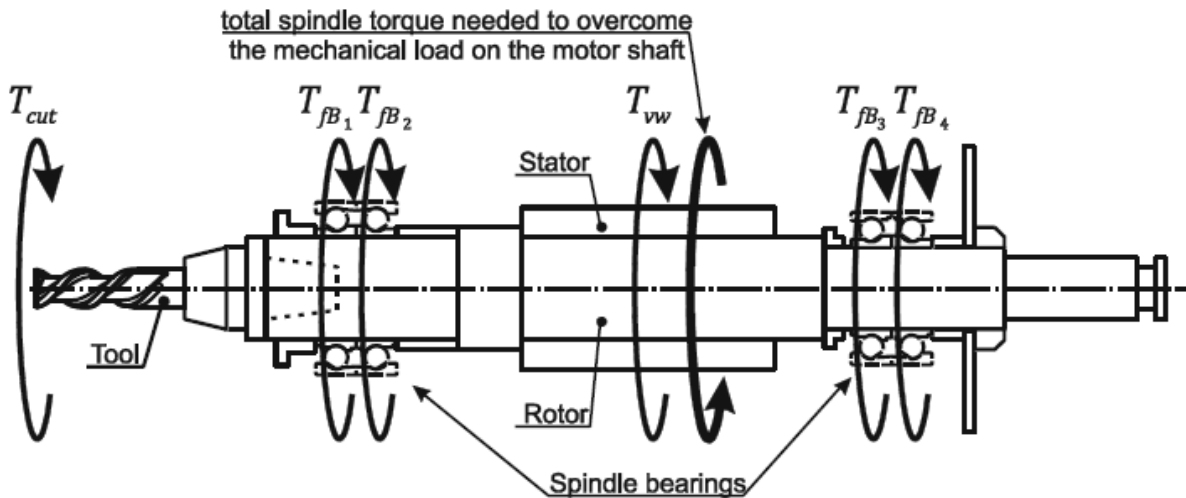
**Figure 3.3.** Analysis of mechanical mechanism of components along force flow



**Figure 3.4.** Power flow in a motorized spindle [66]



In the present work, the model for total spindle power estimation in terms of including all mechanical and electrical sources of power loss in the spindle drive. Figure 3.5 illustrates a typical modern motorized spindle with a simplified presentation of all mechanicals torques that it has to overcome while rotating: friction in bearings,  $T_{fBi}$ ; cutting torque,  $T_{cut}$ ; and windage friction torque,  $T_{vw}$ . [62]



**Figure 3.5.** Free-body schematic diagram of the spindle [66]

As a general rule, the Coulomb friction depends on the material properties and normal forces in the friction contact, whereas the viscous friction depends on the relative motion speed of the bodies in contact and the characteristics of the fluid film separating them. The bearings in motorized spindles are generally designed as pairs of angular contact ball bearings to support the combined radial and axial load. Jones [63] created the first comprehensive analysis of the load deflection of angular contact bearings that took into account gyroscopic effects and the effect of

centrifugal forces on the balls. Empirical expressions for the viscous friction torque in the bearings have been presented by Palmgren [64]. Hertzian theory was used in Houpert's work [65] to calculate the geometrical characteristics of the contact zone and the pressure distribution in the fluid, but it did not account for the effects of centrifugal and gyroscopic forces on rolling elements. Harris [66] provides the most thorough analytical analysis of Coulomb and viscous friction in bearings.

In high-speed machining, the low-friction characteristics of angular contact bearings are particularly crucial. In such a system, the value of the contact angles between the balls and the rings depends on dynamic factors in operating circumstances. A method for the explicit angle computation in angular contact ball bearings has been devised by Antoine et al. [67].

Due to the air's dynamic viscosity, the windage friction torque arises in the space between the rotor and stator. Bossmanns and Tu have examined this issue [68]. In addition to mechanical losses, other kinds of losses resulting from electrical phenomena in the spindle motor are also a part of the power flow in motorized spindles. Fig. 2 presents this power flow as an illustration.

Even though there has been a lot of study in the previous three decades, theoretical models still cannot adequately explain all of the electrical power losses. Among others, Jordan [69] and Hughes [70] describe the mechanisms that give rise to various electrical losses. While Köfler [72] and Agamloh [73] provide some experimental and empirical results of stray loss estimation, Bossmans [71] and Köfler [72] give some semi-empirical formulations to quantify motor core and copper losses.

### 3.3 Theoretical model for the total spindle power calculation

Motorized spindles of double side grinding machine tool use alternate current (AC) induction motors. Therefore, the expressions for mechanical and electrical losses estimation presented in the further text will correspond only to AC induction motors.

The total power,  $P_{AC}$  (in watts), drawn by a symmetrical three-phase induction motor from the electrical network is calculated by the following expression:

$$P_{AC} = \sqrt{3}UI \quad (3.04)$$

where  $U$  is the voltage and  $I$  is the current. Only a portion of the total power, termed as effective electric input power,  $P_{ACeff}$  (in watts), is actually used for overcoming mechanical losses or dissipated as heat. It is expressed as:

$$P_{ACeff} = \sqrt{3}UI |\cos \varphi| \quad (3.05)$$

In Eq. 2.2,  $\cos \varphi$  represents the power factor ( $\varphi$  is the phase angle between the vectors of voltage and the current). In modern electric motor designs, automatic power factor correction units are implemented to achieve  $|\cos \varphi| = 1$ , thus reducing transmission losses and improving voltage regulation.

The total power (in watts) used by a motorized spindle to overcome the mechanical load and compensate for mechanical and electrical losses in the system is expressed by summing up the following components:

$$P_{ACeff} = P_{cut} + P_{ml} + P_{el} \quad (3.06)$$

where

–  $P_{cut} = T_{cut} \omega_M$  is the power providing the cutting torque,  $T_{cut}$ , needed to overcome tangential cutting force in the tool/part contact at a spindle rotational speed,  $\omega_M$  (per second)

–  $P_{ml}$  and  $P_{el}$  represent the total mechanical and electrical power losses, respectively, in the spindle drive system.

### 3.3.1. Mechanical Losses in spindle

Total mechanical losses in the motorized spindle are the consequence of the Coulomb and viscous friction phenomena in contact areas due to the relative motions of the members of kinematic pairs. The total torque required to overcome all sources of mechanical power loss in a motorized spindle is expressed as

$$T_{ml} = \frac{P_{ml}}{\omega_M} = T_{\mu} + T_v \quad (3.07)$$

where  $T_{\mu}$  is the torque to overcome the Coulomb friction forces and  $T_v$  is the torque to overcome the viscous friction forces. This equation is further decomposed according to the physical sources of friction loss into

$$T_{ml} = T_{B\mu l} + T_{B\mu s} + T_{Bv} + T_{vw} \quad (3.08)$$

where  $T_{\mu l}$  is the load-related friction torque in spindle bearings,  $T_{\mu s}$  is the spin-related friction torque in spindle bearings,  $T_{Bv}$  is the viscous friction torque in spindle bearings, and  $T_{vw}$  is the windage friction torque in the air gap between the stator and the rotor.

### **Load-related friction torque in spindle bearings, $T_{B\mu l}$**

The dependency of friction losses in bearings on both load and velocity is of great importance in high-speed machining analyses. Angular contact ball bearings are commonly used in motorized spindles. The most comprehensive theoretical bearing friction model is given by Harris [66].

Bearing friction torque due to the applied load is calculated as

$$T_{B\mu l} = f_l F_a d_{Bb} \quad (3.09)$$

where  $f_l$  is a unitless factor depending upon the bearing design and relative bearing load,  $F_a$  is contact angle related load (newton), and  $d_{Bb}$  is the diameter of a bearing that corresponds to the centers of the balls (in millimeters).

Bearing friction torque due to the applied load can be calculated analytically/numerically. It does not depend on spindle speed, but it is strongly dependent on the bearing geometry and type. The power needed to overcome the bearing friction torque due to the applied load is expressed as

$$P_{B\mu l} = T_{B\mu l} \omega_M = f_l F_a d_{Bb} \omega_M \quad (3.10)$$

Equation 2.7 is simplified to describe the power needed to overcome the bearing friction torque due to the applied load.

$$P_{B\mu l} = K_{B\mu l} \omega_M \quad (3.11)$$

where  $K_{B\mu l}$  is an applied load constant.

For ball bearings, the factor  $f_l$  can be calculated by using the following expression [66]:

$$f_1 = z \left( \frac{F_s}{C_s} \right)^y \quad (3.12)$$

where  $z$  and  $y$  are constants depending on the ball bearing contact angles. For nominal contact angles between  $30^\circ$  and  $40^\circ$  in angular contact ball bearings, the values of these coefficients are  $z=0.001$  and  $y=0.33$  [66].  $F_s$  and  $C_s$  are the static equivalent load and the basic static load (both in newton), respectively.

The basic static load,  $C_s$ , is approximated by using

$$C_s = \phi_s i Z d_b^2 \cos \alpha \quad (3.13)$$

where

–  $\phi_s$  is a unitless parameter depending on the following ratio,  $\gamma_b$  :

$$\gamma_b = \frac{d_b \cos \alpha}{d_{Bb}} \quad (3.14)$$

–  $i$  is the number of rows of rolling elements,

–  $Z$  is the number of balls per bearing,

–  $d_b$  is the ball diameter (in millimeters),

–  $\alpha$  is the ball bearing contact angle (radian)

Static equivalent load is equal to the radial load at radial bearings and to the axial load at thrust bearings. For angular contact ball bearings, it is calculated by using

$$F_s = \max(X_s F_r + Y_s F_a, F_r) \quad (3.15)$$

where  $X_s$  and  $Y_s$  are constants depending on the ball bearing contact angle,  $F_r$  is the radial load (in newton), and  $F_a$  the is axial load (in newton).

Values of  $C_s$  are generally given in manufacturers' catalogues along with the data to enable the calculation of  $F_s$  [66]. Contact angle-related load can be calculated by using the following expression:

$$F_\alpha = \max(0.9F_a \cot \alpha - 0.1F_r, F_r) \quad (3.16)$$

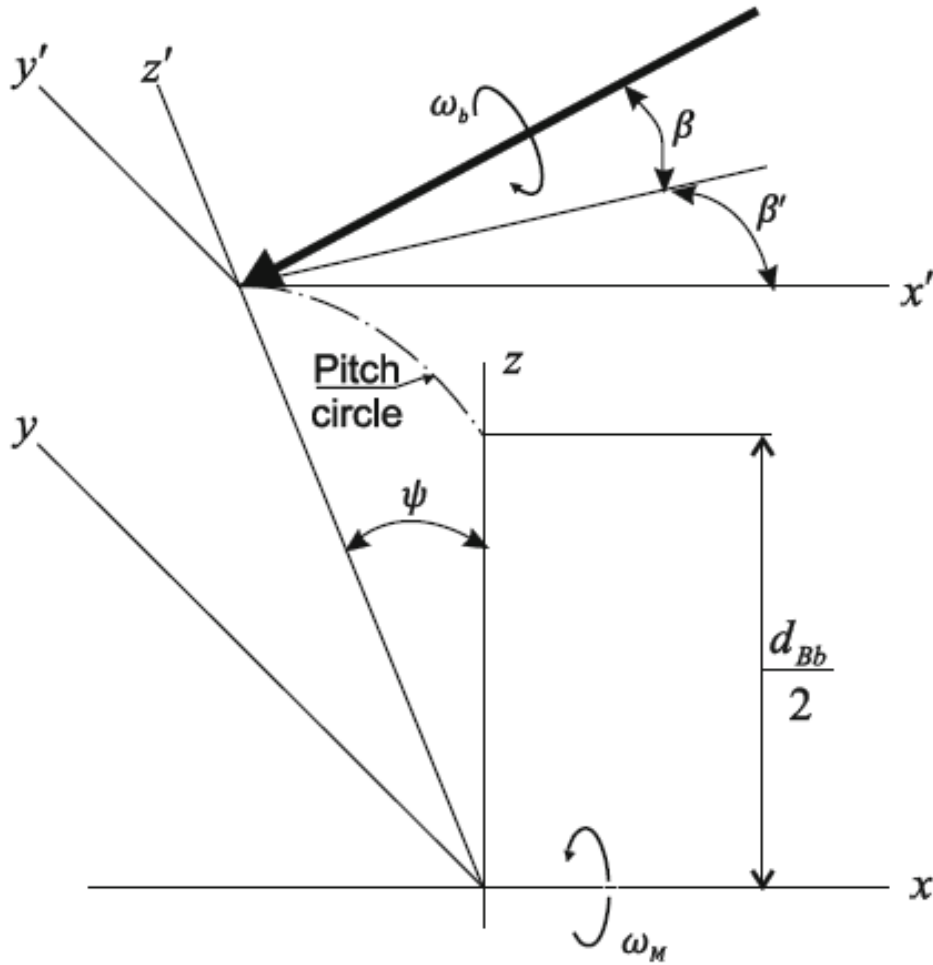
The notion of the bearing contact angle needs to be explained in detail because its value depends on the work conditions of the bearing.

### **The spin-related friction torque, $T_{B\mu s}$**

When a rolling element rotates relative to a deformed surface in the contact areas with raceways, instead of a simple rolling motion, a combination of rolling and sliding motions of the rolling element is generated. Additionally, in the case of angular contact ball bearings, due to the gyroscopic effect, the rolling motion does not occur exactly on a line parallel to the raceway. As a consequence, an additional motion occurs: the “spinning” rotation of the ball. It represents the pure sliding in the contact area, thus contributing significantly to the overall bearing friction power loss.

Detailed analysis of this problem is given by Harris [66]; here, the main conclusions from [66] are applied to the case of spindle bearings. Two variables, the angular speed of ball spinning and the corresponding friction torque, are determined in order to calculate the power loss due to the ball spinning phenomenon.

Figure 3.6 illustrates the speed vector for a single ball in a bearing. The pitch angle  $\beta$  is a consequence of the angular contact bearing design, whereas the yaw angle  $\beta'$  is a consequence of the existence of the gyroscopic motion:



**Figure 3.6.** Ball angular speed vector in a non-zero ball-raceway contact [66]

–  $\omega_b$  is the vector of the resultant angular velocity of the ball (composed of the rolling, sliding, and spinning motion components)

–  $x$  is the axis of the bearing rotation



–  $\psi$  is the current angular position of the ball in the bearing

The angular speed of spinning for a ball in the bearing  $\omega_{spin}$  can be calculated using the following expression [66]:

$$\omega_{spin} = \omega \left( \frac{1 - \gamma'_b \cos \alpha_i}{\gamma'_b} \tan(\alpha_i - \beta) + \sin \alpha_i \right) \quad (3.17)$$

where

–  $\omega_M$  is the angular speed of the motor (spindle) shaft

–  $\alpha_i$  and  $\alpha_o$  are the dynamic values of the bearing contact angle corresponding to the inner and outer raceways, respectively

Finally, we obtain the total power required to overcome the spin-related friction torque for all Nb balls as

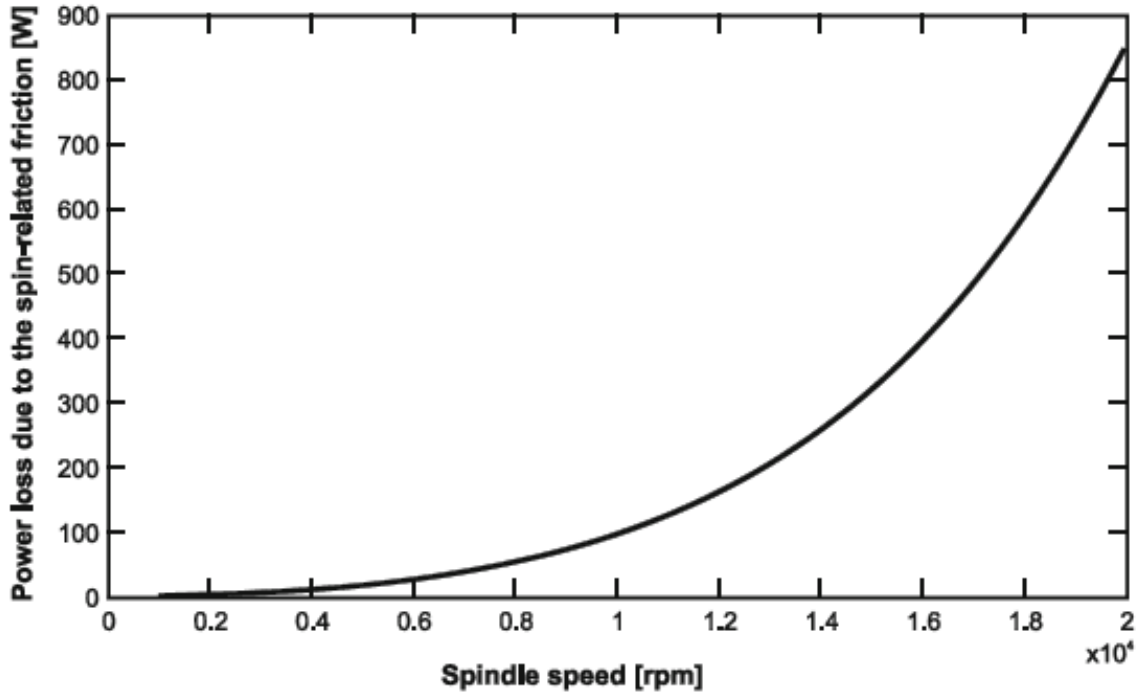
$$P_{B\mu s} = N_b (T_{B\mu si} + T_{B\mu so}) \omega_{spin} \quad (3.18)$$

Power loss due to the spin relation friction in bearings is calculated for spindle speeds up to 20,000 rpm. The calculation is done using a program written in C++. The input parameters for the calculation are taken from the case study given in [74].

Based on the results, the following polynomial model is assumed for the power loss due to the spin-related friction in bearings as a function of the angular speed of the spindle:

$$P_{B\mu s} = K_{0B\mu s} \omega_M + K_{1B\mu s} \omega_M^3 + K_{2B\mu s} \omega_M^5 \quad (3.19)$$

where  $K_{0B\mu s}$  and  $K_{2B\mu s}$  are constants. The fifth-degree term in the polynomial function ensures a better prediction of the power spent due to the spin-related friction during highspeed machining conditions.



**Figure 3.7.** Power loss due to the spin relation friction in bearings vs. spindle speed

### Friction torque due to lubricant viscosity, $T_{Bv}$

The torque due to lubricant viscous friction is given by the following expression [64]:

$$T_{Bv} = \begin{cases} 10^{-7} f_v (vn_M)^{\frac{2}{3}} d_{Bb}^3, & \text{if } vn_M > 2,000 \\ 160 \times 10^{-7} f_v d_{Bb}^3, & \text{if } vn_M < 2,000 \end{cases} \quad (3.20)$$

where  $n$  is kinematic viscosity (centistokes),  $n_M$  is a spindle (motor shaft) rotational speed (revolutions per minute), and  $f_v$  is a factor depending on the type of bearing and method of lubrication (Table 3.1)

**Table 3.1.** Values of  $f_v$  depending on ball bearing type and method of lubrication

Type of lubrication	Grease	Oil mist	Oil bath	Oil bath (vertical shaft) or oil jet
$f_v$	2	1.7	3.3	6.6

#### 2.1.4 Windage friction torque, $T_{nw}$

Another friction source is the air gap between the rotor and stator, i.e., the dynamic viscosity of the air. This problem has been analyzed by Bossmanns and Tu [68]. The rotor and the stator have different surface velocities (the stator has zero surface velocity), and the circumferential velocity profile of air is assumed to be linear between the two boundaries. The power loss due to the windage friction can be expressed as

$$P_{vw} = K_{vw} \omega_M^2 \quad (3.21)$$

where  $K_{vw}$  is the windage friction constant.

#### 3.3.2. Electrical Loss

The losses in electric motors originate in several motor components. They are due to very complex phenomena and are dependent on the type, characteristics, and the application of an

electric motor. Consequently, in spite of numerous research efforts resulting in various methodologies for analytical or experimental determination of these losses, they still represent a domain where many questions are left unanswered.

According to their sources, electrical losses can be divided into four groups [69]: copper losses in the stator,  $P_{SCu}$ ; copper losses in the rotor,  $P_{RCu}$ ; iron losses,  $P_{MFe}$ ; and stray losses,  $P_{stray}$ . Hence, the total losses due to electrical phenomena are calculated as

$$P_{el} = P_{SCu} + P_{RCu} + P_{MFe} + P_{stray} \quad (3.22)$$

### **Copper losses**

Copper losses are a function of the current flowing through the stator winding,  $I_S$  (amperes), and the rotor cage,  $I_R$  (amperes). They can be calculated using the following expressions:

$$\begin{aligned} P_{SCu} &= R_S I_S^2 \\ P_{RCu} &= R_R I_R^2 \end{aligned} \quad (3.23)$$

where  $R_S$  and  $R_R$  (both in ohm) are, respectively, the electric resistances of stator and rotor materials (copper).

### **Iron losses**

Iron losses are the result of hysteresis and eddy currents induced in the stator and rotor by the rotating magnetic field. They can be described as a function of the modulating frequency,  $f$ , as [71]

$$P_{MFe} = K_E f (3f_{rated} + f) \quad (3.24)$$

where  $f_{rated}$  is the rated frequency (corresponding to the rated voltage of the electromotor) and  $K_E$  is the eddy current-related constant.

Taking into account that

$$f = n_M \frac{N_p}{60} = \frac{30\omega_M}{\pi} \frac{N_p}{60} = \frac{\omega_M N_p}{2\pi} \quad (3.25)$$

where  $N_p$  is the number of magnetic pole pairs and  $n_M$  (in revolutions per minute) is the rotational speed of the motor shaft.

### Stray losses

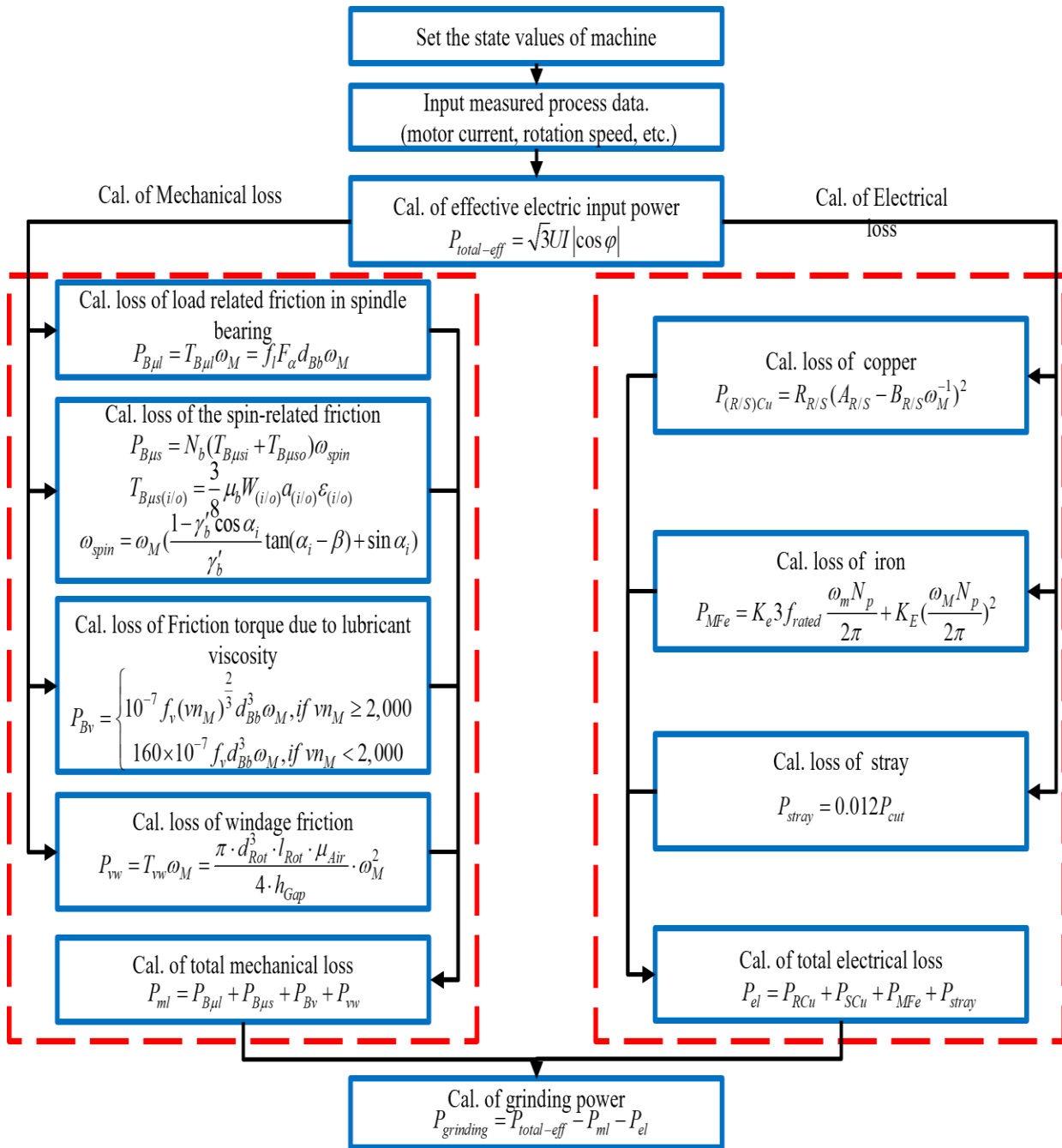
Stray losses represent the same phenomenon as iron losses, but due to the spindle parts outside the motor that are located in the area reached by the magnetic field. These losses are also frequency-dependent, but they are difficult to calculate. There are several methodologies to determine them experimentally or estimate them by performing a finite element analysis [20].

An experimental analysis by Agamloh [73], conducted on about a thousand 60-Hz induction motors, has determined an average stray power loss at 1.2 % of useful power,  $P_{cut}$  (invested in overcoming the cutting torque)

$$P_{stray} = 0.012P_{cut} \quad (3.26)$$

### 3.3.3. Module for power flow

Based on the above analysis and modelling, the calculation of effective grinding power algorithm is shown in figure 3.8. This module was programmed by C++ and connected with database and virtual space layer.



**Figure 3.8.** Flowchart for calculate effective grinding power

## Chapter 4 . DEVELOPMENT OF DIGITAL TWIN-BASED SYSTEM

### 4.1. System architecture

DTRM (Digital Twin Reference Model) is offering a universal reference model to facilitate the implementation of DT in many sectors since it has strong practicability and scalability. Therefore, the system architecture for braking discs with double side grinding machines on the shop floor was built based on DTRM. The intended system is shown in figure 4.1. The system uses five models to actualize connection between various components: physical process layer, virtual process layer, DT-based grinding process data, DT-based application service platform, communication, and data mapping mechanism.

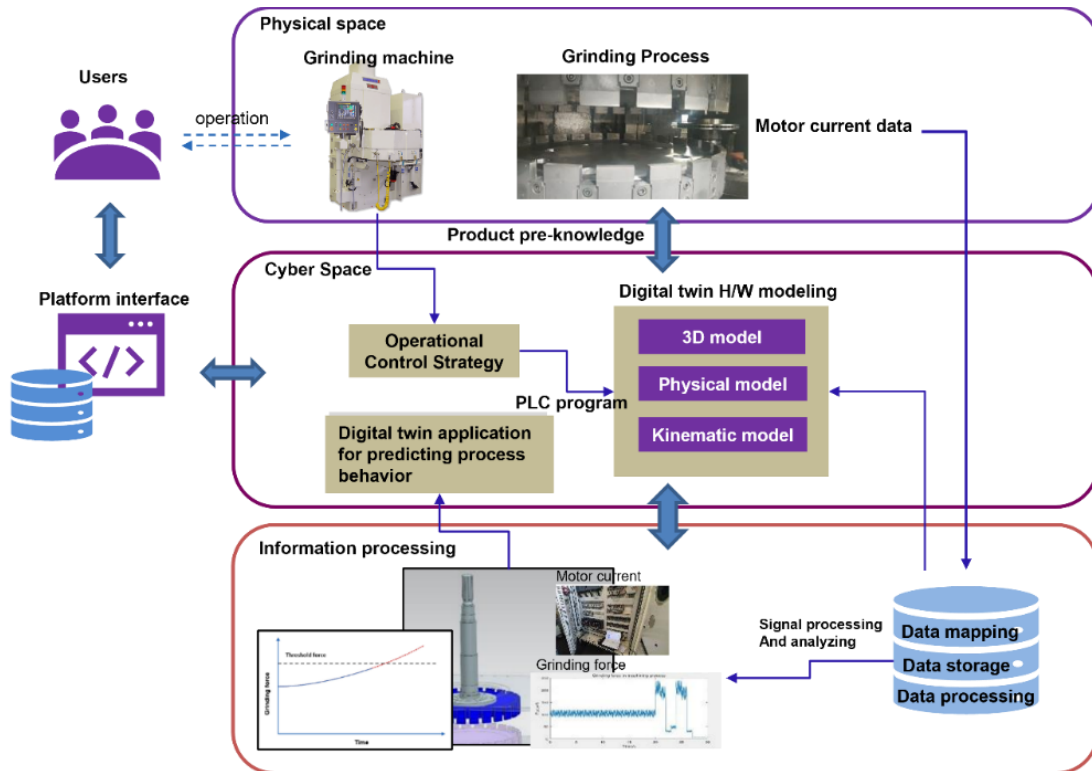
Building a manufacturing system based on DT begins with the physical process layer. The product workshop set up based on combinations of physical process entities may finish the production of brake discs in order to realize the functioning of the physical grinding process. based on the results of the grinding process planning, modeling, and forecast, change the process parameters to satisfy the specification requirement.

As the counterpart to the physical process layer, the virtual process layer should describe the same mechanical mechanism as the actual physical behavior. combined with the communication layer to achieve data mapping for real-time connection between various components.

The foundation of the digital twin system is the application service platform. During the grinding process, it primarily realizes all types of application functions, including process planning, monitoring, prediction, and control.

The link between the physical process layer and the virtual process layer is the communication and data mapping layer. The primary objective of the communication layer is to ensure continuous monitoring and optimization of the grinding process through process state transfer on bi-directional and real-time communication and data mapping. Communication protocols for industrial networks vary, including MTConnet, OPC Unified architecture, TIP/IP, etc.

The primary duties of the data process layer include data gathering, pre-processing, integration, analysis, and so on. A variety of data types were produced by the physical grinding process layer, the virtual grinding process layer, the DT-based application platform, and a combination of these three modules in the data set. The associated data processing function has to be built in order to achieve precise information interaction between virtual and physical space.



**Figure 4.1.** System architecture of digital twin system



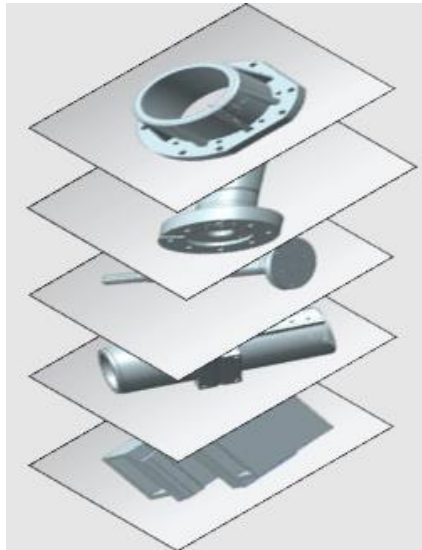
## **4.2. Virtual process layer**

A digital depiction of a physical system is called a virtual model. A digital twin is intended to be a functional and geometric replica of its physical counterpart. Siemens NX 12 was used for the virtual model's creation. Powerful software known as Siemens NX is combined with an application known as Mechatronics Concept Designer (MCD).

MCD assists in giving CAD models that are close to the real physical system physical attributes and movements. Getting the precise system dimensions and building a CAD model of the station's components is the first stage in a CAD implementation. The entire system is broken down into a number of smaller parts. These consist of the following: fixtures, sliding mechanisms, torque tools and motors, workpieces, and control panels. The final assembly is created by combining each of these subassemblies that were designed in Siemens NX. The CAD model's related joints that unite the various parts of subassemblies are equipped with physics, material, and position control attributes to enable motion simulation.

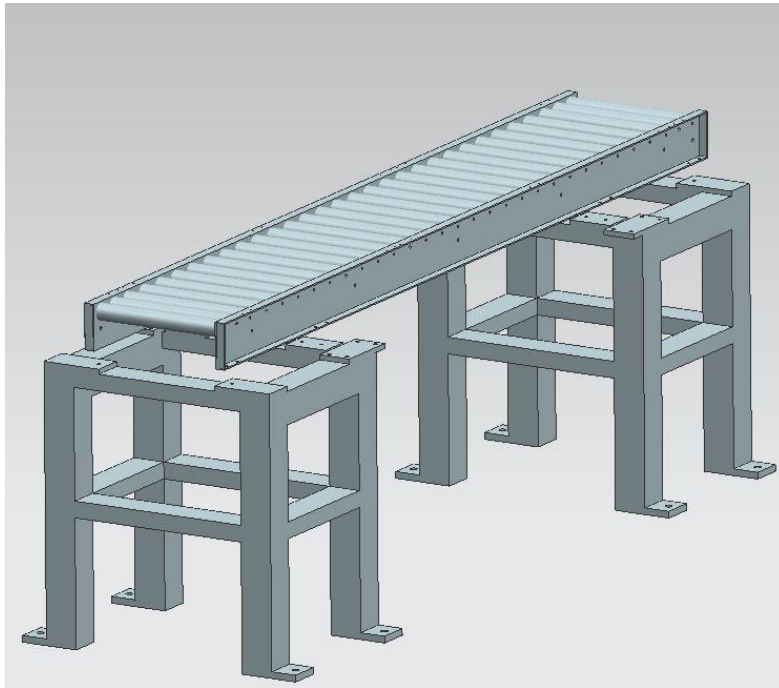
### **4.2.1. Virtual model**

Designing numerous subsystems and putting them together in a way that they mirror the real physical system is the process of creating a virtual model. Siemens NX is used for this. Siemens' Siemens NX software has an add-on called Mechatronics concept designer. Any machine or station under consideration may be created as a 3D model in the MCD by following the appropriate specifications. It enables the user to create the desired model's kinematic simulation in addition to 3D design. A 3D model must be transformed into a mechatronic model for the kinematic simulation by attaching the proper sensors, actuators, or cylinders. According to Figure 4.2, a 3D model of the brake disc grinding production line's component parts was created.

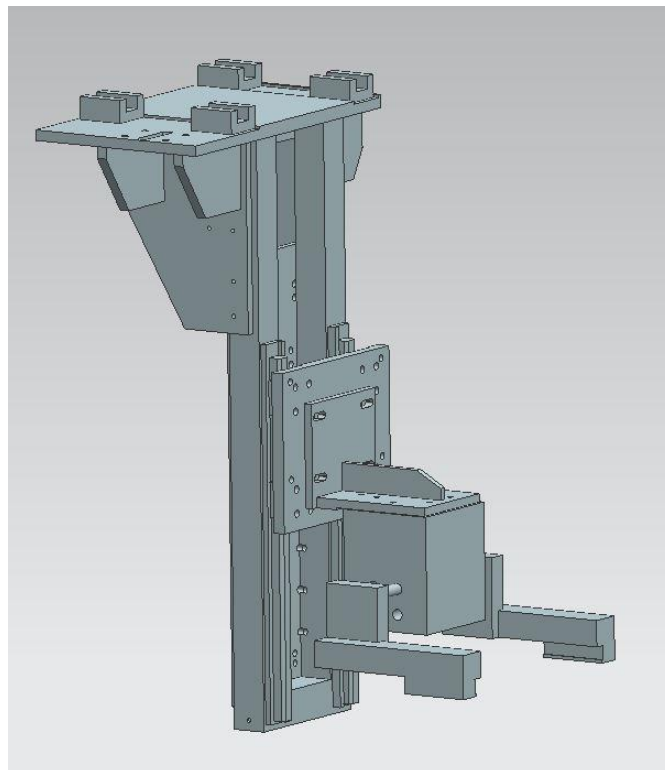


**Figure 4.2.** 3D model of components

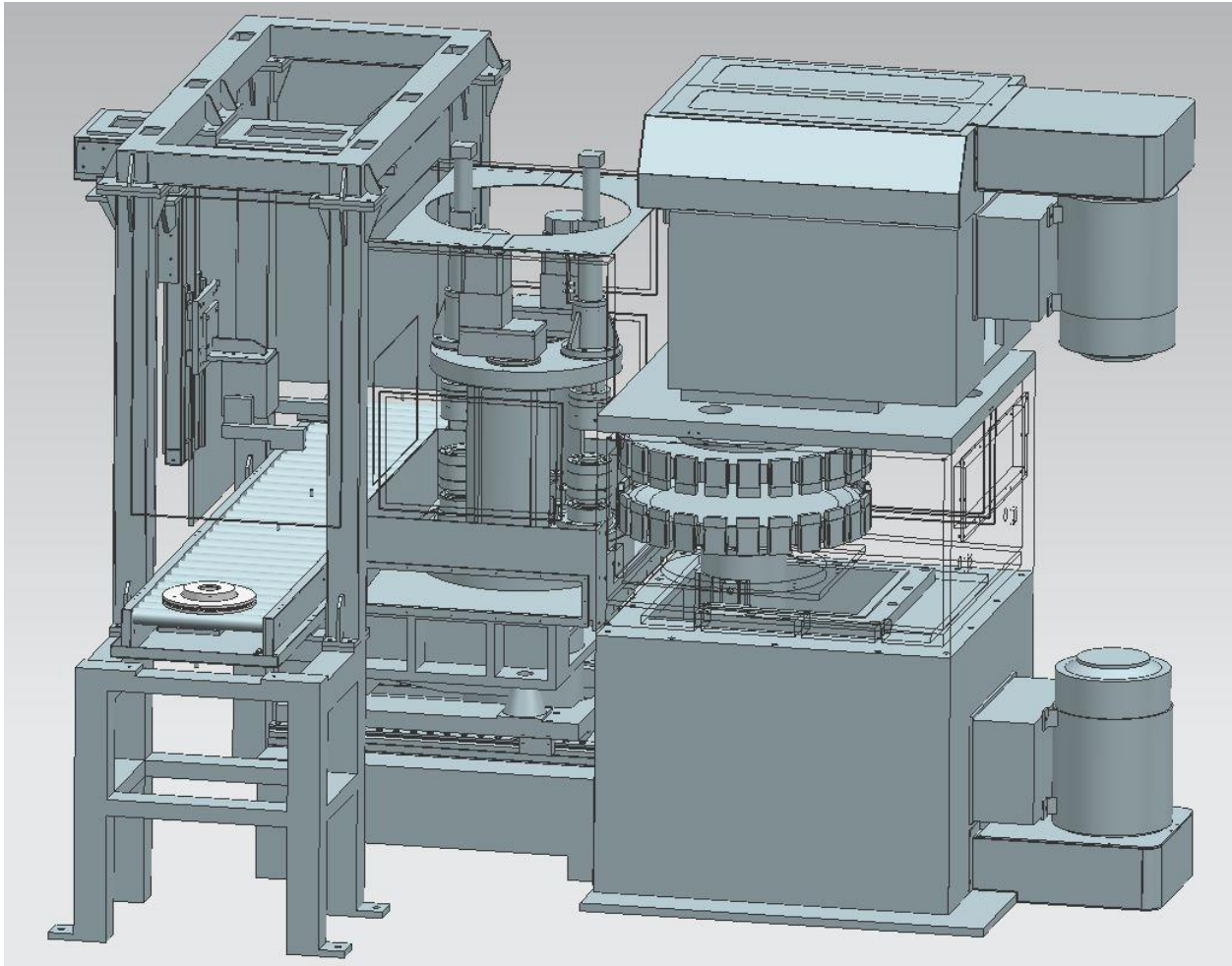
Workpiece, sliding mechanism, torque motor, gripper assembly, and faceplate are implemented as separate subsystems, as depicted in Figures 4.3 and 4.4. The final assembly is made up of all of these subassemblies, as seen in Figure 4.5. Test each subcomponent after it has been properly setup to ensure that it functions as intended.



**Figure 4.3.** sliding mechanism


































**Figure 4.4.** gripper assembly



**Figure 4.5.** Final assembly of the grinding production line

Basic physics must be included in order to construct the mechatronic idea, where stiff bodies are incorporated as seen in Figure 4.6. Joints and restrictions, such fixed and hinge joints, were applied to the virtual model to make it move identically to the real model, see (Figure-4.7).

[-] Conveyor		
<input checked="" type="checkbox"/>  Conveyor		Collision Body
<input type="checkbox"/>  conveyor_left_position		Position Control
<input type="checkbox"/>  conveyor_left_speed		Speed Control
<input checked="" type="checkbox"/>  conveyor_left_wall		Rigid Body
<input type="checkbox"/>  conveyor_right_position		Position Control
<input type="checkbox"/>  conveyor_right_speed		Speed Control
<input checked="" type="checkbox"/>  conveyor_right_wall		Rigid Body
<input checked="" type="checkbox"/>  Product_sink		Object Sink
<input checked="" type="checkbox"/>  Pruduct_out		Collision Sensor
<input checked="" type="checkbox"/>  TransportSurface_left		Transport Surface
<input checked="" type="checkbox"/>  TransportSurface_right		Transport Surface
[-] Brake_disk		
<input checked="" type="checkbox"/>  Brake_disk		Object Source
+ <input checked="" type="checkbox"/>  Brake_disk		Rigid Body
[-] Grip_carry		
<input checked="" type="checkbox"/>  Grip_base		Rigid Body
<input checked="" type="checkbox"/>  grip_carrer_base		Rigid Body
+ <input checked="" type="checkbox"/>  grip_guide_base_1		Rigid Body
+ <input checked="" type="checkbox"/>  grip_guide_base_2		Rigid Body
<input checked="" type="checkbox"/>  grip_guide_base_3		Rigid Body
<input checked="" type="checkbox"/>  grip_guide_base_4		Rigid Body
<input checked="" type="checkbox"/>  grip_left		Rigid Body
<input checked="" type="checkbox"/>  grip_rail_left		Rigid Body
<input checked="" type="checkbox"/>  grip_rail_right		Rigid Body
<input checked="" type="checkbox"/>  grip_right		Rigid Body
<input checked="" type="checkbox"/>  grip_sensor		Rigid Body
<input checked="" type="checkbox"/>  grip_y_axis		Rigid Body
<input checked="" type="checkbox"/>  grip_y_low_position		Collision Sensor
<input checked="" type="checkbox"/>  grip_z_axia_arm		Rigid Body
<input checked="" type="checkbox"/>  grip_z_axis_table		Rigid Body
<input checked="" type="checkbox"/>  Position_be_grip		Collision Sensor
[-] Jig_table		
<input checked="" type="checkbox"/>  index-table		Rigid Body
<input checked="" type="checkbox"/>  index-table spindle		Rigid Body

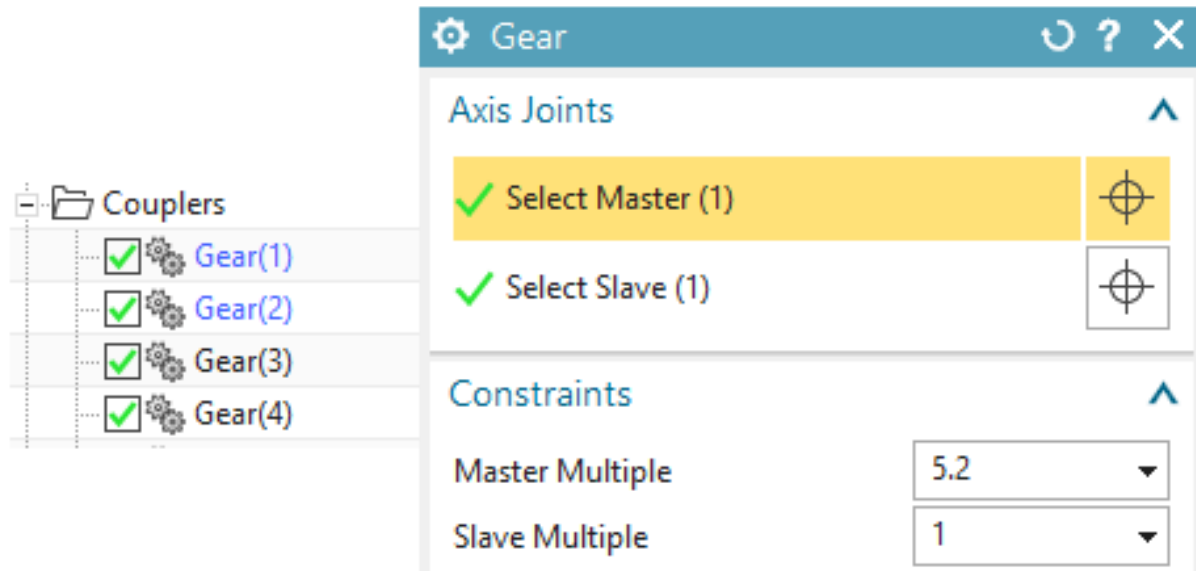
**Figure 4.6.** basic physics of the grinding production line



Joints and Constraints		
	clamp1	Cylindrical Joint
	clamp1_shaft	Sliding Joint
	conveyor_left_wall_FixedJoint(1)	Fixed Joint
	conveyor_left_wall_FixedJoint(2)	Fixed Joint
	conveyor_right_wall_FixedJoint(1)	Fixed Joint
	conveyor_right_wall_FixedJoint(2)	Fixed Joint
	Grip_base_fix	Fixed Joint
	grip_carrer_base_grip_z_axia_arm_FixedJoint(1)	Fixed Joint
	grip_carry	Fixed Joint
	grip_guide_base_1_grip_z_axis_table_FixedJoint...	Fixed Joint
	grip_guide_base_2_grip_z_axis_table_FixedJoint...	Fixed Joint
	grip_guide_base_3_grip_z_axis_table_FixedJoint...	Fixed Joint
	grip_guide_base_4_grip_z_axis_table_FixedJoint...	Fixed Joint
	grip_left_grip_carrer_base_SlidingJoint(1)	Sliding Joint
	grip_rail_left_grip_y_axis_FixedJoint(1)	Fixed Joint
	grip_rail_right_grip_y_axis_FixedJoint(1)	Fixed Joint
	grip_right_grip_carrer_base_SlidingJoint(1)	Sliding Joint
	grip_sensor_FixedJoint(1)	Fixed Joint
	grip_sensor_grip_left_FixedJoint(1)	Fixed Joint
	grip_y_axis	Sliding Joint
	grip_z_axia_arm_grip_z_axis_table_FixedJoint(1)	Fixed Joint
	grip_z_axis_table_grip_y_axis_SlidingJoint(1)	Sliding Joint
	index-table_Index_table_spindle_CylindricalJoi...	Cylindrical Joint
	Index_table_spindle_FixedJoint(1)	Fixed Joint
	Jig_base_1_workspindle_1_FixedJoint(1)	Fixed Joint
	jig_base_2_workspindle_2_FixedJoint(1)	Fixed Joint
	Jig1_carry_disk	Fixed Joint
	Jig1_fix	Fixed Joint
	Jig2_carry_disk	Fixed Joint
	jig2_sensor_jig_base_2_FixedJoint(1)	Fixed Joint
	low_spindle_body_low_wheel_spindle_guide_Fi...	Fixed Joint


















**Figure 4.7.** Joints and constraints of virtual model

The coupling between the different rigid bodies is carried out through gears (Figure 4.8) to achieve that the input and output transmission ratio is fulfilled.



**Figure 4.8.** Model coupling through gears

The movement of each motor has been controlled by position controls (Figure 4.9) to which the target values in ( $^{\circ}$ ) and speed in ( $^{\circ}/s$ ) are giving through the NX MCD input signals.

Signals		
+ <input checked="" type="checkbox"/>	 Control_click_input	Signal Adapter
- <input checked="" type="checkbox"/>	 Signal_Input	Signal Adapter
	<input checked="" type="checkbox"/>  conveyor_start	Signal
	<input checked="" type="checkbox"/>  conveyor_stop	Signal
	<input checked="" type="checkbox"/>  grip_backward	Signal
	<input checked="" type="checkbox"/>  grip_carry	Signal
	<input checked="" type="checkbox"/>  grip_down	Signal
	<input checked="" type="checkbox"/>  grip_forward	Signal
	<input checked="" type="checkbox"/>  grip_get_disk	Signal
	<input checked="" type="checkbox"/>  grip_put_workpiece	Signal
	<input checked="" type="checkbox"/>  grip_release	Signal
	<input checked="" type="checkbox"/>  grip_release_disk	Signal
	<input checked="" type="checkbox"/>  grip_up	Signal
	<input checked="" type="checkbox"/>  process_start	Signal
	<input checked="" type="checkbox"/>  test_control	Signal
	<input checked="" type="checkbox"/>  Workpiece_soindle1_rotation	Signal
+ <input checked="" type="checkbox"/>	 Signal_output	Signal Adapter

**Figure 4.9.** Actuators for twin movement



### 4.3. Data process layer

Connecting several data sources in the manufacturing environment is necessary to create a digital twin for a machining operation. These data types, which come from many data sources, may be categorized into static and dynamic data types. The pertinent data is separated into five groups before being used to generate the digital process twins. These include information about the workpiece, the process, the process, the machine, and the tool. These data serve as the data source for various process models used during machining to ascertain physical correlations. Dynamic data sets are represented by process data. Process data consists of all machine measurement and control signals as well as supplementary sensor data. This information comprises the following:

Actual/ Nominal position data (X, Y, Z, ...)

Motor drive currents (X, Y, Z, ...)

Spindle current/ Power

Spindle Speed

Feedrate/ Grinding depth

PLC information: cooling, hydraulics status

Special process features can be represented in greater depth by incorporating extra sensors, which increases integration, calibration, and processing work. As a result, it is a critical issue to integrate the least number of extra sensors feasible while applying sophisticated computational techniques to establish process characteristics based on control and drive signals.

Static data that are pooled as planning data include tool data, technology data, machine data, and workpiece data. The fundamental development challenges include standardizing semantics, synchronizing dynamic and static data, and unifying data integrity criteria. The procedure for creating and showing the digital process twin as it is being processed will be outlined in the paragraphs that follow. Consider a process model for calculating grinding forces.

#### **4.4. Communication and data mapping layer**

##### **4.4.1 Open platform communications**

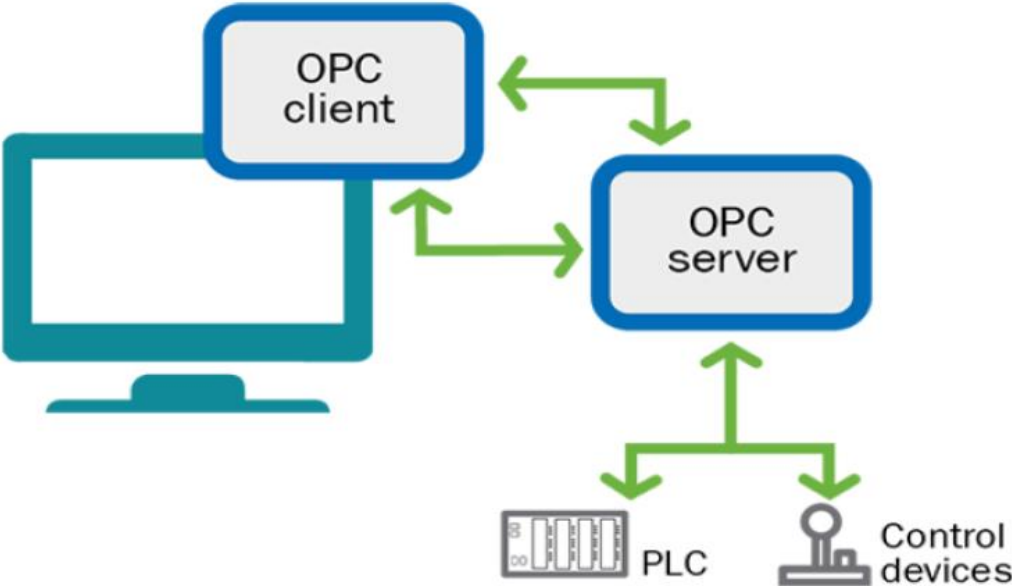
The processing industries of today employ automation to create goods more effectively and consistently. For the entire process to run smoothly as well as for system monitoring purposes, communication is one of the key aspects. Various industrial communication protocols can be used to facilitate this communication between equipment. Depending on the kind of application under consideration and the user demand, an industrial communication protocol can be chosen.

A set of protocols called "Open Platform Communication" (OPC) governs how different industrial devices can communicate with one another. It was established in 1995 with the assistance of Microsoft and several other automation players. Initially, it was referred to as object linking and embedding (OLE), and it was solely applicable to Windows operating systems and process management. Significant advancements were made throughout time, and it eventually became the preferred method of communication across the automation levels of different businesses [75].

OPC is a technology that is built on servers and clients, as shown in figure 4.10. The hardware communication protocol of the PLC is converted into the OPC protocol via the OPC server. For the data transfer between the devices, the client makes use of this OPC server. One or

more servers may be present and reply to customers' requests. These servers' connections to the controllers and devices enable them to carry out the requests made by clients. The communication connection between the programmable logic controller and simulation of the system must be built before implementing software in the loop or hardware in the loop simulation techniques. Information sharing between virtual systems and controllers may be done via OPC.

The task of establishing communication between devices from various suppliers has been made simpler via OPC. This standard makes it possible to transmit data across various controllers, Human Machine Interface (HMI) devices, Supervisory Control and Data Acquisition (SCADA) systems, and simulation tools (such as Siemens MCD, Process Simulate, Factory I/O, etc.). Hoffmann claims that PLC vendors or other parties frequently supply OPC servers (Kepware, Softing, etc). Data communication is possible between this server and many OPC clients [61].



**Figure 4.10.** Open platform communications

As a bridge for communication between DELMIA V6 and GX IEC Developer, Heidari and Salamon employed the OPC. The simulation component was included in DELMIA V6, which also served as an OPC client. The GX IEC Developer was utilized for PLC programming. Using the Beijer OPC server, they were able to connect 110 inputs and outputs to the simulation with success [76].

For the construction of simulation, Hincapié and co-authors employed DELMIA automation as a PLM tool. They utilized OPC to communicate data between the PLC and the PLM tool while they tested the PLM tool in various scenarios [77]. Guerrero and co-researchers employed the pick and place station of the AMATROL mechatronic system to investigate the virtual environment and validate the control code. For the aim of simulating, authors utilized the program Process Simulate. The controller utilized was a Siemens S7-300. OPC was utilized for communication between the S7-300 and Process Simulate [78].

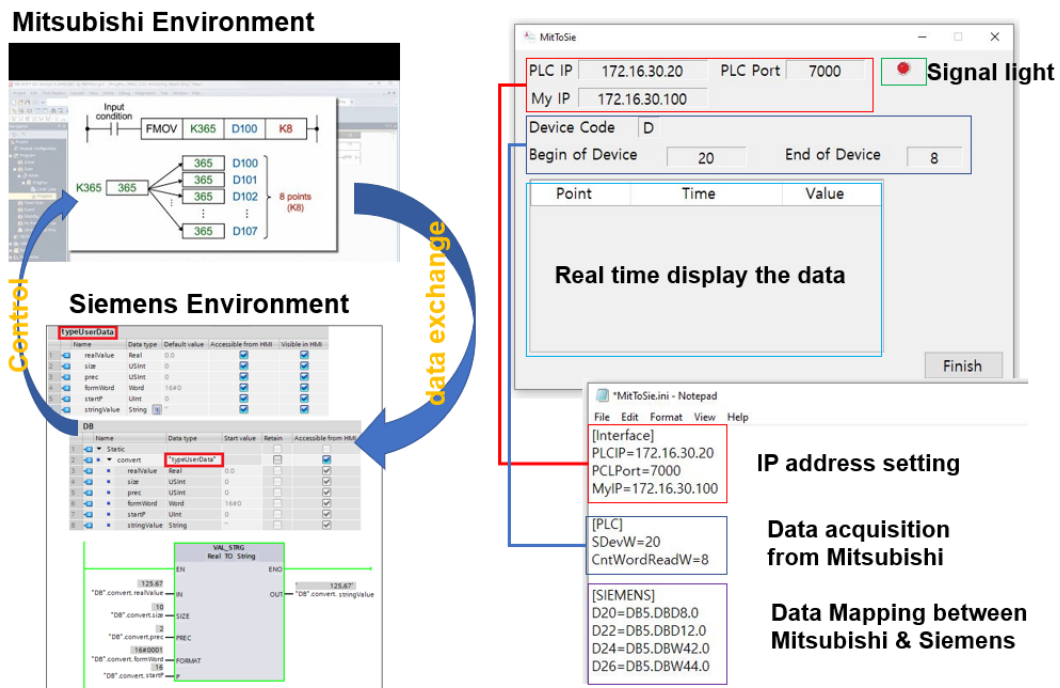
Wischniewski and Freund discussed their modeling work for transportation systems as well as their work on robot and machine interaction. When using the OPC to communicate data between the controller and the virtual system, the authors ran into a few issues. They continued by explaining that when the procedure is quick, the OPC is unable to communicate data between the controller and simulation in real-time, disrupting the entire process. They proposed slowing down simulation and PLC to provide OPC more time to digest information as a remedy to the highlighted issue [79].

Kim and his co-authors talked about virtual plant commissioning for steel production. For data connection between these devices, they connected the HMI and system emulator using OPC

in their solution. The simulation model and control logic were successfully evaluated by the authors [80].

Krause utilized Sandvika to develop a simulation of an LNG plant, and for control purposes, an emulation of the AC 870P/ Melody System, a component of the ABB 800xA Distributed Control System, was employed. Using OPC and a 500ms update time, he was able to effectively establish communication between the simulation model and the controller [81].

Because the developed system is based on Siemens platform, and the factory uses Mitsubishi machine tools, in order to realize the function of HiL. The DT system must be able to connect and transmit data with the PLC. For this purpose, data acquisition and exchange modules were developed



**Figure 4.11.** Data acquisition and exchange module

#### 4.4.2 Data mapping strategy

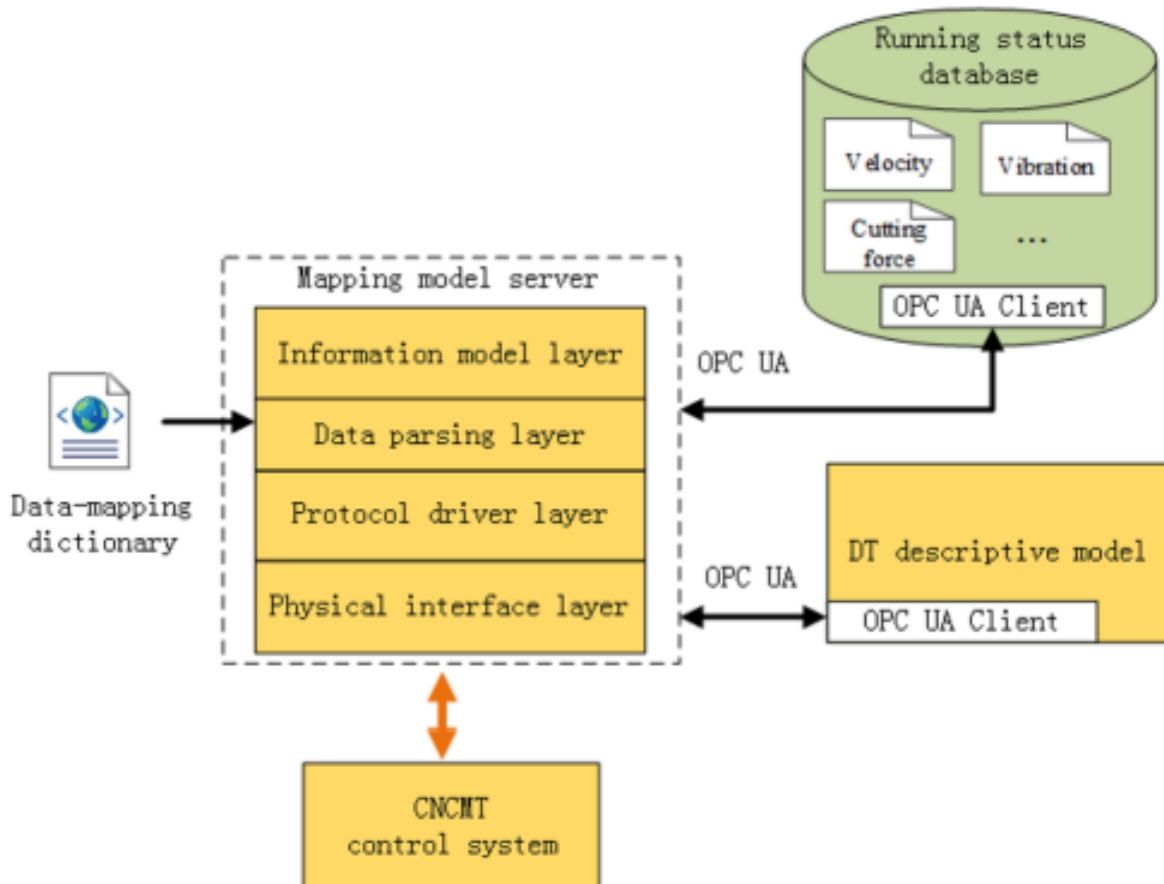
DTMT interacts in real-time with physical CNCMT and functions as a mapping shadow of it. An effective real-time mapping approach between DTMT and CNCMT is therefore required. Additionally, the CNCMT's operating circumstances include kinematics, dynamics, and thermodynamics characteristics according to many topics. CNCMT data is linked, multi-source, and heterogeneous. A mapping model enabling from simple to complicated data with expandability is required to fulfill the DTMT's data-driven applications. Platform-independent interfaces and the interoperability of mapping models continue to be significant. Additionally, the operating data for CNCMT is massive, intricate, unstructured, and updated often, therefore a column database is a good choice for data storage.

As a standardized interface between automation systems, OLE for Process Control Unified Architecture (OPC UA) is frequently utilized in modern times. It makes it possible to move data between the manufacturing floor and the enterprise via a platform-independent interoperability standard. In this study, the mapping methodology is created in accordance with OPC UA specifications.

Running data are acquired from different sensors and the CNC controller, as seen in figure 4.12, and then transferred to the server for the DTMT mapping model. There are four levels in the mapping model server. The compatibility of many interfaces, including RS485, RS232, Wi-Fi, Bluetooth, CAN, and others, is implemented at the physical interface layer. For many interfaces, the protocol driver layer offers a standard read and write technique.

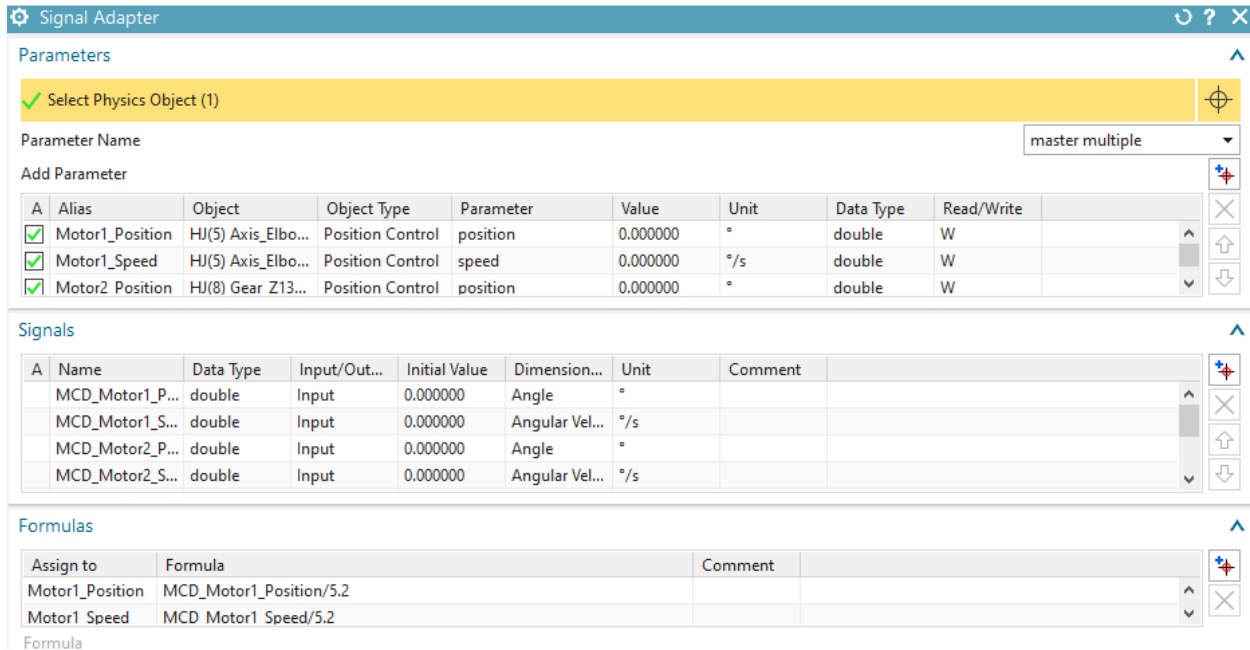
Data processing from various communication protocols is accomplished by the data parsing layer. On the basis of a data-mapping dictionary, the information model layer finally

transforms diverse data into information. The data-mapping dictionary offers comprehensive details on the CNCMT data, including accepted definitions of data items, their meanings, and permitted values. After the mapping model server has parsed and transformed the protocol, running data is published to the DTMT descriptive model and DTMT database. The operating condition loading and system simulation of the DTMT may be carried out by the solver and post-processor in the descriptive model of DTMT. Wrong working conditions and design flaws can be discovered by mining the active data with machine learning methods.



**Figure 4.12.** Digital twin data mapping strategy

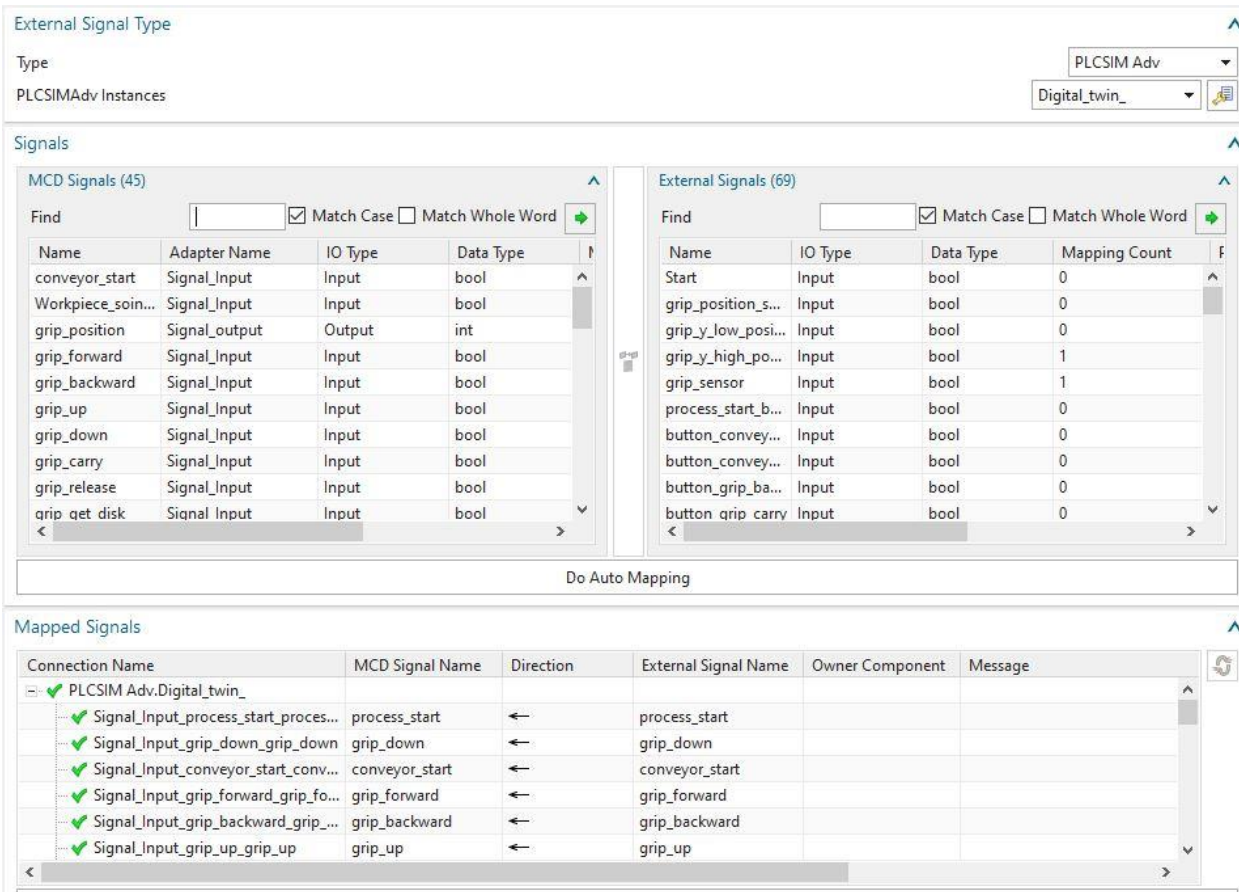
The signal adapter enables the connection between the position control parameters, the input signals, and the required formulae to create a conversion between the value that leaves the PLC and the value that enters the virtual model for the model's proper operation.



**Figure 4.13.** Signals adapter

The mapping between the input signals of the NX MCD with the output signals of the PLC that are carried out through the PLCSIM Advanced software (Figure 4.14).





**Figure 4.14.** Signal Mapping of digital twin

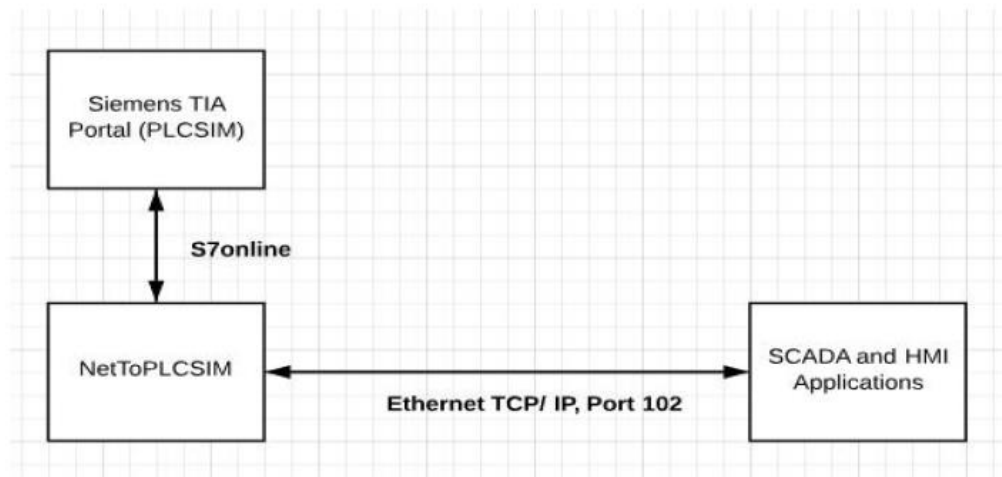
#### 4.4.3 Net to PLCSIM

With the Siemens emulator, Net to PLCSIM is a helpful tool for testing and validating the various systems, such as SCADA and HMI. It provides network connectivity between the user and PLCSIM. When PLCSIM is in operation, this utility establishes network connectivity via the host PC's network interface.

The use of a real PLC is not necessary when using this tool to test client applications. The ability to operate many PLC instances concurrently, allowing for inter-PLC communication, is one of the most valuable features. Also correctly read and written data blocks are supported.

In the initial release of Net to PLCSIM, a library was created utilizing S7ProSim-Com-Object. This interface was sluggish, per the information on the page. The S7-1200 and S7-1500 employ the S7online-interface since they lack the S7ProSim interface, according to the author [82].

For the apps in the Simatic world, S7online-interface leverages the OSI layers. Using the Multipoint Interface (MPI), Profibus, or Transmission Control Protocol (TCP/IP), this interface transfers data from the application layer to the transport layer. As illustrated in the image below [82], the Net to PLCSIM tool utilizes the transport layer (IP/IsoOnTCP) and the S7online-interface for data exchange. In this thesis, control code is tested using the mechatronic model using Net to PLCSIM. Additionally, it connects to Visual Basic. PLCs were employed in many instances, and communication was established satisfactorily.



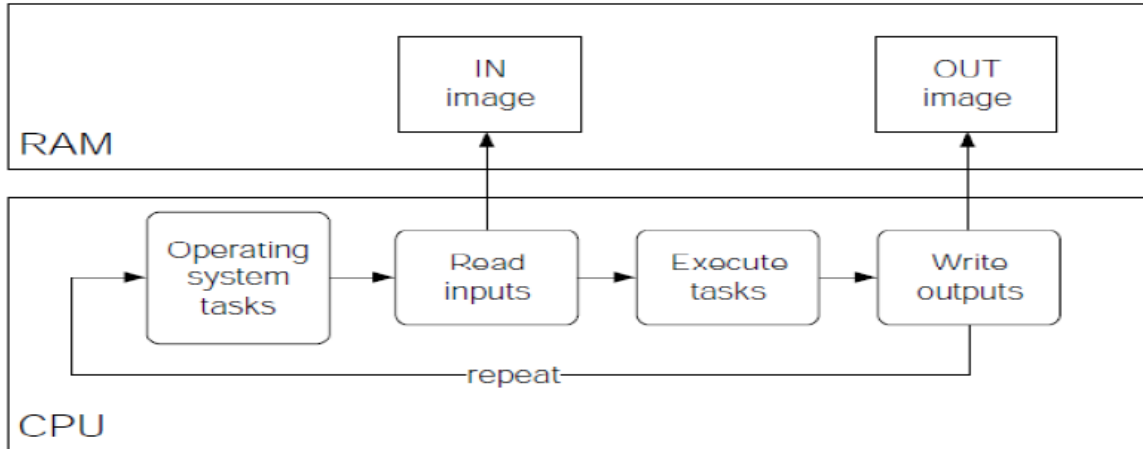
**Figure 4.15.** Net to PLCSIM [82]

#### 4.5 Virtual controller

PLCs are characterized as computers with memory, input/output devices, and a CPU [83]. Depending on the status of the input signals and control logic, the CPU allows the changing of

outputs. PLC may also be referred to as an industrial computer that can operate more basic automation equipment in a demanding industrial setting. PLC is able to handle certain events that are created in real-time as a result of modifications to the manufacturing process or industrial environment. Bolton claims that a PLC is a particular kind of controller built on a microprocessor that uses program memory to store the control program and executes it using timers, counters, arithmetic, and logic. Depending on the program and PLC type employed, this program is then used to control all or a portion of the production operations [84].

A number of communication interfaces, including Ethernet, Profibus, and others, are built into PLCs for connecting to other devices or computers. PLC often adheres to a modular structure, which has several advantages. One advantage is that after deployment, new input and output modules may be combined with the PLC (depending on the type) to accommodate modifications in the initial demand. PLCs are mostly utilized in manufacturing sectors for process control. Special purpose PLCs with extra co-processors can be utilized to manage more complicated production processes and process data quickly and reliably. PLCs are microprocessor-based control systems, therefore by altering the programming of the control code, they may be utilized for a number of purposes and applications.



**Figure 4.16.** Cycle of control code in PLC

Cycles are used to run PLC code that is stored in program memory, as shown in figure 4.16. At the beginning of the scan cycle, it automatically allots some processing time for operating system activities so that everything may be done in real time. The input signals from sensors or other sources are stored in RAM, and the PLC runs the control code in accordance with these signals. The output variables with various values are sent to the out interface when the control code is performed to enable the on/off control of a particular motor, actuator, or other piece of industrial machinery.

The CPU initiates a new scan cycle when the current cycle, which typically lasts a few milliseconds, has finished. This enables the completion of several tasks during a single cycle. Depending on the needs, a variety of languages are available for developing control code. The International Electrotechnical Commission has defined and authorized languages for PLC programming (IEC). The IEC-61131-3 standard was released by this commission to give users adequate instructions and data. This standard also included a list of authoring techniques for

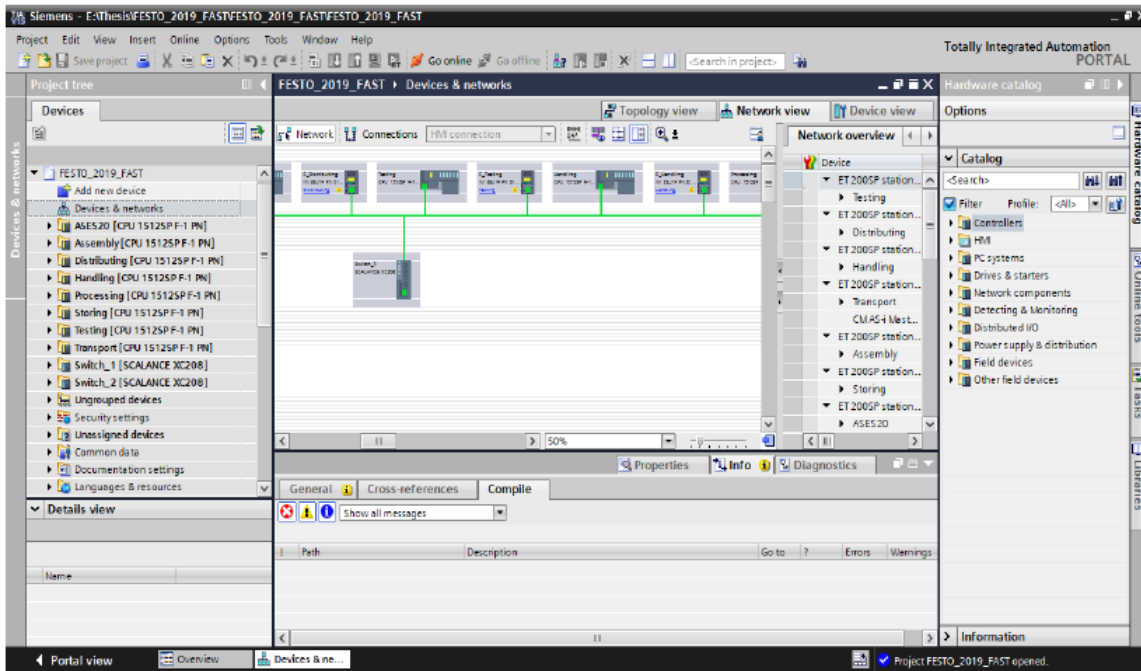
control code. The IEC 61131-3 standard specifies five PLC programming languages, whose names are as follows:

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

The distribution and testing stations in this thesis were both programmed using ladder logic diagrams. This language was created when industrial systems were controlled by electromechanical systems. Along with some functional block diagram components that represent logical expressions, a ladder program may also include contacts (which represent inputs) and coils (which represent outputs). For a large setup, there are approaches that may be used with ladder programming. Timing diagrams, flowchart diagrams, sequence bits, and state diagrams are a few of these approaches. These approaches allow for the creation of well-structured programs, which can aid in troubleshooting efforts.

Siemens provides Totally Integrated Automation (TIA) Portal for PLC programming. Working with motion controllers, drives, and human machine interfaces (HMI) is feasible with this engineering tool. The TIA Portal's user interface is straightforward, which makes it easier to do programming tasks quickly. There is also a library of pre-programmed function blocks that can carry out various tasks. Figure 4.17 depicts this interface, which is simpler to grasp than Siemens Step7 Programming. Another benefit is the ability to combine and program several PLCs in a

single file. A system often consists of several controllers, and it makes programming easier for the programmers if both controllers can be written in the same file.

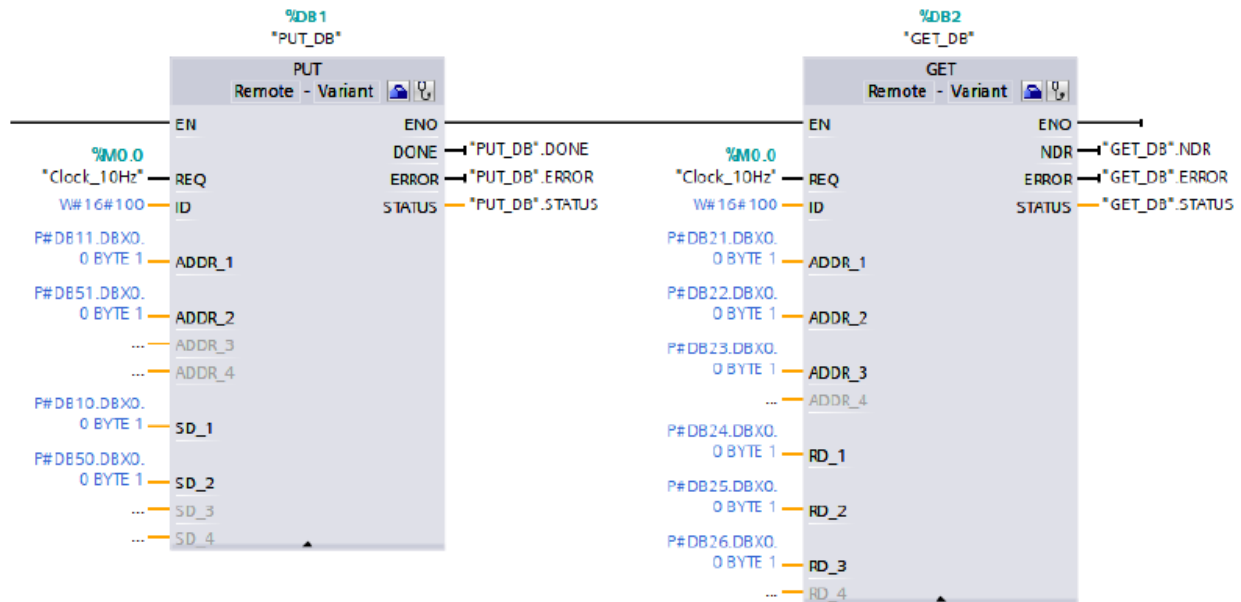


**Figure 4.17.** Virtual controller in TIA Portal

Complete study was needed in order to program the grinding process station based on a digital twin. To fully comprehend how the aforementioned stations worked, a few steps had to be taken. The next stage was to acquaint yourself with the station's real hardware. Signal addressing was compared to the information supplied in the manuals. The TIA portal was equipped with some basic logic, and it was linked to an industrial PLC to see how it operated. The next stage was to develop a sequence diagram of the entire process, taking into account all the signals, after having a general understanding of the system. After the sequence diagram, programming on the TIA site

is quite simple. Finally, by inserting workpieces of various colors and heights into the system, the functionality of both stations was confirmed.

For the safety of the real hardware, communication between distribution controllers and testing stations was also developed, as seen in figure 4.18. It was made possible by utilizing the data blocks utilized in the programming of both stations and the function blocks available in the TIA portal library. Data is sent from one controller to the other controller using the "PUT" function block. The bits that must be transferred to the other controller are designated with SD x. The address of the bits on a different controller to which data will be transferred is ADDR x. These blocks employ a 10Hz clock to obtain or transmit the most recent data in real time.

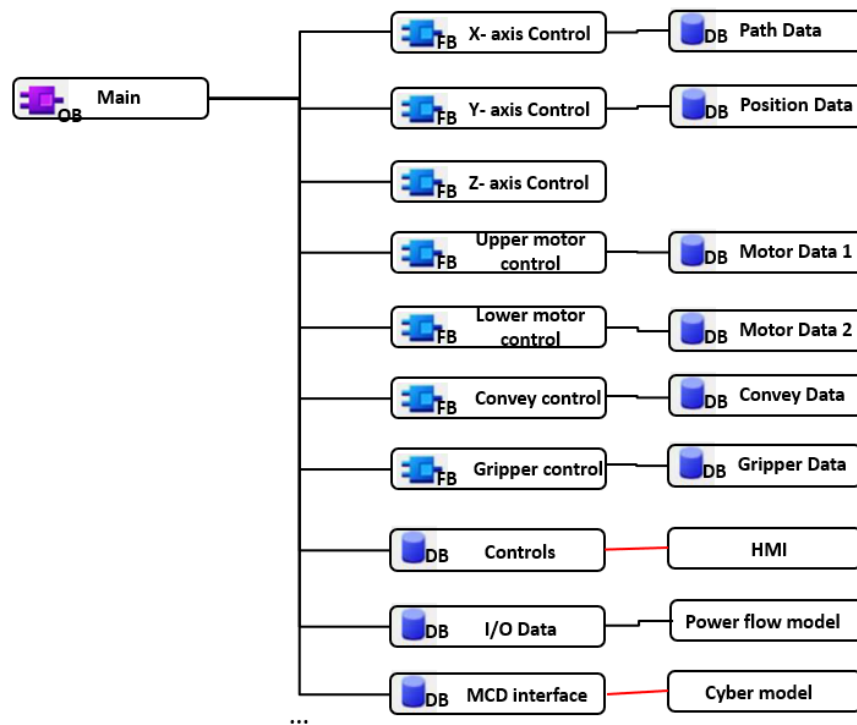


**Figure 4.18.** Data Exchange

There is a "GET" block, similar to the "PUT" block, for obtaining data from other controllers. The address of the bits from which data will be received is indicated in ADDR x. RD

x displays the bits that will be used to save the data from ADDR x for later usage in control logic. Communication between distribution controllers and the testing station was established in this manner. For the sake of the workers' and real hardware's safety, data communication between controllers is essential.

A part of the STEP 7 flowchart for PLC control is shown in Figure 4.19. It is possible to view the organization block's (OB1) cycle main program running continuously when the PLC is operational. All of its functions (FC) are stored in it, including those for controlling the HMI drives and loading the routines (10 in all), which in turn cause the function blocks to be activated and move each of the twin motors in the direction that has been chosen. Data blocks (DB) are where function blocks store their intermediate variables [55].

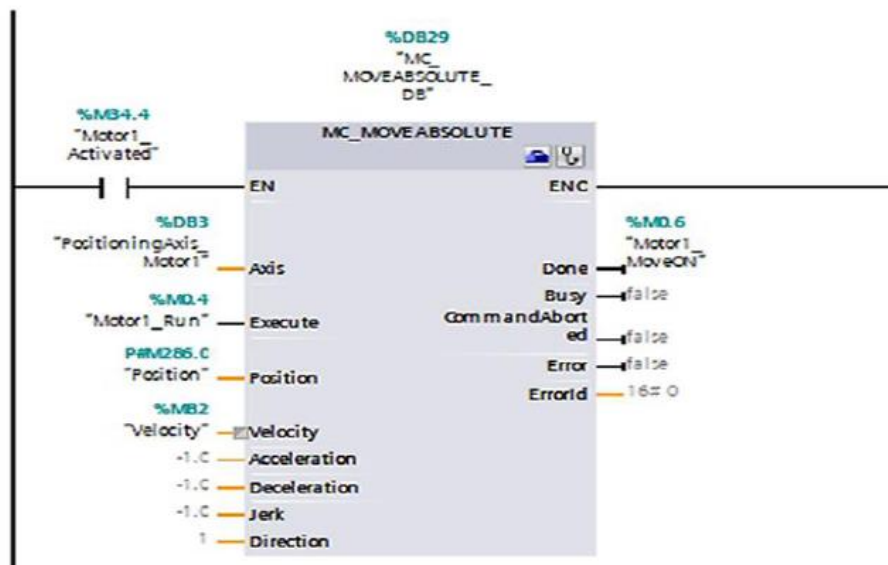


**Figure 4.19.** Virtual PLC encoding



The positioning axis (referred to as "TO Positioning Axis") was the chosen technological object [56]. By using the actual dynamics and mechanics characteristics of the motors, it was set, and the data was recorded in a technical data block.

A movement request is made from the user program using the Motion Control instructions [57] through "Motor1 Run" with absolute coordinates to the destination point specified by the variable "Position" and set speed (Figure-4.20). When the "Busy" flag is set to TRUE, positioning orders are delivered to the axis, which connects to the driver controllers M542 with the aid of pulse generators PTO (Pulse Train Output), from telegram 3 [57]. These drivers act as a power controller (voltage and current), converting the card's pulses into motor steps and controlling the rotation's direction (Figure-4.20). The system alerts the user through "Motor1 MoveON" when the goal position has been attained.

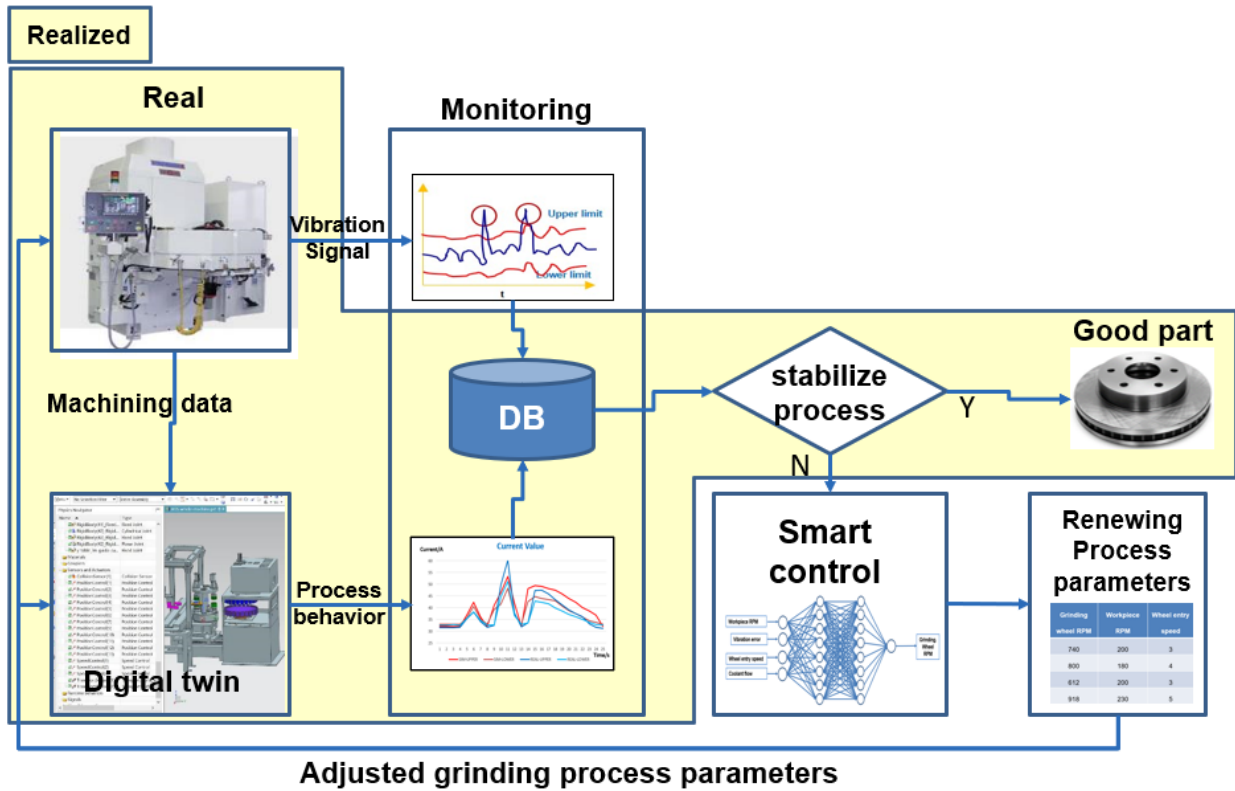


**Figure 4.20.** Motion Control

#### **4.6. Application**

The developed digital twin system is finally applied to the actual factory. There are two application methods, namely SiL and HiL. In the SiL method, the processing parameters and the geometric model parameters of the brake disc are input into the digital twin system, which can predict the process behavior. The operator can adjust the appropriate parameters according to the guidance of the digital twin system to meet the processing requirements. In HiL, the digital twin system is directly connected to the grinding machine in the factory, and real-time process parameters are obtained through PLC. And sensor signals also deliver to the digital twin system. The digital twin system can predict the machining behavior by using real time process parameters, by calculating the grinding force, and then along the power flow to generate current value. And use it as a reference value and compare it with the actual current value. If the deviation between the actual current value and the reference current value exceeds the set threshold, it means that there is a problem with the machine or the processed parts. The system will make an alarm to remind the operator to check the site.

On the other hand, the developed system will be installed as the base platform in the actual plant due to its predicted reliability. Contribute to the development of more applications for the system in the future. The vibration sensor signal will be collected and displayed in real time, by analyzing the stability of the machining process. If there is a problem, the parameters will be adjusted in real time using intelligent control technology, and the adjusted value will be sent to the factory PLC and digital twin system for the next processing and simulation calculation



**Figure 4.21.** Application of implemented system

## Chapter 5 . RESULT AND DISCUSSION

### 5.1.1. Implementation of developed system

The developed digital twin system can be easily applied to a grinding machine for multiple purposes. The developed system is applied to the double-sided vertical type surface grinding machine from Daisho, Japan. The machine is used for the grinding of the automotive brake disc. The developed system is portable. It can be easily installed and relocated from machine to machine. The setting values for monitoring can be changed easily. At first, all the components of the smart system are installed at the real machine at Namyang Nexmo, Korea, as seen in Figure 5.1. The manufacturing of the brake disc consists of several stages. However, our focus is on the stage of the grinding process. The sensors are installed at the grinding process location using mountings. Devices like DAQ and sensors are connected to the developed monitoring system using an ethernet cable. The autonomous monitoring and predicting system is installed into the Control room.

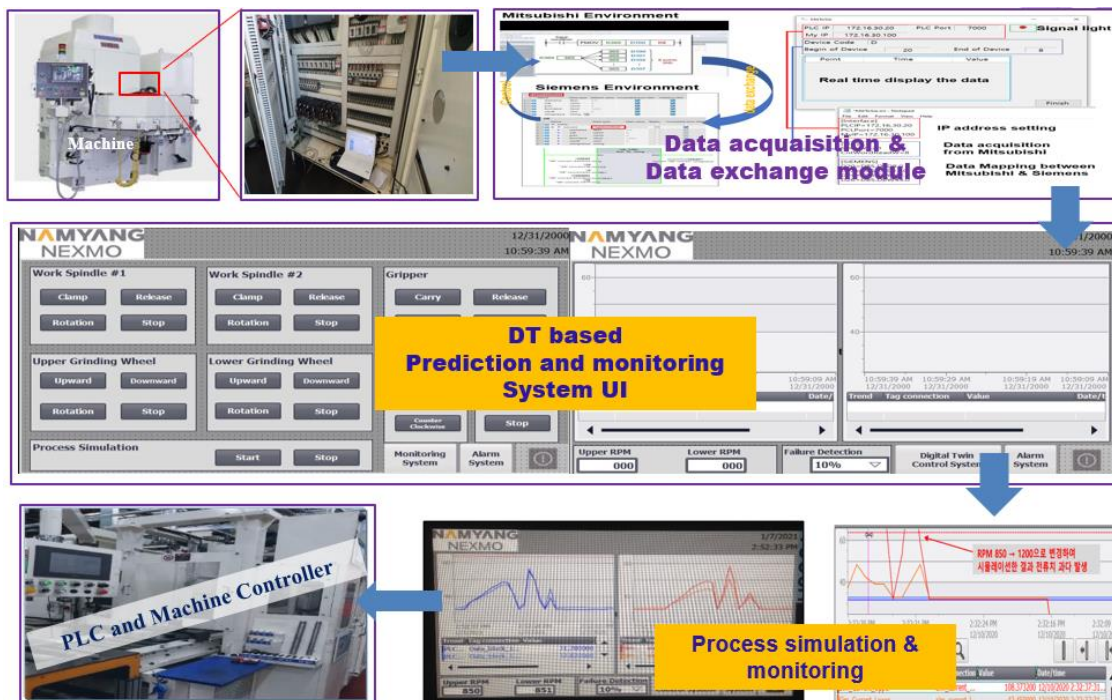
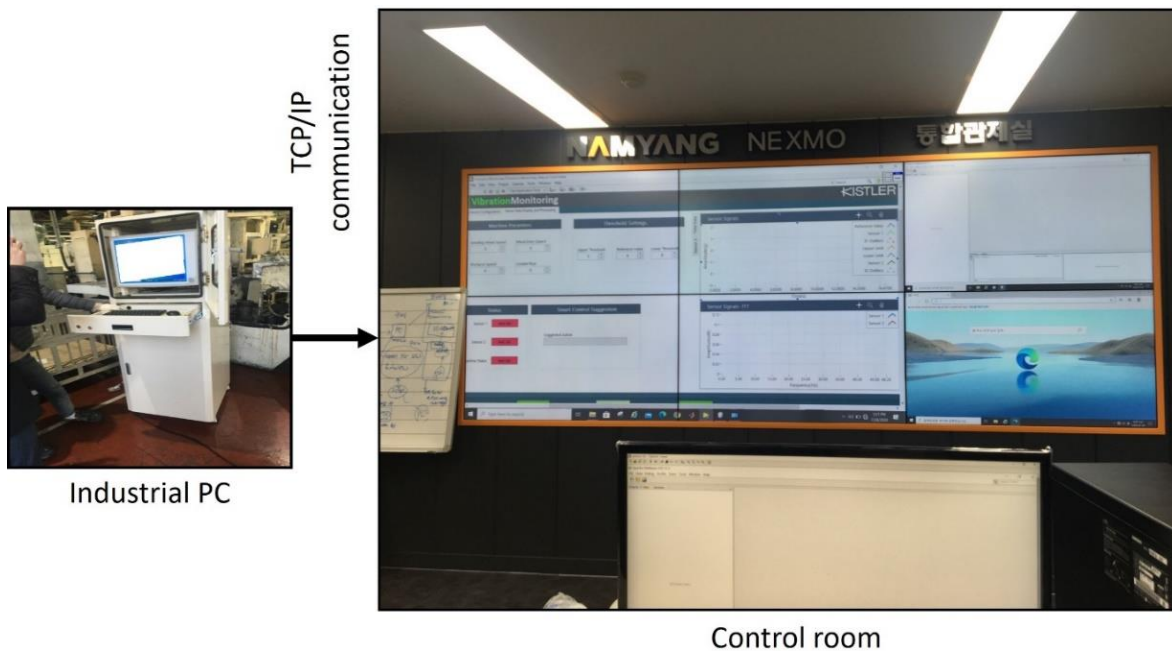


Figure 5.1. Implementation of the system at the site

### 5.1.2. Integrate to the control room

The implemented solution is successfully tested at the industrial PC before being integrated with the company's standard control architecture. TCP/IP is used to transport communication between the control room PC and the industrial PC. Just by sitting in the control room, the plant manager can monitor the behavior of the process. This forecasting technique may potentially be used in the future to test PLC programs. The user double-clicks the program to launch it on the Windows operating system before beginning to utilize the produced system. Then, the gadgets are linked to the program. The program shows the real-time sensor signal after it is attached. It has the potential to significantly alter the factory's overall productivity when fully automated.



**Figure 5.2.** Control room display for the plant manager

## 5.2. Functionality testing of developed system

### 5.2.1. Real factory environment testing

The developed software is converted into the application file and installed into the industrial PC at the machine site. The PC does not require to have any external tools like TIA Portal for its functioning. The developed system is tested for the grinding process of the brake disc. The process parameters information for the brake disc is shown in Table 5.1.

**Table. 5.1.** Process parameter for brake disc grinding

No	Process parameters	Values
1.	Clamp pressure	0.5 ~ 1.5 MPa
2.	Grinding wheel RPM	750 - 850
3.	Workpiece RPM	130 - 250
4.	Feed rate	40/30/20 $\mu\text{m}/\text{sec}$
5.	Spark out time	3.0 sec
6.	Coolant oil concentration	4 $\pm$ 2%

When the start button of the system is pressed, the data acquisition from the sensor initiates, and simultaneously the UI window displays the real-time motor current value. Demonstration of real-time failure instances at our wish is difficult, so it is demonstrated by altering the threshold

limits. In the demonstration scenario, it can be seen that when the vibration amplitude level increases and go beyond threshold limits, the process condition is termed as failure. It is an indication of surface defects like burn marks in the workpiece. Consecutively, the smart control algorithm calculates the difference between the standard and current vibration level and calculates the new process parameters to avoid failure; the new grinding wheel RPM is calculated and delivered to the machine controller through OPC and PLC connection.

### **5.3. Experiment and result**

After the developed digital twin system is correctly connected and installed. By monitoring the real-time processing parameters, including the rotation speed, feed rate motor current, etc., the real-time data process monitoring is shown in the following figure. After the real-time data is obtained in the virtual space, the simulation calculation is carried out, and the predicted spindle motor current value is obtained. The acquisition frequency of the real-time data of the current is once a second. The blue line in the picture is the actual data, and the red is the predicted data. Compare 10 sets of simulated values with actual measured values. The data obtained is shown in the below table.



**Figure 5.3.** Real time experiment of digital twin system

Using mean average value to calculate the accuracy, the formular is:

$$\eta_{Upper} = \frac{\sum_{i=1}^{25} (|SIM\_UPPER - REAL\_UPPER| / REAL\_UPPER)}{25} \quad (5.01)$$

$$\eta_{Lower} = \frac{\sum_{i=1}^{25} (|SIM\_LOWER - REAL\_LOWER| / REAL\_LOWER)}{25}$$

The lowest accuracy measured in the 10 sets of data is used as the effectiveness evaluation of the system. As shown in the below table:



**Table 5.2. Accuracy level of data sets.**

Data Set	Accuracy	
	<i>Upper</i>	<i>Lower</i>
1	95.53	95.15
2	92.78	95.83
3	94.81	95.17
4	92.03	94.87
5	92.78	96.30
6	93.60	96.32
7	92.02	96.00
8	92.93	96.04
9	92.64	96.50
10	92.81	94.95

The minimum accuracy is 92%. Therefore, it can be considered that the developed system is effective to describe the behavior of physical entities.

The result compared between simulation and real data has a certain deviation, although the proposed system calculates the in-process working state of the machine through real-time data analysis. However, it is still the analysis in the static state, the influence of vibration during the grinding process, and the influence of the temperature change of the grindstone and material during the grinding process have not been considered.

## Chapter 6 . CONCLUSION AND FUTURE WORK

This dissertation demonstrates the methodology to develop and implement a digital twin technical in grinding production line for predicting process behavior. The system framework was established based on DTRM. In the physical space layer, the physical entities were rearranged based on the design. Generating the 3D model of grinding production line and give the key corresponding physical parameters to make it have practical significance. The communication module was developed to realize data collection, data processing, and data transfer between physical layer and virtual layer. Due to the characteristics of double side grinding, indirect grinding behavior measurement is proposed and designed in this article. The spindle model was established through analyzing the power losses along the power flow. Deducing the relationship between the processing parameters, the geometric conditions of the workpiece, grinding force and the grinding power, the grinding process model was generated. The developed model was programmed by using C++ and communicate with virtual space layer to get process parameters, machine state. The grinding process behavior can be simulated and predicted in the real time way.

Comparison between predicted result obtained from the proposed digital twin model and experiment, revealed a good agreement between proposed model and practice. With the help of a digital twin, companies can test and validate a product before it even exists in the real world. By creating a replica of the planned production process, a digital twin enables engineers to identify any process failures before the product goes into production.

In further work in the future, the simulation of the process in the virtual space will be optimized, considering the effects of vibration and heat effect. And develop process control and optimization algorithms and use the established digital twin system to control the actual process to improve the rate of quality products and production efficiency.

## Publications

### ❑ Journal

[1] Hong Seok Park; **Bowen Qi**; Duc Viet Dang; Dae Yu Park, “**Development of smart machining system for optimizing feedrates to minimize machining time**”, *Journal of Computational Design and Engineering*, 2018, 5, 299-304, DOI:[10.1016/j.jcde.2017.12.004](https://doi.org/10.1016/j.jcde.2017.12.004)

[2] Hong Seok Park; **Bowen Qi**, “**Autonomous Machining System for Optimizing Feedrates Based on Cutting Force Control**”, *Advances in Transdisciplinary Engineering*, 2017, 6, 108-113, DOI: [10.3233/978-1-61499-792-4-108](https://doi.org/10.3233/978-1-61499-792-4-108)

[3] **Bowen Qi**; Hong Seok Park, “**Data-driven digital twin model for predicting grinding force**”, *IOP Conf. Series: Materials Science and Engineering* 916 (2020) 012092

[4] **Bowen Qi**; Hong Seok Park; Saurabh Kumar; Chang Myung Lee, “**Development of physical-based digital twin system for grinding process prediction and monitoring**” (Submission)

### ❑ Book Chapter

[1] Park, H. S<sup>1</sup>.; **Kumar, S<sup>2</sup>**.; Qi, B<sup>3</sup>., “**Study on the Development of a Deformation Prediction System for 3D Printed Parts in Additive Manufacturing**”, *Recent Trends in Chemical and Material Sciences, Book Pi International.*, 2021, 4, ISBN: [978-93-5547-208-3](https://doi.org/978-93-5547-208-3)

### ❑ Patent

[1] Title: “**Digital Twin Based Grinding Process Control of Brake Disc Manufacturing**”, (Submission)

## ❑ Award

[1] **CDE DX Awards 2021** national software competition, **Silver Award**

## ❑ Program Registration

[1] Title: “**System platform and compatibility NC program interface module development**”,

Registration No. C-2018-000477

[2] Title: “**Cutting force-based feedrate optimization system to minimize machining paths**”,

Registration No. C-2018-026442

[3] Title: “**Tool traversing path minimization system to reduce machining time**”, Registration

No. C-2018-032314

[4] Title: “**Development of NC-program output module to reduce machining time**”,

Registration No. C-2019-018089

[5] Title: “**Intelligent quality management system**”, Registration No. C-2021-003695

[6] Title: “**Mitsubishi and Siemens data exchange module**”, Registration No. C-2021-003694

[7] Title: “**Control Simulation Program for Injection Molding Process Intelligence**”,

Registration No. C-2022-004392

## Reference

1. Boschert, S., & Rosen, R. (2016). Digital twin—The simulation aspect. In *Mechatronic futures*,2016,59-77.
2. Swisher, J. R., Hyden, P. D., Jacobson, S. H., & Schruben, L. W. *A survey of recent advances in discrete input parameter discrete-event simulation optimization*. IIE Transactions, 2004, 36(6), 591–600.
3. Barlas, P., & Heavey, C. *Automation of input data to discrete event simulation for manufacturing: A review*. International Journal of Modeling, Simulation, and Scientific Computing, 2016, 7(1), 1630001.
4. Barricelli, B. R., Casiraghi, E., & Fogli, D. *A survey on digital twin: Definitions, characteristics, applications, and design implications*. IEEE Access, 2019, 7, 167653–167671.
5. Grieves, M. *Digital twin: Manufacturing excellence through virtual factory replication*, White paper, 2014. 1(pp. 1–7).
6. Grieves, M. *Digital twin: Manufacturing excellence through virtual factory replication*. 2015.
7. Grieves, M., & Vickers, J. *Digital twin: Mitigating unpredictable undesirable emergent behavior in complex systems*. In *Transdisciplinary perspectives on complex systems*, 2017, 85–113). Springer.
8. Hochhalter, J. D., Leser, W. P., Newman, J. A., Glaessgen, E. H., Gupta, V. K., Yamakov, V., Cornell, S. R., Willard, S. A., & Heber, G. *Coupling damage-sensing particles to the digital twin concept*,2014,10. National Aeronautics and Space Administration, Langley

Research Center.

9. Wang, S., Liu, Z., Sun, Q., Zou, H., & Yang, F. *Towards an accurate evaluation of quality of cloud service in service-oriented cloud computing*. *Journal of Intelligent Manufacturing*, 2014, 25(2), 283–291.
10. Tao, F., & Qi, Q. *New IT driven service-oriented smart manufacturing: Framework and characteristics*. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 2017, 49(1), 81–91.
11. Panetta, K. *Top 10 strategic technology trends for 2017: Digital twins*, 2017, <https://www.gartner.com/smarterwithgartner/gartner-s-top-10-technology-trends-2017>.
12. Panetta, K. *Top 10 strategic technology trends for 2018: Digital twins*, 2018, <https://www.gartner.com/smarterwithgartner/gartner-top-10-strategic-technology-trends-for-2018>.
13. Panetta, K. *Top 10 strategic technology trends for 2019: Digital twins*, 2019, <https://www.gartner.com/smarterwithgartner/gartner-top-10-strategic-technology-trends-for-2019>.
14. Yoo Ho Son, Goo-Young Kim, Hyeon Chan Kim, Chanmo Jun and Sang Do Noh, *Past, present, and future research of digital twin for smart manufacturing*, *Journal of Computational Design and Engineering*, 2021, 9(1), 1–23
15. Dean, J. Pricing policies for new products. *Harvard Business Review*, 1950, 28(6), 45–53.
16. Rink, D. R., & Swan, J. E. Product life cycle research: A literature review. *Journal of Business Research*, 1979, 7(3), 219–242

17. Nevins, J. L., & Whitney, D. E. *Concurrent design of products and processes: A strategy for the next generation in manufacturing*. McGraw-Hill. 1989.
18. Ryan, C., & Riggs, W. E. *Redefining the product life cycle: The five-element product wave*. *Business Horizons*, 1996, 39(5), 33–40.
19. Hsu, W., & Woon, I. M. *Current research in the conceptual design of mechanical products*. *Computer-Aided Design*, 1998, 30(5), 377–389.
20. Cao, H., & Folan, P. *Product life cycle: The evolution of a paradigm and literature review from 1950–2009*. *Production Planning & Control*, 2012, 23(8), 641–662.
21. Li, J., Tao, F., Cheng, Y., & Zhao, L. *Big data in product lifecycle management*. *International Journal of Advance Manufacturing Technology*, 2015, 81(1–4), 667–684.
22. Tao, F., Cheng, J., Qi, Q., Zhang, M., Zhang, H., & Sui, F. *Digital twin-driven product design, manufacturing and service with big data*. *The International Journal of Advanced Manufacturing Technology*, 2018a, 94(9), 3563–3576.
23. Adolphs, P., Bedenbender, H., Dirzus, D., Ehlich, M., Epple, U., Hankel, M., . . . , & Wollschlaeger, M. *Reference architecture model industrie 4.0 (RAMI 4.0)*. ZVEI and VDI, Status report.
24. Rosen, R., Von Wichert, G., Lo, G., & Bettenhausen, K. D. *About the importance of autonomy and digital twins for the future of manufacturing*. *IFAC-PapersOnLine*, 2015, 48(3), 567–572.
25. Zezulka, F., Marcon, P., Vesely, I., & Sajdl, O. *Industry 4.0 – An introduction in the*

- phenomenon*. IFAC-PapersOnLine, 2016, 49(25), 8–12.
26. Uslar, M., & Sebastian, H. *Model-driven requirements engineering using RAMI 4.0 based visualizations*. 2018, In Modellierung (Workshops).
  27. Zhu, Z., Liu, C., & Xu, X. *Visualisation of the digital twin data in manufacturing by using augmented reality*. Procedia Cirp, 2019, 81, 898–903.
  28. Hankel, M., & Bosch, R. *The reference architectural model industrie 4.0 (RAMI 4.0)*. ZVEI, 2015, 2(2), 4–9.
  29. National Institute of Standard and Technology. "Smart Manufacturing Operations Planning and Control," [http://www.nist.gov/el/msid/syseng/upload/FY2014 SMOPAC ProgramPlan.pdf](http://www.nist.gov/el/msid/syseng/upload/FY2014_SMOPAC_ProgramPlan.pdf).
  30. Kusiak, A. *Smart manufacturing*. *International Journal of Production Research*, 2018, 56(1–2), 508–517.
  31. Kang, H. S., Lee, J. Y., Choi, S., Kim, H., Park, J. H., Son, J. Y., Kim, B. H., & Do Noh, S. *Smart manufacturing: Past research, present findings, and future directions*. *International Journal of Precision Engineering and Manufacturing-Green Technology*, 2016, 3(1), 111–128.
  32. Han, S. *A review of smart manufacturing reference models based on the skeleton meta-model*. *Journal of Computational Design and Engineering*, 2020, 7(3), 323–336.
  33. Rauch, E., & Vickery, A. R. *Systematic analysis of needs and requirements for the design*



- of smart manufacturing systems in SMEs*. Journal of Computational Design and Engineering, 2020, 7(2), 129–144.
34. Tao, F., Anwer, N., Liu, A., Wang, L., Nee, A. Y., Li, L., & Zhang, M. *Digital twin towards smart manufacturing and industry 4.0*. Journal of Manufacturing Systems, 2021, 58, 1–2.
  35. Semeraro, C., Lezoche, M., Panetto, H., & Dassisti, M. *Digital twin paradigm: A systematic literature review*. Computers in Industry, 2021, 130, 103469.
  36. Qi, Q., Tao, F., Hu, T., Anwer, N., Liu, A., Wei, Y., Wang, L., & Nee, A. Y. C. *Enabling technologies and tools for digital twin*. Journal of Manufacturing Systems, 2019, 58, 3–21.
  37. Cheng, J., Zhang, H., Tao, F., & Juang, C. F. *DT-II: Digital twin enhanced Industrial Internet reference framework towards smart manufacturing*. Robotics and Computer-Integrated Manufacturing, 2020a, 62, 101881.
  38. Mourtzis, D., & Vlachou, E. *A cloud-based cyber-physical system for adaptive shop-floor scheduling and conditionbased maintenance*. Journal of Manufacturing Systems, 2018, 47, 179–198.
  39. Tepjit, S., Horváth, I., & Rusák, Z. *The state of framework development for implementing reasoning mechanisms in smart cyber-physical systems: A literature review*. Journal of Computational Design and Engineering, 2021, 6(4), 527–541.
  40. Xu, X. *From cloud computing to cloud manufacturing*. Robotics and Computer-Integrated Manufacturing, 2012, 28(1), 75–86.
  41. Pallis, G. *Cloud computing: The new frontier of internet computing*. IEEE Internet

- Computing, 2010, 14(5), 70–73.
42. Aheleroff, S., Xu, X., Zhong, R. Y., & Lu, Y. (2021). *Digital twin as a service (DTaaS) in industry 4.0: An architecture reference model*. *Advanced Engineering Informatics*, 2021, 47, 101225.
  43. Liu, C., Hong, X., Zhu, Z., & Xu, X. *Machine tool digital twin: Modelling methodology and applications*. 2018, In CIE48 Proceedings.
  44. Ceruti, A., Marzocca, P., Liverani, A., & Bil, C. *Maintenance in aeronautics in an Industry 4.0 context: The role of augmented reality and additive manufacturing*. *Journal of Computational Design and Engineering*, 2019, 6(4), 516–526.
  45. Zhu, Z., Liu, C., & Xu, X. (2019). *Visualisation of the digital twin data in manufacturing by using augmented reality*. *Procedia Cirp*, 2019, 81, 898–903.
  46. Dhiman, H., & Röcker, C. *Middleware for providing activity-driven assistance in cyber-physical production systems*. *Journal of Computational Design and Engineering*, 2021, 8(1), 428–451.
  47. Tao, F., Zhang, Y., Cheng, Y., Ren, J., Wang, D., Qi, Q., & Li, P. *Digital twin and blockchain enhanced smart manufacturing service collaboration and management*. *Journal of Manufacturing Systems*. 2020
  48. Popović, K., & Hocenski, Z. *Cloud computing security issues and challenges*. In *The 33rd International Convention MIPRO*, 2010, 344–349 IEEE.
  49. Liu, C., Jiang, P., & Jiang, W. *Web-based digital twin modeling and remote control of*

- cyber-physical production systems*. Robotics and Computer-Integrated Manufacturing, 2020a, 64, 101956.
50. Erdős, G., Paniti, I., & Tipary, B. *Transformation of robotic workcells to digital twins*. CIRP Annals, 2020, 69(1), 149–152.
  51. Liu, C., Vengayil, H., Lu, Y., & Xu, X. *A cyber-physical machine tools platform using OPC UA and MTConnect*. Journal of Manufacturing Systems, 2019a, 51, 61–74.
  52. Zhang, H., Liu, Q., Chen, X., Zhang, D., & Leng, J. *A digital twin-based approach for designing and multi-objective optimization of hollow glass production line*. IEEE Access, 2017, 5, 26901–26911.
  53. 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
  54. IEEE Standard for System and Software Verification and Validation, IEEE Std 1012-2012 (Revision of IEEE Std 1012-2004), 2012, pp. 1-223.
  55. P.G. Maropoulos, D. Ceglarek, *Design verification and validation in product lifecycle*, CIRP Annals, Vol. 59, Issue. 2, 2010, pp. 740-759.
  56. 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
  57. 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*
58. 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
  59. 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
  60. Jasmin Dzinic, Charlie Yao, *Simulation-based Verification of PLC Programs*, Chalmers University of Technology, 2013, 1-33 p, Gothenburg, Sweden
  61. Peter Hoffmann, *On Virtual Commissioning of Manufacturing Systems*, University of South Wales, 2016
  62. Saurabh Aggarwal & Nenad Nešić & Paul Xirouchakis, *Cutting torque and tangential cutting force coefficient identification from spindle motor current*, Int J Adv Manuf Technol (2013) 65:81–95
  63. Jones AB (1960) *A general theory for elastically constrained ball and radial roller bearings under arbitrary load and speed conditions*. ASME J Basic Eng 82:309–320
  64. Palmgren A (1964) *Grundlagen der Wälzlagertechnik*, 3rd edn. Franckische Verlagshandlung, Stuttgart
  65. Houpert L (1997) *A uniform analytical approach for ball and roller bearings calculations*. J Tribol—T Asme 119(4):851–858
  66. Harris TA (2001) *Rolling bearing analysis*, 4th edn. Wiley- Interscience, New York
  67. Antoine JF, Abba G, Molinari A (2006) *A new proposal for explicit angle calculation in*

- angular contact ball bearing. *J Mech Des* 128(2):468–478.
68. Bossmanns B, Tu JF (2001) *A power flow model for high speed motorized spindles—heat generation characterization*. *J Manuf Sci Eng—Trans ASME* 123(3):494–505
  69. Jordan HE (1994) *Energy-efficient electric motors and their applications*, 2nd edn. Plenum, New York
  70. Hughes A (2006) *Electric motors and drives: fundamentals, types and applications*, 3rd edn. Elsevier, Amsterdam
  71. Bossmanns B (1997) *Thermo mechanical modeling of motorized spindle systems for high speed milling*. Purdue University, West Lafayette
  72. Köfler H (2003) *Stray load losses in induction machines. A review of experimental measuring methods and a critical performance evaluation*. Paper presented at the International Conference on Renewable Energy and Power Quality (ICREPQ)
  73. Agamloh EB (2010) *An evaluation of induction machine stray load loss from collated test results*. *IEEE Transactions on Industry Applications* 46(6):2311–2318
  74. Avram IO (2010) *Machine tool use phase: modeling and analysis with environmental considerations*. Swiss Federal Institute of Technology Lausanne, Lausanne
  75. Novotek OPC and OPC UA Explained, <https://www.novotek.com/en/solutions/keeware-communication-platform/opc-and-opc-ua-explained>.
  76. Ali Heidari, Oliver Salamon, *Virtual Commissioning of an Existing Manufacturing Cell at Volvo Car Corporation Using DELMIA V6*, Chalmers University of Technology, 2012
  77. 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.
78. 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.
  79. 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
  80. 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 System
  81. Herbert Krause, *Virtual commissioning of a large LNG plant with the DCS" 800XA" by ABB*, 2007,
  82. NetToPLCsim - *Network extension for PLCSIM*, <http://nettoplcsim.source-forge.net>
  83. E. Dummermuth, O. Chesterland, Programmable Logic Controller, 3,942,158, 473,149.
  84. W. Bolton, *Programmable Logic Controllers*, Fifth ed. Newnes, Amsterdam, 2009.
  85. Visual Components – A world leader in 3D simulation, <https://www.visualcomponents.com/about-us/>
  86. In-House Solutions Introduces OCTOPUZ Robotic and Simulation Software, 2014
  87. Aksel Øvern, *Industry 4.0 - Digital Twins and OPC UA*, Norwegian University of Science and Technology, 2018, 1-111 p
  88. Siemens System Overview - I/O Systems - Siemens, <http://w3.sie->

mens.com/mcms/distributed-io/en/ip20-systems/et-200sp/system-over-view/pages/default.aspx.

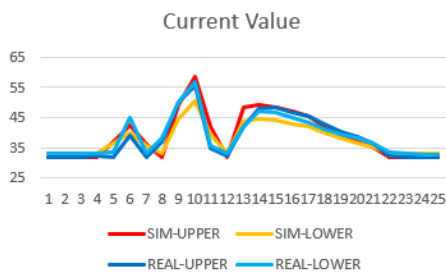
89. Siemens Distributed Controller based on SIMATIC ET 200SP, <https://w3.siemens.com/mcms/programmable-logic-controller/en/distributed-controller/et200sp-based/pages/default.aspx>
90. NetObjex How Can Digital Twin Technology Benefit Your Organization? <https://www.netobjex.com/how-can-digital-twin-technology-benefit-your-organization/>
91. Modelling, Simulation and Optimization in a Data rich Environment, Foundation EU-MATHS-IN, European Service Network of Mathematics for Industry, Netherlands, 1-44 p.
92. Siemens. SIMATIC S7-1500 Motion Control V4.0 en el TIA Portal V15 [Manual de funciones].2017;Availablefrom:  
[https://support.industry.siemens.com/cs/attachments/109749262/s71500\\_motion\\_control\\_function\\_manual\\_es-ES\\_es-ES.pdf](https://support.industry.siemens.com/cs/attachments/109749262/s71500_motion_control_function_manual_es-ES_es-ES.pdf).
93. Siemens. Motion Control Avanzado Digitalización. 2018; Available from:  
<https://assets.new.siemens.com/siemens/assets/api/uuid:9c34c2e2-0f2d-4559-8537-ab2d00dccb06/ws-mc-avanzado-documentacion-2019.pdf>.
94. Leadshine. M542 Economical Microstepping Driver [Datasheet]. Available from:  
<http://www.leadshine.com/uploadfile/down/m542v2m.pdf>.
95. Siemens. SIMATIC WinCC Engineering V15.1: Comunicación [Manual de sistema]. 2018; Availablefrom:  
[https://support.industry.siemens.com/cs/attachments/109755215/WCC\\_Communication\\_es-ES.pdf](https://support.industry.siemens.com/cs/attachments/109755215/WCC_Communication_es-ES.pdf).
96. Siemens and B. Viktor. 2016. Digitalization in Machine Engineering Siemens MCD and

- Cadenas smart catalog components. Available from:  
[https://www.cadenas.de/files/iform/PDF/presentations/2016/en/Industry-Forum\\_2016\\_Psol\\_Siemens\\_Braun\\_EN.pdf](https://www.cadenas.de/files/iform/PDF/presentations/2016/en/Industry-Forum_2016_Psol_Siemens_Braun_EN.pdf).
97. Siemens. SIMATIC Programar con STEP 7 [Manual]. 2006; Available from:  
[https://support.industry.siemens.com/cs/attachments/45531107/s7pr\\_d.pdf](https://support.industry.siemens.com/cs/attachments/45531107/s7pr_d.pdf).
98. Siemens. SIMATIC WinCC Professional The SCADA system inside TIA Portal [System overview]. 2018; Available from:  
<https://assets.new.siemens.com/siemens/assets/api/uuid:fe7ffa96-7714-43c4-a4da-dc98315e6079/dffa-i10492-001.pdf>.
99. Siemens. SIMATIC S7-1500 S7-PLCSIM Advanced [Function manual]. 2016; Available from:  
[https://support.industry.siemens.com/cs/attachments/109773484/s7-plcsim\\_advanced\\_function\\_manual\\_en-US\\_en-US.pdf](https://support.industry.siemens.com/cs/attachments/109773484/s7-plcsim_advanced_function_manual_en-US_en-US.pdf).
100. Siemens. SIMATIC S7-1500 [Catálogo]. 2013; Available from:  
<https://www.technical.cat/PDF/SIEMENS/PLC/siemens-simatic-plc-s7-1500-catalogo.pdf>.
101. Siemens. SIMATIC HMI devices Comfort Panels [Operating Instructions]. 2018; Available from:  
[https://support.industry.siemens.com/cs/attachments/49313233/hmi\\_comfort\\_panels\\_operating\\_instructions\\_enUS\\_en-US.pdf](https://support.industry.siemens.com/cs/attachments/49313233/hmi_comfort_panels_operating_instructions_enUS_en-US.pdf).



## Appendix : Experiment data

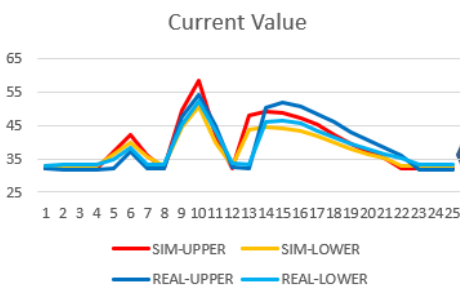
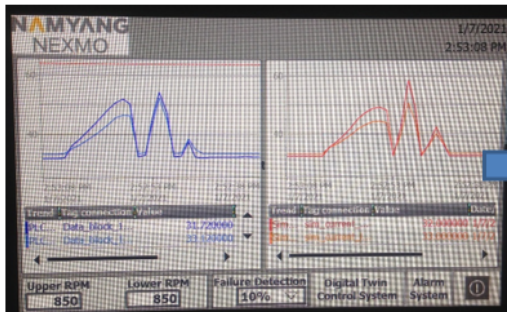
### Dataset 1 of the DT system



### Collected current values

Time	Simulation value		Real DATA		Accuracy	
	SIM-UPPER	SIM-LOWER	REAL-UPPER	REAL-LOWER	Upper	Lower
1	32	33	31.92	33.29	0.997494	0.991289
2	32	33	32.03	33.29	0.999063	0.991289
3	32	33	31.95	33.27	0.998435	0.991885
4	32	33	32.2	33.29	0.993789	0.991289
5	37.01	36.34	32.03	33.46	0.844521	0.913927
6	42.33	39.89	39.16	44.89	0.91905	0.888617
7	36.01	35.67	32.03	33.15	0.875741	0.923982
8	32	33	36.89	38.23	0.867444	0.863196
9	49.43	44.64859	50.05	50.05	0.987612	0.89208
10	58.36705	50.67799	55.45	56.88	0.947393	0.890963
11	42.01	39.67	34.67	35.62	0.78829	0.8863
12	32	33	32.11	33.12	0.996574	0.996377
13	48.32556	43.89516	41.91	42.03	0.846921	0.955623
14	49.43133	44.63558	48.12	46.99	0.972749	0.949895
15	48.53736	44.02707	48.51	46.88	0.999436	0.939144
16	47.07405	43.04466	46.77	45.14	0.993499	0.953581
17	45.29874	41.86944	45.23	43.32	0.99848	0.966515
18	42.16664	39.77893	42.92	41.3	0.982447	0.96317
19	39.55149	38.02118	40.4	39.55	0.978997	0.961345
20	37.1188	36.40144	38.52	38.09	0.963624	0.955669
21	35.35767	35.23726	36.63	36.66	0.965265	0.961191
22	32	33	32.34	33.71	0.989487	0.978938
23	32	33	31.86	33.24	0.995606	0.99278
24	32	33	31.78	32.82	0.993077	0.994516
25	32	33	31.66	32.82	0.989261	0.994516
Mean absolute deviation					0.95537	0.95152

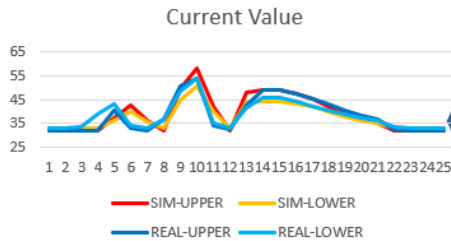
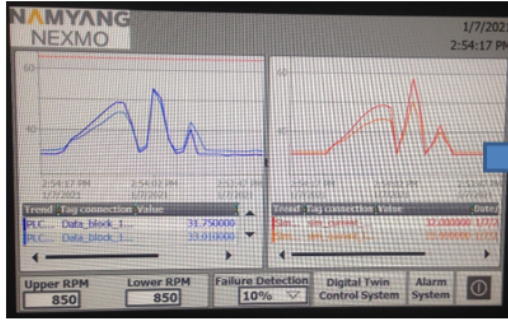
### Dataset 2 of the DT system



### Collected current values

Time	Simulation value		Real DATA		Accuracy	
	SIM-UPPER	SIM-LOWER	REAL-UPPER	REAL-LOWER	Upper	Lower
1	32	33	31.97	33.07	0.999062	0.997883
2	32	33	31.83	33.24	0.994659	0.99278
3	32	33	31.81	33.24	0.994027	0.99278
4	32	33	31.72	33.18	0.991173	0.994575
5	37.01	36.34	31.95	34.7	0.841628	0.952738
6	42.35339	39.92393	37.31	38.29	0.864825	0.957328
7	36.01908	35.67	32.31	33.41	0.885203	0.932356
8	32	33	32.11	33.15	0.996574	0.995475
9	49.46946	44.62	47.58	45.09	0.960289	0.989576
10	58.54554	50.67799	54.29	52.39	0.921615	0.967322
11	42.19203	39.78532	44.92	43.2	0.93927	0.920956
12	32	33	32.68	33.63	0.979192	0.981267
13	48.07	43.73635	32.09	33.24	0.502026	0.684225
14	49.15876	44.43808	50.34	46.21	0.976535	0.961655
15	48.7624	44.16291	51.94	46.49	0.938822	0.949944
16	47.27918	43.1684	50.73	45.56	0.931977	0.947507
17	45.32864	41.8478	48.29	43.23	0.938676	0.968027
18	42.16664	39.74587	45.93	41.41	0.918063	0.959813
19	39.53453	38.00891	43.09	39.5	0.917487	0.962251
20	37.1073	36.39314	40.54	37.84	0.915326	0.961764
21	35.35012	35.22633	38.32	36.35	0.922498	0.969087
22	32	33	36.04	35.12	0.887902	0.939636
23	32	33	31.92	33.29	0.997494	0.991289
24	32	33	31.69	33.15	0.990218	0.995475
25	32	33	31.69	33.24	0.990218	0.99278
Mean absolute deviation					0.92779	0.95834

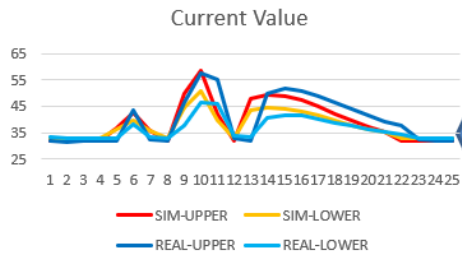
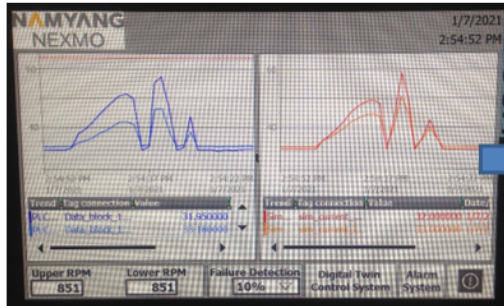
### Dataset 3 of the DT system



### Collected current values

Time	Simulation value		Real DATA		Accuracy	
	SIM-UPPER	SIM-LOWER	REAL-UPPER	REAL-LOWER	Upper	Lower
1	32	33	31.86	33.12	0.995606	0.996377
2	32	33	31.92	33.18	0.997494	0.994575
3	32	33	31.78	33.27	0.993077	0.991885
4	32	33	31.83	33.88	0.994659	0.848765
5	37.01	36.34	40.34	42.92	0.917452	0.846692
6	42.33	39.94093	32.73	33.94	0.706691	0.82319
7	36.0555	35.70293	32	33.07	0.873266	0.920383
8	32	33	36.72	36.55	0.87146	0.902873
9	49.46946	44.62	50.64	48.57	0.976885	0.918674
10	58.36705	50.59178	53.96	53.96	0.918327	0.937579
11	42.16919	39.78532	34.25	35.03	0.768783	0.86425
12	32	33	32.31	33.1	0.990405	0.996979
13	48.10638	43.73635	42.53	41.55	0.868884	0.94738
14	49.15876	44.43808	48.96	45.87	0.99594	0.968783
15	48.7624	44.16291	49.13	45.82	0.992518	0.963835
16	47.24489	43.14359	47.42	43.96	0.996307	0.981428
17	45.29874	41.8478	45.31	42.03	0.999751	0.995665
18	42.16664	39.76239	42.87	40.45	0.983593	0.983001
19	39.53453	38.00891	40.51	38.66	0.97592	0.983159
20	37.1073	36.39314	38.35	37.22	0.967596	0.977785
21	35.35012	35.22633	36.75	36.27	0.961908	0.971225
22	32	33	32.68	33.55	0.979192	0.983607
23	32	33	31.75	33.15	0.992126	0.995475
24	32	33	31.75	32.98	0.992126	0.999394
25	32	33	31.75	33.01	0.992126	0.999697
<b>Mean absolute deviation</b>					<b>0.94808</b>	<b>0.95171</b>

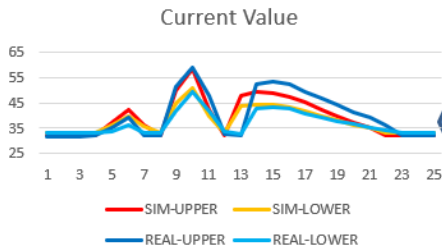
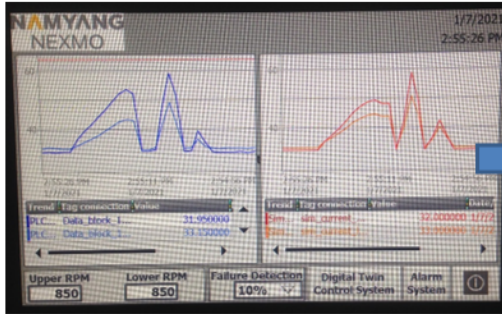
### Dataset 4 of the DT system



### Collected current values

Time	Simulation value		Real DATA		Accuracy	
	SIM-UPPER	SIM-LOWER	REAL-UPPER	REAL-LOWER	Upper	Lower
1	32	33	31.86	33.21	0.995606	0.993677
2	32	33	31.47	32.9	0.983159	0.99696
3	32	33	31.78	33.1	0.993077	0.996979
4	32	33	31.86	33.07	0.995606	0.997883
5	37.02134	36.34822	31.86	33.07	0.837999	0.90087
6	42.42371	39.92393	43.63	38.21	0.972352	0.955144
7	36.01	35.67	32.65	33.55	0.89709	0.936811
8	32	33	31.81	33.15	0.994027	0.995475
9	49.70719	44.73459	46.55	38.01	0.932176	0.823084
10	58.66487	50.67799	57.5	46.46	0.979741	0.909212
11	42.16919	39.73577	55.36	46.21	0.761727	0.859895
12	32	33	33.15	33.8	0.965309	0.976331
13	48.10638	43.73635	31.89	33.27	0.49149	0.685412
14	49.39226	44.52251	49.77	40.73	0.99241	0.906887
15	48.7624	44.10847	51.66	41.74	0.94391	0.943257
16	47.27918	43.11881	50.93	41.83	0.928317	0.969189
17	45.29874	41.82619	48.88	40.28	0.926734	0.961614
18	42.16664	39.72937	46.71	38.77	0.902733	0.975255
19	39.53453	37.99666	44.1	37.76	0.896475	0.993733
20	37.09581	36.37654	41.46	36.19	0.894737	0.994846
21	35.35012	35.22088	39.39	35.17	0.897439	0.998553
22	32	33	37.56	34.44	0.85197	0.958188
23	32	33	32.42	33.15	0.987045	0.995475
24	32	33	31.78	33.07	0.993077	0.997883
25	32	33	31.81	33.15	0.994027	0.995475
<b>Mean absolute deviation</b>					<b>0.92033</b>	<b>0.94872</b>

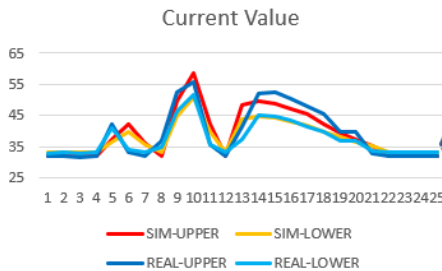
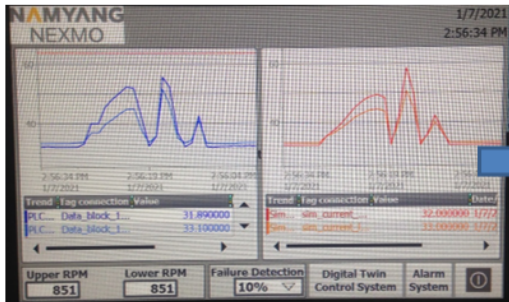
### Dataset 5 of the DT system



### Collected current values

Time	Simulation value		Real DATA		Accuracy	
	SIM-UPPER	SIM-LOWER	REAL-UPPER	REAL-LOWER	Upper	Lower
1	32	33	31.69	33.18	0.990218	0.994575
2	32	33	31.69	33.1	0.990218	0.996979
3	32	33	31.69	33.01	0.990218	0.999697
4	32	33	31.95	33.12	0.998435	0.996377
5	37.04407	36.36469	35.23	33.83	0.948508	0.925076
6	42.35339	39.89	39.39	36.35	0.924768	0.902613
7	36.01	35.67	32.39	33.27	0.888237	0.927863
8	32	33	32.06	33.12	0.998129	0.996377
9	49.74696	44.76332	51.49	41.74	0.966148	0.927568
10	58.66487	50.67799	59.1	49.32	0.992637	0.972466
11	42.07807	39.68641	48.03	41.88	0.876079	0.947622
12	32	33	32.7	33.55	0.978593	0.983607
13	48.10638	43.71	32.23	32.79	0.507404	0.666972
14	49.39226	44.55073	52.22	42.78	0.945849	0.958608
15	48.7624	44.13567	53.65	43.4	0.908898	0.983049
16	47.27918	43.11881	52.19	42.73	0.905905	0.990901
17	45.29874	41.82619	49.41	40.99	0.916793	0.9796
18	42.16664	39.74587	46.69	39.47	0.903119	0.993011
19	39.53453	37.99666	44.19	37.79	0.894649	0.994531
20	37.09581	36.37654	41.32	36.49	0.897769	0.996891
21	35.35012	35.22088	39.25	35.46	0.90064	0.993257
22	32	33	36.35	34.28	0.88033	0.96266
23	32	33	32.14	33.07	0.995644	0.997883
24	32	33	32.11	33.12	0.996574	0.996377
25	32	33	31.97	33.29	0.999062	0.991289
<b>Mean absolute deviation</b>					<b>0.92779</b>	<b>0.96303</b>

### Dataset 6 of the DT system

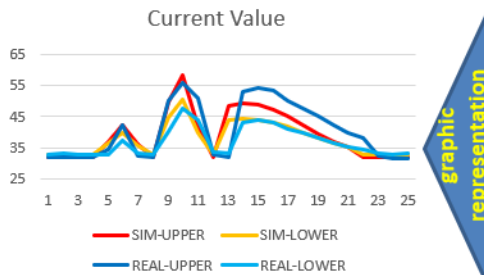
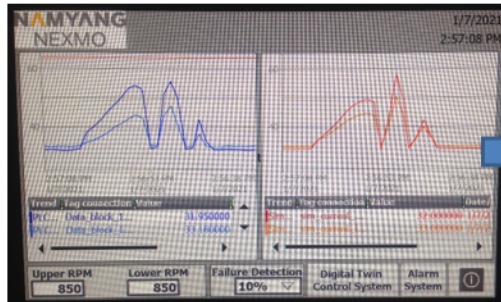


### Collected current values

Time	Simulation value		Real DATA		Accuracy	
	SIM-UPPER	SIM-LOWER	REAL-UPPER	REAL-LOWER	Upper	Lower
1	32	33	31.86	32.93	0.995606	0.997874
2	32	33	31.86	32.98	0.995606	0.999394
3	32	33	31.69	32.93	0.990218	0.997874
4	32	33	31.83	33.12	0.994659	0.996377
5	37.07825	36.38947	42.36	41.01	0.875313	0.887332
6	42.33	39.90696	33.15	34.19	0.723077	0.832789
7	36.01	35.67657	31.89	33.21	0.870806	0.925728
8	32	33	36.8	34.75	0.869565	0.94964
9	49.74696	44.79209	52.39	46.32	0.949551	0.967014
10	58.60517	50.72117	55.84	51.6	0.95048	0.982968
11	42.01	39.68641	35.57	35.43	0.818949	0.879864
12	32	33	32.03	33.27	0.999063	0.991885
13	48.28893	43.81561	41.52	37.34	0.836972	0.826577
14	49.39226	44.55073	52.02	45	0.949486	0.990016
15	48.7624	44.13567	52.44	44.83	0.92987	0.984512
16	47.27918	43.14359	50.28	43.26	0.940318	0.997309
17	45.29874	41.82619	48.12	41.46	0.94137	0.991168
18	42.1438	39.72937	45.34	39.64	0.929506	0.997745
19	39.51759	37.98442	39.84	36.66	0.991907	0.963873
20	37.09581	36.37654	39.84	36.66	0.93112	0.922268
21	35.32	35.21	32.96	33.55	0.928398	0.950522
22	32	33	31.78	33.29	0.993077	0.991289
23	32	33	31.89	33.15	0.996551	0.995475
24	32	33	31.92	33.15	0.997494	0.995475
25	32	33	31.97	33.15	0.999062	0.995475
<b>Mean absolute deviation</b>					<b>0.93592</b>	<b>0.96322</b>



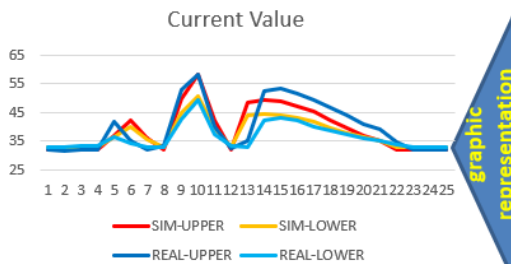
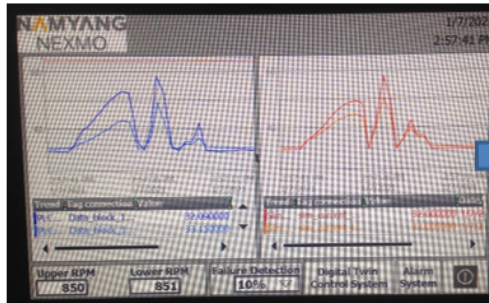
### Dataset 7 of the DT system



### Collected current values

Time	Simulation value		Real DATA		Accuracy	
	SIM-UPPER	SIM-LOWER	REAL-UPPER	REAL-LOWER	Upper	Lower
1	32	33	31.86	33.01	0.995606	0.999697
2	32	33	32	33.07	1	0.997883
3	32	33	31.95	32.82	0.998435	0.994516
4	32	33	31.92	32.79	0.997494	0.993596
5	37.02134	36.34822	34.58	32.9	0.9294	0.895191
6	42.35339	39.90696	42.42	37.28	0.99843	0.929534
7	36.07377	35.70293	32.45	33.18	0.888328	0.923962
8	32	33	32.03	32.98	0.999063	0.999394
9	49.62777	44.67722	50.17	40.2	0.989192	0.888626
10	58.13	50.42	55.89	47.42	0.959921	0.936736
11	42.03267	39.67	51.09	43.88	0.822718	0.904057
12	32	33	32.9	33.6	0.972644	0.982143
13	48.36223	43.8421	32.2	33.29	0.498067	0.683025
14	49.39226	44.52251	52.97	42.95	0.932457	0.963387
15	48.72478	44.10847	54.38	44.07	0.896006	0.999127
16	47.24489	43.11881	53.17	42.92	0.888563	0.995368
17	45.26886	41.8046	50.14	41.1	0.902849	0.982856
18	42.12099	39.71289	47.56	39.86	0.885639	0.996309
19	39.51759	37.98442	45.25	38.01	0.873317	0.999327
20	37.08434	36.36827	42.25	36.44	0.877736	0.998032
21	35.32	35.21544	39.86	35.2	0.886101	0.999561
22	32	33	38.09	34.58	0.840116	0.954309
23	32	33	32.37	33.35	0.98857	0.989505
24	32	33	31.78	33.01	0.993077	0.999697
25	32	33	31.72	33.21	0.991173	0.993677
<b>Mean absolute deviation</b>					<b>0.9202</b>	<b>0.95998</b>

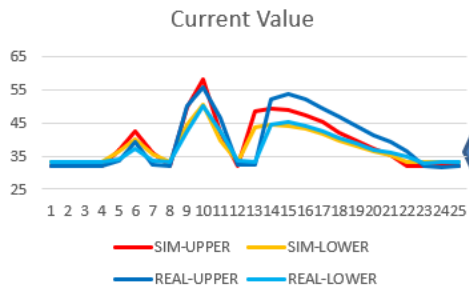
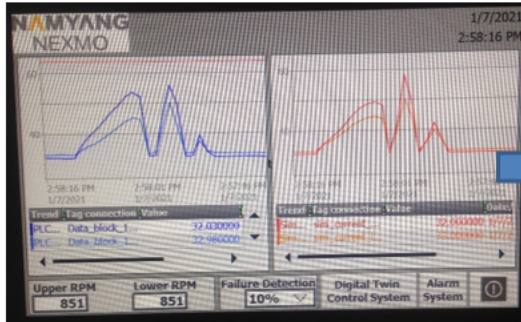
### Dataset 8 of the DT system



### Collected current values

Time	Simulation value		Real DATA		Accuracy	
	SIM-UPPER	SIM-LOWER	REAL-UPPER	REAL-LOWER	Upper	Lower
1	32	33	31.95	32.9	0.998435	0.99696
2	32	33	31.72	32.96	0.991173	0.998786
3	32	33	31.95	33.18	0.998435	0.994575
4	32	33	32	33.24	1	0.99278
5	37.01	36.34	41.86	36.58	0.884138	0.993439
6	42.33	39.89	35.2	34.42	0.797443	0.841081
7	36.02817	35.67	32.03	32.9	0.875174	0.915805
8	32	33	33.38	32.9	0.958658	0.99696
9	49.78679	44.79209	52.92	42.53	0.940793	0.946812
10	58.13	50.42	58.39	49.16	0.995547	0.974369
11	42.01	39.68641	40.06	37.14	0.951323	0.931438
12	32	33	32.31	33.18	0.990405	0.994575
13	48.32556	43.81561	35.01	32.82	0.619664	0.664972
14	49.39226	44.55073	52.53	42.22	0.940268	0.944796
15	48.7624	44.10847	53.28	42.9	0.91521	0.971831
16	47.24489	43.11881	51.54	42.19	0.916665	0.977985
17	45.29874	41.8046	49.35	40.12	0.917908	0.958011
18	42.1438	39.72937	46.46	38.82	0.907099	0.976575
19	39.51759	37.98442	44.02	37.42	0.897719	0.984917
20	37.09581	36.37654	40.93	35.99	0.906323	0.98926
21	35.32	35.21	39.02	35.01	0.905177	0.994287
22	32	33	34.53	33.74	0.92673	0.978068
23	32	33	31.97	33.12	0.999062	0.996377
24	32	33	31.97	33.12	0.999062	0.996377
25	32	33	32	32.98	1	0.999394
<b>Mean absolute deviation</b>					<b>0.9293</b>	<b>0.96042</b>

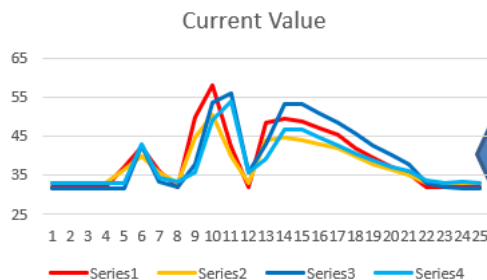
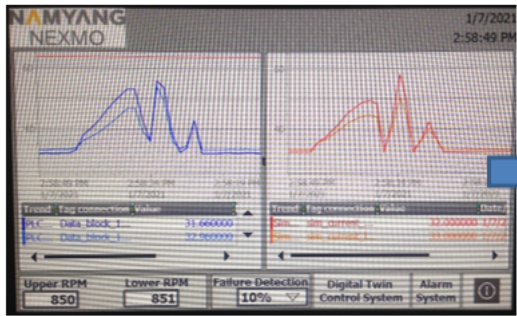
### Dataset 9 of the DT system



### Collected current values

Time	Simulation value		Real DATA		Accuracy	
	SIM-UPPER	SIM-LOWER	REAL-UPPER	REAL-LOWER	Upper	Lower
1	32	33	31.92	33.07	0.997494	0.997883
2	32	33	31.97	33.24	0.999062	0.99278
3	32	33	31.89	33.18	0.996551	0.994575
4	32	33	31.89	33.1	0.996551	0.996979
5	37.01	36.34	33.55	34.08	0.89687	0.933685
6	42.35339	39.89	39.22	37.22	0.920107	0.928264
7	36.07377	35.70293	32.34	33.41	0.884546	0.93137
8	32	33	32.06	33.07	0.998129	0.997883
9	49.54853	44.64859	50.08	42.31	0.989388	0.944727
10	58.13	50.42	55.95	50.25	0.961037	0.996617
11	42.03267	39.68641	46.63	42.39	0.901408	0.936221
12	32	33	32.59	33.66	0.981896	0.980392
13	48.32556	43.8421	32.34	33.07	0.505703	0.674264
14	49.39226	44.55073	52.33	44.61	0.943861	0.998671
15	48.72478	44.10847	53.73	45.25	0.906845	0.974773
16	47.24489	43.11881	52.22	44.13	0.904728	0.977086
17	45.26886	41.8046	49.18	42.28	0.920473	0.988756
18	42.12099	39.72937	46.94	40.31	0.897337	0.985596
19	39.51759	37.98442	44.1	38.66	0.89609	0.982525
20	37.07288	36.36827	41.41	36.92	0.895264	0.985056
21	35.32	35.21544	39.16	35.88	0.901941	0.981478
22	32	33	36.35	34.73	0.88033	0.950187
23	32	33	31.86	32.98	0.995606	0.999394
24	32	33	31.72	33.01	0.991173	0.999697
25	32	33	31.92	33.1	0.997494	0.996979
<b>Mean absolute deviation</b>					<b>0.9264</b>	<b>0.96503</b>

### Dataset 10 of the DT system



### Collected current values

Time	Simulation value		Real DATA		Accuracy	
	SIM-UPPER	SIM-LOWER	REAL-UPPER	REAL-LOWER	Upper	Lower
1	32	33	31.78	33.07	0.993077	0.997883
2	32	33	31.78	32.96	0.993077	0.998786
3	32	33	31.61	32.98	0.987662	0.999394
4	32	33	31.75	32.87	0.992126	0.996045
5	37.01	36.34	31.72	33.01	0.833228	0.899121
6	42.33	39.90696	42.78	42.95	0.989481	0.929149
7	36.0555	35.68974	33.52	34.36	0.924359	0.9613
8	32	33	32	33.32	1	0.990396
9	49.74696	44.84974	37.93	35.74	0.688453	0.745111
10	58.13	50.46287	53.62	49.21	0.91589	0.97454
11	42.16919	39.75227	55.95	54.01	0.753694	0.736017
12	32	33	35.74	35.88	0.895355	0.919732
13	48.32556	43.86861	43.01	39.22	0.876411	0.881473
14	49.39226	44.55073	53.26	46.88	0.92738	0.950314
15	48.72478	44.10847	53.17	46.85	0.916396	0.941483
16	47.21065	43.11881	50.87	44.61	0.928065	0.966573
17	45.23903	41.8046	48.59	42.76	0.931036	0.977657
18	42.12099	39.72937	45.62	40.62	0.923301	0.978074
19	39.50066	37.98442	42.53	38.74	0.928772	0.980496
20	37.05	36.36	40.23	37.31	0.920955	0.974538
21	35.32	35.21	37.98	36.1	0.929963	0.975346
22	32	33	32.98	33.63	0.970285	0.981267
23	32	33	31.92	33.18	0.997494	0.994575
24	32	33	31.78	33.21	0.993077	0.993677
25	32	33	31.75	33.15	0.992126	0.995475
<b>Mean absolute deviation</b>					<b>0.92807</b>	<b>0.94954</b>