MINISTRY OF EDUCATION AND SCIENCE OF UKRAINE
NATIONAL AVIATION UNIVERSITY
Faculty of Aeronavigation, Electronics and Telecommunications
Department of computer integrated complexes

ADMIT TO DEFENSE
Head of the graduating department
__________ Viktor M. Sineglazov
“___ _”_________________2023y.

QUALIFICATION WORK
(EXPLANATORY NOTE)

OF THE GRADUATE OF THE EDUCATIONAL DEGREE
“BACHELOR”

Specialty 151 "Automation and computer-integrated technologies"
Educational and professional program "Computer-integrated technological processes and production"

Theme: Computer aided design of industrial automation systems based on programmable logic controllers and microcontrollers

Performer: student of FAET-404 group Kokhanevych Taras Sergiyovych
Supervisor: PhD, Associate Professor, Professor of Aviation Computer-Integrated Systems Department Sergeyev Igor Yuriyovych

Norm controller: ___________ Filyashkin M.K.
(sign)

Kyiv – 2023

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ

Факультет аеронавігації, електроніки та телекомунікацій

Кафедра авіаційних комп'ютерно-інтегрованих систем

ДОПУСТИТИ ДО ЗАХИСТУ

Завідувач випускової кафедри

______ Віктор СИНЄГЛАЗОВ

“____” _____________2023 р.

КВАЛІФІКАЦІЙНА РОБОТА
(ПОЯСНЮВАЛЬНА ЗАПИСКА)

ВИПУСКНИКА ОСВІТНЬОГО СТУПЕНЯ

“БАКАЛАВР”

Спеціалність 151 "Автоматизація, та комп’ютерно-інтегровані технології"

Освітньо-професійна програма "Комп’ютерно-інтегровані технологічні процеси і виробництва"

Тема: Автоматизована система проектування систем промислової автоматизації на базі програмованих логічних контролерів і мікроконтролерів

Виконавець: студент групи КП-404Ба Коханевич Тарас Сергійович

Керівник: Кандидат технічних наук, доцент, професор кафедри авіаційних комп’ютерно-інтегрованих систем Сергеєв Ігор Юрійович

Нормоконтроллер: ___________ Філяшкін М.К.

(підпис)

Київ – 2023

NATIONAL AVIATION UNIVERSITY

Faculty of Aeronautics, Electronics and Telecommunications

Department of aviation computer-integrated systems

Educational degree: Bachelor

Specialty 174 "Automation, computer-integrated technologies and robotics"

Educational and professional program "Computer-integrated technological processes and production"

APPROVED

Head of department

_________ Sineglazov V.M

“____” ___________2023.

TASK

For the student’s thesis

by: Kokhanevych Taras Sergiyovych

1. **Thesis topic** (project topic) “ Computer Aided design of industrial automation systems based on programmable logic controllers and microcontrollers”
2. **Deadline for an execution of a project**: from May 10 of 2023 to June 7 of 2023
3. **Initial data for the project**: To study and review the process of designing of industrial automation systems based on PLCs and microcontrollers, describe types, features and purpose of programmable logic controllers and microcontrollers.
4. **Contents for explanatory note:**

1. General analysis of Computer aided design (CAD). 2. Analysis of Computer aided manufacturing (CAM). 3. Analysis of PLC programming. 4. Analysis of design methodology and architecture of industrial automation systems. 5. Analysis

of CAD software such as AutoCAD and SolidWorks. Analysis of Model Based Design (MBD) for industrial automation systems.

5. **List of required graphic material**: figures, tables and diagrams.

6. **Calendar schedule-plan:**

№	Task	Execution term	Execution mark
1.	Getting the task	01.04.2023 – 02.04.2023	Done
2.	Formation of the purpose and main objectives of the study	02.04.2023 – 14.04.2023	Done
3.	Analysis of existing methods	15.04.2023 – 30.04.2023	Done
4.	Theoretical consideration of problem solving	01.05.2023 – 05.05.2023	Done
5.	Analysis of algorithm for designing an industrial automation system using CAD software	06.05.2023 – 25.05.2023	Done
6.	Preparation of an explanatory note	26.05.2023 – 03.06.2023	Done
7.	Preparation of presentation and handouts	04.06.2023 – 06.06.2023	Done

7. **Task issue date:** 01 “April” 2023.

Supervisor: __________________________ Sergeyev I.Y

(sign)

Task is taken for completion by: ________________ Kokhanevych T.S

(sign)

НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ

Факультет аеронавігації, електроніки та телекомунікацій

Кафедра авіаційних комп’ютерно-інтегрованих комплексів

Освітній ступінь: Бакалавр

Спеціальність 151 "Автоматизація та комп’ютерно-інтегровані технології"

Освітньо-професійна програма "Комп’ютерно-інтегровані технологічні процеси і виробництва"

ЗАТВЕРДЖУЮ

Завідувач кафедри

_____ Віктор СИНЄГЛАЗОВ

“____” __________2023 р.

ЗАВДАННЯ

на виконання кваліфікаційної роботи студента

Коханевич Тарас Сергійович

1. **Тема роботи** “Автоматизована система проектування систем промислової автоматизації на базі програмованих логічних контролерів і мікроконтролерів”
2. **Термін виконання роботи**: з 10.03.2023 по 7.06.2023

3. **Вихідні дані до роботи**: процес проектування систем промислової автоматизації на основі ПЛК та мікроконтролерів,

4 Зміст пояснювальної записки (перелік питань, що підлягають розробці):

1. Загальний аналіз систем автоматизованого проектування (САП). 2. Аналіз систем автоматизованого виробництва (САМ). 3. Аналіз програмування ПЛК. 4. Аналіз методології проектування та архітектури систем промислової автоматизації. 5. Аналіз програмного забезпечення САПР, такого як AutoCAD

і SolidWorks. Аналіз Модельного проектування (МП) для систем промислової автоматизації.

5 **Перелік обов'язкового графічного матеріалу**: графіки, таблиці, зображення. діаграми.

6 **Календарний план-графік:**

№	Завдання	Термін виконання	Відмітка про виконання
1.	Отримання завдання	01.04.2023 -02.04.2023	Виконано
2.	Формування мети та основних завдань дослідження	02.04.2023 – 14.04.2023	Виконано
3.	Аналіз існуючих методів	15.04.2023 – 30.04.2023	Виконано
4.	Теоретичний розгляд вирішення поставлених завдань	01.05.2023 – 30.04.2023	Виконано
5.	Аналіз алгоритму проектування систем промислової автоматизації з використанням необхідного програмного забезпечення	01.05.2023 – 05.05.2023	Виконано
6.	Оформлення пояснювальної записки	26.05.2023 – 03.06. 2023	Виконано
7.	Підготовка презентації та роздаткового матеріалу	04.06.2023 – 06.06.2023	Виконано

7 Дата видачі завдання ___ «01» березня 2023р.

Керівник: ________________________ Сергеєв І.Ю

(підпис)

Завдання прийняв до виконання: _______________ Коханевич Т.С

(підпис)

ABSTRACT

Explanatory note of qualification work “Automated system for designing of industrial automation systems based on programmable logic controllers and microcontrollers”

PLC CONTROLLER, COMPUTER-AIDED DESIGN, ARTIFICIAL INTELLIGENCE, COMPUTER-AIDED MANUFACTURING, IIoT.

Investigation object – CAD software and its elements, PLC controllers.

Research subject – Industrial automation systems

Purpose of qualification paper - To study and analyze principle of work and architecture of industrial automation systems and computer aided design.

Research method - Comparative analysis, processing of literary journals, digital graphic modeling.

Theoretical research was consisted from deep analyzing of architecture and algorithms of industrial automation systems and computer aided design.

The results of research showed us that computer aided design allows the designer to increase drastically his effectiveness and capabilities while designing an industrial automation system using tools that CAD provides.

The results of qualification paper can be used for acquaintance with computer aided design, its features and useful tools which can be used by future specialists in order to gain knowledge for the practical activities using CAD tools

РЕФЕРАТ

Пояснювальна записка до кваліфікаційної роботи "Автоматизована система проектування систем промислової автоматизації на базі програмованих логічних контролерів та мікроконтролерів"

ПЛК-контролер, автоматизоване проектування, штучний інтелект, автоматизоване виробництво, IIoT.

Об'єкт дослідження - програмне забезпечення САП та його елементи, ПЛК-контролери.

Предмет дослідження - Системи промислової автоматизації

Мета кваліфікаційної роботи - Вивчити та проаналізувати принцип роботи та архітектуру систем промислової автоматизації та автоматизованого проектування.

Метод дослідження - порівняльний аналіз, обробка літературних джерел, цифрове графічне моделювання.

Теоретичні дослідження полягали в глибокому аналізі архітектури та алгоритмів систем автоматизації виробництва та автоматизованого проектування.

Результати досліджень показали, що автоматизоване проектування дозволяє проектувальнику різко підвищити свою ефективність і можливості при проектуванні системи промислової автоматизації, використовуючи інструменти, які надає САП.

Результати кваліфікаційної роботи можуть бути використані для ознайомлення з автоматизованим проектуванням, його особливостями та корисними інструментами, які можуть бути використані майбутніми фахівцями з метою отримання знань для практичної діяльності з використанням засобів САП.

Table of Contents

INTRODUCTION

In the ever-evolving landscape of industrialization and manufacturing, the advent and progression of industrial automation systems stand as a testament to human ingenuity and our persistent pursuit of efficiency and effectiveness. This thesis explores the broad and dynamic realm of industrial automation, delving into its history, current state, and potential trajectory.

Industrial automation systems refer to the wide array of computer-aided, automated technologies that are designed to monitor and control the operation of industrial production and manufacturing processes. These systems have proven pivotal in reducing human error, enhancing productivity, and improving overall process efficiency. From simple mechanized assembly lines of the early 20th century to today's complex, interconnected systems driven by artificial intelligence (AI) and Internet of Things (IoT) technologies, industrial automation has revolutionized the manner in which industries function.

The use of industrial automation systems spans across a multitude of sectors such as automotive, pharmaceuticals, food and beverage, oil and gas, and many more. Despite their ubiquitous presence, these systems remain an area of active research and development, presenting a rich field of study. The evolution of these systems not only mirrors the growth and transformation of industrial sectors but also provides insights into the changing socio-economic aspects associated with industrial growth.

The comprehensive exploration of this significant field will furnish stakeholders with a detailed understanding of the complexities and possibilities inherent in industrial automation systems. In doing so, the thesis will inform decisions around the adoption, design, and regulation of such systems, shaping future industrial practices in a rapidly changing world.

CHAPTER 1. Industrial automation systems. Basic overview and history.

In the vast panorama of technological evolution, industrial automation stands as a significant chapter that has remarkably reshaped manufacturing and production processes worldwide. The roots of industrial automation trace back to the First Industrial Revolution, but its true emergence and maturation span the 20th and 21st centuries, marked by rapid technological advancements.

The genesis of industrial automation can be traced back to the late 18th and early 19th centuries during the First Industrial Revolution. The advent of steam-powered machines marked the shift from manual labor to mechanized processes, particularly in the textile industry. However, these were basic, standalone systems and lacked the level of sophistication we associate with automation today.

The Second Industrial Revolution, occurring between the late 19th century and early 20th century, introduced electrical power into the mix. The shift from steam to electricity fostered the development of assembly lines, a quintessential example being Henry Ford's automobile manufacturing plant in 1913. Ford's assembly line showcased how efficiency and productivity could be dramatically improved by streamlining and standardizing the production process, setting the stage for future advancements in automation.

The mid-20th century heralded the Third Industrial Revolution, characterized by the introduction of electronics, telecommunications, and eventually computers into industrial processes. This period saw the first programmable logic controllers (PLCs), invented in the late 1960s by engineer Richard Morley. These devices, capable of being programmed to control industrial processes, marked a significant step towards modern automation systems.

Simultaneously, industrial robots started appearing on production floors, most notably the Unimate, developed by George Devol and Joseph Engelberger in the 1950s.

Initially used for repetitive tasks in hazardous environments, these machines paved the way for robotic automation in numerous industries.

The late 20th and early 21st century, known as the Fourth Industrial Revolution or Industry 4.0, brought about a fusion of physical and digital technologies. With the proliferation of the internet and advances in computing power, automation systems became increasingly intelligent, interconnected, and capable of handling complex tasks.

One notable development of this era is the Internet of Things (IoT), allowing machines to communicate and cooperate with each other and with humans in real time. This interconnectedness facilitated the emergence of 'smart factories,' where automation systems are not just mechanized but can adapt and make decisions based on real-time data.

Meanwhile, advances in artificial intelligence (AI) and machine learning have further pushed the boundaries of what automation systems can do. These technologies have enabled systems to analyze vast amounts of data, make predictions, and adjust operations to optimize productivity and efficiency.

Today, industrial automation systems have become indispensable in numerous sectors, including automotive, pharmaceutical, food and beverage, and oil and gas industries, among others. The journey of industrial automation has been one of continuous evolution and adaptation, driven by the ever-changing needs of industries and the unending progress of technology. The history of industrial automation is not merely a chronicle of technological advancements but a testament to human ingenuity and our unceasing strive for efficiency, productivity, and progress.

In the rapidly progressing industrial landscape of the 21st century, automation systems have evolved to meet the multifaceted demands of industries worldwide. Today's systems are characterized by digitalization, interconnectedness, and intelligence, reflecting the Fourth Industrial Revolution's ethos or Industry 4.

A modern industrial automation system typically comprises multiple interconnected elements, including sensors, actuators, controllers, human-machine interfaces (HMIs), and network systems. These elements work cohesively to control and optimize industrial processes, from production and assembly lines to quality control and maintenance.

A key player in this modern landscape is the Programmable Logic Controller (PLC), an evolution of the original 1960s' design, which has become increasingly powerful and versatile. Modern PLCs are capable of managing complex operations, often in real-time and harsh environments, making them a mainstay in many industries.

Similarly, industrial robots have come a long way from their early incarnations. Advanced robotics now includes collaborative robots or "cobots", designed to work safely alongside human operators, and autonomous mobile robots (AMRs) that can navigate independently within a facility. These robotic systems perform tasks ranging from precision assembly and packaging to heavy lifting and transportation, enhancing productivity and safety.

The Industrial Internet of Things (IIoT) plays a critical role in modern automation systems. IIoT devices collect and exchange data, enabling greater connectivity and interoperability between equipment, systems, and workers. This heightened connectivity facilitates real-time monitoring and control of industrial processes, leading to increased operational efficiency and predictive maintenance capabilities.

Further enhancing the efficacy of modern automation systems is the incorporation of artificial intelligence (AI) and machine learning technologies. These technologies allow systems to analyze vast amounts of data, identify patterns, make predictions, and adjust operations to optimize productivity, quality, and efficiency. For instance, AI-powered quality control systems can inspect and detect defects with a level of accuracy and speed beyond human capability.

Similarly, digital twin technology, another product of the Industry 4.0 revolution, allows the creation of a virtual replica of a physical system or process. This virtual model can be used to simulate, predict, and optimize system performance, leading to improved product design, process optimization, and predictive maintenance.

The concept of Cyber-Physical Systems (CPS) also characterizes modern industrial automation. CPS are integrations of computation, networking, and physical processes, where embedded computers and networks monitor and control the physical processes with feedback loops. This technology has given rise to "smart factories," where cyber-physical systems monitor physical processes, create a virtual copy of the physical world, and make decentralized decisions.

All these elements combine to form the backbone of Industry 4.0. However, it's crucial to note that modern industrial automation is not just about technology. It also involves the transformation of organizational structures, work processes, and workforce skills. The human factor remains crucial, both in terms of operating and maintaining automation systems and in terms of the broader decision-making processes required to implement and manage these systems effectively.

In conclusion, modern industrial automation systems are increasingly intelligent, adaptable, and interconnected. They have moved far beyond the realm of simple mechanization to become complex ecosystems, embodying the digital and physical convergence. The future promises even more advancements, with emerging technologies such as quantum computing and 5G promising to push the boundaries of what industrial automation can achieve.

1.1 Types of industrial automation systems and their general purpose and usage in modern industries.

Modern industrial automation systems are engineered to serve a multitude of purposes that form the backbone of contemporary industries. They facilitate a range of operations, from enhancing productivity, improving product quality, to reducing production costs and ensuring worker safety. These systems vary widely in their complexity, capabilities, and application, reflecting the diversity and specificity of industrial needs.

Fixed Automation Systems

Fixed or "Hard" automation systems are used for high volume, repetitive tasks. This category includes machinery designed to perform specific tasks without variation, like assembly lines in an automotive plant or filling lines in a beverage factory. Fixed automation systems are generally characterized by high capital investment, high production rates, and a long-life cycle.

Key features of fixed automation systems include their high-speed production capability and their relative inflexibility when it comes to modifying or changing the sequence of operations. However, their efficiency and reliability in the context for which they were designed make them invaluable for many industrial applications.

Programmable Automation Systems

Programmable automation systems allow for the change or reprogramming of operation sequences to accommodate different product configurations. These systems are generally used in industries with medium-scale production, where the product or parts may change periodically.

A quintessential example is the Programmable Logic Controller (PLC). PLCs are microprocessor-based devices that can be programmed to perform control functions for a wide range of industrial applications. They are highly adaptable and capable of managing complex operations in real-time.

Flexible Automation Systems

Flexible or "Soft" automation systems are designed for low to medium production volumes with a high variety of products. They can be easily adjusted to handle various products without losing significant time for changeover. Robotics, especially collaborative robots (cobots) and autonomous mobile robots (AMRs), often play a crucial role in flexible automation systems.

Cobots can work safely alongside humans and be programmed to perform a variety of tasks. AMRs, on the other hand, can independently navigate through facilities, performing transportation tasks with high adaptability. These robotics systems enhance flexibility, efficiency, and safety in industrial environments.

Industrial Internet of Things (IIoT)

IIoT devices form a significant subset of modern automation systems. By collecting and exchanging data, these devices foster greater connectivity and interoperability among equipment, systems, and human operators. Real-time monitoring and control of industrial processes, increased operational efficiency, and predictive maintenance capabilities are key features of IIoT-based systems.

Cyber-Physical Systems and Smart Factories

Modern automation also incorporates Cyber-Physical Systems (CPS), marking the convergence of computation, networking, and physical processes. CPS leads to the creation of 'smart factories,' where cyber-physical systems monitor physical processes, create a virtual copy of the physical world, and make decentralized decisions.

AI and Machine Learning in Automation

Artificial intelligence (AI) and machine learning technologies are increasingly being incorporated into industrial automation. These technologies allow automation systems to analyze large amounts of data, identify patterns, make predictions, and adjust operations. They enhance various aspects of industrial processes, from predictive maintenance to quality control.

Digital Twin Technology

Digital twin technology, another modern automation feature, enables the creation of a virtual replica of a physical system or process. This virtual model can be used to simulate, predict, and optimize system performance, leading to improved product design, process optimization, and predictive maintenance.

In conclusion, the purpose of modern industrial automation systems extends far beyond simple mechanization. They are designed to enhance productivity, ensure product quality, reduce production costs, and increase worker safety, among other things. The diversity of these systems, from fixed and programmable to flexible automation, coupled with technologies like IIoT, AI, machine learning, and digital twins, offer boundless opportunities for industries to improve their processes and outcomes. As we continue to push the boundaries of technology, the capabilities and potential of these systems promise to expand even further.

Industrial automation systems are integral to a wide variety of industries, enhancing efficiency, precision, and safety in a multitude of processes and operations. The manufacturing industry is perhaps the most notable beneficiary of these systems, where automation has revolutionized the production of goods, with industries such as automotive, aerospace, electronics, and pharmaceuticals all heavily relying on automated production lines for assembly, quality control, and packaging tasks. Automotive manufacturers, for instance, make use of sophisticated robotics and automation systems for tasks ranging from welding and painting to assembly and inspection.

In the aerospace industry, automation is employed in the assembly of complex aircraft components, ensuring precision and consistency while also reducing the risk of human error. The electronics industry, particularly semiconductor manufacturing, is another field where automation plays a crucial role, with highly precise and fast robots used in the assembly of delicate and minute electronic components. The pharmaceutical industry leverages automation in drug discovery, testing, production, and packaging processes, ensuring strict adherence to quality and safety standards.

Food and beverage industry is another major user of industrial automation systems, which are employed in a variety of tasks including processing, packaging, and labeling, helping companies to meet stringent food safety standards, improve productivity, and reduce waste. In the chemical and petrochemical industry, automation systems are used in process control to regulate variables such as pressure, temperature, and flow in chemical reactors and other equipment, significantly enhancing safety and efficiency.

Automation also plays a vital role in the utilities sector, particularly in energy production and distribution. In power plants, automated control systems are used to regulate processes such as power generation and distribution, ensuring stable and reliable supply of electricity. In the oil and gas industry, automation is used in drilling operations, pipeline monitoring, and in controlling refining processes.

Moreover, automation systems are being increasingly used in the agriculture industry, where they help to automate tasks such as planting, irrigation, fertilizing, and harvesting, enhancing productivity and sustainability in the process. In the logistics and warehousing industry, automation systems, often incorporating robotics and AI technologies, are used to enhance efficiency in tasks such as sorting, picking, packing, and transporting goods.

In the healthcare industry, automation is utilized in numerous applications including laboratory analysis, patient data management, and even in surgical procedures with the help of robotic surgery systems. In the construction industry, automation systems are used in various applications ranging from prefabrication of building components to autonomous construction equipment.

Lastly, in the mining industry, automation is increasingly being implemented in areas such as drilling, blasting, loading, hauling, and ore processing, improving safety and productivity in what is traditionally a labor-intensive industry. These examples only scratch the surface of the wide array of industries where industrial automation systems are employed, and with continuous advancements in technology, the scope of their application is set to expand even further in the future.

1.2 The designing process of an industrial automation system.

Designing industrial automation systems involves integrating many different elements to create a cohesive, efficient, and productive system. With advancements in artificial intelligence and machine learning, it's now possible to automate some parts of this process. Here's how an automated system might work for designing industrial automation systems based on programmable logic controllers (PLCs) and microcontrollers. We can divide this process step-by-step:

System Requirement Gathering: The first step in any design process is understanding the requirements. An AI system could facilitate this process by providing a simple interface for entering information about the needed industrial automation system. This could include details about the processes to be automated, the expected outputs, and any constraints.

System Design: Once the requirements are understood, the automated system can then design the automation system. This involves selecting the appropriate PLCs or microcontrollers, designing the control logic, specifying the communications protocols and other hardware requirements, and mapping out the network architecture. Advanced AI systems could potentially use machine learning algorithms to optimize this process based on previous successful designs.

Simulation and Testing: Before the actual implementation, the system would simulate the whole setup to check for any possible flaws or inefficiencies. Machine learning algorithms could learn from these simulations to improve future designs.

Code Generation: Depending on the PLCs and microcontrollers used, code generation could also be automated. This involves translating the control logic into the programming language used by the PLCs and microcontrollers.

Documentation Generation: Creating the necessary documentation is a vital part of the design process. An automated system could generate this documentation based on the design, including the system overview, the design rationale, the specifications of the PLCs and microcontrollers, the control logic, and the programming code.

Maintenance and Updates: Post-implementation, the system can help with maintenance and updates. Machine learning algorithms could monitor the system's performance and suggest updates or modifications to improve efficiency or address problems.

Such an automated design system could greatly speed up the design process, improve the efficiency and effectiveness of the designs, and reduce the chance of human error.

1.2.1 Automated processes in designing of industrial automation systems.

Designing an automated industrial system can also be assisted by various automated processes.

Before a system is built, it can be modeled and simulated using software. This allows designers to test different configurations and parameters in a virtual environment. There are also automated tools that can optimize the design by running simulations with different parameters and selecting the best ones.

Modeling and simulation are critical parts of the design process for industrial automation systems. These methods are used to understand how the system will behave before it is physically implemented, allowing designers to identify and correct potential issues early in the design process.

Here's a general overview of how the modeling and simulation process might work:

- Defining the System: The first step is to define the system that will be modeled. This involves identifying the components of the system, their properties, and how they interact with each other. For example, in an automated assembly line, components might include machines, conveyor belts, sensors, and controllers, and interactions might include how machines perform tasks based on signals from sensors and controllers.

- Creating the Model: Once the system is defined, a model is created. The model is a mathematical or logical representation of the system, and it describes how the system behaves. There are many different ways to model systems, depending on their complexity and the level of detail required. For instance, a system could be modeled using equations that describe the behavior of the components, or using a state machine that describes the different states the system can be in and the transitions between them.

- Implementing the Model in Simulation Software: The model is then implemented in simulation software. This software takes the model and uses it to simulate the behavior of the system over time. There are many different types of simulation software available, ranging from general-purpose tools to software designed specifically for industrial automation systems.

- Running Simulations: Once the model is implemented, simulations can be run. A simulation involves setting the initial conditions of the system, running the simulation for a certain period of time, and then observing the results. The results might include things like the state of the system at different times, the performance of the system, or any problems that occurred.

- Analyzing the Results: After a simulation is run, the results are analyzed. This might involve looking for potential issues, assessing the performance of the system, or verifying that the system behaves as expected. If any issues are found, the model might need to be refined, or the design of the system might need to be adjusted.

- Iterating the Process: Typically, the modeling and simulation process is iterative. After analyzing the results, new simulations are run with adjusted models or different initial conditions. This process is repeated until the designers are satisfied with the performance and behavior of the system.

By using modeling and simulation, designers can experiment with different designs, test how the system reacts to different conditions, and optimize the system before it's physically built. This can save significant time and resources in the design process, and it can result in a more effective and reliable automation system.

1.2.2 Automating programming process of PLC controllers while designing an automation system

Automating the programming of Programmable Logic Controllers (PLCs) in the context of designing an automation system involves employing code generation tools, model-based design strategies, and programming software that leverages standardized industrial languages. Code Generation Tools: These are software tools that can generate code automatically based on specific input parameters or predefined templates. One example is Structured Text (ST) code generators which can automatically generate ST code for PLCs based on higher-level models or designs. This speeds up the coding process and reduces the possibility of errors that could occur during manual coding.

Model-Based Design: This is a method of designing control systems where a system model is used as the central artifact in the design process. Software like MATLAB/Simulink can be used to design and simulate the control system, then automatically generate PLC code from the model. This ensures that the implemented control system accurately reflects the designed model.

Standardized Industrial Languages: The IEC 61131-3 standard defines five languages for programming PLCs: Ladder Diagram (LD), Function Block Diagram (FBD), Structured Text (ST), Instruction List (IL), and Sequential Function Chart (SFC). Tools that support these languages allow for a certain degree of code reusability, which can speed up the PLC programming process. They also often include libraries of predefined function blocks that can simply be dropped into a program, which can greatly reduce the amount of code that needs to be written manually.

Integrated Development Environments (IDEs): IDEs such as Siemens TIA Portal, Rockwell's Studio 5000, or Schneider Electric's EcoStruxure Machine Expert provide extensive libraries of ready-to-use functions and function blocks, graphical programming interfaces, and built-in simulation capabilities. This not only speeds up the development process but also allows for early testing and debugging, thus reducing the time spent on troubleshooting during commissioning.

PLC Open Libraries: PLCopen is an organization that aims to create efficiency in industrial automation by harmonizing PLC programming standards. Their XML based format allows the interchange of PLC program data between different programming environments, making the process easier and more standardized.

While automating the PLC programming process can greatly speed up the development of an automation system and reduce errors, it's important to note that it still requires a high level of expertise to design the system, to choose the right tools and strategies, and to ensure that the generated code accurately implements the desired behavior. It's also essential to thoroughly test the implemented control system to ensure its safe and reliable operation.

1.2.3 Computer-Aided Manufacturing as an essential part of designing an automated industrial system.

Computer-aided manufacturing (CAM) is an important aspect of modern industrial automated systems. Essentially, CAM uses computer software and hardware to automate the manufacturing process, increasing efficiency, accuracy, and flexibility. This automation covers all stages - from planning, management, control to the physical production of goods. CAM works closely with computer-aided design (CAD) systems, where CAD develops a 3D model of the product and CAM uses this model to control equipment to produce the physical product. Tight integration between CAD and CAM optimizes the entire production process, reduces errors and increases speed.

CAM is widely used in computer numerical control (CNC) machines. These are electromechanical devices that manipulate tools around a different number of axes (usually three or five) with high precision according to instructions encoded in a program. The coded instructions are prepared using a CAM system, where engineers enter the parameters required to perform manufacturing operations. This can include drilling holes in specific locations, cutting out sections, molding complex structures, and so on. Once the program is created, it can be loaded into a CNC machine, which then performs operations with minimal human intervention. The advantage of such a system is that it can produce complex shapes and structures with high precision, and it is also capable of repeating the process as many times as needed without deviation.

CAM also plays a crucial role in the implementation of advanced manufacturing technologies, such as additive manufacturing or 3D printing. In this case, the CAM software cuts the 3D model into thin layers and then controls the 3D printer to apply the material layer by layer, gradually creating the final product. The software can control various aspects of the printing process, such as the print head path, material application rate, and print bed temperature, ensuring optimal printing conditions.

CAM systems are also crucial for the implementation of flexible manufacturing systems (FMS). An FMS consists of a group of workstations interconnected by an automated material handling system that is controlled by a central computer system. Each workstation in an FMS is typically equipped with a CNC machine or industrial robot, and a central computer controls the operation of the entire system. The computer uses CAM software to plan the production process, schedule operations at each workstation, control the material handling system, and monitor system performance. This allows the FMS to quickly adapt to changes in product design or production schedule, increasing flexibility and efficiency.

Finally, it's important to note that CAM systems are not just for machine control. They also play an important role in other aspects of the manufacturing process. For example, they can help with production planning and scheduling, resource management, quality control, and performance data analysis. In this way, CAM systems help manufacturers optimize their operations, reduce waste, improve quality, and increase efficiency. As technology advances, CAM systems are becoming more sophisticated and powerful, further enhancing their role in industrial automation systems. However, despite the many advantages, there are also challenges to overcome, such as the high initial cost, the need for skilled personnel to operate and maintain the system, and the need to integrate the CAM system with other systems in the organization. However, the benefits usually outweigh these challenges, making CAM an invaluable tool in the world of industrial automation.

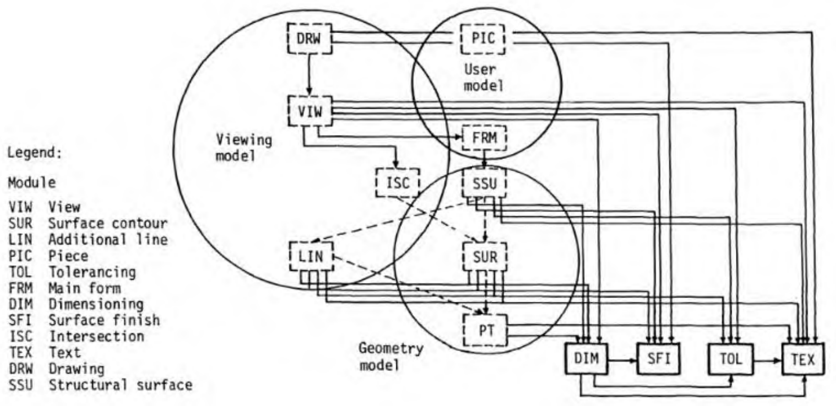


Figure1.1 Interconnection of the technology model with the geometry and drawing models.

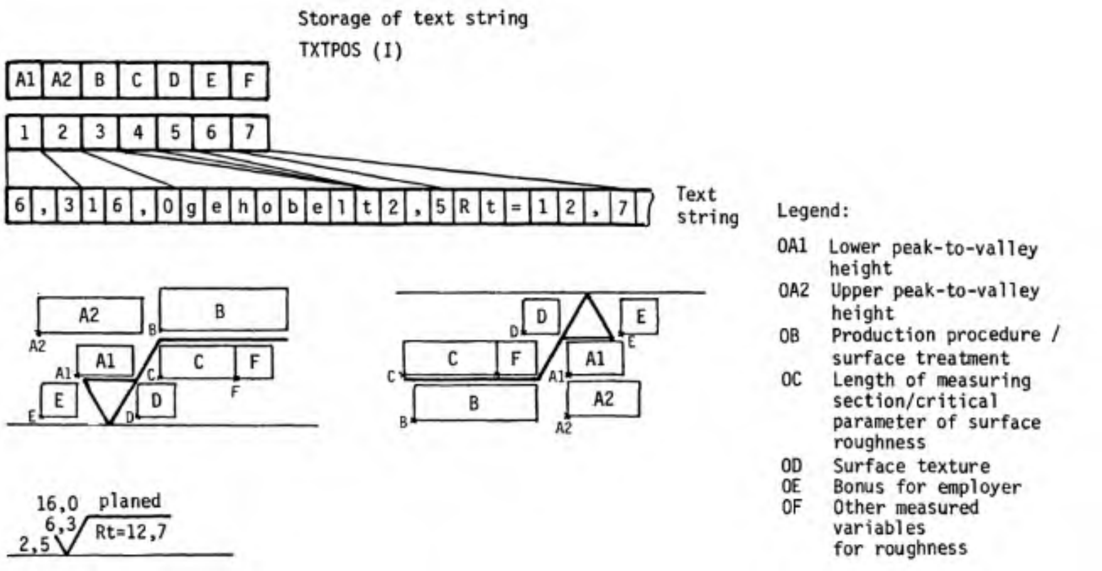


Figure1.2 Structure of surface finish data based on ISO 1302.

1.2.4 Model-Based Design for an industrial automation system.

Model-Based Design (MBD) tools form an integral part of designing an industrial automation system, streamlining the development process and facilitating effective collaboration among various stakeholders. MBD is a mathematical and visual method of managing complex systems, which involves the use of models as an integral part of the design process for systems and components. In the realm of industrial automation, these tools enable the simulation, visualization, and analysis of systems behavior and performance prior to the physical implementation.

At the core of MBD is the idea of a "model," a digital representation of the system being designed, serving as the central hub around which the design process revolves. A model could represent anything from an entire production line to a single PLC program. Software such as MATLAB/Simulink, EPLAN, and LabVIEW, among others, are used to create these models, leveraging a range of graphical programming languages and simulation capabilities.

These models can then be used to simulate the system's behavior and performance, allowing designers to assess the system's efficiency, identify potential problems, and make necessary adjustments before physically implementing the system. In this way, MBD facilitates risk-free virtual experimentation, reducing the likelihood of expensive and time-consuming revisions during the physical implementation phase.

A major advantage of MBD is its inherent ability to automatically generate code for PLCs or microcontrollers based on these models. This process, known as code generation, transforms high-level system models into low-level code that can be executed on specific hardware. This significantly accelerates the programming process, enhances consistency between the model and the final implemented system, and minimizes the potential for manual coding errors. For instance, with MATLAB/Simulink, control algorithms can be developed and validated in a simulation environment, and the Simulink PLC Coder can then be used to automatically generate IEC 61131-3 compliant code that can be deployed on a PLC.

Another aspect of MBD is its role in fostering effective communication and collaboration among stakeholders. By providing a clear visual representation of the system, MBD allows stakeholders from various disciplines to better understand the system and its behavior, ensuring that everyone is on the same page and making the design process more efficient. This is particularly important in the context of industrial automation, where a system may involve a variety of disciplines, including mechanical engineering, electrical engineering, control engineering, and operations management.

MBD tools also enable the design of fault-tolerant and safety-critical systems. For instance, fault detection, isolation, and recovery (FDIR) strategies can be modeled and simulated to ensure the safety and reliability of the system.

In addition, MBD facilitates the documentation of the design process, which is essential for quality assurance, troubleshooting, and future system improvements. Since the model serves as a comprehensive representation of the system, it can act as a valuable source of information for system documentation.

MBD tools are powerful assets in the design of industrial automation systems, enabling efficient system design, rigorous simulation, automatic code generation, and effective communication among stakeholders. As the complexity and sophistication of industrial automation systems continue to increase, the role of MBD in their design and implementation is set to become even more significant.

1.2.5 Automated testing and validation process while designing an industrial automation system.

Automating testing and validation in the design of industrial automation systems is a critical component to ensuring the system's reliability and safety. The complex nature of these systems, involving numerous interconnected components, necessitates thorough and efficient testing methodologies. Automated testing involves using software tools and strategies to execute tests and validate the system's behavior against predetermined criteria, thereby streamlining the process and reducing the potential for human error.

Model-Based Design (MBD) tools, such as MATLAB/Simulink, play a key role in automating testing and validation. With MBD, engineers can create a digital model of the automation system and simulate its behavior in various operating conditions. This allows for comprehensive testing of the system's functionality, performance, and safety without the need for physical prototypes, thereby reducing costs and accelerating the development process.

Moreover, MBD tools often provide capabilities for automating the execution of these simulations and for systematically verifying the results against predefined specifications. For example, Simulink Verification and Validation, an extension to MATLAB/Simulink, enables engineers to automate model checking, simulate requirements traceability, and generate reports, thereby streamlining the validation process.

Another aspect of automated testing involves the use of Hardware-In-the-Loop (HIL) and Software-In-the-Loop (SIL) techniques. In HIL testing, the physical hardware of the automation system is integrated with a real-time simulation of the system's behavior, allowing for rigorous testing of the hardware components in a controlled and repeatable environment. On the other hand, SIL testing involves executing the system's software or firmware in a virtual environment, enabling the detection of software-related issues before the physical implementation stage.

Automated testing also encompasses the use of formal verification methods, which involve mathematically proving the correctness of a system with respect to a formal specification. Tools such as model checkers and theorem provers can automate this process, providing rigorous guarantees about the system's behavior. For example, SMT solvers, which are tools for solving satisfiability modulo theories problems, can be used to automatically verify properties of PLC programs.

Automated code review tools are also essential for ensuring the quality and safety of the automation system's software. These tools can automatically check the system's source code against coding standards, best practices, and specific safety criteria, highlighting potential issues and suggesting improvements. This can greatly enhance the efficiency of the code review process and reduce the likelihood of coding errors slipping into the final system.

Furthermore, Continuous Integration (CI) and Continuous Deployment (CD) practices, which involve automatically building and testing the system's software whenever changes are made, are becoming increasingly prevalent in the design of industrial automation systems. Tools like Jenkins, Travis CI, and CircleCI provide capabilities for setting up automated build and test pipelines, enabling early detection of integration issues and enhancing the system's overall quality.

Finally, it is crucial to note that while automated testing can greatly enhance the efficiency and reliability of the testing process, it is not a silver bullet. Manual testing and inspection remain essential for validating certain aspects of the system's behavior, especially those involving complex or unpredictable real-world conditions. Additionally, a high level of expertise is required to effectively design and implement automated testing strategies, and to interpret and act upon the results.

CHAPTER 2. Computer-Aided Design (CAD) as crucial process for designing an automation system.

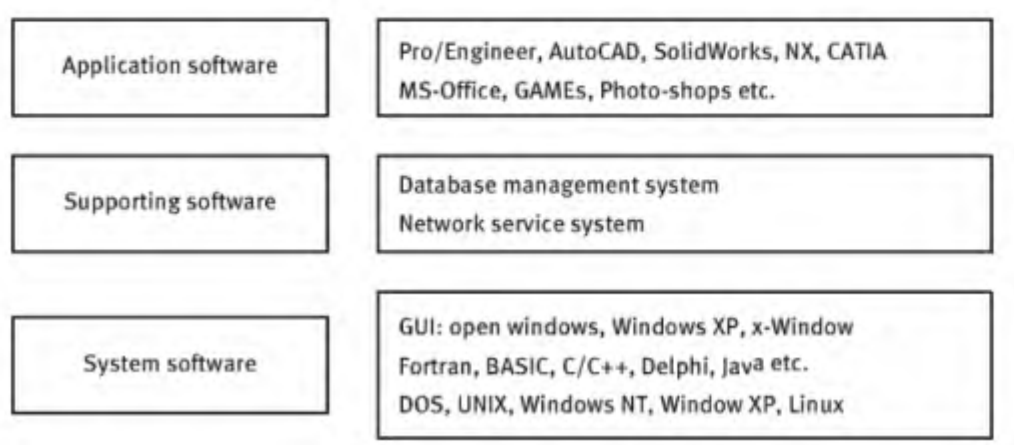


Figure 1.3 Software configuration of CAD systems.

Computer-Aided Design (CAD) plays a pivotal role in the design of industrial automation systems. CAD software is used to create precise 3D models of the automation system, allowing engineers to visualize the system, its components, and its operation in a simulated environment. Using CAD, they can design and analyze individual components as well as the overall system layout.

At the beginning of the CAD process, engineers input the specifications and constraints of the automation system into the software. The software then allows them to build a digital representation of the system, creating each component in 3D space. These components could include mechanical structures, robotic arms, conveyors, sensors, actuators, and more.

Once these individual components have been designed, they can be assembled virtually in the CAD software to form the complete system. Here, engineers can check the system for any potential design errors or interferences between components. They can also analyze the system's performance, including its motion dynamics and strength under various loads, using built-in simulation tools.

As changes are made to the design, the CAD software automatically updates the 3D model, allowing engineers to instantly see the impact of their design decisions. Furthermore, the software can generate detailed 2D drawings, which are essential for manufacturing the components.

In addition to this, CAD can be integrated with other software systems used in the design process. For example, it can be combined with Computer-Aided Manufacturing (CAM) software to streamline the process of turning the CAD designs into physical components. It can also be linked with Programmable Logic Controller (PLC) programming software to aid in the design of the control system.

Here is a list of commonly used CAD software as an example: **AutoCad, SolidWorks, Inventor, CATIA, PTC Creo**.

The subject of Computer-Aided Design (CAD) in designing industrial automation systems is an expansive and profound one. CAD refers to the use of computer software to aid in the creation, modification, analysis, or optimization of a design. In the context of industrial automation system design, CAD can be used to create a detailed 3D model of the system, simulate its behavior under different operating conditions, and optimize the design for performance, cost, or other criteria.

The history of CAD dates back to the 1960s when the first CAD programs were developed. At first, CAD software was used for 2D drafting and only available on large, expensive mainframe computers. However, with the advancement of technology, CAD software has evolved to include 3D modeling and is now accessible on personal computers. Notable milestones in the history of CAD include the development of Sketchpad by Ivan Sutherland in 1963, the launch of AutoCAD by Autodesk in 1982, and the introduction of parametric modeling by PTC with their Pro/ENGINEER software in 1988.

An example of the use of CAD in industrial automation system design could be the design of a robotic assembly line in a manufacturing plant. The CAD software can be used to create a 3D model of the robots, conveyor belts, workstations, and other components of the assembly line. The software can also be used to simulate the movement of the robots and the flow of products on the conveyor belts, helping the designers to identify and resolve any potential issues before the system is physically built.

The advantages of using CAD in designing industrial automation systems are numerous. CAD allows for more precise and detailed designs compared to manual drafting. It also enables designers to easily modify the design and see the impact of the changes immediately. CAD software often comes with tools for automated design analysis, such as finite element analysis and computational fluid dynamics, which can provide valuable insights into the performance of the system. Additionally, CAD models can be directly used for Computer-Aided Manufacturing (CAM), reducing the time and effort required to transition from design to production.

On the other hand, there are also disadvantages associated with the use of CAD. The software can be expensive to purchase and maintain, and it requires a significant amount of training to use effectively. There is also the risk of becoming overly reliant on the software, leading to a lack of understanding of the fundamental principles of design.

Moreover, while CAD software can simulate many aspects of the system's behavior, it cannot perfectly replicate the real-world conditions the system will be subjected to, which can lead to discrepancies between the simulated and actual performance.

Despite these challenges, the advantages of using CAD in designing industrial automation systems far outweigh the disadvantages. The ability to create detailed 3D models, simulate the system's behavior, and optimize the design, coupled with the seamless transition to CAM, makes CAD an invaluable tool in the field of industrial automation system design. As technology continues to advance, we can expect CAD software to become even more powerful and versatile, further enhancing its role in this field.

As an example of CAD modelling I’d like to show you one of the methods used in geometric modelling systems in CAD called Wireframe.

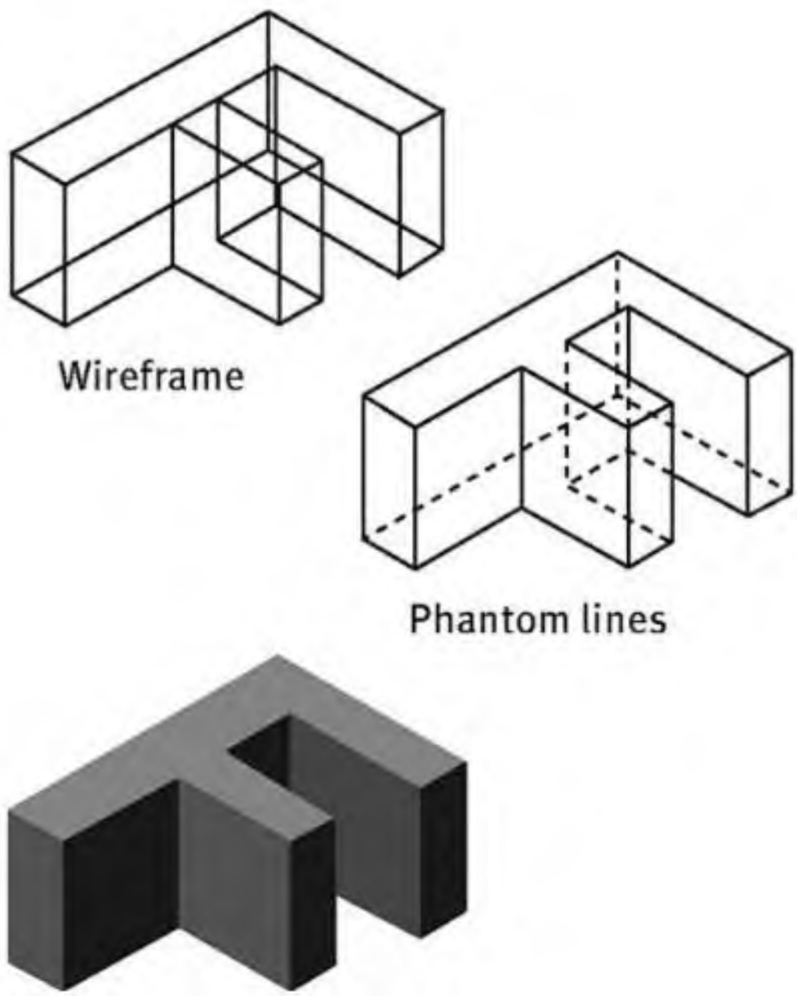


Figure 1.4 A wireframe model.

A wireframe model represents the shape of a solid object with its characteristic lines and points. They are used to define complex solid objects. The designer makes a wireframe model of a solid object, and then the CAD operator reconstructs the object, including detailed analysis. Wireframe models require less memory space and CPU capacity.

A wireframe model is created by specifying each edge of the physical object where two mathematically continuous smooth surfaces meet, or by connecting an object's constituent vertices using straight lines or curves. The straight lines and arcs are normal curves. Here, three special curves are expressed in matrix mode.

Curves

Hermilton and Furguson curve:

$$r(u) = \begin{bmatrix} 1 & u & u^2 & u^3 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -3 & 3 & 2 & -1 \\ 2 & -2 & 1 & 1 \end{bmatrix} \begin{bmatrix} r(0) \\ r(1) \\ r'(0) \\ r'(1) \end{bmatrix}$$

Where $0 \le u \le 1$

Bezier cruve:

$$r(u) = \begin{bmatrix} 1 & u & u^2 & u^3 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ -3 & 3 & 0 & 0 \\ 3 & -6 & 3 & 0 \\ -1 & 3 & -3 & 1 \end{bmatrix} \begin{bmatrix} V_0 \\ V_1 \\ V_2 \\ V_3 \end{bmatrix}$$

Where $0 \le u \le 1$

B-spline curve:

$$r_i(u) = \begin{bmatrix} 1 & u & u^2 & u^3 \end{bmatrix} \frac{1}{6} \begin{bmatrix} 1 & 4 & 1 & 0 \\ -3 & 0 & 3 & 0 \\ 3 & -6 & 3 & 0 \\ -1 & 3 & -3 & 1 \end{bmatrix} \begin{bmatrix} V_i \\ V_{i+1} \\ V_{i+2} \\ V_{i+3} \end{bmatrix}$$

Where $0 \le u \le 1$

2.1AutoCAD as one of the best tools for computer-aided design and how can it be used when designing an industrial robotic manufacturing line.

AutoCAD, a prominent Computer-Aided Design (CAD) software developed by Autodesk, plays an instrumental role in designing a robotic assembly line in a manufacturing plant. The process begins by setting the project parameters and constraints which include factors like the available workspace, the type of product to be manufactured, the required production rate, safety considerations, and budget. With these parameters set, designers can begin drafting the layout of the assembly line using AutoCAD's 2D drafting features. This layout typically includes the position and orientation of each robot, the path of the conveyor belts, the locations of workstations, and other relevant elements.

Once the 2D layout has been established, designers can move on to creating 3D models of the individual components of the assembly line. AutoCAD's 3D modeling features allow designers to construct detailed and accurate models of the robots, conveyor belts, workstations, and other elements. These models can then be assembled into a complete 3D model of the assembly line, providing a comprehensive visual representation of the system.

Next, the 3D model can be used for simulation and analysis. For instance, designers can simulate the movement of the robots and the flow of products on the conveyor belts to identify any potential issues. This could include collisions between robots, inefficiencies in the product flow, or areas where the robots cannot reach. These simulations can provide invaluable insights into the performance of the assembly line and help designers to optimize the system.

AutoCAD's features also extend to drafting the necessary electrical wiring and control schematics for the robotic assembly line. This is an important step in ensuring all electronic components and control systems of the robotic line function in harmony. These comprehensive schematics are critical for proper installation, operation, and maintenance of the system.

Furthermore, AutoCAD's collaboration features can be highly beneficial during the design process. For example, the software allows multiple designers to work on the same project simultaneously, facilitating teamwork and enhancing productivity. Additionally, the 3D model of the assembly line can be shared with stakeholders, providing them with a clear understanding of the proposed system and allowing them to provide feedback.

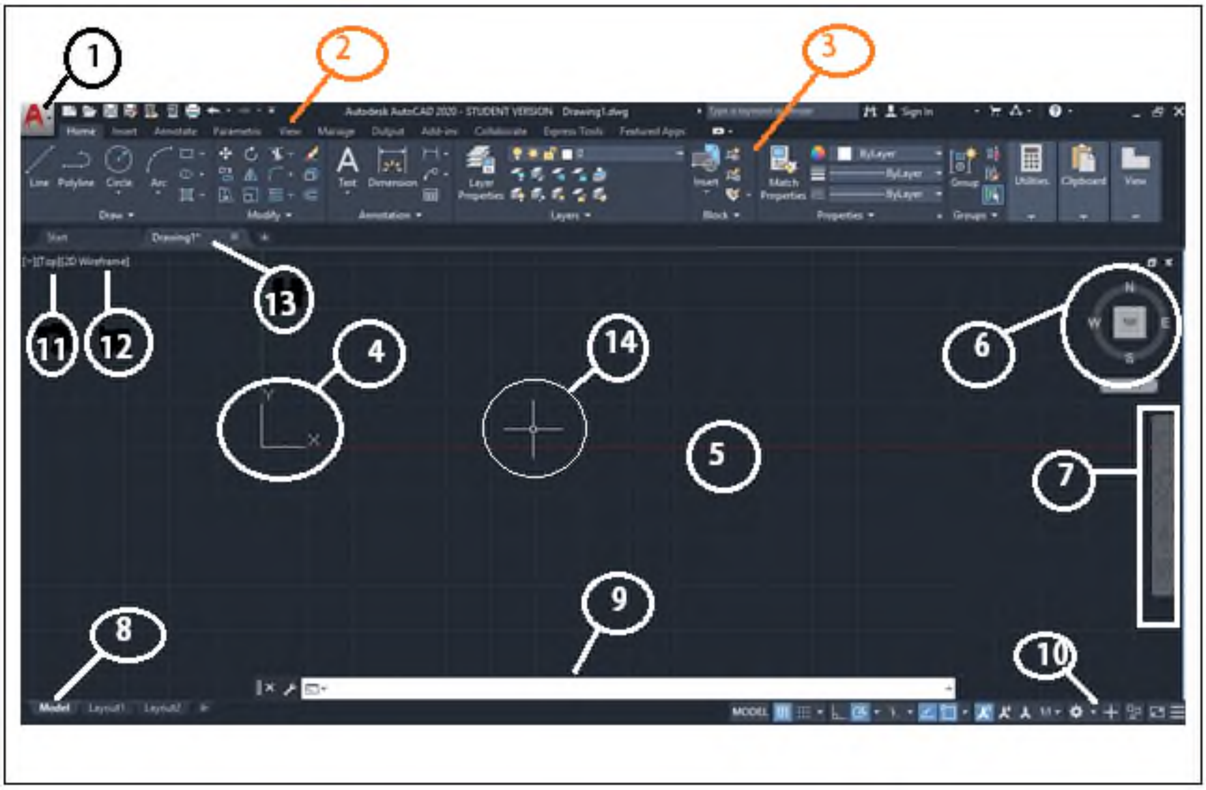


Figure 1.5 AutoCAD interface.

The parts of the AutoCAD 2020 display are displayed in the numbers.

The name of the numbered parts are listed below:

1. Application menu/button.

It is present at the upper-left corner of the workspace. To close the application menu, we can click anywhere outside the application button or window.

The button is shown in the image marked as number 1.

2. Quick Access Toolbar

The Quick Access Toolbar is located at the top of the application window and right of the application menu. It consists of the set of frequently used commands. We can add and remove the commands according to the requirements.

3. Ribbon Panel

It provides access to the dialog box related to that panel. If the ribbon panel disappears, then we need to enter the 'ribbon' on the command line to display the ribbon panel on the workspace.

4. User Coordinate System (UCS)

The UCS is the active coordinate system that represents the XY plane in 2D and XYZ planes in 3D. It acts as the direction in the X, Y, and Z-axis for drawing and modeling. We can control the origin and orientation of the UCS to make drawings according to the specific points and coordinates. We can also work with drawing aids such as Grid and the ortho mode.

The 2D UCS is shown in the given image:

The 3D UCS is shown in the given image:

5. Model Space / Work Space / Drawing Window

It is defined as the area to create 2D and 3D drawings, models, and objects. We can create using different commands according to the requirements.

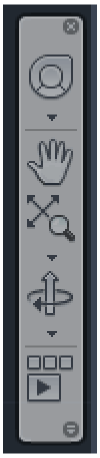
6. View Cube

The View Cube is termed as the navigation tool that is displayed when we are working on a 2D or 3D model space. We can switch between the isometric view and the standard view of our drawings or model.

7. Navigation Bar

The Navigation Bar is used to access the navigation tools. It is a user interface element, where we can access both unified (common tools) and product-specific tools (unique product tools).

The navigation bar is shown in the below image:



8. Model Layout Tab

A layout is defined as a 2D working environment for creating the drawing sheets.

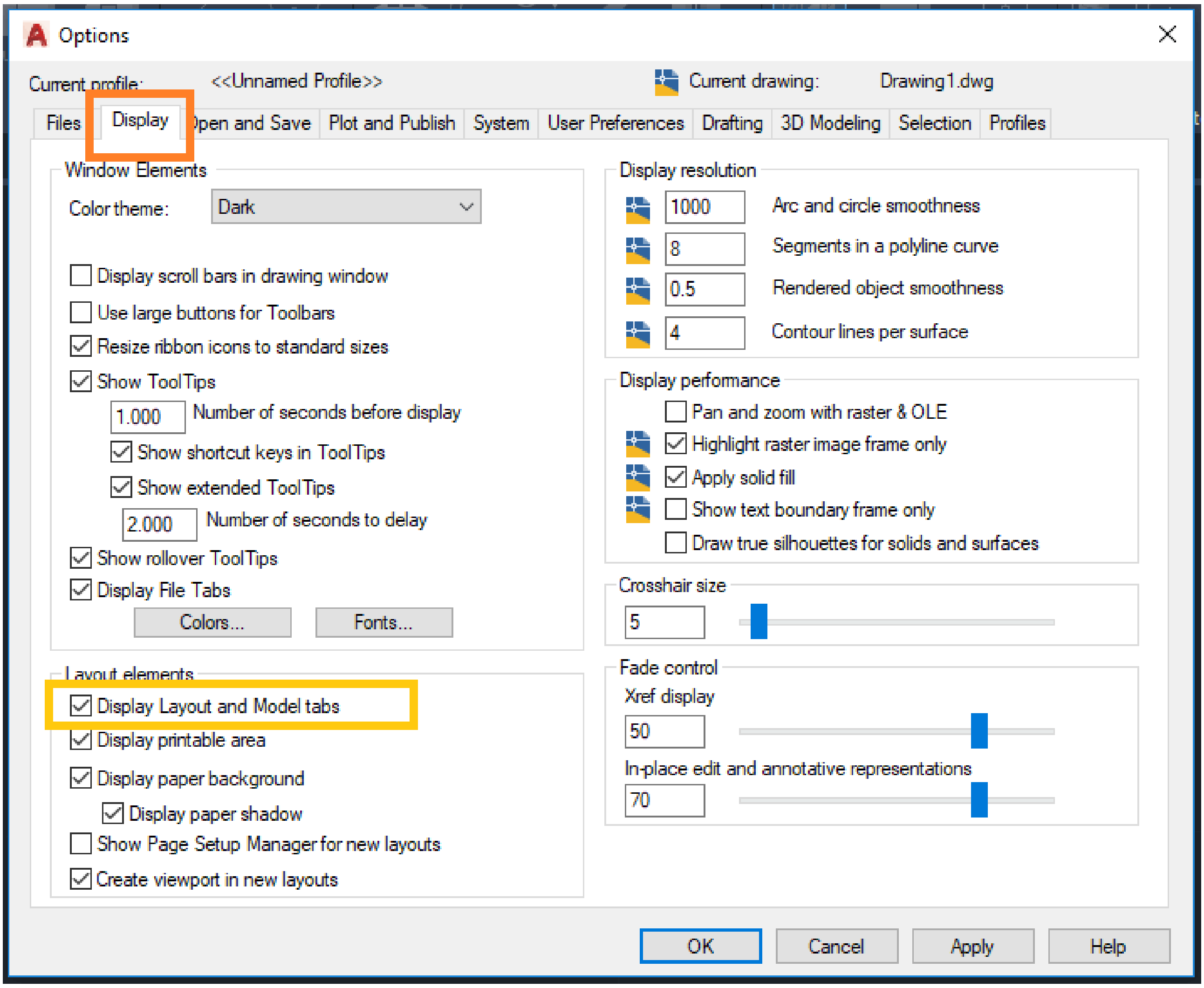
The model tab is the screen where we create the 2D and 3D drawings.

If the model and layout tab is not visible, then follow the steps given below:

a. We need to click on the AutoCAD option displayed on the top left corner of the screen, shown as

b. Click on the 'Options' button at the bottom.

c. Then on the 'display' option on the top, select the option 'Display Layout and Model tabs' and then click 'Ok' as shown below:



9. Command Line Window (shortcut key- ctrl+9)

The command line window is used to write the commands. We need to press 'Enter' after typing any particular command. The AutoCAD also displays the steps after each command on the Command line. We need to select the option and press 'Enter' after each step.

10. Status Bar

The status bar displays the drawing tools that affect the drawing environment. It provides quick access to most of the commonly used drawing tools. It includes options such as ISODRAFT, ORTHOMODE, AUTOSNAP, OSNAP, etc.

11. View Control

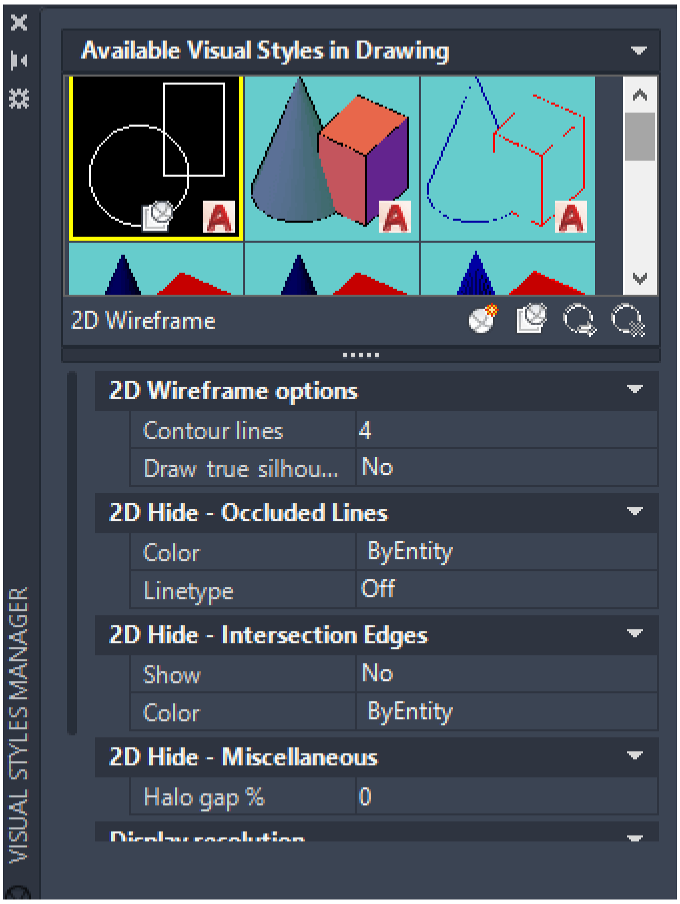
The View Control is displayed on the left corner of the viewport. It provides a suitable way to change views, visual settings, and styles. It includes options such as top, bottom, left, right, etc.

12. Visual Style Control

The Visual Style Control is the customized setting that controls the structure view of the models or 3D drawings created on the Viewport. The options include 2D Wireframe, Realistic, Shaded, Conceptual, Hidden, etc.

We can also enter the visualstyles on the command line to display the options of the visual style control.

After entering on the command line, the box will appear displaying clear information with figures for each option. It is shown in the below image:



13. File Tab

The file tab consists of the current drawing files opened on the screen.

We can also type FILETAB command in the command line to turn on turn on the file tabs.

To close the file tab, we can type FILETABCLOSE command in the command line window.

We can click on the '+' sign to add a new tab on the File Tab.

14. Mouse Cursor

The cursor used to draw figures, etc. is known as the mouse cursor.

Finally, once the design of the assembly line has been finalized, AutoCAD can facilitate the transition to manufacturing. The software can generate detailed assembly instructions, bill of materials, and other documents necessary for construction. In some cases, the 3D model can even be used directly for Computer-Aided Manufacturing (CAM), reducing the time and effort required to transition from design to production.

In conclusion, AutoCAD plays an essential role in designing a robotic assembly line in a manufacturing plant. Its robust 2D and 3D drafting features, simulation and analysis capabilities, collaboration tools, and seamless transition to manufacturing make it an invaluable tool in this process. It allows designers to create a detailed and accurate design of the assembly line, optimize the system for performance and efficiency, effectively communicate the design to stakeholders, and smoothly transition to manufacturing. As such, AutoCAD continues to be a cornerstone of modern industrial automation system design.

2.2 SolidWorks as another useful tool implemented in designing of industrial automation systems.

SolidWorks, a solid modeling computer-aided design (CAD) software, is often used to visualize, simulate, and optimize assembly line designs before they're implemented in a real-world manufacturing environment.

Conceptualization and Design

The first step involves creating an initial conceptual design of the assembly line. Using SolidWorks, engineers can sketch out the general layout of the manufacturing floor, including the placement of each machine, conveyor belt, workstation, and other components. The software's easy-to-use interface and powerful sketching tools make it simple to make changes and adjustments as needed during this initial design phase.

Detailed 3D Modeling

After the general layout is finalized, the next step is to create detailed 3D models of each component. SolidWorks excels at this with its robust 3D modeling capabilities. Each machine, conveyor system, robot, and other parts of the assembly line can be modeled in high detail, including all the nuts, bolts, and other minor components. These 3D models can then be assembled together in the software to create a complete, detailed model of the assembly line. This helps engineers to visually inspect the entire assembly line, understand how different components fit together, and identify any potential issues.

Simulation and Optimization

Once the detailed 3D model is complete, SolidWorks can be used to simulate the operation of the assembly line. The software's motion study capability allows engineers to animate the movement of the machines, simulate the flow of products on the conveyor belts, and identify any potential collisions or bottlenecks. This is a critical step in the design process because it allows potential problems to be identified and fixed before the assembly line is actually built.

Collaboration and Presentation

SolidWorks also features strong collaboration tools that allow multiple engineers to work on the same project at once. Furthermore, the software can generate photorealistic renderings and animations of the assembly line, which can be used to present the design to stakeholders, clients, or other members of the project team.

Transition to Manufacturing

Finally, SolidWorks can help to facilitate the transition from design to manufacturing. The software can generate a detailed bill of materials (BOM) that lists all the parts and components needed to construct the assembly line. Additionally, SolidWorks can output detailed drawings and assembly instructions for use in the manufacturing process.

In conclusion, SolidWorks is a powerful tool for designing automated assembly lines in a manufacturing plant. Its robust 3D modeling capabilities, simulation tools, and ease of use make it an ideal choice for this complex task. By allowing engineers to visualize, simulate, and optimize their designs before they're built, SolidWorks helps to reduce errors, save time, and ensure that the final assembly line operates as efficiently as possible.

2.3 Design methodology used for industrial automation system.

Designing an industrial automation system is a complex task that requires a systematic approach to ensure that the final system meets the desired requirements and performance criteria. A common design methodology involves the following stages:

The first step in the design methodology is understanding the requirements of the system. This involves identifying the tasks that the system needs to perform, the conditions under which it needs to operate, and the performance criteria it needs to meet. This step often involves discussions with stakeholders, including process owners, operators, and maintenance personnel.

Once the requirements have been defined, the next step is creating a conceptual design of the system. This involves identifying the main components of the system, such as sensors, actuators, controllers, and communication networks, and outlining how they will interact to perform the required tasks. This stage often involves creating flowcharts, block diagrams, or other types of schematic representations.

After the conceptual design is complete, the next step is creating a detailed design of the system. This involves specifying the exact types and models of components to be used, defining the control algorithms for the controllers, and designing the layout of the system, including the placement of components and routing of cables. This stage often involves creating detailed drawings, circuit diagrams, and software code.

Before the system is built, it's important to simulate and test the design to identify any potential issues or areas for improvement. This can involve using software to simulate the behavior of the system under different conditions, performing static or dynamic analysis of the system, and testing the control algorithms in a virtual or physical testbed.

Once the design has been tested and validated, the next step is implementing the system. This involves purchasing and installing the components, wiring up the system, programming the controllers, and commissioning the system. Commissioning involves testing the system under real-world conditions to ensure that it operates correctly and meets the performance criteria.

After the system is commissioned, it moves into the operation and maintenance phase. This involves monitoring the system to ensure that it continues to operate correctly, performing regular maintenance to prevent failures, and troubleshooting and repairing any issues that arise.

The final step in the design methodology is evaluating the performance of the system and iterating on the design as needed. This involves collecting and analyzing data on the system's performance, identifying areas where the system can be improved, and making modifications or updates to the design as needed.

This methodology provides a systematic approach to designing an industrial automation system. By following these steps, designers can ensure that the final system meets the desired requirements and performance criteria, operates reliably and efficiently, and can be easily maintained and updated over time.

2.4 Industrial automation system architecture

Industrial automation systems that utilize Programmable Logic Controllers (PLCs) often follow a layered architecture. This architecture allows the various components and systems to interact effectively, resulting in a cohesive, efficient automation process. Here's a typical architecture of such a system:

1. Field Level (Process): The lowest layer, this consists of the physical devices that interact directly with the environment. It includes sensors and actuators. Sensors are used to gather data such as temperature, pressure, humidity, proximity, and so on. This data is then used to monitor the state of the environment. Actuators, on the other hand, are used to interact with the environment, based on the control signals they receive. Examples include motors, pumps, heaters, and valves.

2. Control Level: This is where the PLCs reside. PLCs are industrial-grade computers used for automation of electromechanical processes. They receive inputs from the sensors, process these inputs based on the programmed logic, and generate outputs to control the actuators. The PLC can control the system in real-time, responding to sensor inputs and adjusting control outputs as needed.

3. Supervisory Level: This level includes supervisory computers or systems such as SCADA (Supervisory Control And Data Acquisition) systems or Human Machine Interface (HMI) devices. They provide an interface for human operators to monitor and control the process. They can display real-time data, historical trends, alarms and warnings, and offer controls to override automatic control when necessary.

4. Information Level (Network Level): This is the highest level in the architecture and is responsible for managing and distributing information across the entire plant or facility. It involves computers and servers that may be running Manufacturing Execution Systems (MES), or Enterprise Resource Planning (ERP) systems. These systems help in planning, tracking, analyzing, and managing operations in the manufacturing plant, often integrating data from multiple sources and providing a holistic view of the operations.

Communication: Crucial to this architecture is the communication between the various levels. This communication is facilitated by industrial networks that could either be wired (like Ethernet) or wireless. These networks must be robust and reliable due to the critical nature of the information being transmitted.

This layered architecture provides an organized framework for the operation of industrial automation systems. It ensures that each component performs its designated function while seamlessly interacting with other components for a harmonious operation. The architecture is flexible and scalable, allowing for easy expansion or modification as the needs of the industrial process change.

2.5 Hardware and software implementation in industrial automation systems with PLCs.

The implementation of an automation system is a multifaceted process that includes both hardware and software aspects. These components work together to form a complete system that can effectively automate a variety of industrial processes.

Hardware Implementation:

The hardware of an industrial automation system typically includes a range of components, including PLCs, microcontrollers, sensors, actuators, human-machine interfaces (HMIs), and communication networks.

2.2.5 Example of PLC controller programming.

Develop PLC Programming Examples on Industrial Automation according to the logic given below,

A Saw, Fan and oil pump all go ON when a start button is pressed.

If the saw has operated less than 20s, the oil pump should go off when the saw is turned off and the fan is to run for an additional 5s after the shutdown of the saw.

If the saw has operated for more than 20s, the fan should remain on until reset by a separate fan reset button and the oil pump should remain on for an additional 10 s after the saw is turned off.

Write a program that will implement this process.

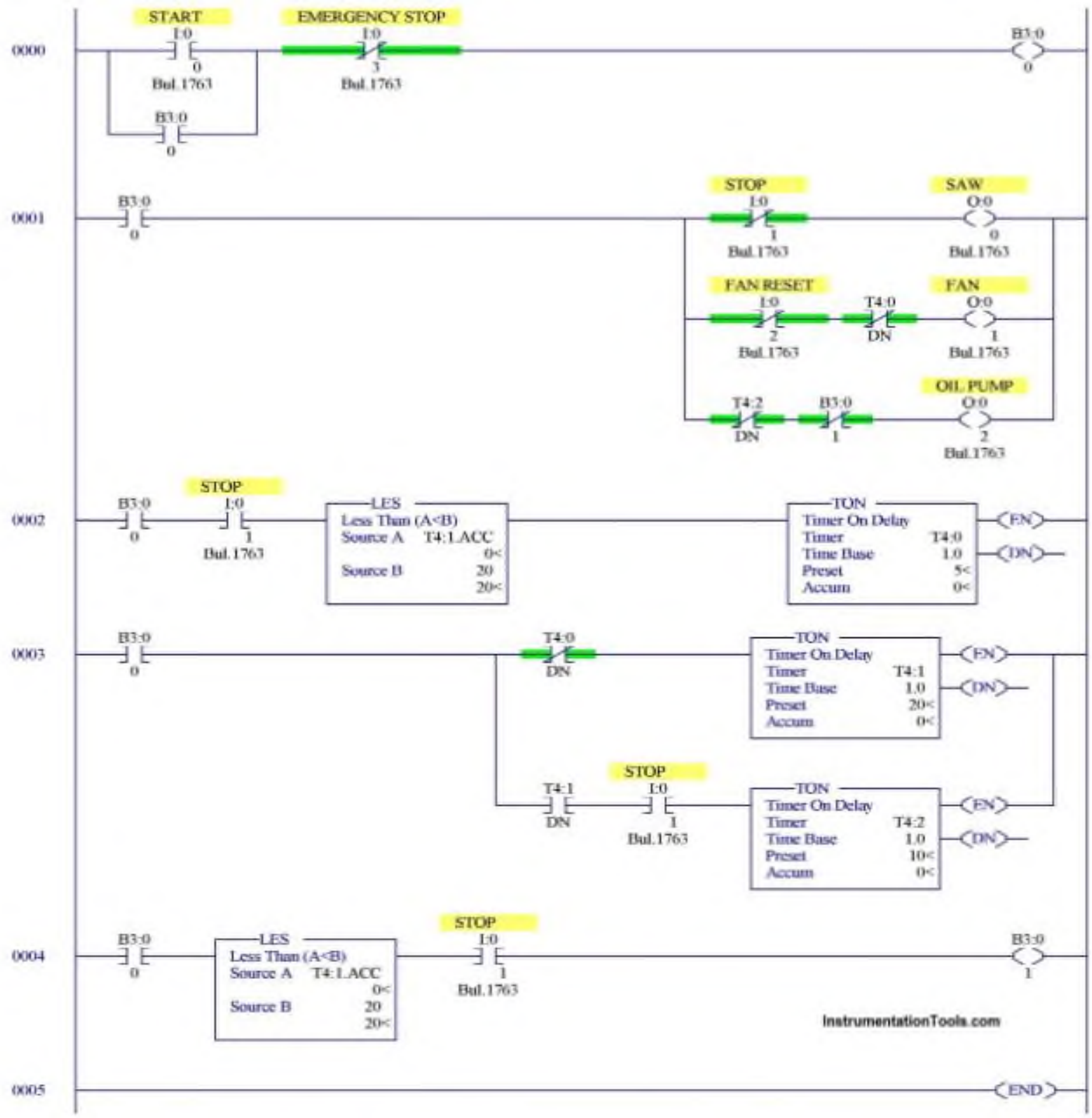


Figure 1.6 Logical scheme of a saw and oil pump operation

<u>Program Description:</u>

Rung 0000:

Start/Emergency Stop PB latched with memory B3:0/0.

Rung 0001:

B3:0/0 enabled to turn on Saw (O: 0/0), Fan (O: 0/1) and Oil pump (O:0/2).

Normally closed contact of Stop switch is in series Saw output to turn off.

Fan reset switch and Timer T4:0 is connected to turn off Fan when condition meets.

Timer T4:2 done a bit and memory bit is to turn off the oil pump.

Rung 0002:

When the stop is pressed, according to the logic mentioned in point 2, Fan output (O: 0/2) needs to turn off after 5s.

Comparator block restricts the timer T4:0 to run after the 20s of Saw operation.

Rung 0003:

Timer T4:1 runs when the start is pressed. When the stop is pressed at any point after the 20s, Saw output will go off.

After 10s, the oil pump will go off. This operation is done by Timer T4:2. Timer T4:0 done bit is used to restrict the Timer T4:1 operation when T4:0 is ON.

Rung 0004:

Less than a comparator block is used to perform the logic mentioned in point 2, to turn off Fan when saw output operation was less than 20s.

Program Output:

Now we see the simulation of above ladder logic for different conditions as mentioned below.

PLCs and Microcontrollers: These are the brains of the system, where the control logic is executed. The choice between a PLC or a microcontroller depends on the application. PLCs are typically used in larger systems due to their robustness, ease of programming, and built-in functionalities. Microcontrollers, on the other hand, are often used in smaller, more specialized systems due to their lower cost and greater flexibility.

Sensors and Actuators: Sensors gather information about the state of the system, such as temperature, pressure, position, and speed. Actuators perform actions based on commands from the controller, such as moving a part, opening or closing a valve, or adjusting a temperature. The choice and placement of sensors and actuators are crucial to the functionality of the system.

Communication Networks: These networks facilitate communication between different parts of the system. They may be wired (like Ethernet) or wireless, and they must be reliable and robust due to the critical nature of the information being transmitted.

Human-Machine Interfaces (HMIs): These provide a way for human operators to interact with the system. They typically include displays for monitoring system status, as well as controls for manual intervention when necessary.

Software Implementation:

The software side of an automation system is primarily concerned with creating the control logic that dictates how the system responds to various inputs. This often involves the following steps:

Program Design: This involves outlining the logic that the controller will follow. This is often based on a careful analysis of the system's requirements and typically involves creating flowcharts or state diagrams to visualize the control process.

Coding: Once the control logic has been defined, it is translated into code that can be executed by the PLC or microcontroller. Most PLCs are programmed using languages defined by IEC 61131-3 standard, which include Ladder Diagram (LD), Structured Text

(ST), Function Block Diagram (FBD), Instruction List (IL), and Sequential Function Chart (SFC). Microcontrollers, on the other hand, are typically programmed in languages like C or C++.

Testing and Debugging: After the code has been written, it must be thoroughly tested to ensure it behaves as expected. This involves simulating various inputs and checking that the output is correct. Any issues that are discovered are then corrected through debugging.

Commissioning: Once the code has been tested and validated, it is loaded onto the PLC or microcontroller and the system is put into operation. Further testing is often performed during this stage to ensure the system behaves correctly in a real-world environment.

Both the hardware and software implementation processes are iterative and may need to be repeated multiple times as issues are discovered and corrected. The end result is a robust and reliable automation system capable of efficiently controlling a wide range of industrial processes.

2.6 Testing and validation of industrial automation system.

Testing and validation are critical phases in the development of an industrial automation system. They ensure the system performs as intended, is reliable and safe, and complies with all relevant industry standards and regulations. Here's a broad overview of these processes:

Unit Testing: This is the first level of testing where each component of the system (such as sensors, actuators, PLCs, or HMIs) is tested independently to ensure that it functions correctly. For software components, unit testing typically involves running the software with a variety of inputs and checking the outputs against expected results.

After each component has been tested individually, they are then integrated and tested as a system. This allows for the detection of issues that only appear when components interact with one another. It could involve testing the communication between devices, or the operation of a control loop.

System Testing: In this phase, the entire system is tested as a whole. This includes not just the individual components and their interactions, but also the system's interaction with the external environment. Tests are performed under various conditions to validate that the system performs as expected in all intended scenarios.

Acceptance Testing: This type of testing is performed to determine whether the system meets all of the specified requirements. It includes both functional requirements (what the system should do) and non-functional requirements (how well the system should do it). Acceptance testing often involves the system's end-users or clients to ensure the system is acceptable for its intended use.

Simulation Testing: This is particularly useful when testing complex automation systems. The real-world conditions are replicated in a controlled simulation environment where various parameters can be adjusted and monitored. This allows for stress testing the system under different conditions and validating the system behavior.

While testing aims to find defects in the system, **validation** is the process of evaluating the system during or at the end of the development process to determine whether it satisfies the specified requirements. Validation ensures that 'you built the right system'.

Safety and Regulatory Compliance Testing: In many industries, there are specific safety standards and regulations that automation systems must comply with. This type of testing ensures the system is safe to operate and meets all regulatory requirements.

All these stages are crucial to ensure that the automation system is reliable, efficient, and safe. Regular testing and validation help in early detection of problems, which reduces the risk of failure when the system is operational, ultimately leading to lower costs and better system performance.

CONCLUSION

The application of advanced technologies like Computer-Aided Design (CAD), Computer-Aided Manufacturing (CAM), simulation software, and others, significantly enhance the design process by improving efficiency, accuracy, and reducing time-to-market. Automation allows for more complex and intricate designs to be developed and tested in a virtual environment before physical implementation, thereby reducing the risk of errors and rework. It also enables designers to explore more design options and optimize the design based on various parameters. Additionally, automation allows for better documentation and easier modification of designs, enhancing traceability and adaptability to changes. As industrial automation systems become more complex and interconnected, automating the design process becomes increasingly important to manage this complexity and ensure the systems are designed optimally for their intended purpose. This not only leads to more efficient and effective automation systems but also paves the way for innovation and development of next-generation industrial systems. Thus, the automation of the design process is not merely an advantageous move, but rather a necessary stride towards a more efficient, innovative, and productive industrial future.

REFERENCES

1. Computer Aided Design (CAD) and Automation. Sunny K. Tuladhar 2019-10-22.
2. Computer Aided Design and Design Automation ByWai-Kai Chen, 2009.
3. Computer-Aided Design and Manufacturing: Methods and Tools U.Rembold, R.Dillmann, 2012.
4. Computer Aided Design in Control and Engineering Systems P. Martin Larsen, N. E Hansen, 2014.
5. Computer Aided Design in Power Engineering by Zlatan Stojkovic, 2012.
6. Computer Aided Design: Fundamentals and System Architectures J. Encarnacao, E. G. Schlechtendahl · 2012.
7. Principles of Computer-aided Design, Philip Steadman, Joe Rooney, 1987.
8. Operational Methods in Computer Aided Design, Cornelius T. Leones, 2000.