DAS Departamento de Automação e Sistemas
CTC Centro Tecnológico
UFSC Universidade Federal de Santa Catarina

The virtualization at the Commissioning Station in a Hybrid Learning Factory

Relatório submetido à Universidade Federal de Santa Catarina

como requisito para a aprovação na disciplina

DAS 5511: Projeto de Fim de Curso

Filipe Macedo de Carvalho

Erlangen, Janeiro de 2014

The virtualization at the Commissioning Station in a Hybrid Learning Factory

Filipe Macedo de Carvalho

Esta monografia foi julgada no contexto da disciplina
DAS5511: Projeto de Fim de Curso
e aprovada na sua forma final pelo
Curso de Engenharia de Controle e Automação

Prof. ***Marcelo Ricardo Stemmer***

____
Assinatura do Orientador

Banca Examinadora:

Dr.-Ing. Hans-Peter Kagerer / Siemens AG
Orientador na Empresa

Prof. Marcelo Ricardo Stemmer
Orientador no Curso

Prof. <nome do professor avaliador>
Avaliador

<nome aluno 1>
<nome aluno 2>
Debatedores

Resumo Estendido

É um fato que produção e processos de automação estão se tornando cada vez mais inseparáveis em quase todos os ambientes de manufatura. Na competição atual das empresas procura-se produzir uma gama de produtos com pequenos ciclos de inovação, que sejam curtos, contínuos e rapidamente implementados. Além disto, o incremento da complexidade nos processos industriais requer um tempo de resposta cada vez mais curto, seja devido a variações na demanda, sazonalidades, abastecimento ou tendências do mercado. Portanto os pontos críticos referentes á implementação de um novo processo precisam ser mensurados no fator tempo e custo.

Uma forma de abordar esta problemática é através da virtualização destes sistemas automatizados, permitindo que tarefas de um ambiente real possam ser simuladas em um ambiente virtual. Assim o fator custo e tempo poderão ser reduzidos, mensurados e simulados enquanto paralelamente o ambiente real está em operação. Com esta abordagem é possível previamente remover erros, simular falhas e novos cenários, bem como melhorar a qualidade do processo e segurança.

Atualmente a Siemens AG é o maior conglomerado de engenharia na Europa e evidentemente se encontra dentro da esfera do problema abordado. Suas principais áreas de atuação são: Indústria, Energia e Medicina, sendo que em Erlangen localiza-se um dos seus principais escritórios ligados ao setor industrial, dos quais se podem destacar os segmentos de Automação Industrial, Drivers, Tecnologia para construção, Soluções para indústria e Mobilidade.

Diante deste desafio a Siemens em Nuremberg construiu uma máquina piloto em conjunto com a Festo[1] para estudar as possibilidades da virtualização, bem como usá-la para outros projetos pertinentes a sua área de atuação. A máquina, chamada Athene, é uma fábrica hibrida de aprendizado que ocupa um espaço enxuto, mas que desempenha todas as funções elementares de um sistema automatizado de produção. Por se tratar de uma máquina hibrida pode assumir

[1] Empresa fornecedora de equipamentos elétricos e pneumáticos para automação industrial: http://www.festo.com

diversos formatos para se adaptar ao processo, permitindo assim uma gama maior de aplicação e estudo.

O foco da Athene é ensinar processos básicos presentes em indústrias farmacêuticas e alimentícias. De modo que aprendizes venham adquirir conhecimento prático em programação, automação e documentação de fábrica por meio da tecnologia de controle de temperatura, nível de enchimento, pressão e fluxo de bombas e válvulas.

Um dos exemplares da Athene está em Erlangen no Setor Industrial I CS EH BB e é chamada de *Backbone Factory*, pois foi customizada para atender melhor as necessidades dos trabalhos desenvolvidos nesta divisão.

Esta monografia examina as possibilidades da virtualização de uma das estações da *Backbone Factory*, onde um modelo tridimensional foi construído utilizando o software *Tecnomatix Plan Simulation* desenvolvido pela Siemens para modelar, simular, analisar, visualizar e aperfeiçoar sistemas e processos de produção como o fluxo de material e logística. O CLP[2] que controla esta estação no ambiente real é simulado através dos programas *S7-PLCSIM* e *WinCC Flexible Runtime*. A conexão do CLP simulado com o ambiente virtual é feita via *OPC-Interface*[3], sendo possível testar o código para esta estação no ambiente virtual criado.

[2] Controlador Lógico Programável

[3] Software criado pela OPC Foundation para interoperabilidade em Automação Industrial

Abstract

Beyond question, production and automation processes are becoming more and more inseparable in all manufacturing environments. It has been required, since the high level of competition among companies nowadays, the production of a variety of products with short, continual and promptly implemented innovation cycles. Besides, the increased complexity of industrial processes postulates a faster way of response for variations such demand, seasonality, supply or in market tendencies.

A way to address this issue would be achievable by the virtualization of those automated systems; it allows the execution of tasks from a real environment in a virtual one. Thereby, the time and cost factors will be reduced, measured and simulated while the operation occurs in the real environment at the same time. Based on this approach, it is possible to remove mistakes previously, simulate failures and new scenarios, as well as to improve the process and security qualities.

Currently, Siemens AG is the largest engineering conglomerate in Europe, and evidently it is within the scope of this addressed problem. Before this challenge, Siemens has built in Nuremberg, together with Festo, a pilot machine to study the potentialities of virtualized processes, as well as to study projects related to its area of expertise.

The machine named Athene is a hybrid learning fabric which occupies a slim room, and even then it performs all the elementary functions of an automated production system. One of Athene's copies has been set at the Industrial Sector I CS EH BB, in Erlangen, and it is called Backbone Factory, since it has been customized to better assist the necessities of projects developed in this division.

This present paper analyses the virtualization possibilities for one of the Backbone Factory stations, where a tridimensional model has been built by using the Tecnomatix Plan Simulation software, which Siemens developed in order to model, simulate, analyze, visualize and improve production systems and processes, such as material flow and logistics. The PLC, which controls the station in a real environment, is simulated by using S7-PLCSIM and WinCC Flexible Runtime programs. The connection between the simulated PLC with the virtual environment is done through

the OPC-Interface, thus it is possible to test the related code to this station directly in a virtual environment.

Contents

List of Figures

List of Tables

Simbology

I CS EH BB - Industry Customer Services Expert House BackBone

PLC - Programmable Logic Controller

VR - Virtual Reality

PLM - Product Lifecycle Management

VDI - Verein Deutscher Ingenieure

WinCC - Windows Control Center

STEP - STandard for the Exchange of Product

HMI – Human Machine Interface

OPC – OLE for Process Control

OLE - Object Linking and Embedding

OPC UA - OPen Connectivity Unified Architecture

XML - Extensible Markup Language

ERP - Enterprise Resource Planning

MES - Manufacturing Execution Systems

DCOM - Distributed Component Object Model

COM - Component Object Model

DA – Data Access

SOAP - Single Object Access Protocol

DP – Default Position

AP – Active Position

MPS - Modular Production System

Chapter 1: Introduction

Based on the competition of international production networks, the pressure to increase the efficiency of production systems has grown significantly. Production and process automation have become more and more inseparable in almost all manufacturing environments and it towards to produce broader ranges of products with shorter innovation cycles.

In addition, the number of technical components and the complexity of industrial arrangements are increasing and it demands a quicker response time in the production. Therefore mistakes and disturbances during a project implementation in automatic systems need to be measure to preview costs and deadlines.

Those requirements can be managed only by using appropriate Digital Factory tools in the context of a product lifecycle, which allows reusing data, supports an effective cooperation between different departments, and provides up-to-date and relevant data to every user who needs it.

Simulating the complete material flow - including all relevant production, storage, and transport activities – has been recognized as a key component of the Digital Factory in the industry and currently widely used and accepted. Cutting inventory and throughput time by 20–60% and enhancing the productivity of existing production facilities by 15–20% can be achieved in real-life projects. [1]

For this reason Virtual Commissioning is able to provide alternatives to nowadays industry problems, since it is capable of building a virtual environment based on the real factory to provide a range of tools to analyze models with stochastic process, to manage simulation experiments, to calculate distribution for sample values, and to determinate optimized system parameters.

In this scenario, Siemens AG developed means to achieve the needs of their automated system by using Plant Simulation, providing all necessary functionality to model, analyze, and maintain large and complex systems in an efficient way.

Based on these accurate modeling capabilities and statistical analysis capabilities, typically an accuracy of at least 99% of the throughput values is achieved with Plant Simulation models in real-life projects depending on the level of detail. Considering the price of production equipment, a return on costs investment to introduce simulation is quite often already achieved after the first simulation project.

Visualizing the complete model in the Plant Simulation 3D environment allows an impressive 3D presentation of the system behavior. Log files can be used to observe the simulation in a Virtual Reality (VR) environment. The support of a Siemens PLM Software gives a common look-and-feel and easy access to 3D graphics which were created in other tools like digital product design or 3D factory layout design tools. [1]

To provide all this background for their technicians and engineers, Siemens AG built, together with Festo, a hybrid learning factory with powerful tools, such as PLC and robot programming systems, simulations systems, EPLAN and CAD programs, vision system, data matrix encoding, drive technology and a range of sensors and actuators.

The present paper aims to study the possibilities of Virtual Commissioning in one of the stations of this learning factory using Siemens PLM Software to visualize the complete model in the Plant Simulation 3D environment.

In the following sections, the concepts of factory simulation and hybrid learning factory are explained in the context of this project. Then, the objectives of this work are presented and at the end of this chapter the structure of the work is detailed.

1.1: Characterization of Digital Factory

For some years now, globalized markets have been demanding that the production in a variety of sectors might be able to provide new and individualized products in even shorter cycles. In order to cope commercially with the ever more complex product and production development processes, new approaches and instruments have become necessary, e.g. digital mock-up (DMU) and virtual reality (VR).

In this context the concept of the digital factory supports an early parallelization and also, as far as possible, digital processing of product development and production planning right up to virtual start-up and operation including holistically integrated data management. A digital product development has been routine practiced in many companies for some time now; the focus, at the moment, is on production planning. [2]

The comprehensive planning and operation approach of the digital factory has already become a central innovation issue in the automotive and aviation industry but also in other advanced and globalized sectors. The term is defined as follows:

Digital factory is the generic term for a comprehensive network of digital models, methods and tools – including simulation and 3D visualization – integrated by a continuous data management system. [3]

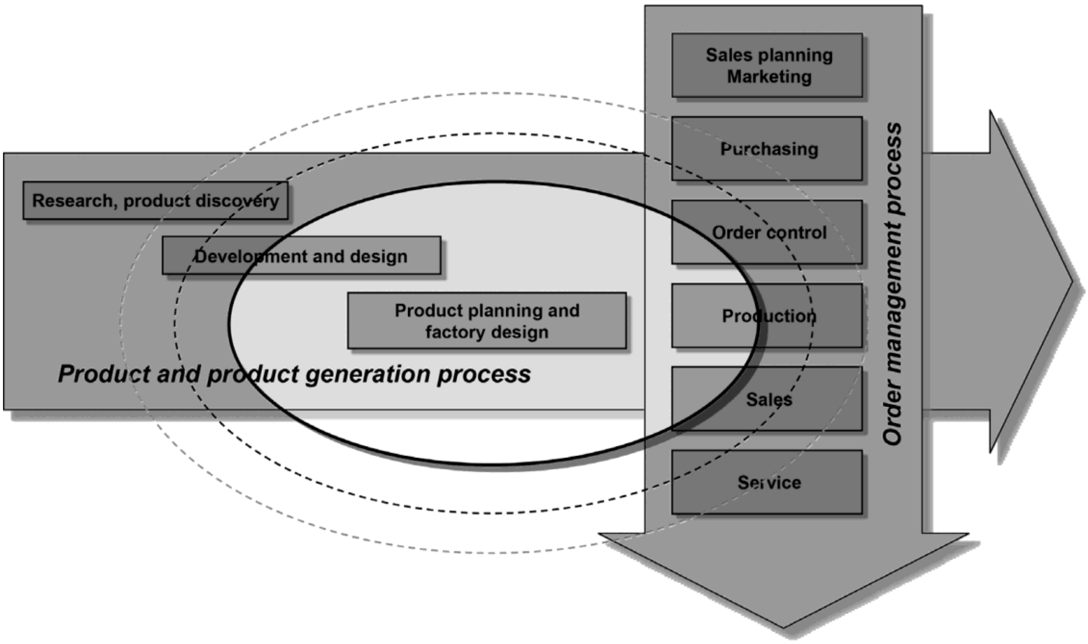


Figure 1 - Focus of the digital factory. Source: [2]

Its aim is the holistic planning, evaluation and ongoing improvement of all the main structures, processes and resources of the real factory in conjunction with the product.

Production planning, in this context, refers to both the planning of the processes as well as the planning of the production systems. The objective is to secure products, production processes and production sequences in an early phase of development, using digital models and tools accompany the production development and, then, to monitor and improve the real production on a constant basis with virtual instruments.

One of these virtual instruments is the software Tecnomatix detailed in section 2.4:, which is a comprehensive portfolio of digital manufacturing solutions; it provides the necessary tools for virtualization, from process layout and design, process simulation and validation, to manufacturing execution.

Virtualization is one of the phases of digital factory and, according to the VDI, it is defined as the methodical transfer of a real object to a computer-integrated processing of digital models with visualization options for its real behavior by simulation. Virtualization is the simulation of what is planned with the aid of digital models including the real behavior of a function system.

In this work the virtualization will be used on the context of a real and already built plant, which is introduced in the next section.

1.2: Hybrid Learning Factory

In order to provide a learning scenario for new employees and special training of employees with several years' service, Siemens I&S[4] in Nuremberg developed, together with Festo, a machine called Athene. It is a hybrid learning factory because it is based on the idea of modular hybrid production with Stand-alone part handling stations.

Hybrid industries are the industries which require systems and solutions for both process and production engineering. The most striking examples are the food and the pharmaceutical industry.

This factory teaches how to handle media used in the pharmaceutical and food industries. Automation specialists acquire practical knowledge in programming and plant documentation. By means of highly modern technology, trainees and engineers monitor temperature, filling level, pressure and flow rate values in order to control pumps, butterfly valves and ball valves in the hybrid learning factory "Athene".

[4] Siemens Industrial Solutions and Services

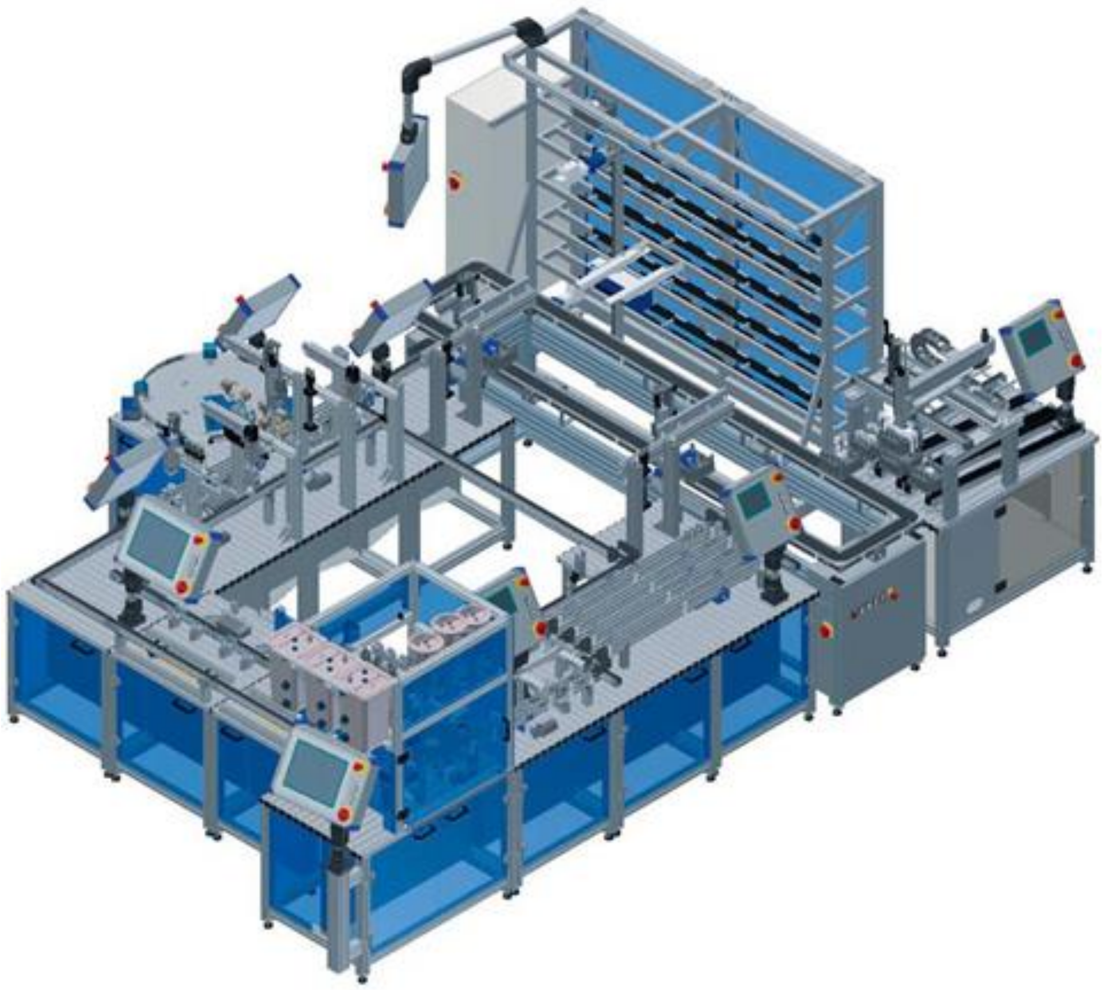


Figure 2 - Hybrid learning factory "Athene". Source: [5]

Before the realization of this work, other academic papers [6] [7] were already conducted to document and virtualize the machine present in the I CS EH BB Siemens Erlangen, which is a variation of Athene, due to that the name was changed to Backbone Factory and it will be deeper described in the Chapter 3:.

1.3: Objectives of this work

Much has been already done, in the sense to provide a learning scenario for students and specialists in the Backbone Factory. Many projects are running in parallel with different subjects like virtualization, programming, drivers and simulation.

However, a high-resolution 3D model to make the complete virtualization is still desired. With this accomplishment a greater number employees in Siemens would be able to improve their knowledge without having to be present at the real installation, and more projects could be run at the same time, topic and hardware but without interfering each other, since the model can be installed and tested in different computers.

Some previous students have already virtualized some parts of the machine, however, this process takes time to be finished. This work is directed to the study and implementation of virtualization in the Backbone Factory, more specific in the Commissioning station.

The objectives of this work are:

- Perform a depth study of the state of the art of factory simulation;
- Provide a preliminary study of the software involved in the process of virtualization;
- Obtain a 2D model of Commissioning station;
- Import the CAD files of Commissioning station to make a 3D model based on the behavior of the 2D model;
- Make a Standalone version of this model to control individually each part of the machine;
- Integrate the virtual environment with a simulated PLC.

1.4: Structure of the document

Much of this work is dedicated to the virtualization from one of the Athene's stations, the Commissioning. Nevertheless a great effort was given to understand the principles of factory simulation, plan simulation and software to achieve the complete virtualization from this part of the machine.

The Chapter 2: presents the technical background necessary to realize this work. In this chapter a brief review of factory simulation and software like S7-PLCSIM, WinCC Flexible and Tecnomatix is presented. A deeper discussion of the main work theme, virtualization, will be carried in the Chapter 4:.

The Chapter 3: presents the description of the Backbone Factory, with its hardware and software characteristics. A more detailed description of the Commissioning station will be done in section 3.1:, also describing its hardware and software characteristics.

The Chapter 4: presents the work procedure. In this chapter the approaches taken to apply the virtualization are discussed, exploring what is feasible and what does not apply. The results are dedicated and discussed in Chapter 5:.

The Chapter 6: presents the conclusion of the developed work, and suggestions for improving the system and also for continuing this work.

Chapter 2: State of Art

In this chapter, a review of the techniques and background knowledge for the development of this academic work is presented. This chapter emphasizes on concepts of virtualization, which are the main subject of this work.

This chapter begins with a brief discussion about digital factory, which is the context of this case study. Further, some basics concepts about the software are showed, which were necessary to the development of this work. The reader, who is already familiar with the concepts here presented, can skip this chapter.

2.1: Factory Simulation

Simulation is a generally accepted technology for the planning, implementation, and operation of technical systems. Several trends in the economy have led to ever-shorter development and planning cycles, such as:

- increasing product complexity and variety
- increasing quality demands in connection with high cost pressure
- increasing demands regarding flexibility
- shorter product life cycles
- shrinking lot sizes
- increasing competitive pressure

These trends have also led to more complex technical systems and more intelligent strategies to fulfil these demands. The simulation has to offer many advantages and it will continue to grow since it can supply many possibilities of examining:

- systems which do not (yet) exist,
- real, existing systems without direct intervention in operation,
- multiple design variants with a minimum of effort,
- the system behavior over long periods (time lapse) and

- Ramp-up[5] processes, transient phases and transitions between well-defined operating states.

As described in section 1.1:, digital factory is early production planning and factory design closely coordinated with all corporate processes. One of its tools is factory simulation, aiming to support all the benefits listed above.

Although the objectives is to reproduce the real system with its process in a model, in factory simulation the model accuracy should not be as detailed as possible, but as detailed as necessary to fulfil the given targets.

2.1.1: Definitions

Before going deeper in some subjects it is important to present some definitions directly related to the purpose of this work:

- **Model** is a simplified replica of a planned or real system with its processes in another system. It differs in important properties only within specified tolerance from the original.
- **System** is defined as a separate set of components which are related to each other.
- **Process** is the totality of mutually influencing operations within a system by which material, energy or information is transformed, transported or stored.
- **Simulation** is the reproduction of a real system with its dynamic processes in a model. The aim is to reach transferable findings for the reality. In a wider sense, simulation means preparing, implementing, and evaluating specific experiments with a simulation model.
- **Simulation run** is the image of the system behavior in the simulation model within a specified period.
- **Experiment** is a targeted empirical study of a model behavior by repeated simulation runs with systematic variation of arguments.
- **Visualization** refers to the generation of a graphic illustration of data and objects by the transformation of data to symbolic and geometric information. It does not necessarily have to be computer-based.

[5] Increase in production ahead of anticipated increases in product demand.

- **Virtualization** is the methodical transfer of a real object to computer-integrated processing of digital models with visualization options for its real behavior by simulation. Virtualization is the simulation of what is planned with the aid of digital models including the real behavior of a function system.
- **Virtual reality** is the application of a high-tech man-machine interface which uses innovative terminal devices to integrate the user in a three-dimensional, internal computer world. It thus includes all technologies for the definition and real-time processing of an internal, three-dimensional computer model for the human senses which allows users to immerse themselves into the model and to manipulate it directly as a result of multimodal feedbacks initiated by the model (interaction). [4] [8]

2.1.2: Uses

Simulation can be used during, planning, implementation and operational phases, as well as in the operative level, in which continues monitoring and controlling the material flow, incoming goods, production and outgoing goods.

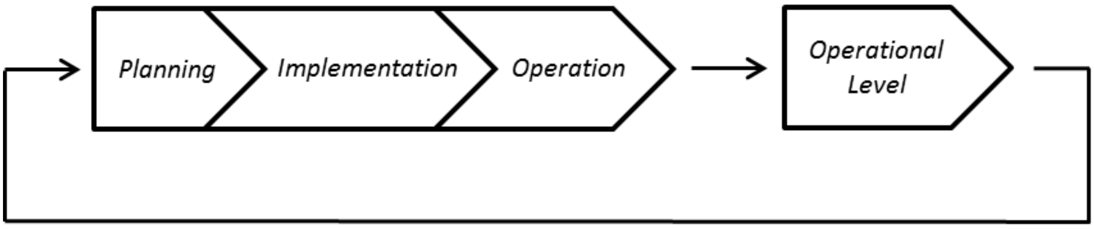


Figure 3 - Simulation’s life cycle

The simulation can be summarized in 3 phases:

- **Planning phase**

The possible level of the simulation model accuracy is determined by the planning depth already achieved and the targets of the investigation. It can help on the identification of bottlenecks in derivation of potential improvement and to visualize alternatives for decision-making.

During planning phase, tests of arguments regarding capacity, effectiveness of control, performance limits, bottlenecks, throughput speed, and volume of stocks can be done to predict minimum and maximum of utilization.

- **Implementation phase**

It has typical application areas such: performance tests in the real plant and also for future requirements; simulation of exceptional system conditions like accidents; and improvement of employee training regarding the handling of emergencies and disturbances.

- **Operational phase**

Here it can be done the comparative valuation of short-term and situation dependent process sequence variations, such as, the examination in advance of useful strategies and responses to disturbances and alterations as well as the examination of long-term development trends.

Typical issues are: testing of control alternatives; review of emergency strategies and accident programs; proof of quality assurance and fault management; and dispatching of orders and determination of the probable delivery dates.

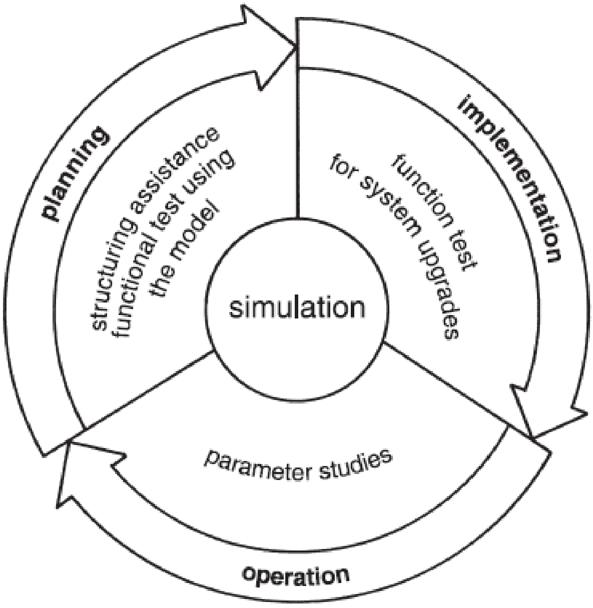


Figure 4 - Application areas for simulation. Source: [3]

2.1.3: Procedure of Simulation

According to [4] and [1] the following approach is recommended:

1. Formulations of problems
2. Test of the simulation-worthiness
3. Formulation of targets

4. Data collection and data analysis
5. Modeling
6. Execute simulation runs
7. Result analysis and result interpretation
8. Documentation

- **Formulation of problems**

Together with the customer of the simulation, the simulation expert must formulate the requirements for the simulation. The result of the formulated problem should be a written agreement (e.g., a technical specification), which contains concrete problems that will be studied using simulation.

- **Test of the simulation-worthiness**

To assess the simulation-worthiness it can be examined for example: the lack of analytical mathematical models; high complexity; inaccurate data; gradual exploration of system limit; and repeated use of the simulation model.

- **Formulation of targets**

Each company aims at a system of targets. It usually consists of a top target (such as profitability), that splits into a variety of sub targets, which interact with each other. The definition of the target system is an important preparatory step. Frequent targets for simulations are for example:

  - Minimizing processing time
  - Maximizing utilization
  - Minimizing inventory
  - Increasing in-time delivery

All defined targets must be collected and analyzed statistically at the end of the simulation runs, which implies a certain required level of detail for the simulation model. As a result, they determinate the range of the simulation study.

- **Data collection and data analysis**

The data required for the simulation study can be structured as a system load data, an organizational data and a technical data.

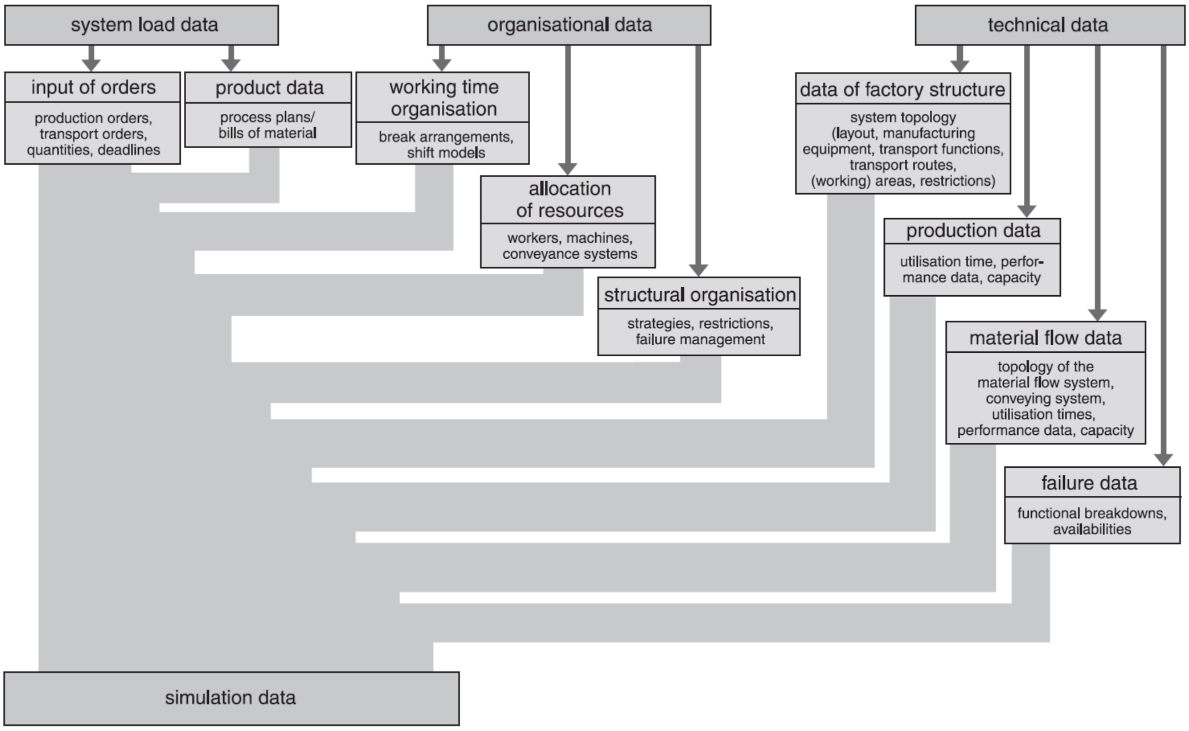

system load data
input of orders
production orders, transport orders, quantities, deadlines
product data
process plans/ bills of material
organisational data
working time organisation
break arrangements, shift models
allocation of resources
workers, machines, conveyance systems
structural organisation
strategies, restrictions, failure management
technical data
data of factory structure
system topology (layout, manufacturing equipment, transport functions, transport routes, (working) areas, restrictions)
production data
utilisation time, performance data, capacity
material flow data
topology of the material flow system, conveying system, utilisation times, performance data, capacity
failure data
functional breakdowns, availabilities
simulation data

Figure 5 - Data collection overview. Source [4]

- **Modeling**

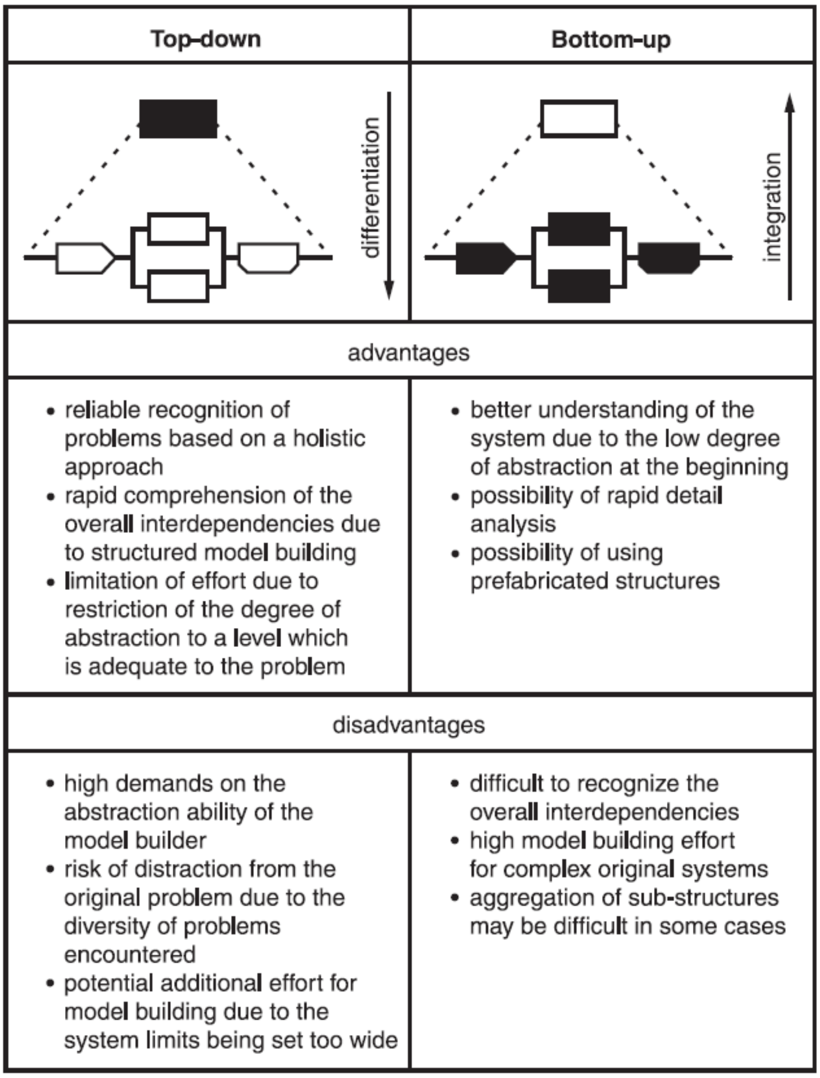
The modeling phase includes building and testing the simulation model. Modeling usually consists in two stages:

1. Deriving an iconic model from the conceptual model.
2. Transferring the model into a software model.

  - First Modeling Stage

First, the user has to develop a general understanding of the simulated system. Based on the objectives to be tested, decisions about the accuracy of the simulation need to be made. Based on the accuracy of the simulation, essencial decisions are taken about which aspects you want to simplify. The first modeling stage covers two activities: analysis (top-down) and abstraction (bottom-up). The advantages and disadvantages can be seen in the Figure 6.

Using the system analysis, the complexity of the system in accordance with the original investigation targets will be dissolved by meaningful dissection of the system into its elements. By abstraction, the amount of the specific system attributes will be decreased as far as it is practical to form an essential limited image of the original system. Typical methods of abstraction are reduction (elimination of not relevant details) and generalization (simplification of the essential details).


differentiation
integration

Top-down	Bottom-up
advantages	
• reliable recognition of problems based on a holistic approach • rapid comprehension of the overall interdependencies due to structured model building • limitation of effort due to restriction of the degree of abstraction to a level which is adequate to the problem	• better understanding of the system due to the low degree of abstraction at the beginning • possibility of rapid detail analysis • possibility of using prefabricated structures
disadvantages	
• high demands on the abstraction ability of the model builder • risk of distraction from the original problem due to the diversity of problems encountered • potential additional effort for model building due to the system limits being set too wide	• difficult to recognize the overall interdependencies • high model building effort for complex original systems • aggregation of sub-structures may be difficult in some cases

Figure 6 - First modeling stage approach. Source [4]

- Second Modeling Stage

A simulation model will be built and tested. The result of modeling has to be included in the model documentation to make further changes of the simulation model possible. In practice, this step is often neglected, so that models due to the lack of documentation of functionality cannot be used. Therefore, there is a needing for commenting the models and the source code during programming. In this way the explanation of the functionality is still available after the programming is finished.

- **Execute simulation runs**

Depending on the objectives of the simulation study, the experiments based on a test plan will be realized. In the test plan, the individual experiments output data, arguments of the model, objectives, and expected results are determinate. It is also important to define a time span for the simulation experiments, based on the findings of the test runs. Computer runs spanning several hours or frequent repetitive experiments for the statistical coverage are not uncommon. In these cases it is helpful to check if it is possible to control the experiments by a separate programmed object (batch runs).

The realization times for the experiments can be relocated partly in the night hours, so the available computing capacity can be utilized optimally. Input and output data and the underlying parameters of the simulation model must be documented for each experiment.

- **Result analysis and result interpretation**

The values, which will change in the modeled system, are derived from the simulation results. The correct interpretation of the simulation results significantly influences the success of a simulation study. If the results contradict the assumptions made, it is necessary to analyze what influences are responsible for the unexpected results. It is also important to realize that complex systems often have a ramp up phase. This phase may run differently in reality and in the simulation. Therefore, the results obtained during the ramp up phase are often not transferable to the modeled system and might have no influence for the evaluation (Exception: the Ramp up phase of the original system has to be fully modeled).

- **Documentation**

For the documentation of a simulation study, the form of a project report is recommended. The documentation should provide an overview of the timing of the study and document the work carried out. The documentation of failed system variants and constellations is the matter of interest in this context. The core of the project report should be a presentation of the simulation results based on the required specifications of the customer. Resulting from the simulation study it makes sense to include proposals for actions in the documentation. Finally, is recommended to describe the simulation model in its structure and its functionality.

2.2: S7-PLCSIM

SIMATIC by Siemens is one of the world's leading industrial automation systems. It includes a range of industrial automation products designed to accommodate a large variety of tasks. Flexible and cost-effective, the SIMATIC system is ideally suited to manage the ever-growing demands the manufacturing and process industry requires from today’s machines and systems.

One of the products in the SIMATIC portfolio is the SIMATIC STEP 7 Software, a package that supports the complete engineering workflow during a project. It has an extensive range of useful tools and numerous sophisticated functions that support throughout the automation project and reduce the complexity of PLC programming, such as:

- Configuring: SIMATIC Manager, Hardware configuration
- Development and debugging of IEC languages: LAD, FBD, STL, SCL, Graph
- Commissioning: Integrated system diagnostics and online functionality
- Operation and maintenance: P-Diag, TeleService.

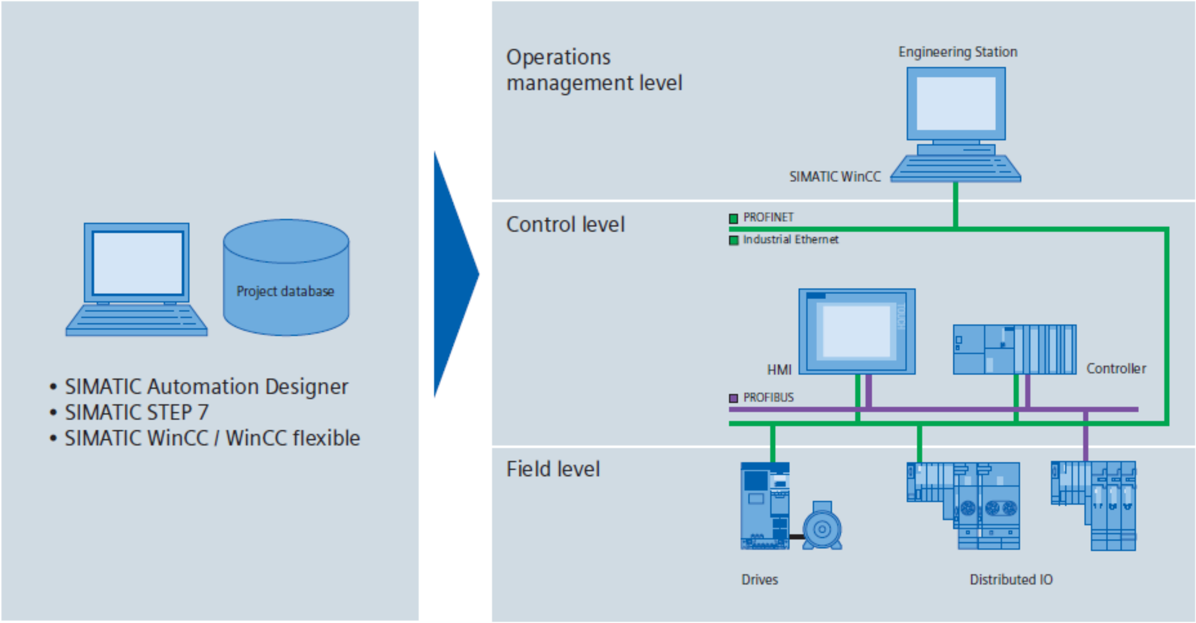


Figure 7 - SIMATIC integrated automation

S7-PLCSIM is a component of the STEP 7 software and it simulates a controller for functional testing of user blocks and programs for S7-300 and S7-400 (SIMATIC Controllers) on the programming device/PC. Online access and test functions of the programming tools can be executed in exactly the same way as if it was with a real controller. This allows the entire program test to be executed on-site in the development office. The facility to simulate the communication via MPI, PROFIBUS DP and TCP/IP is new and ensures a high degree of flexibility in the simulation. [9]

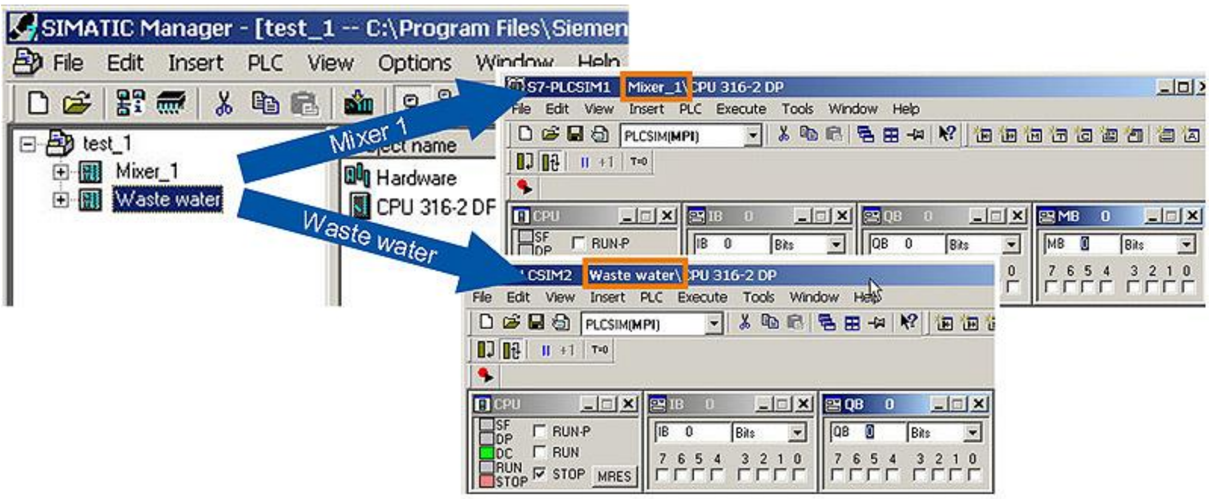

SIMATIC Manager - [test_1 -- C:\Program Files\Siemen
File Edit Insert PLC View Options Window Help
test_1
Mixer_1
Waste water
Object name
Hardware
CPU 316-2 DP
Mixer 1
Waste water
S7-PLCSIM1 Mixer_1\CPU 316-2 DP
File Edit View Insert PLC Execute Tools Window Help
PLCSIM(MPI)
CPU
SF
DP
RUN-P
IB 0
QB 0
MB 0
Bits
7 6 5 4 3 2 1 0
CSIM2 Waste water\CPU 316-2 DP
File Edit View Insert PLC Execute Tools Window Help
PLCSIM(MPI)
CPU
SF
DP
DC
RUN
STOP
RUN-P
RUN
STOP
MRES
IB 0
QB 0
Bits
7 6 5 4 3 2 1 0

Figure 8 - Screenshot of S7-PLCSIM. Source [9]

2.3: WinCC Flexible

Maximum transparency is essential for the operator who works in an environment where processes are becoming more complex, and requirements for machine and plant functionality are increasing. The Human Machine Interface (HMI) provides this transparency. [10]

The HMI system represents the interface between man (operator) and process (machine/plant). The PLC is the actual unit which controls the process. Hence, there is an interface between the operator and WinCC flexible (at the HMI device) and an interface between WinCC flexible and the PLC. An HMI system assumes the following tasks:

- **Process visualization**

The process is visualized on the HMI device. The screen on the HMI device is dynamically updated. This is based on process transitions.

- **Operator control of the process**

The operator can control the process by means of the GUI. For example, the operator can preset reference values for the controls or start a motor.

- **Displaying alarms**

Critical process states automatically trigger an alarm, for example, when the set point value is exceeded.

- **Archiving process values and alarms**

The HMI system can log alarms and process values. This feature allows you to log process sequences and to retrieve previous production data.

- **Process values and alarms logging**

The HMI system can output alarms and process value reports. This allows you to print out production data at the end of a shift, for example.

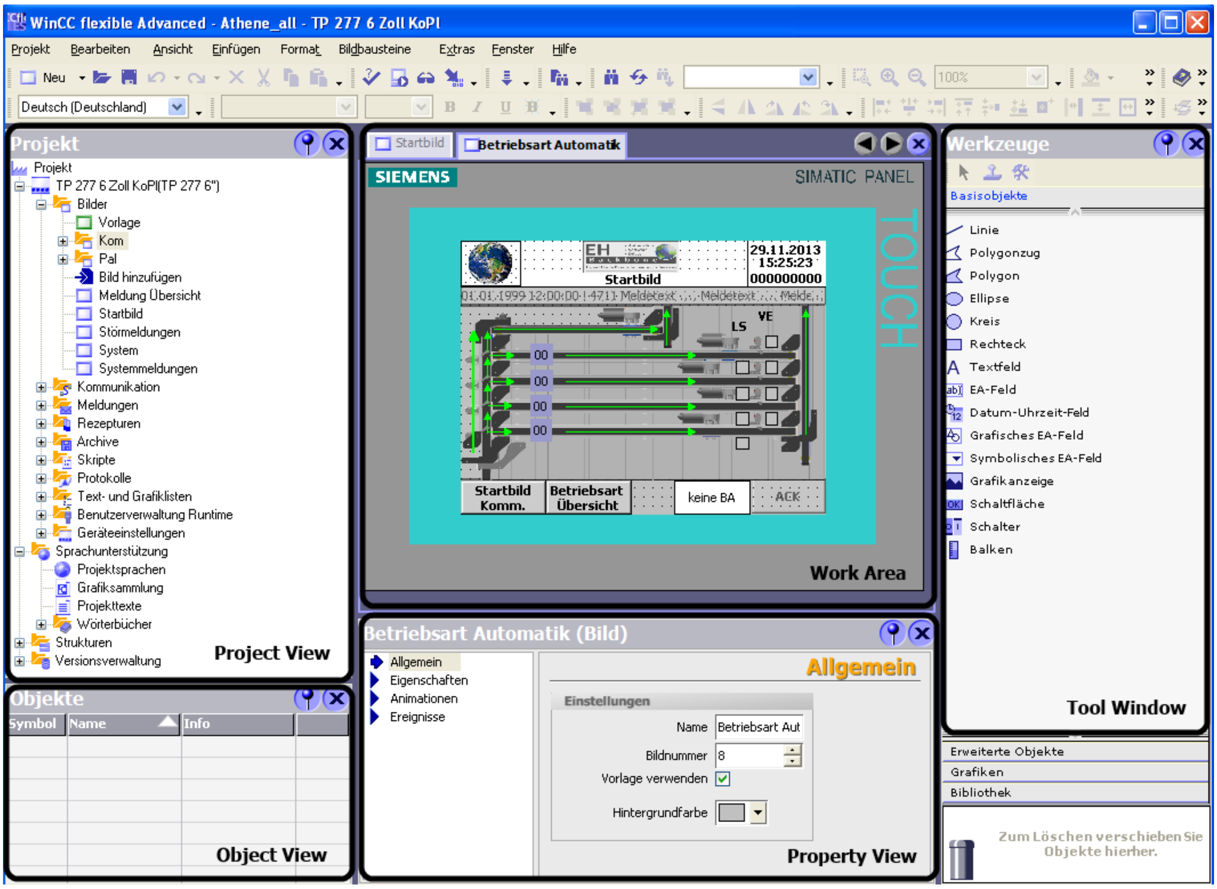
- **Process and machine parameter management**

The HMI system can store the parameters of processes and machines in recipes. For instance, these parameters can be downloaded in one pass from the HMI device to the PLC in order to change over the product version for production.

WinCC flexible is the HMI software for future-proof machine-oriented automation concepts with comfortable and highly efficient engineering, it is consisted of 3 components:

1. **WinCC flexible Engineering System**: software for handling all essential configuring tasks. The WinCC flexible edition determines which HMI devices in the SIMATIC HMI spectrum can be configured.
2. **WinCC flexible Runtime**: software for process visualization. The project can be executed in process mode in Runtime.
3. **WinCC flexible options**: allows expanding the standard functionality of WinCC flexible.

The WinCC flexible workbench opens on the screen of the programming computer when someone creates a new or opens an existing project in WinCC flexible. The project structure is visualized and the project is managed in the "Project View."



Name Betriebsart Aut

Bildnummer 8

Vorlage verwenden ☑

Hintergrundfarbe

Figure 9 - WinCC Flexibe Screenshot Project View

WinCC flexible provides a special editor for each configuring task. For example, to configure the GUI of an HMI device in the "Screens" editor or use the "Discrete Alarms" editor to configure alarms.

In runtime, the operator can control and monitor the process. This involves in particular the following tasks:

- Communication with the automation systems.
- On-screen visualization of images
- Operating the process, for example, by setting setpoint values or opening and closing valves.
- Archiving of current runtime data, e.g. process values and alarm events.

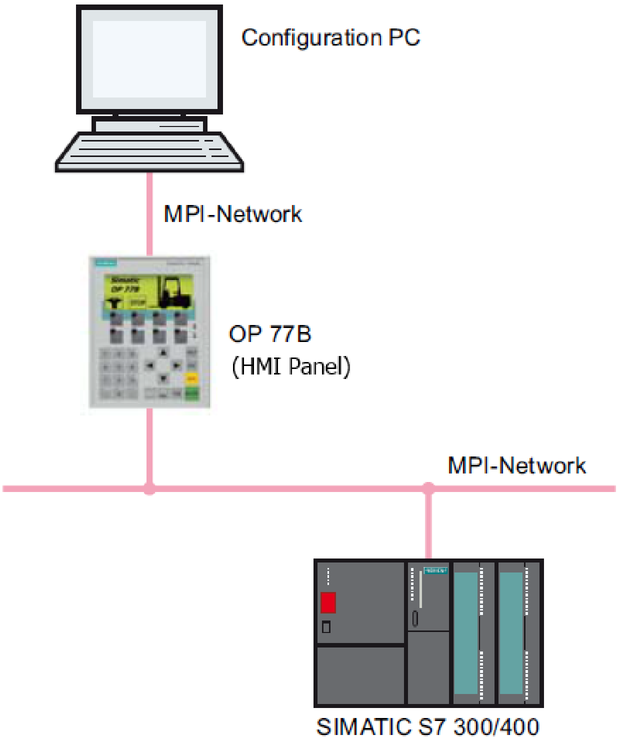


Figure 10 – Overview WinCC connections with HMI

2.4: Tecnomatix Plan Simulation

Plant Simulation is a discrete-event simulation tool that helps to create digital models of logistic systems (such as production), so that the user can explore a system's characteristics and optimize its performance. These digital models allow to run experiments and what-if scenarios without disturbing existing production systems or – when used in the planning process – long before the real production systems are installed. Extensive analysis tools, such as bottleneck analysis, statistics and charts let evaluate different manufacturing scenarios. The results provide the information needed to make fast, reliable, smarter decisions in the early stages of production planning. [11]

Some features:

- Optimize systems for reduced energy usage
- Simulation of complex production systems and control strategies
- Object-oriented, hierarchical models encompassing business, logistics and production processes
- Dedicated application object libraries for fast and efficient system modeling

- Graphical outputs for analysis of throughput, resources and bottlenecks
- Automatic bottleneck detection, Sankey diagrams and Gantt charts
- 3D online visualization and animation
- Integrated neural networks and experiment handling
- Automated system optimization via genetic algorithms
- Value stream mapping and simulation
- Open system architecture supporting multiple interfaces and integration capacities (ActiveX, CAD, Oracle SQL, ODBC, XML, Socket, OPC, etc.)

There are 3 main areas that Tecnomatix provides for its users in order to model, simulate, analyze, visualize and improve production systems:

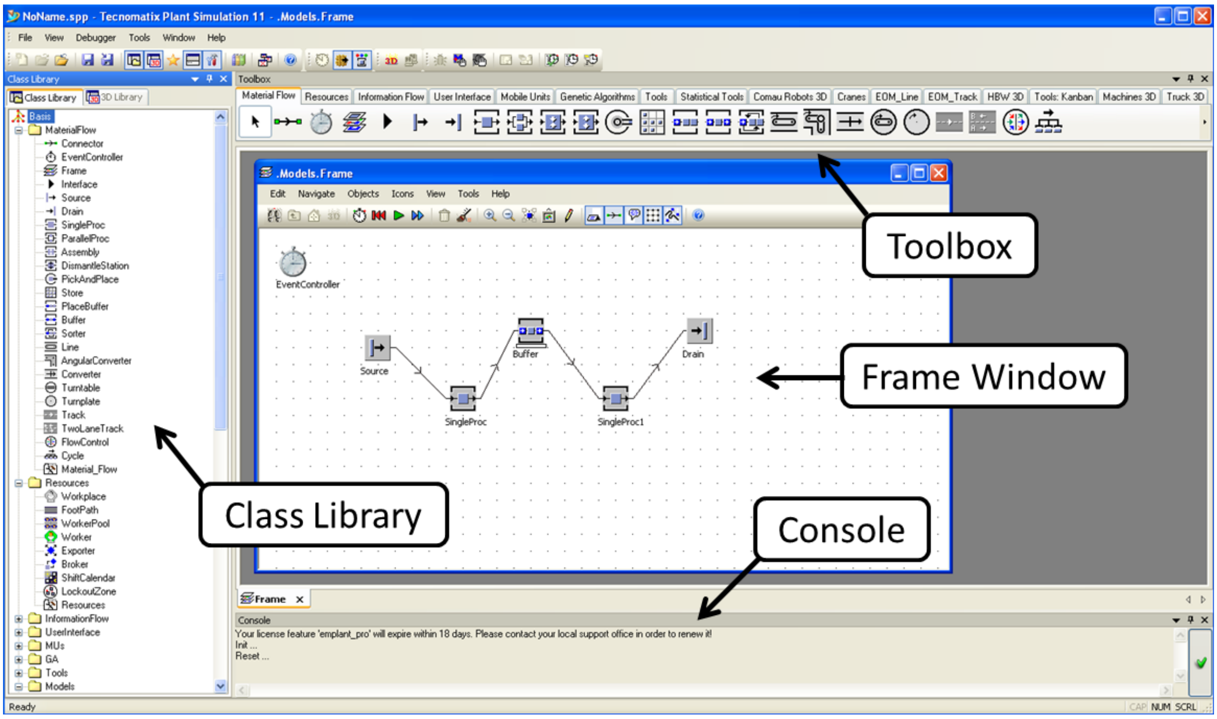
1. Modeling manufacturing processes

Plant Simulation enables to create well-structured, hierarchical models of production facilities, lines and processes. This is achieved through powerful object-oriented architecture and modeling capabilities that enable to create and maintain even highly complex systems, including advanced control mechanisms.

Plant Simulation's user interface follows Microsoft Windows standards, making it easy to get familiar with and to productive quickly. Simulation models can be quickly created by using components from application object libraries dedicated to specific business processes, such as assembly or car body manufacturing processes. Users can choose from predefined resources, order lists, operation plans and control rules.

Complex and detailed simulations can be handled, understood and maintained much better than in conventional simulation tools by using Plant Simulation architectural advantages like capsulation, inheritance and hierarchy.

The program works with object-oriented programming and therefore carries the concepts of objects, classes, instances and inheritance. The figure bellow shows the class library where all objects are for the simulation, here it can derive and duplicate classes, create frames and load objects from other simulations models. The console provides information during the simulation like error messages or printed messages. The toolbox is a quick access to the class library.



```
Your license feature 'emplant_pro' will expire within 18 days. Please contact your local support office in order to renew it!
Init ...
Reset ...
```

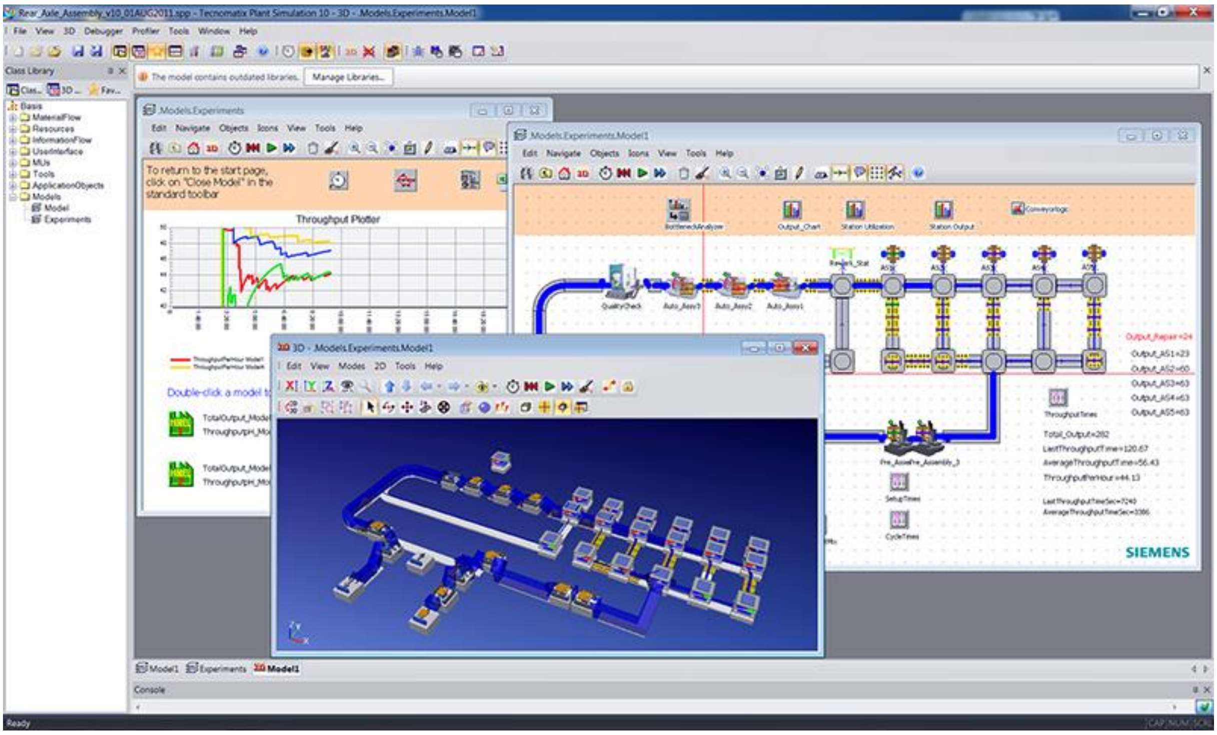
Figure 11 – Tecnomatix program window

2. Simulating and analyzing system performance

Plant Simulation models are used to optimize throughput, relieve bottlenecks and minimize work-in-process. The simulation models take into consideration internal and external supply chains, production resources and business processes, allowing to analyze the impact of different production variations. Statistical analysis, graphs and charts display the utilization of buffers, machines and personnel. Extensive statistics and charts can be generated to support dynamic analysis of performance parameters including line workload, breakdowns, idle and repair time and proprietary key performance factors.

3. Model visualization

In addition to the highly efficient 2D model view of Plant Simulation, models may be visualized in a 3D virtual environment using Plant Simulation's included libraries or your own CAD data. The result is impressive 3D virtual models that are synchronized at all times with their 2D counterparts, allowing you the flexibility to choose the appropriate method of visualization without compromising simulation and analysis needs.


Rear_Axle_Assembly_v10_01AUG2011.spp - Tecnomatix Plant Simulation 10 - 3D - .Models.Experiments.Model1
File View 3D Debugger Profiler Tools Window Help
Class Library
Basis
MaterialFlow
Resources
InformationFlow
UserInterface
MUs
Tools
ApplicationObjects
Models
Model
Experiments
Manage Libraries...
.Models.Experiments
Edit Navigate Objects Icons View Tools Help
Throughput Plotter
.Models.Experiments.Model1
3D - .Models.Experiments.Model1
Edit View Modes 2D Tools Help
Station Utilization
Station Output
ThroughputTimes
SetupTimes
CycleTimes
SIEMENS
Console
Ready

The model contains outdated libraries.

To return to the start page, click on "Close Model" in the standard toolbar

Output_Repair=24
Output_AS1=23
Output_AS2=60
Output_AS3=63
Output_AS4=63
Output_AS5=63

Total_Output=282
LastThroughputTime=120.67
AverageThroughputTime=56.43
ThroughputPerHour=44.13

Figure 12 – Tecnomatix Plant Simulation Screenshot. Source [11]

As Plant Simulation is a discrete event-controlled simulation program, it only inspects those points in time, at which events take place within the simulation model. In reality, on the other hand, time elapses continually. When watching a part move along a conveyor system, you will detect no leaps in time. The curve for the distance covered, and the time it takes to cover it, is continuous, it is a straight line.

On the other hand, a discrete, event-controlled simulation program only takes points in time (events) into consideration that are of importance to the further course of the simulation. Such events might be, for instance, a part entering or leaving the station, or also a part moving from one to another machine. Any movements in between are of little interest to the simulation as such. It is only important that the entrance and the exit (Out) events are displayed correctly. When a part enters a material flow object, Plant Simulation calculates the time until it exits the object and enters an exit event into the list of scheduled events of the Event Controller for this point in time.

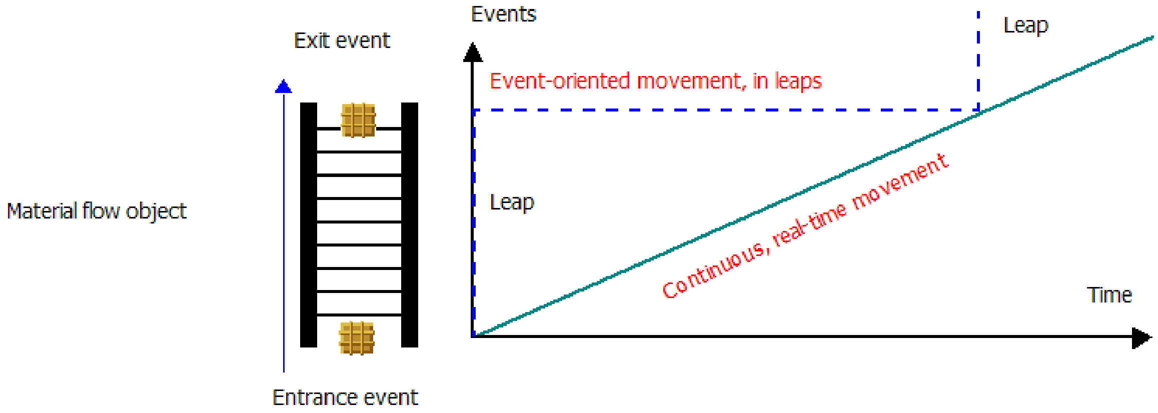

Exit event
Material flow object
Entrance event
Events
Event-oriented movement, in leaps
Leap
Leap
Continuous, real-time movement
Time

Figure 13 - Plant Simulation as event-controlled. Source [12]

Thus, the simulation time that the Event Controller displays, leaps from event to event. This happens as soon as an event is processed.

2.5: OPC Interface

OPC is a set of industrial standards for systems interconnectivity, providing a common interface for communications among different products and different vendors. It was developed in 1996 by the automation industry as a standard specification that would allow the interoperability in automation by creating and maintaining open specifications that standardize the communication of acquired process data, alarm and event records, historical data, and batch data to multi-vendor enterprise systems and between production devices. Production devices include sensors, instruments, PLCs, RTUs, DCSs, HMIs, historians, trending subsystems, alarm subsystems, and more as used in the process industry, manufacturing, and in acquiring and transporting oil, gas, and minerals.

The OPC Foundation was created to maintain the standard and has since overseen the introduction of a series of standard specifications (such as OPC data access). Today the OPC Foundation states that OPC UA is no longer an acronym for OLE for process control, but OPC UA is an acronym for OPen Connectivity Unified Architecture.

Classic OPC provides standard specifications for data access (DA), historical data access (HDA), and alarms and events (A&E). These OPC specifications are widely accepted by the automation industry and it's based on aging Microsoft-

COM[6]/DCOM[7]-technology, which has led to the development of new OPC Foundation specifications known as OPC UA (Unified Architecture).

The main goal of OPC UA is to keep all the functionality of Classic OPC, while switching from Microsoft-COM/DCOM-technology to state of-the-art Web services technology. OPC UA becomes platform-independent by using web service technology, thus it can be applied in scenarios where Classic OPC is not used today. OPC UA can be seamlessly integrated into Manufacturing Execution Systems (MES) and Enterprise Resource Planning (ERP) systems, running not only on Unix/Linux systems using Java, but also on controllers and intelligent devices having specific real-time capable operation systems. [13]

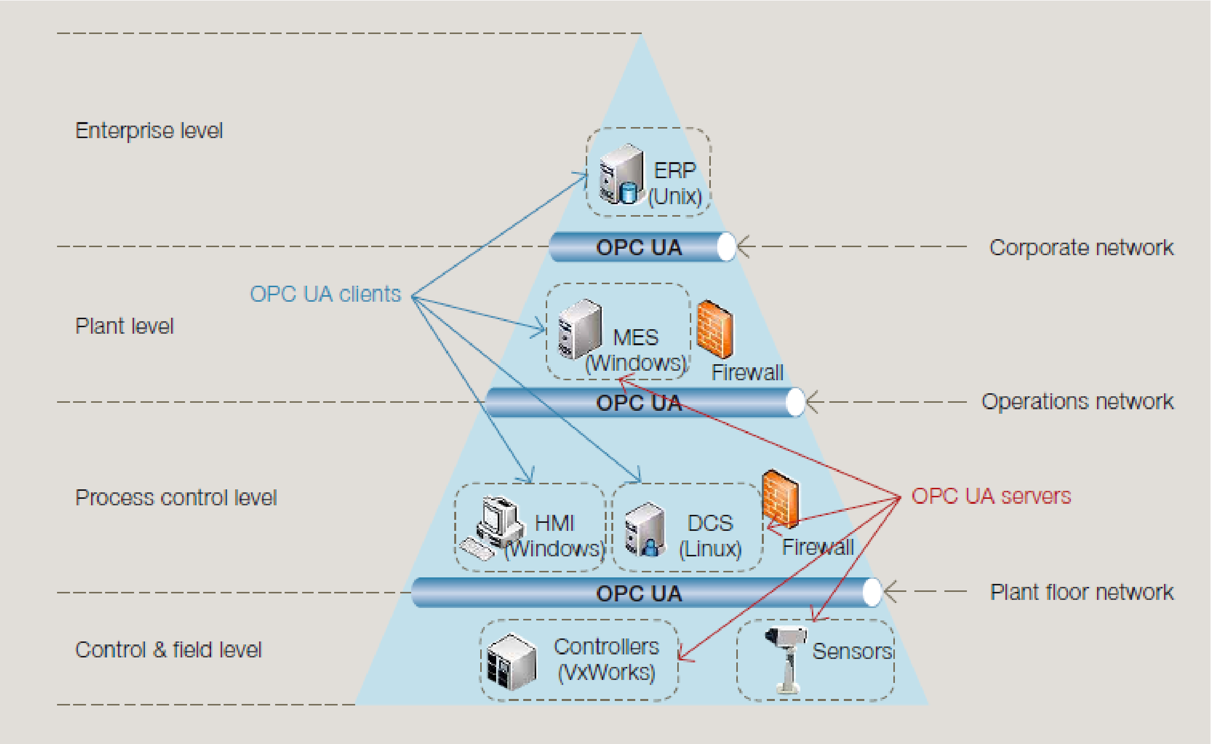


Figure 14 - OPC applications within the automation pyramid. Source [13]

In principle, communication via OPC is a server-client form of communication. All OPC servers have one or more data connections via which the system-specific data of the automation or visualization system is made available to the OPC server. The OPC server converts this data in compliance with the OPC standard and makes

[6] COM was introduced by Microsoft in 1993 to allow component software to communicate between different applications.

[7] DCOM was introduced by Microsoft allowing software components to communicate even when distributed across a network.

it available via the OPC interface. Other systems working as OPC clients can exchange data with the OPC server via this common interface. [14]

Data exchange between an OPC DA server and an OPC DA client runs on the same PC (local) via the COM interface. In the case of different PCs, OPC communication runs via the DCOM interface of the system.

OPC XML has been added to OPC and data exchange with OPC XML is made via SOAP protocol. This provides platform-independent data exchange and also enables data transfer via the Internet. The server-client architecture of OPC XML is retained. It needs an OPC XML gateway for exchanging data between an OPC DA client and an OPC XML DA server.

Multi Panels and Mobile Panels from Siemens running WinCC can also act optionally as OPC XML DA servers. Communication via DCOM with the OPC XML DA server of the Multi Panel or Mobile Panel is not possible, because the realtime operating system Windows CE has no DCOM interface. Therefore, an OPC XML gateway is always required for communication between a PC (OPC DA client) and a multi panel or mobile panel (OPC XML DA server). The OPC XML gateway implements communication from XML to COM or DCOM. This mapping with OPC will be detailed in section 4.4:.

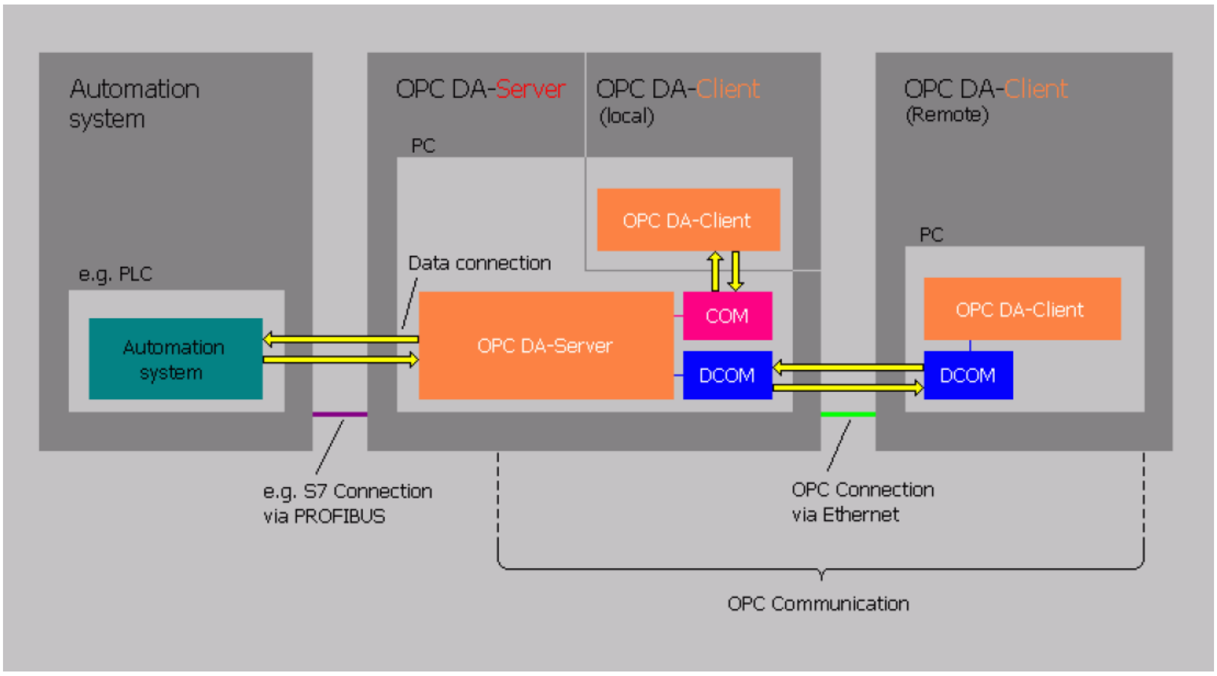


Figure 15 – Connections in OPC Interface

Chapter 3: Description of the Machine

The Backbone Factory placed in the I CS EH BB Siemens Erlangen was already introduced in section 1.2:, but in this chapter a deeper discussion about functionalities, hardware and software will be presented.

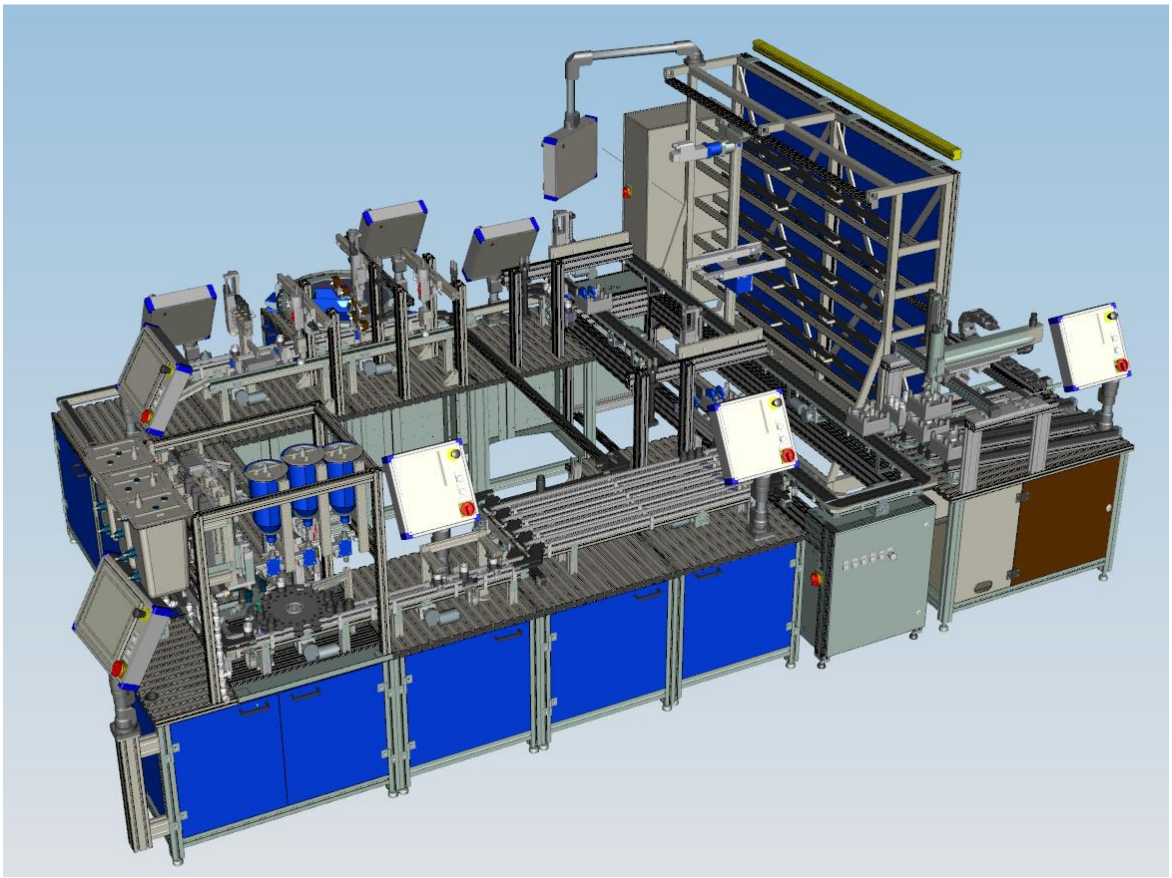


Figure 16 – The Backbone Factory

The main objective of this factory is to provide an environment where people could take part in training sessions and consolidate their previous theoretical knowledge with practical experience of process and production automation. Therefore the subject chosen for this machine was to teach how to handle media used in the pharmaceutical industry, since it is a hybrid industry that requires systems and solutions for both process and production engineering.

A scenario to manipulate bottles that are field with caps was built based on tasks related to the pharmaceutical industry such as: mixing, filtering, temperature control, dosing, filling, packing, storing, delivering and recycling.

The factory is composed of 11 stations, which are: Modular Production System; Bottling; Quality Gate; Commissioning; Palletizer; Transport; In/Out; High Bay Racking; De-Palletizer; Recycling Solid; and Recycling Fluid. Their symbols can be seen in Figure 17 and their position in Figure 18. Generally the stations are divided in 4 zones:

- **Zone 1: Production automation**

The station MPS is responsible for this area and here the caps are produced, tested and fed to the bottles by the filling station. Programmable logic controllers monitor and control the production process. Various sensors record the end positions of the actuators or identify and differentiate among the work-pieces. Typical actuators used in automation production ensure fast and precise movement, such as linear cylinders, swivel cylinders, motors, parallel grippers or vacuum suction cups.

Zone 2: Filling and Packing

Not only dosing, filling, capping and packing, but also the acquisition, storage and reliable management of product and quality data are key tasks in this section. The position and fill level of each bottle and the state of completion of each lot are recorded by various optical sensors and a high-speed camera. The bottles are either filled with solids or colored liquids or with both of them.

After the containers have been metered and filled in the first stage of the production automation process, the next work step is to close them with plastic lids using a rotary gripper. There is also a separate production area for the lids themselves, in which PLCs control and monitor production. The production data can be fully tracked by means of RFID tags in the bottle caps. The stations Bottling, Quality Gate, Commissioning and Palletizer run this zone.

- **Zone 3: Transport and Logistics**

Once the bottles have been filled and quality checked, they are packed in six packs and palletized. Empty six-packs have to be conveyed to the order compilation station or placed in storage. Completed six-packs are delivered just in time, or stored in an interim facility. A highly accurate industrial two-axis handling system places two lots of three bottles on a mini pallet. A high speed camera checks the six packs to make sure they are complete. These are then either dispatched or stored in the appropriate high bay warehouse.

The stock-keeping, either chaotic or systematic, is done in miniature. A Cartesian 3-axis handling system puts items into stock and takes them out again. Intelligent servo motors with integrated servo amplifiers, controllers and a PROFINET bus interface drive two toothed belt axes. Should the items not be put into stock but supplied just in time, a pneumatic 3-axis handling system with a linear gripper places the six packs on the outgoing goods ramp. Here the stations Transport, High Bay Racking and In/Out deal with the transport and logistics.

- **Zone 4: Process Automation**

Throughout the process all communication is via an AS-interface[8] or PROFINET[9]. The stations De-Palletizer, Recycling Solid and Fluid acts as a closed system that reflects the production process in recycling tract. Seals are removed from returns, liquids and solids sucked out and discharged, while the packaging and the contents are sent back to the manufacturing section

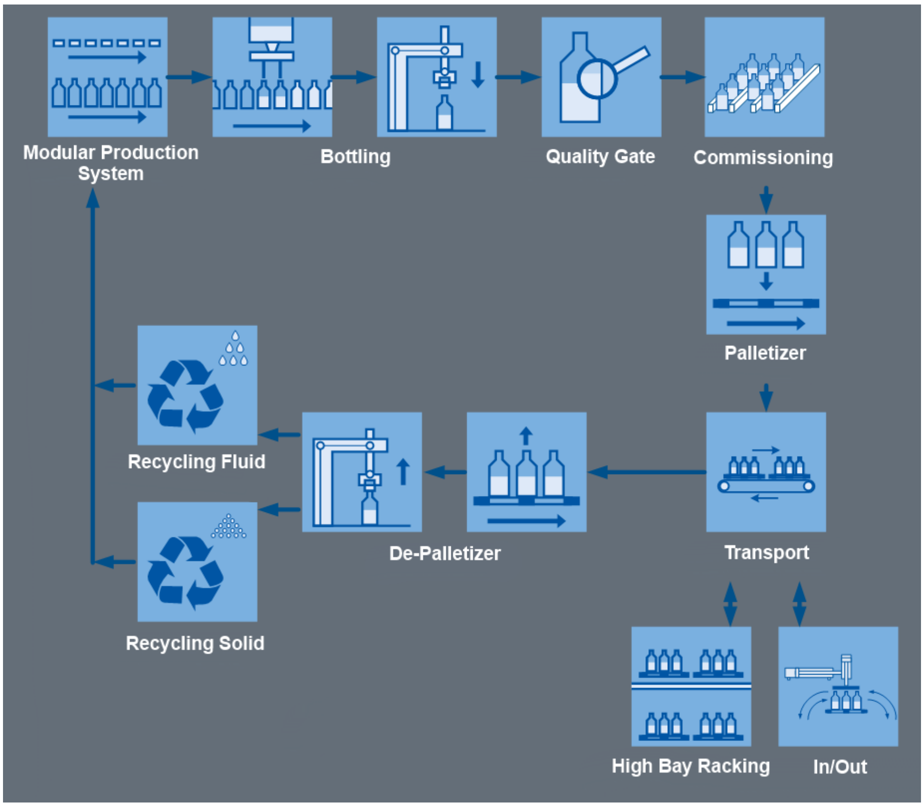


Figure 17 – Backbone Factory’s route

[8] Protocol for networking sensors and actuators in automation systems.
[9] Standard for industrial automation using a computer network, such as TCP/IP and Ethernet.

3.1: Stations

In this section each station will be summarized to understand a little bit more their specific functions, hardware and software.

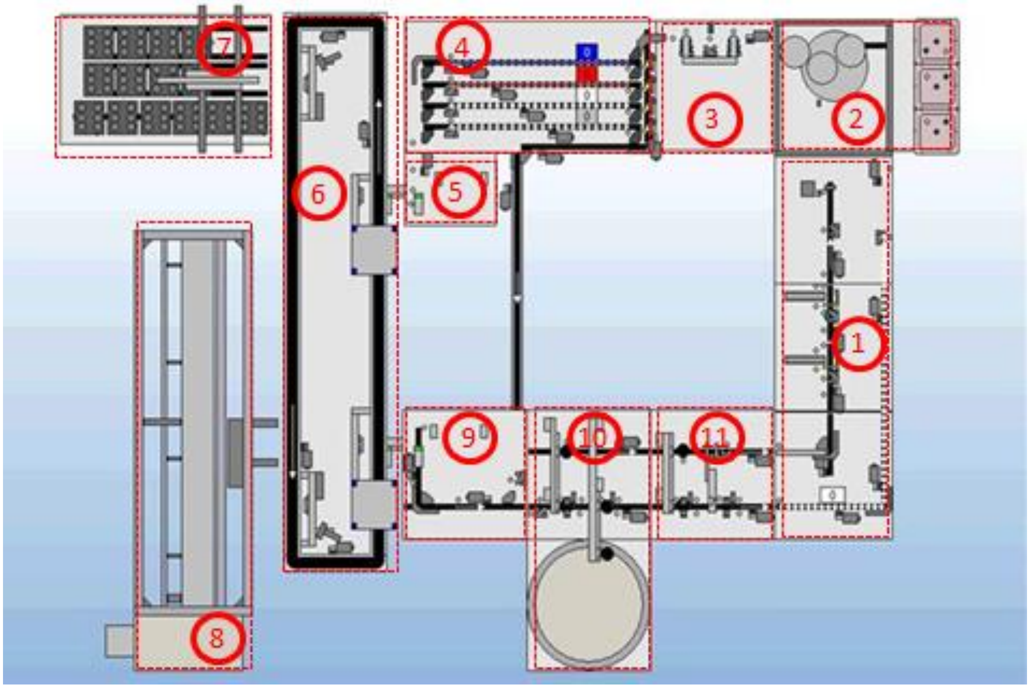


Figure 18 – Backbone Factory's Stations

1. Modular Production System Station

After the Recycling Station the bottles are transported through a conveyor to MPS station and stocked there temporarily. After the receipt of order the wished amount of bottles, depending on the booking, arrive through the separator to the Bottling station.

Meanwhile the lids are transported by another conveyor to get sorted. Thereby they are distinguished by color and status of the lid. If the lid color, which is captured by a camera, does not match to the standard, the lid is dumped by a separator. The same happens considering the status of the lid.

The appropriate lids are moved over a conveyor to the gripper. The gripper grabs, depending on the order, the wished amount of lids, one after another, and transports them to the Bottling station. [17]

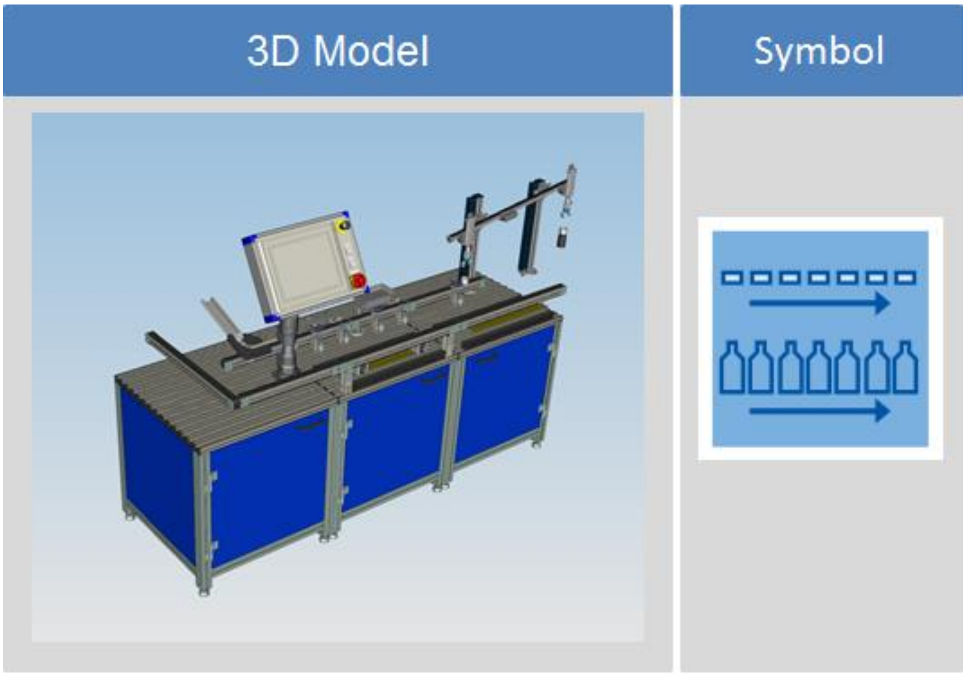


Figure 19 – MPS Station

2. Bottling Station

After the order is assumed, the empty bottles are transported by a conveyor from the MPS station to the Bottling Station. Over a second conveyor the bottles are directed to a rotatory table and released for the filling. The first stop on the rotatory table is recognized with the help of a sensor and, subsequently, the rotatory table is rotated to the next stop. During the rotation another sensor controls the height of the bottles. If the bottle is too big or there is still a lid on it, the unit stops. The rotatory table has 12 positions.

The sensor on stop 2 recognizes the bottle and weights it with the load cell. Afterwards, the bottle is filled with a liquid according to the order. The rotatory table is again rotated and moves step by step until stop 4, where the sensor recognizes the bottle. There the bottle is filled with solid, according to the order, and forwarded to the next step (stop 5). At the stop 5, the sensor gives a signal to the MPS station to charge a lid. This lid is only put on the bottle and then transported to the next stop. At stop 7, the gripper is controlled by a sensor signal and screws the lid on the bottle. At stop 8, the RFID sensor writes on the chip inside the lid. After the recognition of the bottle is made by a sensor in stop 9, the bottle leaves the Bottling station over the conveyor, which also provides the supply of bottles.

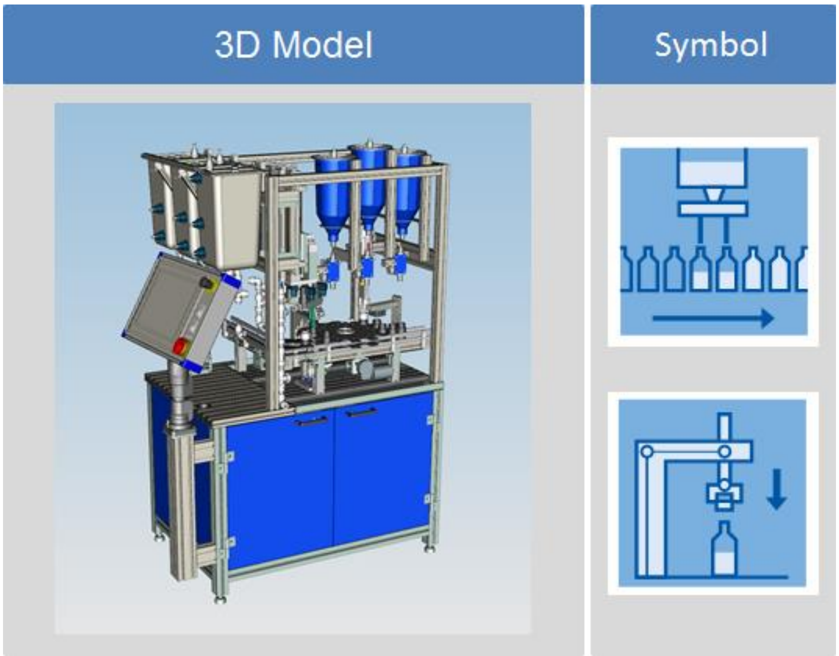


Figure 20 – Bottling Station

3. Quality Gate Station

At the Quality Gate, the content of the bottles is recognized by the first reading device. Afterwards the content is checked by a color scanner so as to confirm if it matches with the order. Whether the content does not match with the order, the product is transferred to recycling; otherwise, it is read off by a second reading device and then delivered to the according Commissioning line.

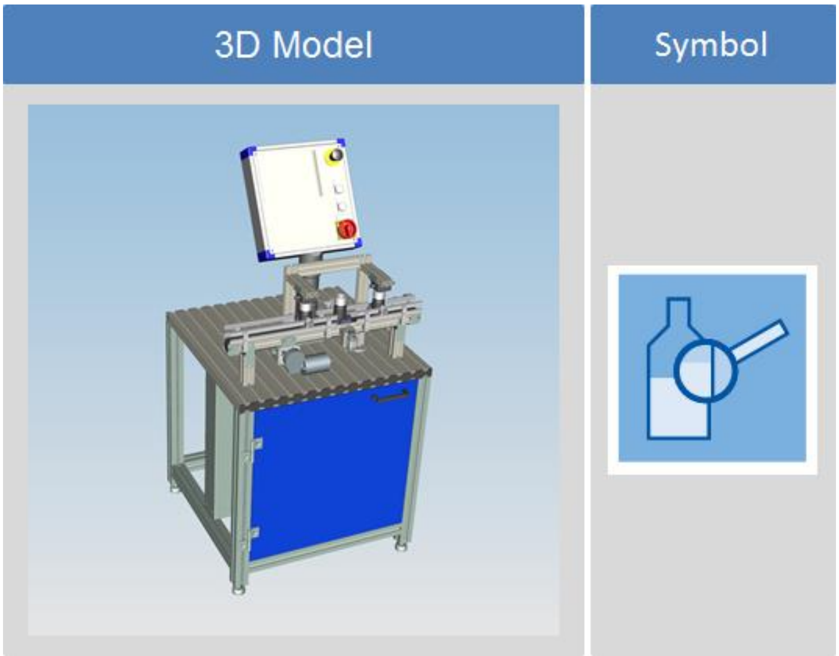


Figure 21 – Quality Gate Station

4. Commissioning Station

Since the present station is the focus of this work, a more in-depth discussion will be developed in the section 3.2: concerning its hardware and software.

The station gets a few bottles from the Quality Gate with the information whether the bottle has to be recycled or finished. If the bottle has to be recycled, it goes over the Commissioning station to the Recycling Station. If the bottle matches with the specification, it goes to the respective commissioning line, according to the content (fluid or solid). The first three lines are for bottles filled with fluid content and the fourth one transports bottles with solid content.

Finally the bottles are located in the separator that only passes the bottles if there is no more bottles on the Palletizer or if the bottles belong to the same order as the bottles already on the Palletizer.

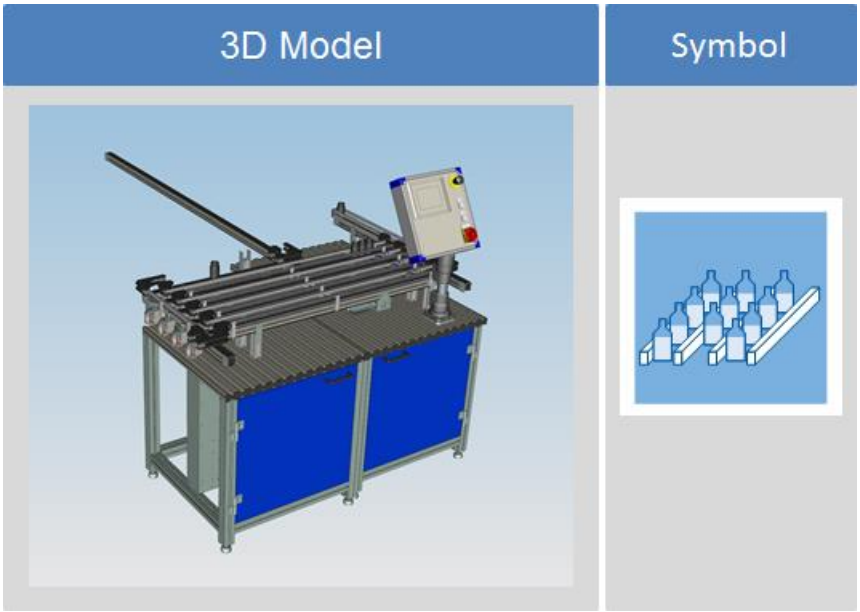


Figure 22 – Commissioning station

This description represents the default operation mode, it was changed though to fit better the objectives in the Backbone Factory, this subject will be explained in section 3.2:.

5. Palletizer Station

The station gets bottles transferred from the Commissioning station and waits until an order is completed. As soon as an order is finished and an empty pallet is left on the Transport station, the gripper arm takes maximum three bottles and places them on the pallet.

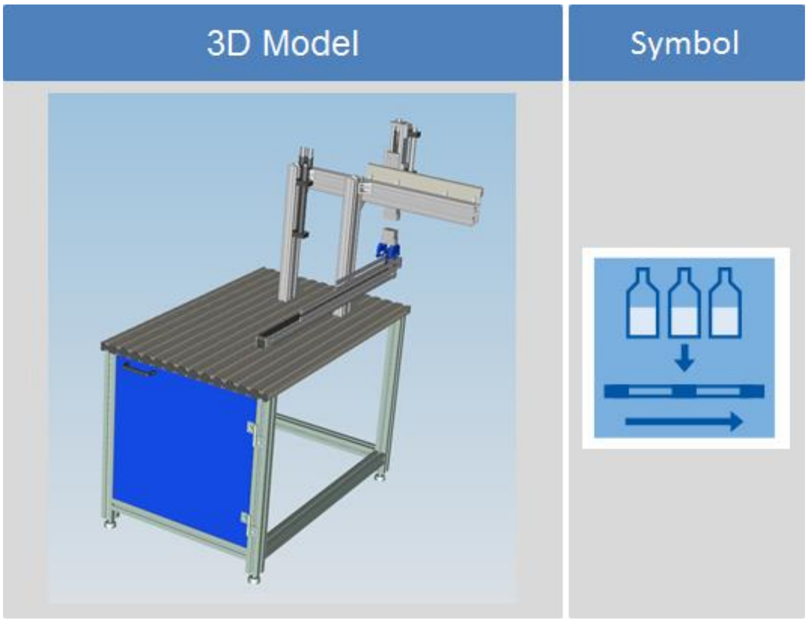


Figure 23 – Palletizer Station

6. Transport Station

Via the Transport station the pallets are transported to the individual station nearby. There are single slides on the line where the pallets are attached. If the machine is turned on, the lines are in constant movement.

To carry out the work progress of the bottles, as demanded in the order, there are RFID readers at all the following stations, which read the saved code located inside the bottle lid.

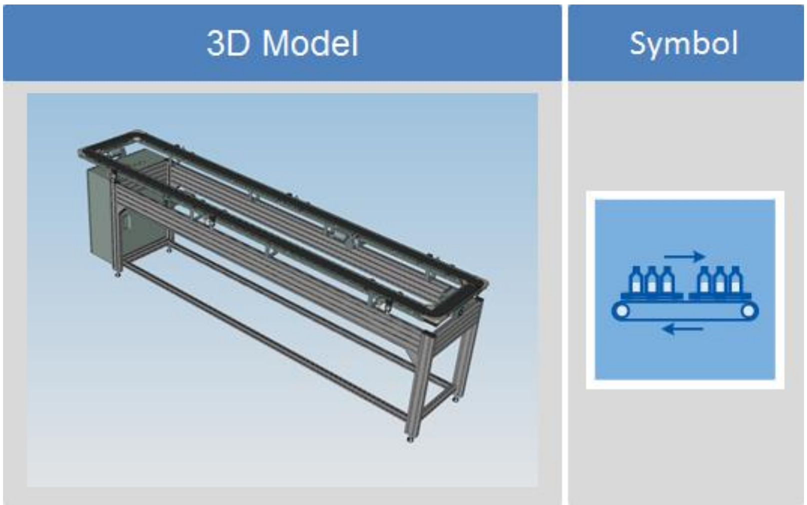


Figure 24 – Transport Station

At the next station, the so called In/Out station, the slides with pallets are unloaded or rather loaded, if the order requires it to.

The following station is the High Bay Racking station. Here the pallets are transferred or rather released from stock. Like in the In/Out station, the order here is also given by the QR-Code of the pallet.

The last station is the De-Palletizer. In this phase, the bottles are lifted from the pallet to be delivered to the recycling station. To register the amount of bottles located on the pallets, there are two inter-stations on the Transport station, where the pallets charge condition is captured by cameras. The process in these inter-stations occurs before the Palletizer and the De-Palletizer.

7. In/Out Station

The In/Out station serves to remove or rather reintroduce pallets. The removal of a pallet happens through a pneumatic gripper arm with three axes, which lays the pallet, depending of its state (i.O/n.i.O), on the roll over ramp. Each ramp has place for maximum four pallets.

In addition pallets can also be reintroduced to the process at this station. A sensor recognizes if a pallet is laid on the stock conveyor. This pallet is then transported by the gripper arm to an already forthcoming slide.

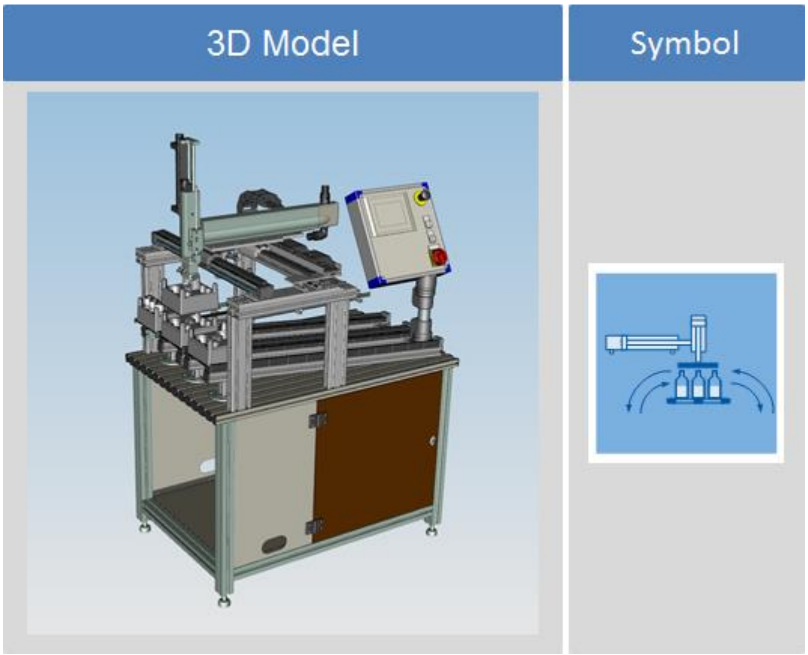


Figure 25 – In/Out Station

8. High Bay Racking Station

The High Bay Racking Station serves for the identification of the content and for the storage. It is possible to store up to 40 pallets, which are taken from the Transport station and also settled back on it. Whether a pallet should be stored, it is provided in the information of the order. This information can be determined by the QR code which is on every single pallet.

If a pallet needs to be stored, the crane goes to the collecting area on the Transport station and gets by means of the telescope arm the pallet on the sledge of the crane. The crane directs the pallet to the stock, where it will be placed, soon after, the crane is ready to receive a new order again. With an outplacement order this process takes place exactly in reverse order.

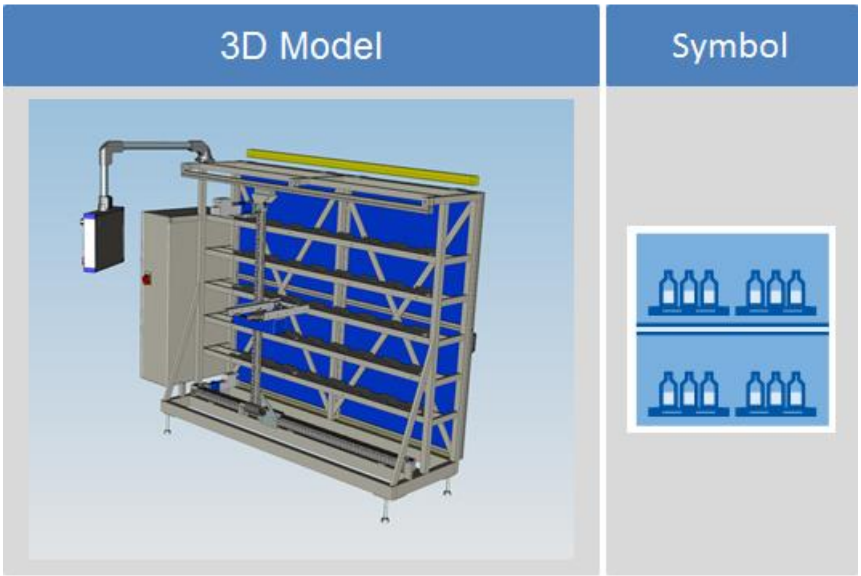


Figure 26 – High Bay Racking Station

9. De-Palletizer Station

The De-Palletizer, after the signal of another component, directs the container from the pallets placed on Transport station to the Recycling station. This process is possible with the help of a gripper arm, which can grab groups of 3 bottles and place them on a conveyor. The conveyor directs the bottles through another line to the Recycling. In addition, there is a separator, whose the function is keeping the correct queue of the bottles from the Commissioning station. An own SPS, with touch panel, controls this process. The gripper arm, named “beforehand”, is controlled pneumatically, while the conveyor, on the other hand, are is controlled electrically.

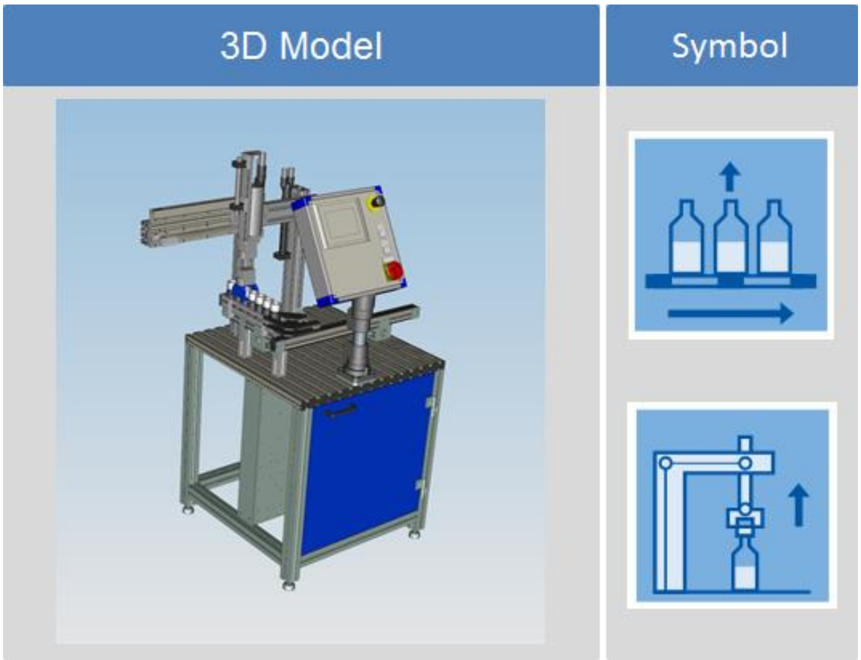


Figure 27 – De-Palletizer Station

10. Recycling Solid Station

A light barrier recognizes if a bottle is queuing to get recycled. If there is a bottle under the RFID, it will be recognized by this sensor and, then, the saved code written inside the lid will be readout. The readout is done in order to know if the bottle is filled with liquid or solid content. Next, an electrical switch is opened and the bottle is directed to the accordingly lid gripper for liquids or solids. Now, by the light barrier, it is identified whether there is another bottle in the queue or not. The lid is taken by the gripper and placed on an extra conveyor belt. If the gripper arm is in the starting position, the lid removal opens and the bottle is send to the bottle gripper, where a sensor also recognizes if the bottles are arranged in the right position. After, the bottle will be emptied by a reservoir, which is located over the rotary plate. The bottle is again placed on the conveyor. As soon as the bottle gripper is back in its starting position, it opens the bottle drain.

The rotary plate for separation starts to work as soon as there are balls inside the reservoir. The color of the balls is recognized by a color sensor. Furthermore, to save the position of these balls and to assort them into the correct recycling bin, the sprockets of the rotary plate are counted by a sensor. The sorting occurs by compressed air, which means that the balls are blown from below into the drum.

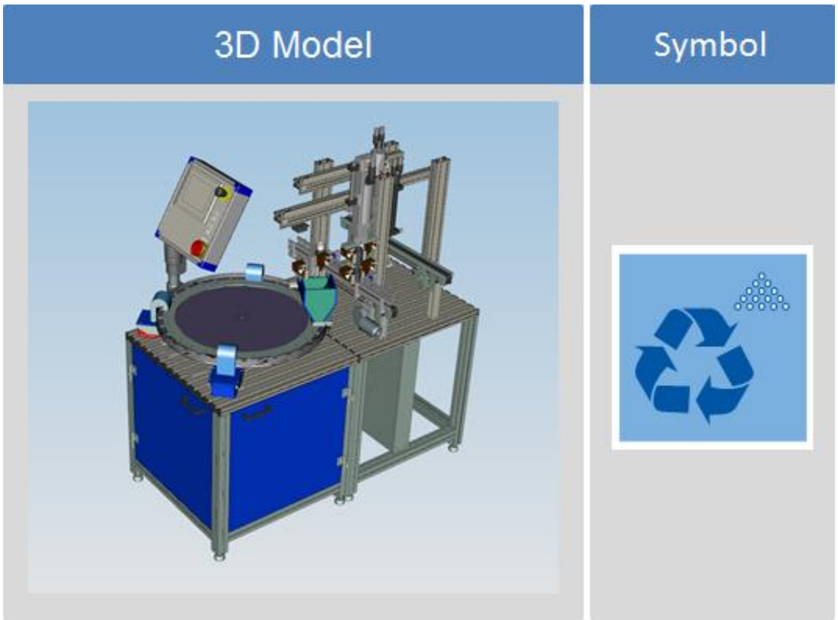


Figure 28 – Recycling Solid Station

11. Recycling Fluid Station

A light barrier recognizes if there is a bottle queuing or not. The lid is taken off by the lid gripper and placed on an extra conveyor. If the gripper arm is located in its starting position, the lid removal opens and the bottle is directed to the evacuator. There is also a sensor that recognizes if the bottle is arranged in the right position. Next, the evacuator is lowered and the liquid is pumped down. Afterwards the bottle drain opens and the bottle is directed to the MPS station.

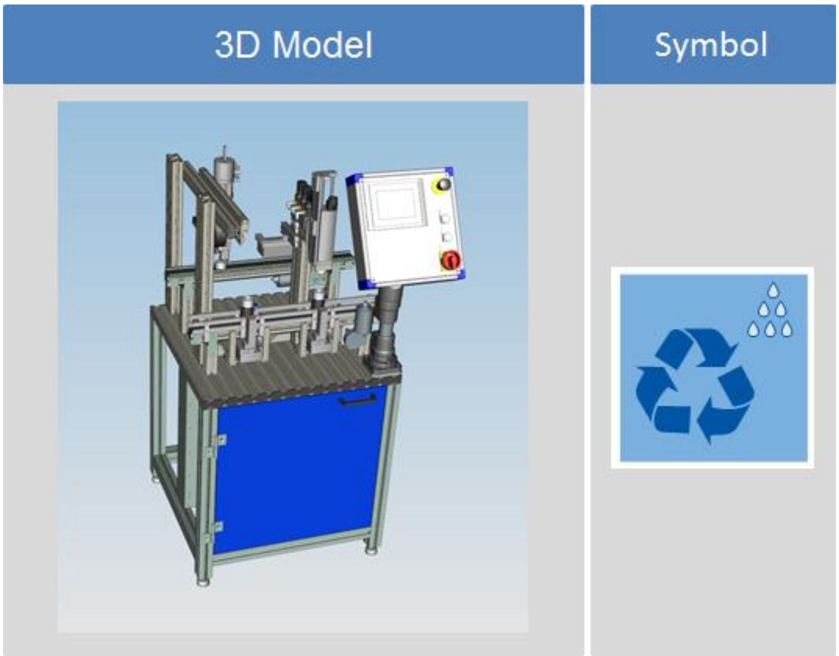


Figure 29 – Recycling Fluid Station

3.2: Commissioning Station

Regarding the matter of this station the hardware and software are more detailed in the following parts, this allows the presentation of sufficient technical data of the material flow (topology, performance), production and structure (layout, manufacturing equipment, transport function) in order to obtain an accurate simulation data.

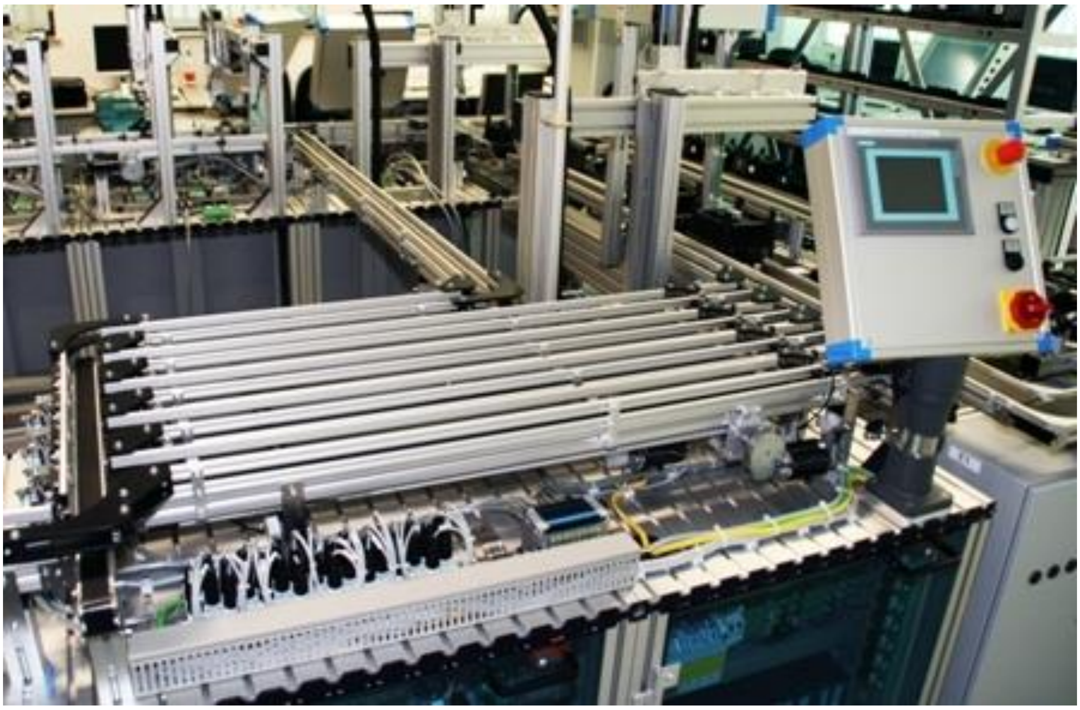


Figure 30 – Commissioning station in the Backbone Factory

3.2.1: Hardware

The hardware in this is station is mainly built with profiles, conveyors belt, motors, optical sensor and pneumatic cylinders. Besides it is equipped with:

- Power supply: PS 307 10A
- CPU: SIMATIC S7-300 CPU 319-3 PN/DP
- Digital input module: DI16xDC24V
- Digital output module: DO16xDC24V/0,5A
- Communication module: CP 343-2 P
- Touch panel: TP 277 6"

The following pictures and table will summarized just the hardware which have a functionality in the virtualization, since they change the station's state and don't have just a structural function.

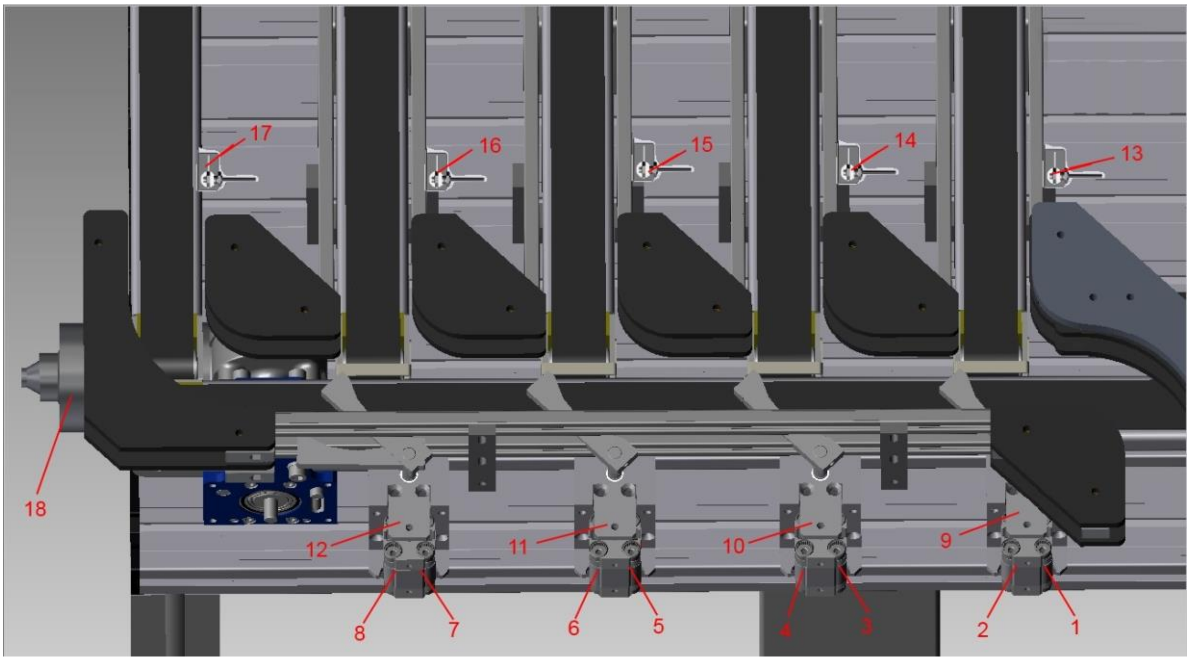


Figure 31 – Start view from Commissioning Station. Source [16]

The figure above shows almost all the items which compose this station. One of them is the Festo compact ADN pneumatic cylinder located in 9, 10, 11 and 12. This cylinder has a rail coupled on it to address the bottle to the respective line and it has a mode of Double-acting operation with a Default Position (DP) and an Active Position (AP). In DP there is no barrier and, in the AP, the rail makes a barrier, so the bottle moving through the line from Quality gate station can be direct to the respective line.

To indicate the AP and DP there are 2 sensors in each pneumatic cylinder, so they can transfer this information to the PLC. A simply diagram can be seen in the Figure 32. The

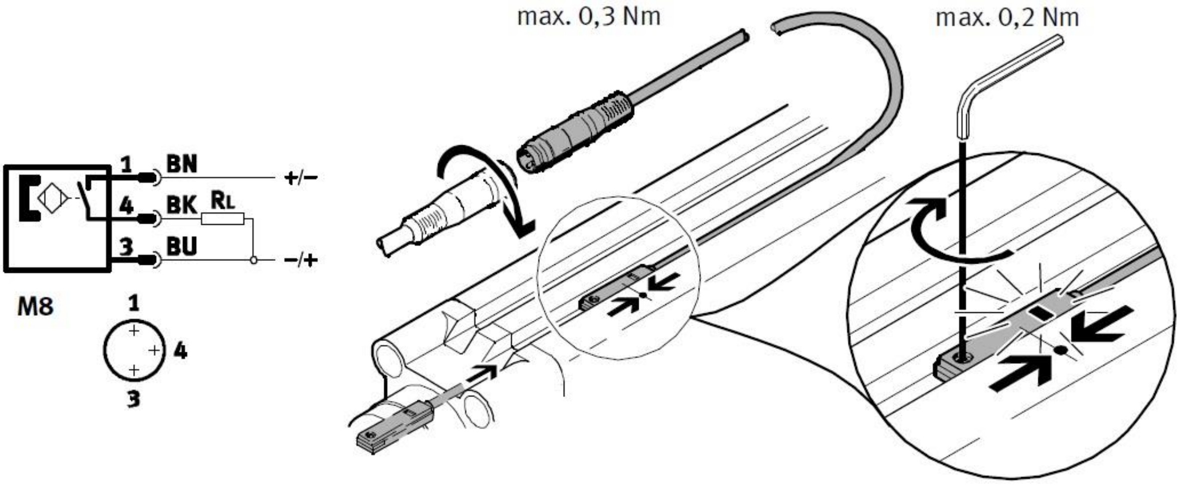


Figure 32 – Electric and mounting diagram from sensor SME-8-S-LED-24

The 36V-DC-Motors showed in Figure 33 are used to move the lines while the speed is controlled by the PLC in this station.

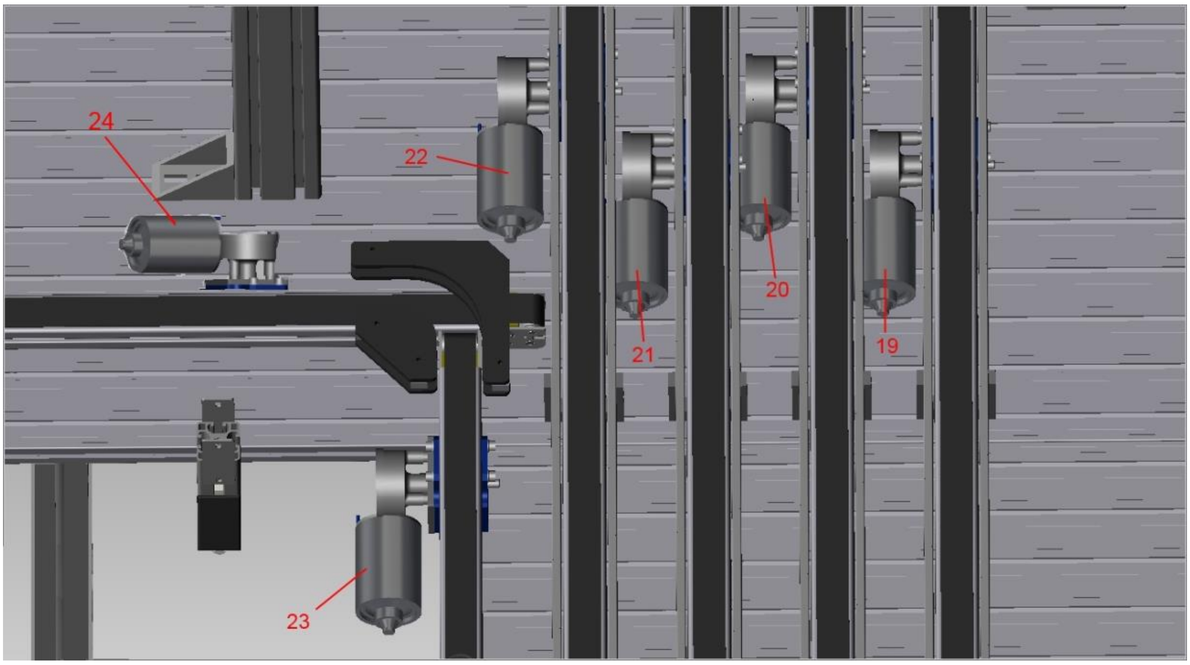


Figure 33 - Middle view from Commissioning Station

The Figure 34 shows a different kind of cylinder in 33, 34, 35 and 36. This cylinder has a Single-acting operation mode and, therefore, just the DP, which also has a coupled sensor to identify this position. There is a barrier when this cylinder is at DP, so the bottles will be stopped and the optical sensor in front of the cylinder will be set. When the cylinder is activated it leads just one bottle to the next part and when the optical sensor is no longer activated, the cylinder comes back to DP. The mechanical design of the rail just allows the passage of 1 bottle per time, because when the cylinder is not in the DP the next bottle is stuck in its rail.

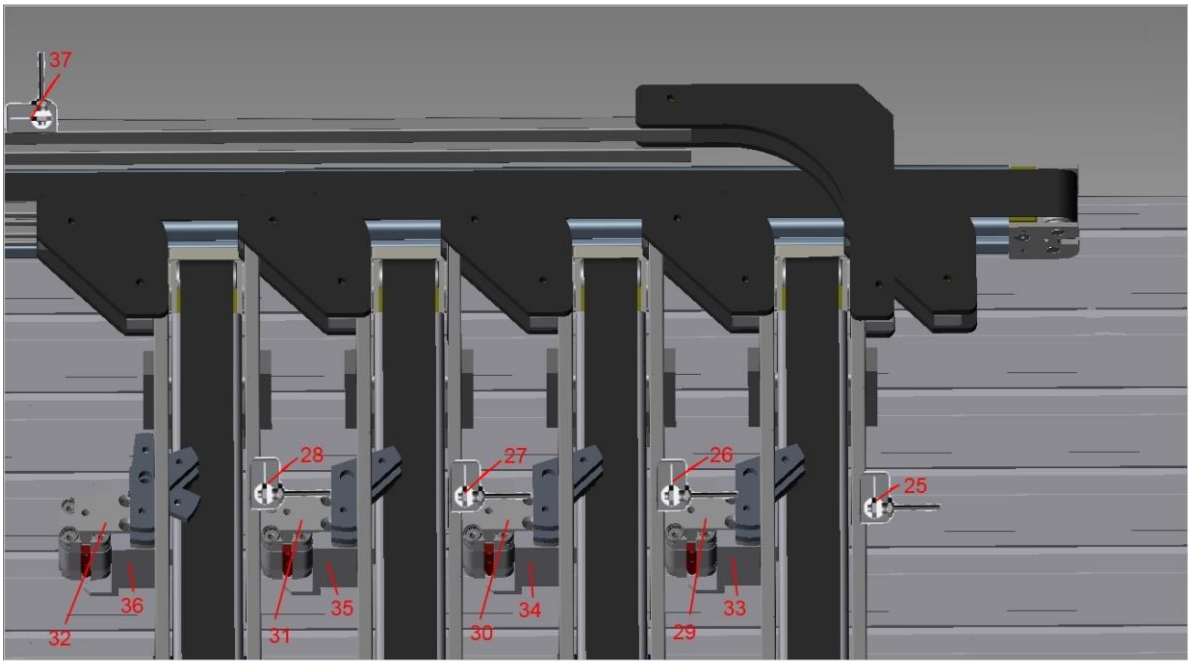


Figure 34 - End view from Commissioning Station

The table below shows the numbered items from the pictures above, with a short description, function and where they are connected in the PLC (Address starting with E means Input and starting with A means Output).

Table 1 – Hardware description from Commissioning Station

Number	Description	PLC Address	Function
1	SME-8-S-LED-24	E304.0	Sensor 1 for DP
2	SME-8-S-LED-24	E304.1	Sensor 1 for AP
3	SME-8-S-LED-24	E304.2	Sensor 2 for DP
4	SME-8-S-LED-24	E304.3	Sensor 2 for AP
5	SME-8-S-LED-24	E305.4	Sensor 3 for DP
6	SME-8-S-LED-24	E305.5	Sensor 3 for AP
7	SME-8-S-LED-24	E305.6	Sensor 4 for DP
8	SME-8-S-LED-24	E305.7	Sensor 4 for AP
9	Pneumatic Cylinder	A309.0	Commissioning Line 1
10	Pneumatic Cylinder	A309.1	Commissioning Line 2

11	Pneumatic Cylinder	A309.2	Commissioning Line 3
12	Pneumatic Cylinder	A309.3	Commissioning Line 4
13	Optical Sensor	E305.0	Counter Line 1
14	Optical Sensor	E305.1	Counter Line 2
15	Optical Sensor	E305.2	Counter Line 3
16	Optical Sensor	E305.3	Counter Line 4
17	Optical Sensor	E306.4	Counter Line 5
18	36V-DC-Motor	A4.5	Line Quality Gate
19	36V-DC-Motor	A4.6	Line 1
20	36V-DC-Motor	A4.7	Line 2
21	36V-DC-Motor	A5.0	Line 3
22	36V-DC-Motor	A5.1	Line 4
23	36V-DC-Motor	A5.2	Line 5
24	36V-DC-Motor	A5.3	Line De-Palletizer
25	Optical Sensor	E306.0	Bottle End Line 1
26	Optical Sensor	E306.1	Bottle End Line 2
27	Optical Sensor	E306.2	Bottle End Line 3
28	Optical Sensor	E306.3	Bottle End Line 4
29	SME-8-S-LED-24	E307.4	Sensor 1 for DP
30	SME-8-S-LED-24	E307.5	Sensor 2 for DP
31	SME-8-S-LED-24	E307.6	Sensor 3 for DP
32	SME-8-S-LED-24	E307.7	Sensor 4 for DP
33	Pneumatic Cylinder	A310.4	Separator Line 1
34	Pneumatic Cylinder	A310.5	Separator Line 2
35	Pneumatic Cylinder	A310.6	Separator Line 3
36	Pneumatic Cylinder	A310.7	Separator Line 4
37	Optical Sensor	E306.5	Counter De-Palletizer

3.2.2: Software

This section is most important to further understand how the station operates in a deeper level; besides, this study will support afterwards the path to build the model for the virtualization.

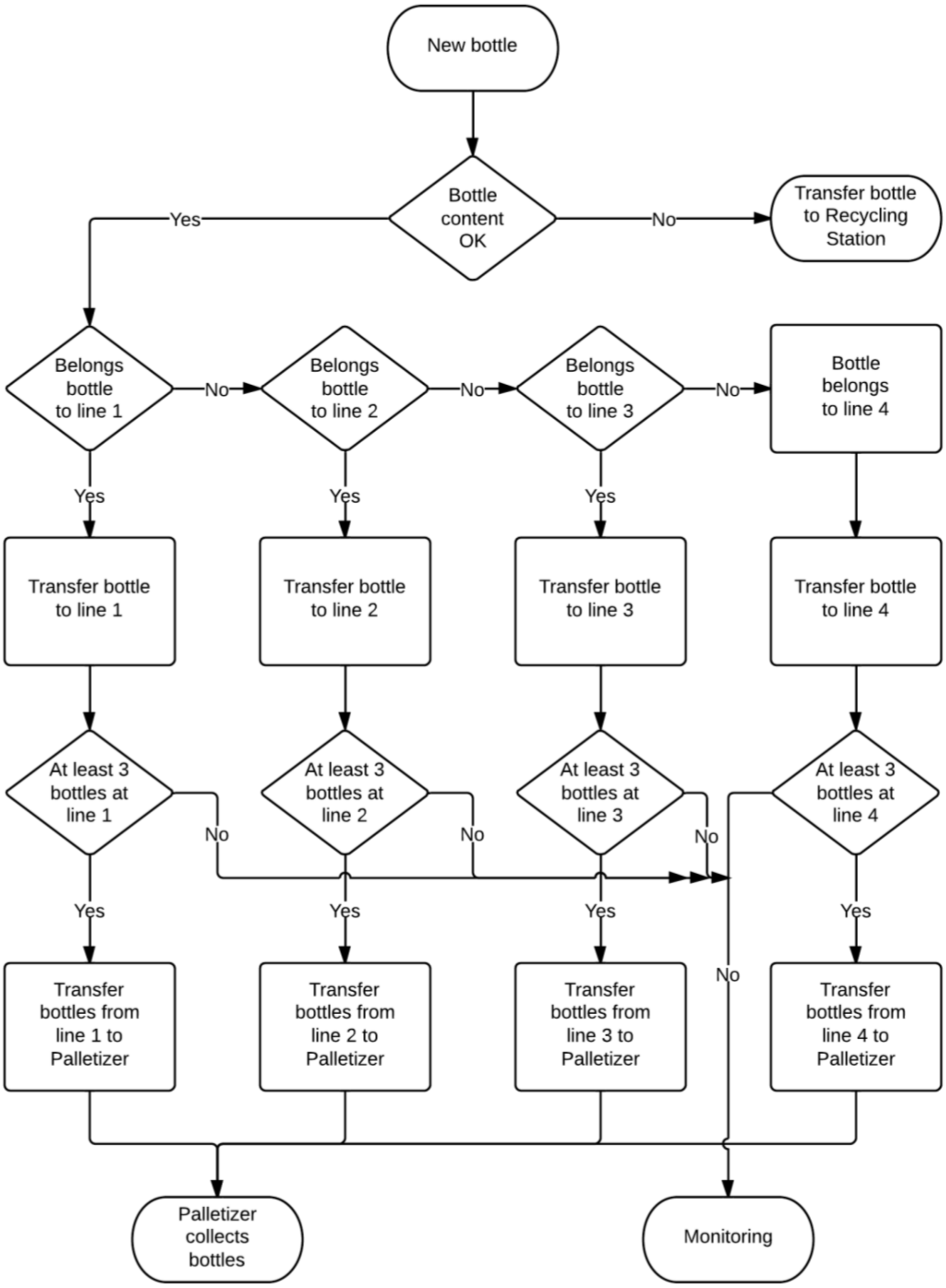

New bottle
Bottle content OK
Yes
No
Transfer bottle to Recycling Station
Belongs bottle to line 1
Belongs bottle to line 2
Belongs bottle to line 3
Bottle belongs to line 4
Transfer bottle to line 1
Transfer bottle to line 2
Transfer bottle to line 3
Transfer bottle to line 4
At least 3 bottles at line 1
At least 3 bottles at line 2
At least 3 bottles at line 3
At least 3 bottles at line 4
Transfer bottles from line 1 to Palletizer
Transfer bottles from line 2 to Palletizer
Transfer bottles from line 3 to Palletizer
Transfer bottles from line 4 to Palletizer
Palletizer collects bottles
Monitoring

Figure 35 – Commissioning station’s flowchart

In the Figure 35, the flowchart demonstrates, in a high level, the route for each bottle. Everything starts when a new bottle comes from Quality Gate. If the bottle content is not in proper conditions, it goes directly to the recycling, passing through line 5, De-Palletizer and then Recycling station.

If the content is appropriate the Commissioning station sorts the item according to its line and, thereafter, the bottle is counted by the first sensor in the line, for example, number 13 showed in Table 1. It keeps going until reaches another bottle stopped in the line or the cylinder rail, where it stays as far as 3 bottles are in the same line to, then, be transferred to the Palletizer.

In the Backbone Factory the content in the bottle are caps that were put by the Bottling station and they have different colors like blue, green, yellow and red. Each bottle has a sort of data carried in its RFID history, which is complemented by each station. Usually each bottle has data like ID, Color (according to its caps), State (solid or fluid), Target Place (station), Status (OK or not OK) and amount (milliliter for fluid, for example).

As the purpose of the machine is study and development, many changes were taken to provide a better scenario and make the stations work independently from each one. For that reason some functionalities are still in development and tasks were made simple with this change to be improved gradually. Therefore, the data for each bottle is just the color and status. Another part of the software is dedicated to HMI and is located also in the PLC as described in section 2.3:. This was not changed as mentioned above but the navigation through screens was limited but the user can still see the operational status from the Commissioning station though.

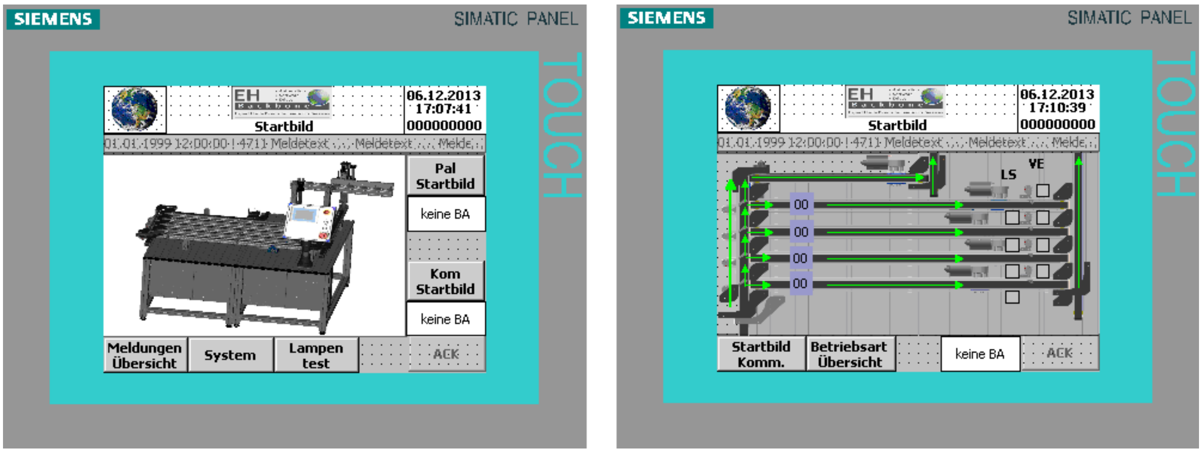


Figure 36 – WinCC screenshot from Commissioning station

Chapter 4: Implementation

This chapter is dedicated to approach the implementation of the work described in previous chapters. The virtualization process in the Commissioning station will be divided in four sections.

In the first section the focus is presenting the planning phase for modelling in 2D and the steps to implement it using the software Tecnomatix.

In the second section it will be imported the CAD files of the Commissioning station to work according the model built in 2D. Here it will be possible to visualize the real flow of the machine, where bottles, sensors and actuators take part of the scenario 3D.

In the third section the model 3D built in the second section will work in a standalone version, which means every action in the station will need to be control by an external agent and every status will be also communicated.

The last section is used to explain the mapping from the 3D model with Step7 through OPC interface. Here, the complete model will be controlled through a simulated PLC and monitored by the HMI Panel in WinCC.

4.1: 2D Model

Before the study was done in section 3.2: to understand the parts of the machine and how it operates, the work got started with the planning phase. This section will be the base also for the 3D model, since the 2D model is responsible to control the actions and status visualized in the 3D model.

According to the State of Art in section 2.1.3:, it is recommended to follow the procedures bellow for simulation:

1. **Formulations of problems**

The main objectives of this work were already presented in section 1.3:, where it is desired a high-resolution 3D model to make the complete virtualization of the Commissioning station.

Since many projects are running in parallel the idea of virtualization can bring more employees to work in the Backbone Factory, so they can work independently in different computers. In this regard, the software used in this work are required in each computer.

The model must emulate the station as close as possible of the reality, considering, of course, the limitations of the software Tecnomatix and CAD files. All inputs and outputs used to control and check status of the station need to be found in this model, so an external user can manage them to make tests in his software.

2. Test of the simulation-worthiness

It was examined the inaccurate data and repeated use of the simulation model in order to optimize the model.

3. Formulation of targets

This procedure was made for a simulation process, but it can be used to base the virtualization. This section actually is defined by the user, which will run the model and, then, decide its targets to, thereafter, collect and analyze them statistically.

4. Data collection and data analysis

The Figure 5 shows that the data simulation can be divided in 3 sections: system load, organizational and technical data. For the virtualization, it is needed to focus just on the technical data since, the other two are related to a strategic level.

Unfortunately, not all the data are available like functional breakdowns, performance data, utilization time and availabilities related to production and failure data. However, data of factory structure and material flow were discriminated to fill in some option in the model built in Tecnomatix (system topology, transport routes and conveying systems). These data can be seen and modified by accessing the model in Tecnomatix and selecting the respective features.

5. Modeling

A general understanding of the simulated system was developed through some versions in the beginning and, over time, it was possible to manage how accurate the model could be without disagree with the targets of the problem in the first place.

With the aid of system analysis, the complexity of the system was resolved by breaking it down into its individual elements like lines, bottles, actuators and sensors. Therefore it was chosen a Bottom-up approach in which incrementally synthesizing the whole by proceeding from the detail, that means detailing as close as possible each part of the elements in the station to get, at the end of the process, a model by integrating its parts.

With the results from simulation, it was possible to raise questions about the accuracy of the model and fixing it according to the simplification of indispensable details and also omission of unimportant details.

6. **Execute simulation runs**

As Plant Simulation is a discrete event-controlled simulation program, it only inspects those points in time, at which events take place within the simulation model so it's not possible to define a time span for the simulation experiments.

7. **Result analysis and result interpretation**

After passing these steps the following model in Figure 37 was built in Tecnomatix to virtualize the real scenario. This version is not standalone, so the model (Blue Square) controls itself using methods and variables (Yellow Square). The Event Controller (Green Square) defines when the next event occurs according to the parameters of the methods. This model will be improved in the next sections.

The stations around Commissioning station are represented with source and drain blocks. The bottles have different colors and are produced randomly by the Quality Gate station. They go through the Line (L_QG) until reach the respective pneumatic cylinder (Red Streak) which will direct to the next line (L1, L2, L3, L4). If the bottle has a black color, the content is not a proper one and it goes to Recycling station passing through De-Palletizer station.

In the lines L1-L4 there are 2 red streaks, the first one represents the optical sensor for counting and the second one the pneumatic cylinder that makes the packing with 3 bottles.

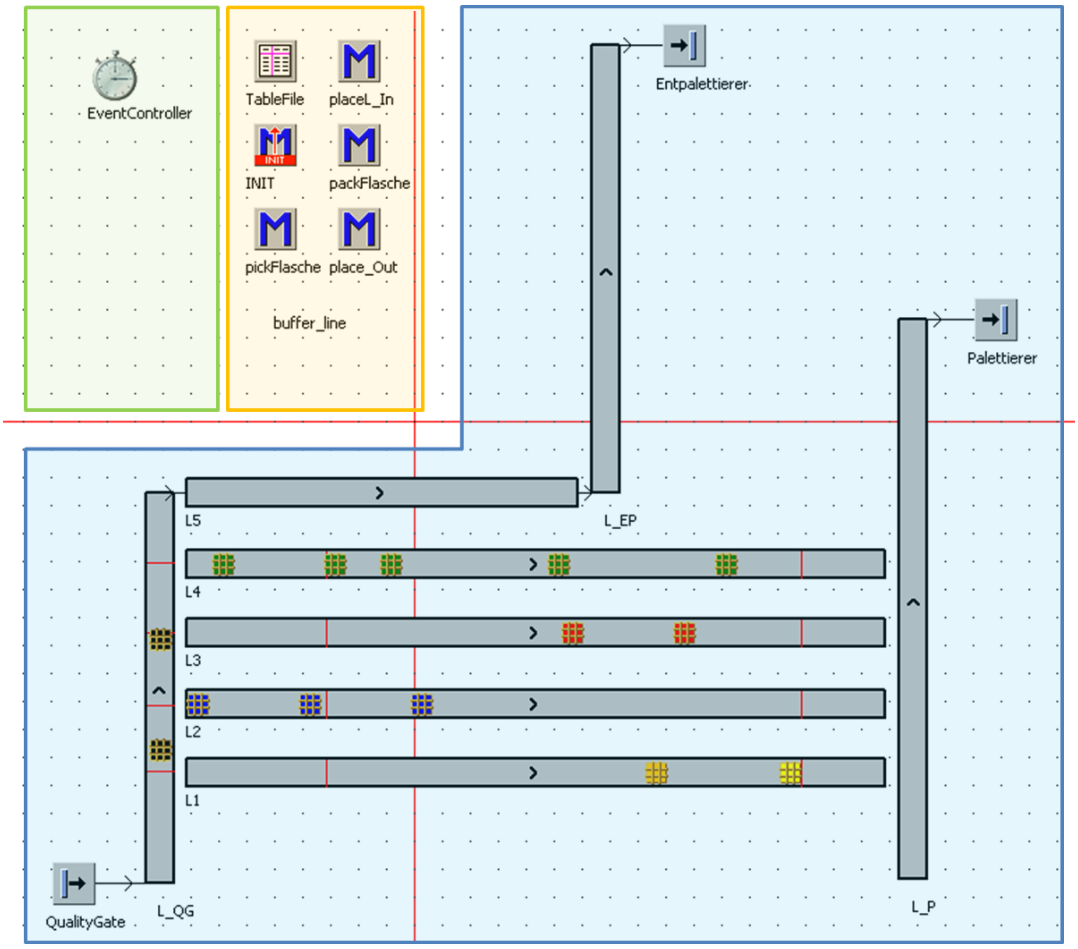


Figure 37 – Model 2D with Tecnomatix

This model helps to understand a little bit better how the station works but it has to be improved since not all the hardware are specified like objects, so it is not possible to control and check status from them with the methods available in Tecnomatix.

8. Documentation

This paper represents itself as the main documentation for this work, providing an overview of the timing of the study and documenting the work carried out with its structure and its functionality.

Other documents such as videos, presentation, software, codes and literature are also dedicated to this topic and will be delivered at the end of this work.

4.2: 3D Model

In this section it will be present just the point 7 (Result analysis and result interpretation) of the simulation procedure discussed in the previous section. The other points are the same for this work.

The figure bellow shows the 3D model built in Tecnomatix where the CAD file was imported and assigned to each object in the 2D model. Tecnomatix has a default 3D graphic for each object and it needed to be changed to fulfill the specifications of the CAD file and represent the real installation.

In the figure bellow it's possible to see the bottles being carried out through the lines according to their color, although it is not possible to see in the 3D model. They run according to 2D model and the methods in Figure 37, so if a change is made on the 2D model it reflects on the 3D model.

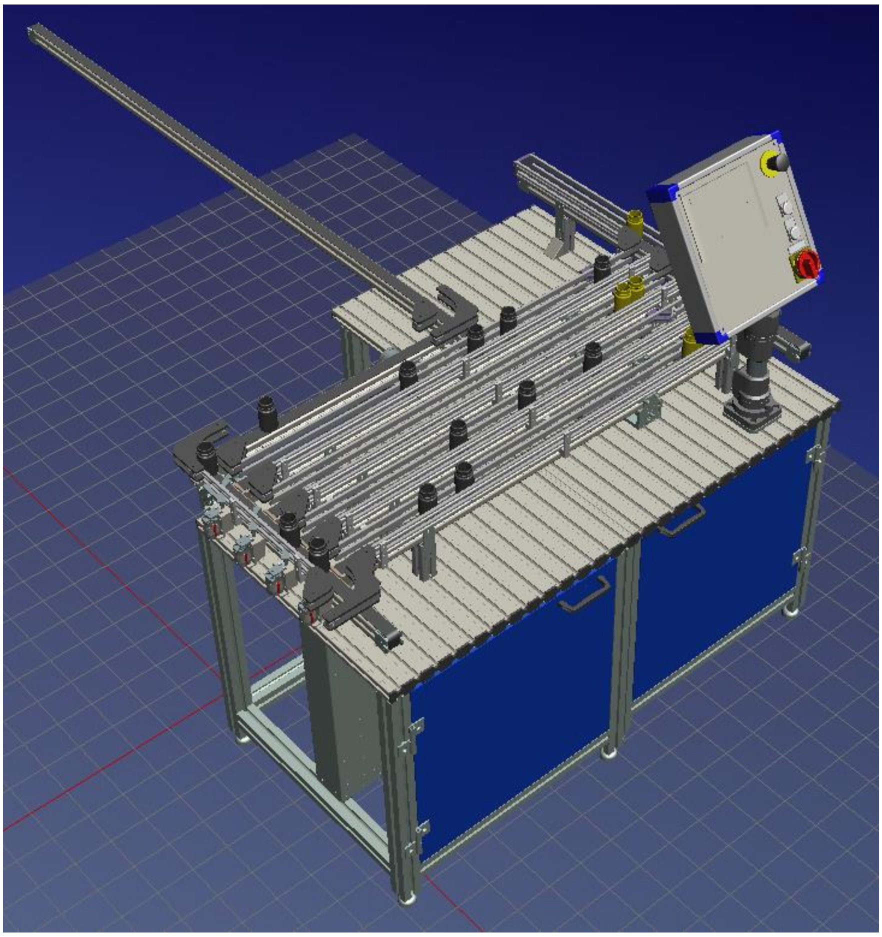


Figure 38 – Model 3D with Tecnomatix

The CAD file for each object was isolated and configured so it could represent the real flow in the machine. As it was described in Table 1 each item has its own graphic, so the movement from pneumatic cylinders, motors and check status from the optical sensors can be configured, for instance.

The Figure 39 shows an example of the configuration of the Line L1. On the left side, it is the default Tecnomatix CAD for the 3D model according to the 2D model in Figure 37. This object needed to be changed by the object on the right side that represents the real line in the Commissioning station. It was possible for example to configure the path where the bottle would follow and also animate it.

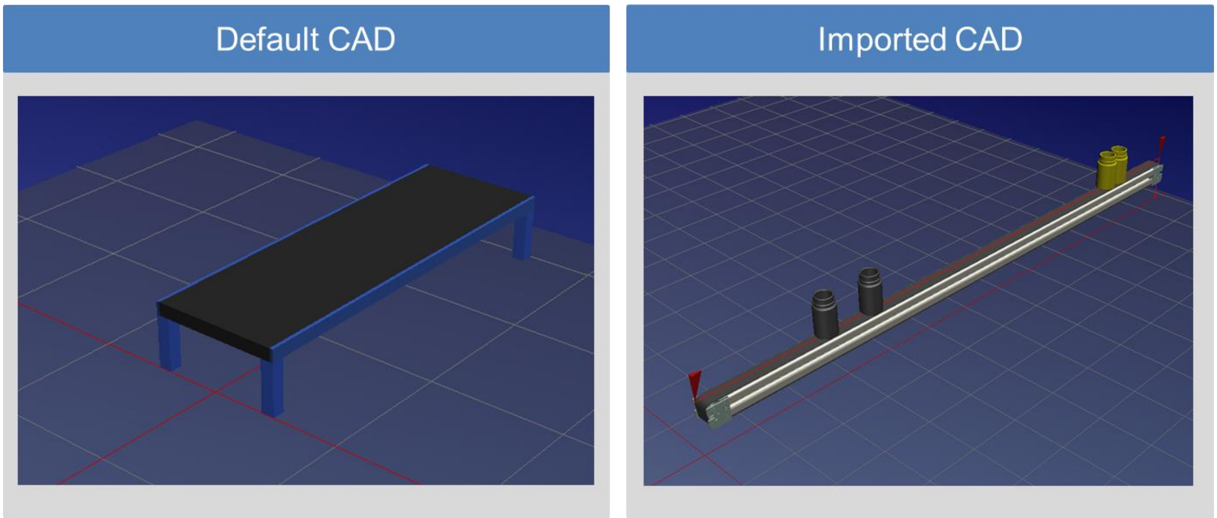


Figure 39 – Import CAD file

This process was made on the entire station for each hardware on the 2D model. Later, the other parts were imported, the ones which do not participate actively on the simulation, but have a structural function.

In general, only a limited selection of objects is available in Tecnomatix to represent the real installation, and the basic behavior provided by the objects in practice is not sufficient to generate realistic system models.

To extend this object limitation, Plant Simulation offer the programming object-oriented language called SimTalk. With the SimTalk, it is possible to modify the basic behavior of individual objects by controlling structures and language constructs (conditions, loops) and by using the standard methods, which are built-in and have basics functionalities.

For that reason it was needed to program methods using SimTalk to simulate with better accuracy what happens in the real installation; an example of that would be if a bottle reaches a barrier it has to stop or be transferred to another line.

4.3: Standalone Version

Many improvements were done in the previous versions of the program to better cope with the specifications of the work. The main goal here was to model the components of the station so they could represent with accuracy the real installation.

The Figure 40 shows some changes in the model such as:

- Methods: A Generator block which calls a method in regular distributed intervals, in this case the method Check Status that is responsible to update the inputs and outputs in the model (Yellow Square). Also individual methods were created to model the actions in the model like controlling the movement from the separator and commissioner.
- Hardware: each hardware described in Table 1 was model as an object so it could be accessed by a user or PLC. This enabled the possibility to simulate the movement in the 3D model and to access individually each object. The separator (Vereinzeler V1-V4) and commissioner (Weiche W1-W4) were modeled as a Pick-and-Place robot, so it could perform with better accuracy the movement from the pneumatic cylinder (Blue Square). Sensors were also modeled with a bottom, where the user can simulate a bottle passing through it, although the sensor is activated automatically if a bottle passes by.
- Checkbox: several checkboxes were created to individualize the command for each object in the model. For example if the check box L_QG is activated, then the line L_QG is turned on, otherwise, it is turned off. This will help latter the mapping with OPC because each checkbox will be assigned with the respective input and output in the PLC (Green Square).
- 3D Model: some changes were also done in the 3D model to make it lighter, so it could run faster in a normal computer. This was achieved by grouping some hardware that had just structural function. The resolution of the model was also increased with the new version of the Tecnomatix and some objects now have their movements in action.

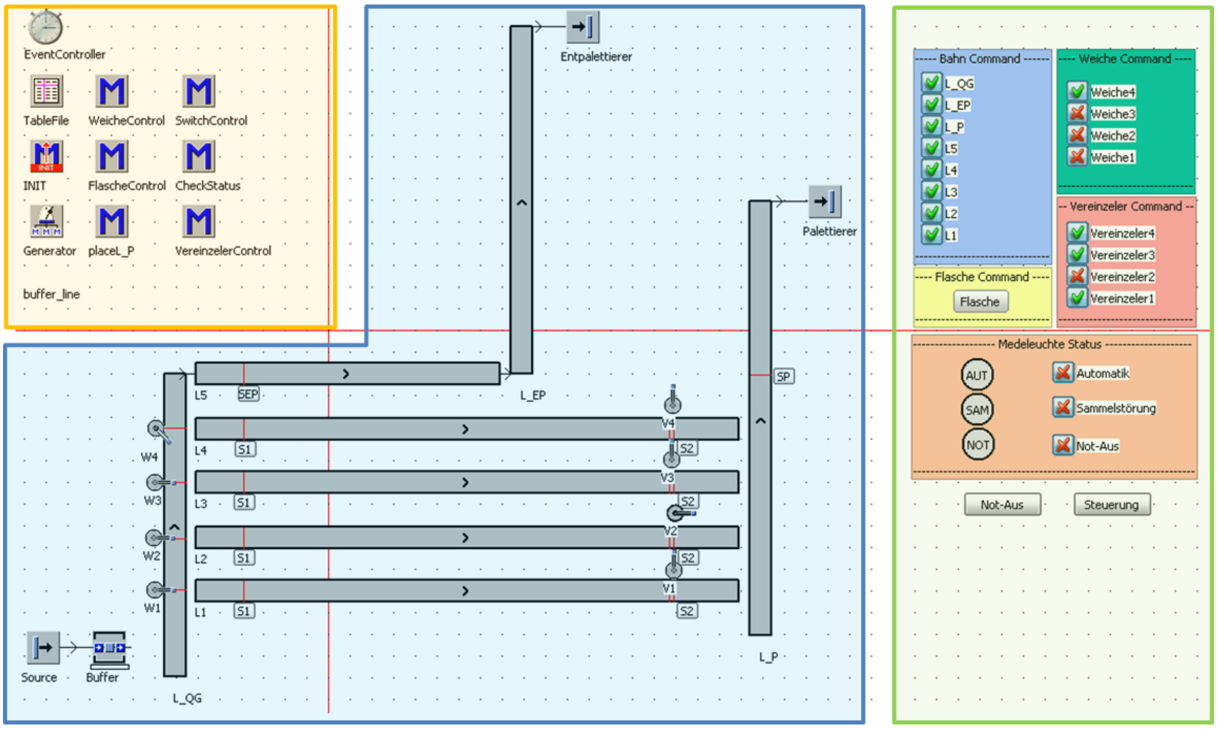


Figure 40 – Final version model 2D Tecnomatix

The purpose of the standalone version is to shape the model so it could represent the real installation, where it would match the variables inside the program with the real PLC address.

The model should be capable of reading the output of the simulated PLC, interpret the signal and change the correspondent variable inside the program. Then the program will answer according to this new scenario and update the 2D/3D model, and it will also change its outputs that are connected to the PLC inputs. This way the model no longer controls itself but just change its scenarios according to the commands from the PLC or user.

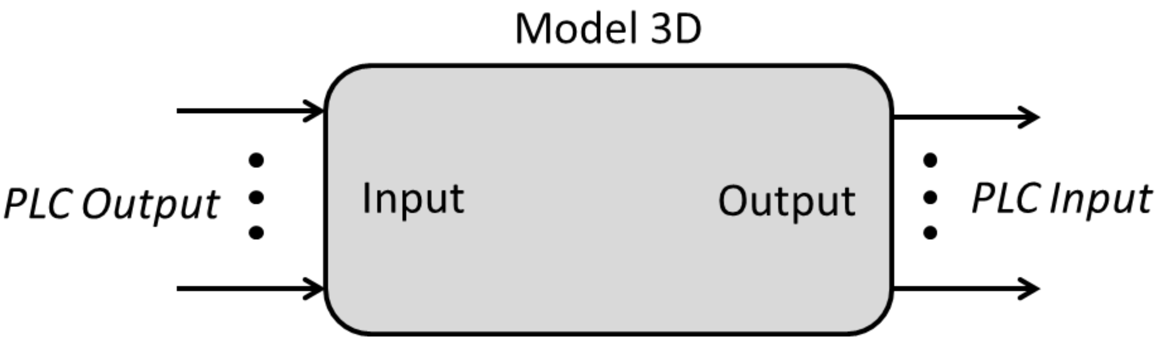


Figure 41 – Input and output model 3D schema

4.4: Mapping with OPC

This section is dedicated to explain the link between Tecnomatix Plant Simulation and the S7-PLCSIM in connection with WinCC Flexible throughout an OPC server. This communication interface was selected because of the fact that WinCC is capable of adapting the integrated OPC server functionality and that Tecnomatix has an OPC client already to be configured in its software.

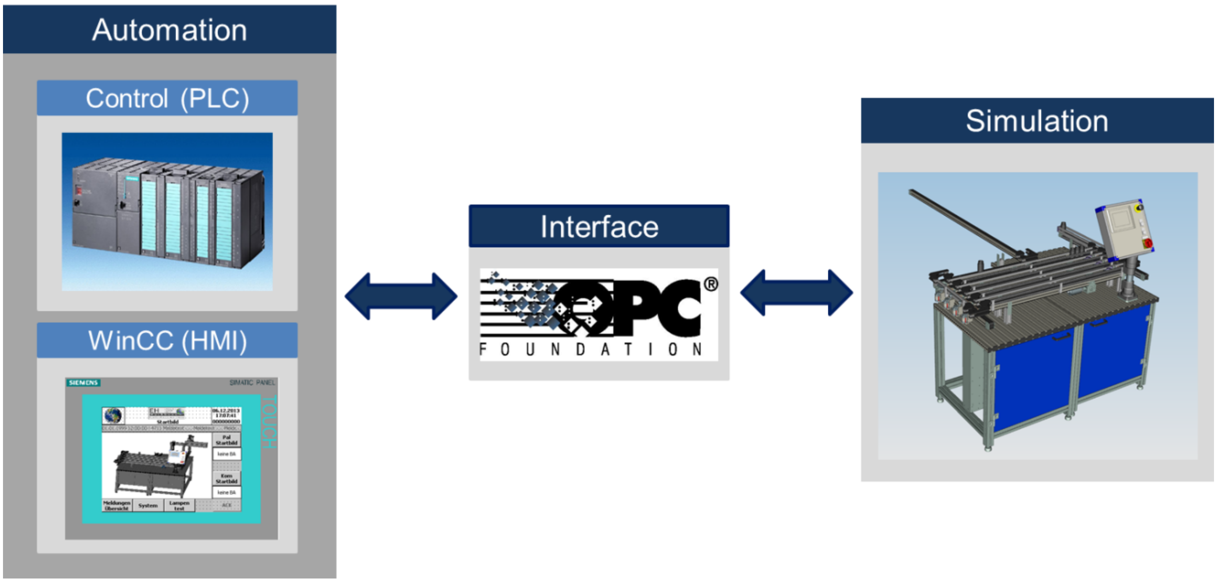


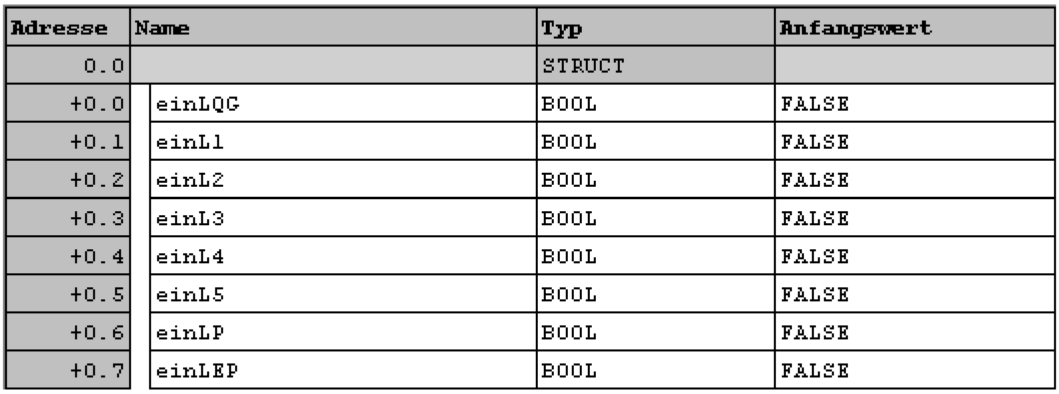
Figure 42 – OPC interface explanation

The user must have the software OPC Demo Client and set some parameters to make the link between the automation and simulation according to the guide [18].

In general, the Figure 42 shows how the linkage between the simulated model and the control part works. In the automation area, the real PLC is simulated by the S7-PLCSIM, and the HMI panel by the WinCC Flexible. Here, the command flow starts in the PLC passing through WinCC and OPC, until it reaches the simulation area (Tecnomatix). After receiving this command, Tecnomatix will update its variables and send a message back to the PLC with the new status.

To make that happens, every input and output from the Step7 and WinCC program must be listed and configured in the OPC according to its type. The OPC will call a specific method in Tecnomatix every time a change is made on those variables, so they could be updated. Therefore a mapping with OPC was done to couple those variables.

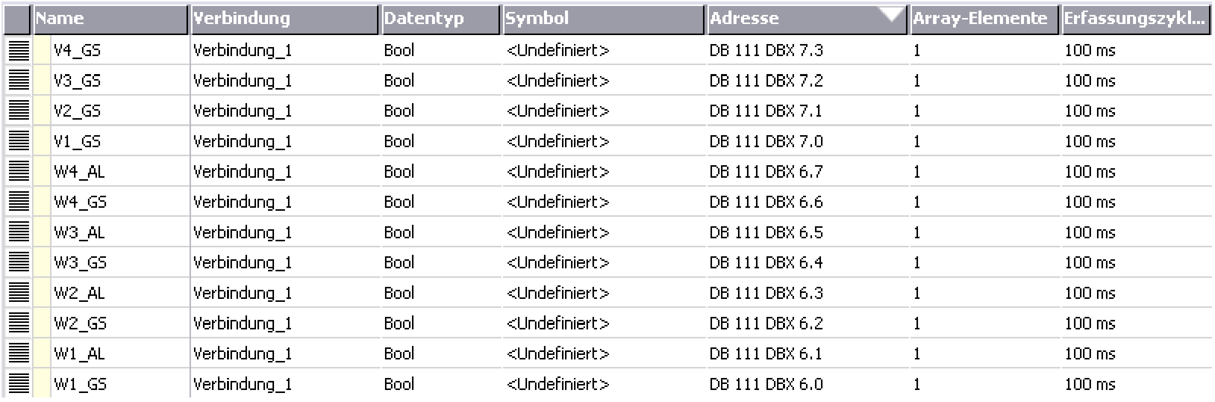
First, in the Step7 program, a data block for communication needs to be created, in which all relevant variables for the simulation under Tecnomatix will be configured.



Adresse	Name	Typ	Anfangswert
0.0		STRUCT	
+0.0	einLQG	BOOL	FALSE
+0.1	einL1	BOOL	FALSE
+0.2	einL2	BOOL	FALSE
+0.3	einL3	BOOL	FALSE
+0.4	einL4	BOOL	FALSE
+0.5	einL5	BOOL	FALSE
+0.6	einLP	BOOL	FALSE
+0.7	einLEP	BOOL	FALSE

Figure 43 – Communication variables in Step7 program

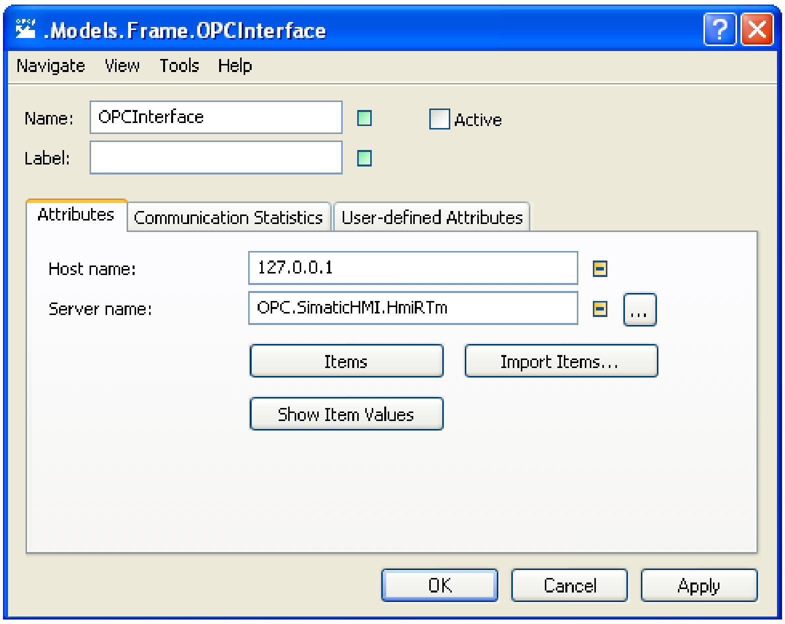
The next step is to add these variables from Step7 in the WinCC program. Thereunto each one is specified by its type, address, connection and cycle time to update it. They are also used to monitor the station in the interface built in WinCC, which later can be loaded in a real panel.



Name	Verbindung	Datentyp	Symbol	Adresse	Array-Elemente	Erfassungszykl...
V4_GS	Verbindung_1	Bool	<Undefiniert>	DB 111 DBX 7.3	1	100 ms
V3_GS	Verbindung_1	Bool	<Undefiniert>	DB 111 DBX 7.2	1	100 ms
V2_GS	Verbindung_1	Bool	<Undefiniert>	DB 111 DBX 7.1	1	100 ms
V1_GS	Verbindung_1	Bool	<Undefiniert>	DB 111 DBX 7.0	1	100 ms
W4_AL	Verbindung_1	Bool	<Undefiniert>	DB 111 DBX 6.7	1	100 ms
W4_GS	Verbindung_1	Bool	<Undefiniert>	DB 111 DBX 6.6	1	100 ms
W3_AL	Verbindung_1	Bool	<Undefiniert>	DB 111 DBX 6.5	1	100 ms
W3_GS	Verbindung_1	Bool	<Undefiniert>	DB 111 DBX 6.4	1	100 ms
W2_AL	Verbindung_1	Bool	<Undefiniert>	DB 111 DBX 6.3	1	100 ms
W2_GS	Verbindung_1	Bool	<Undefiniert>	DB 111 DBX 6.2	1	100 ms
W1_AL	Verbindung_1	Bool	<Undefiniert>	DB 111 DBX 6.1	1	100 ms
W1_GS	Verbindung_1	Bool	<Undefiniert>	DB 111 DBX 6.0	1	100 ms

Figure 44 – Communication variables in WinCC program

In Tecnomatix some parameters need to be set by using the toolbox OPCInterface showed in Figure 45. The first step is to configure the OPC-Client where some fields like hostname (127.0.0.1 or local host) and server name need to be filled. The OPC.SimaticHMI.HmiRTm calls the name of the WinCC-Runtime-OPC-Servers.



.Models.Frame.OPCInterface

Name: OPCInterface

Label:

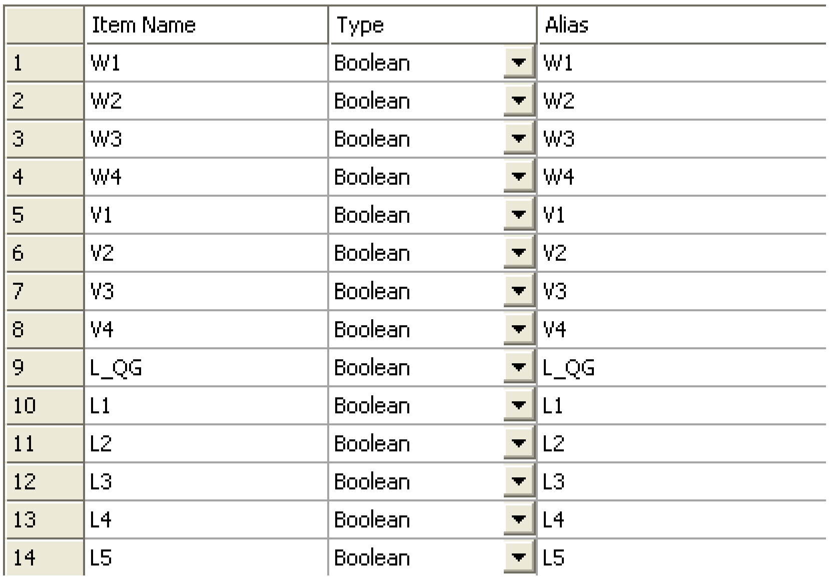
□ Active

Host name: 127.0.0.1

Server name: OPC.SimaticHMI.HmiRTm

Figure 45 – OPC-Client configuration

Later, the variables from Tecnomatix need to be mapped with the variables from WinCC like the figure bellow shows. These variables now can be accessed and updated by commands using SimTalk and they control the objects present in the 2D model.

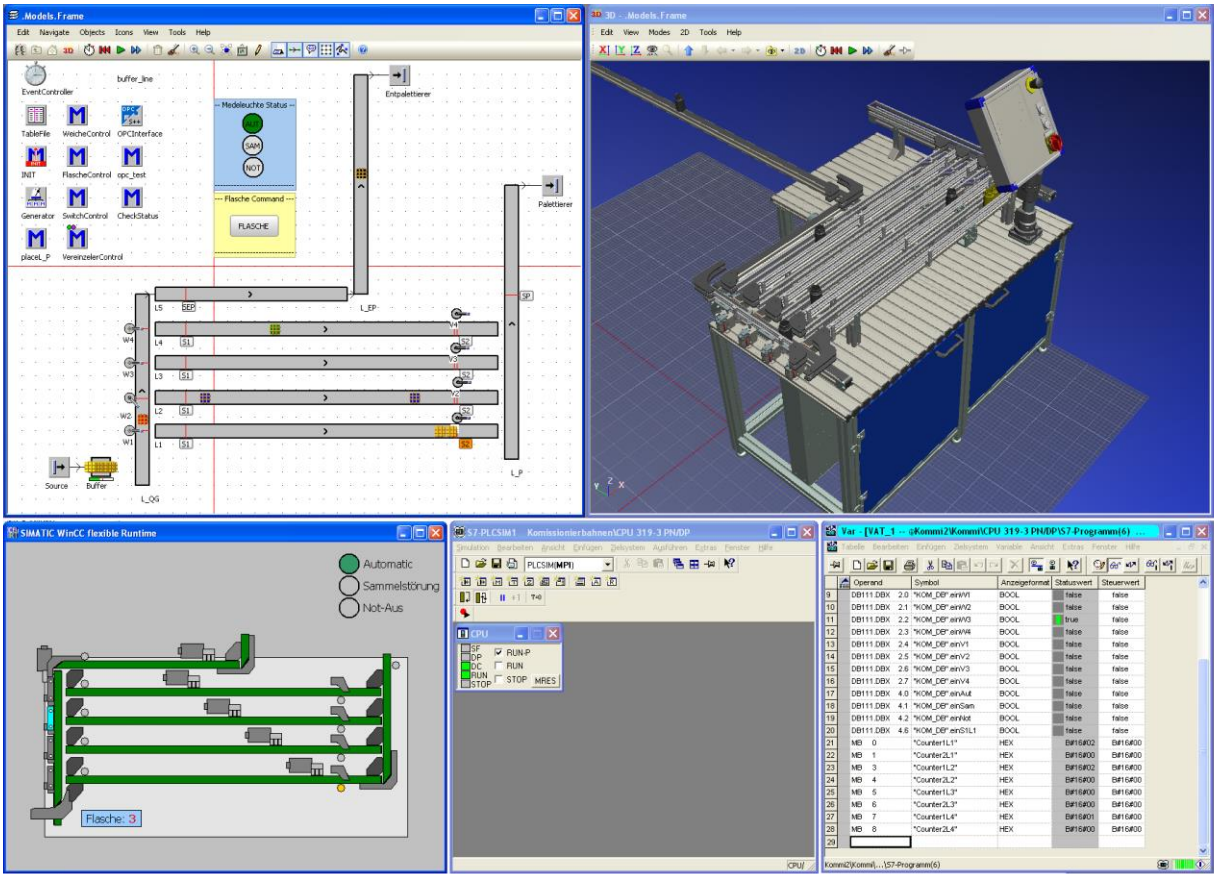


	Item Name	Type	Alias
1	W1	Boolean	W1
2	W2	Boolean	W2
3	W3	Boolean	W3
4	W4	Boolean	W4
5	V1	Boolean	V1
6	V2	Boolean	V2
7	V3	Boolean	V3
8	V4	Boolean	V4
9	L_QG	Boolean	L_QG
10	L1	Boolean	L1
11	L2	Boolean	L2
12	L3	Boolean	L3
13	L4	Boolean	L4
14	L5	Boolean	L5

Figure 46 – Communication variables in Tecnomatix

Chapter 5: Results

At the end of this work the virtualization of the Commissioning Station was completed by creating an accurate 3D model in Tecnomatix and the model was totally integrated with S7-PLCSIM and WinCC Flexible. It was possible to simulate the PLC in a virtual machine and also the HMI Panel from WinCC, while the Commissioning Station could be seen in Tecnomatix.



	Operand		Symbol	Anzeigeformat	Statuswert	Steuerwert
9	DB111.DBX	2.0	"KOM_DB".einW1	BOOL	false	false
10	DB111.DBX	2.1	"KOM_DB".einW2	BOOL	false	false
11	DB111.DBX	2.2	"KOM_DB".einW3	BOOL	true	false
12	DB111.DBX	2.3	"KOM_DB".einW4	BOOL	false	false
13	DB111.DBX	2.4	"KOM_DB".einV1	BOOL	false	false
14	DB111.DBX	2.5	"KOM_DB".einV2	BOOL	false	false
15	DB111.DBX	2.6	"KOM_DB".einV3	BOOL	false	false
16	DB111.DBX	2.7	"KOM_DB".einV4	BOOL	false	false
17	DB111.DBX	4.0	"KOM_DB".einAut	BOOL	false	false
18	DB111.DBX	4.1	"KOM_DB".einSam	BOOL	false	false
19	DB111.DBX	4.2	"KOM_DB".einNot	BOOL	false	false
20	DB111.DBX	4.6	"KOM_DB".einS1L1	BOOL	false	false
21	MB	0	"Counter1L1"	HEX	B#16#02	B#16#00
22	MB	1	"Counter2L1"	HEX	B#16#00	B#16#00
23	MB	3	"Counter1L2"	HEX	B#16#02	B#16#00
24	MB	4	"Counter2L2"	HEX	B#16#00	B#16#00
25	MB	5	"Counter1L3"	HEX	B#16#00	B#16#00
26	MB	6	"Counter2L3"	HEX	B#16#00	B#16#00
27	MB	7	"Counter1L4"	HEX	B#16#01	B#16#00
28	MB	8	"Counter2L4"	HEX	B#16#00	B#16#00
29						

Figure 47 –Final model’s screenshot

The Figure 47 shows the final model running with all software integrated and fully communicating. This is an improved version from section 4.3: where the integration with OPC substituted the checkboxes. This version is not integrated with other stations since the Backbone Factory is not fully virtualized, therefore to simulate the bottles coming, in the 2D model, there is a button that, when pressed, makes a new bottle appears on the model and it’s possible to see the objects moving through the 3D model and also in the HMI Panel showed in Figure 48.

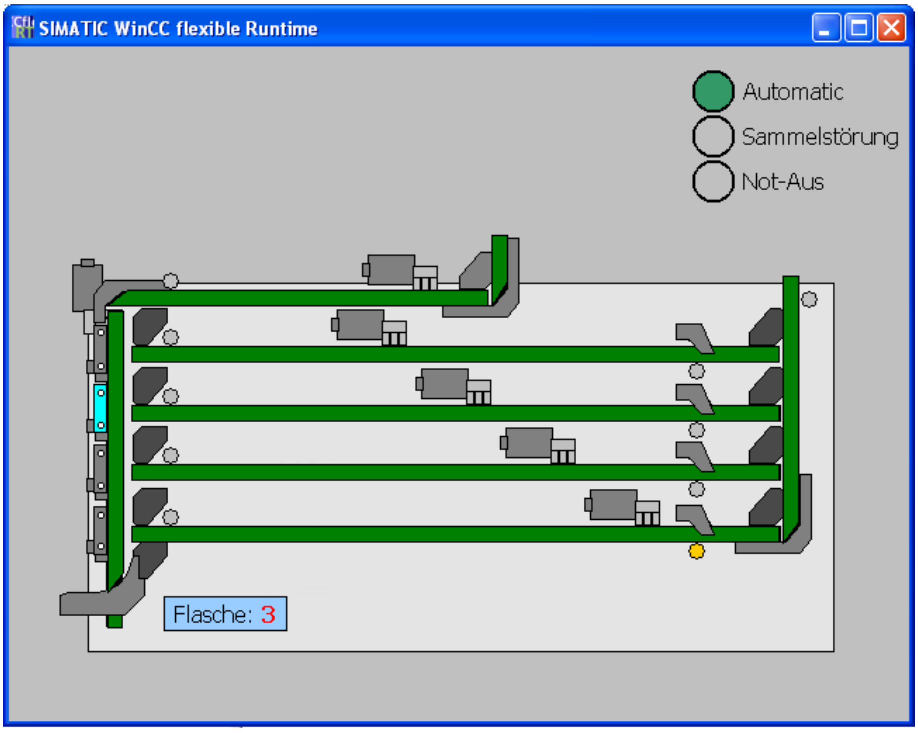


Figure 48 – WinCC interface in HMI Panel

This interface in WinCC was built at the end of this work because it was not planned in the beginning. Unfortunately, the Step7 and WinCC code were not ready, since another team was making some changes in the Backbone Factory. Therefore, a simple program was written to control the Commissioning Station. Even then, all variables and objects are listed in the program, and if a new code needs to be tested, the user just needs to specify the address from each variable that is present in his data block for communication, as it was like showed in Figure 44.

In the WinCC interface, the objects have 2 colors that indicate when it is turned on or off. The bottles are specified by number, one to five, also with different colors because each line has a specific type of colored bottle. There are 3 lights which inform the actual control status of the machine.

The 3D model was also improved to better represent the bottle’s flow, although it was not possible to represent the colors and the bottles button, however, since it is a limitation from Tecnomatix and representations like these might only be possible in future releases.

Chapter 6: Conclusions and Perspectives

This present work is a small part of the whole project of virtualization from the Backbone Factory and it was possible to see that the objectives described in section 1.3: were accomplished during the presentation of this paper.

At the beginning of this work a technical background was presented to show the main tools and concepts used to base the process of virtualization. This involved the first part of the project, where it was possible to learn the idea of Plan Simulation and how to accomplish the objects using tools like S7-PLCSIM, WinCC Flexible and Tecnomatix.

After that, it was necessary to have a general knowledge of the Backbone Factory and a deep study of the Commissioning Station to understand its hardware and software. Collecting sufficient technical data of the material flow, production and structure in order to obtain an accurate simulation data.

Therefore, simple programs were built to explore what is feasible and what does not apply in the process of virtualization in order to achieve, at the end, a complete and validated program.

A more detailed result will be further raised with the complete virtualization of the machine, yet the same can more easily be done with other stations by following the path of this work since its success is concrete.

At the end, it was possible to have a high-resolution 3D model to make the complete virtualization from Commissioning Station, but results from simulation experiments can raise further questions which the model is not detailed enough to answer. Ergo it might be a good idea to develop incrementally refined models or sub-models, or even to represent individual parts of the system with different degrees of detail in the model. Undoubtedly, these subjects are potential propositions for future works.

References

[1] Bangsow, S.: Fertigungssimulationen mit Plant Simulation und SimTalk - Anwendung und Programmierung mit Beispielen und Lösungen. Hanser Verlag, München, Wien, 1. Aufl. Aufl., 2008.

[2] Verein Deutscher Ingenieure - Richtlinie 4499 Part 1: Digital factory – Fundamentals, Düsseldorf, 2008.

[3] Bracht,U. and Masurat,Th: The digital factory between vision and reality. In: Computers in Industry, 2005.

[4] Verein Deutscher Ingenieure - Richtlinie 3633 Part 1 Draft: Simulation of systems in materials handling, logistics and production - Fundamentals. Düsseldorf, 2010.

[5] Festo: AFB factory – Hybrid production http://www.festo-didactic.com/us-en/products/learning-factories,cim-fms-systems/afb-factory-hybrid-production/customised-solutions-afb-factory-hybrid-production.htm accessed November 21, 2013.

[6] Nickl, A. “Virtuelle Inbetriebnahme automatisierter Systeme”. Bachelorarbeit, I CS EH BB, Erlangen, Siemens AG, 2013.

[7] Ludwig, M. "Migrationskonzept einer Schulungsanlage für Factory Automation von der SIMATIC „Classic“ Engineering Plattform auf das Totally Integrated Automation Portal V12" Bachelorarbeit, I CS EH BB, Erlangen, Siemens AG, 2013.

[8] Verein Deutscher Ingenieure - Richtlinie 3633 Part 11: Simulation of systems in materials handling, logistics and production - Simulation and visualization. Düsseldorf, 2009.

[9] Siemens AG: STEP 7 Engineering Software SIMATIC S7-PLCSIM http://www.automation.siemens.com/mcms/simatic-controller-software/en/step7/simatic-s7-plcsim/Pages/Default.aspx accessed November 29, 2013.

[10] WinCC flexible 2008 Compact/Standard/Advanced User's Manual, 2008.

[11] Siemens AG: Tecnomatix Plant Simulation Product Overview http://www.plm.automation.siemens.com/en_us/products/tecnomatix/plant_design/plant_simulation.shtml accessed November 29, 2013.

[12] Siemens PLM Software Inc: Tecnomatix Plant Simulation 11 Help, Version 11.0.0, 2013.

[13] OPC Foundation: Devices, Draft Version 0.75, 2008

[14] FAQ WinCC flexible Options: What options are there in WinCC flexible for using a Multi Panel or Mobile Panel as an OPC server?, March, 2012.

[15] Festo: Trends in automation, The Festo customer magazine, 2.2011

[16] Siemens AG: Schulungsanlage Factory Automation Anlage, 2013

[17] Siemens AG: Dokumentation Factory Automation - Handbuch für die Factory Automation Anlage v0.1

[18] Siemens AG: Konfiguration der OPC-Schnittstelle in TX Plant Simulation & STEP7, Andreas Nickl, 2013.