

Digital Twins for Education and Study of Engineering Sciences

Serge Zacher

University of Applied Sciences Darmstadt, Faculty EIT, Germany, info@szacher.de

Abstract: Generally a digital twin is a software-model as a prototype for some product, ordered by a customer. It shall be used on all stages of industrial production of this ordered product, comparing the actual state with the model and correcting the differences. A “virtual world” will be created from the “real world” and both “worlds” communicate with each other during production. At the educational institutions of engineering like universities, trainings centres or schools the digital twins are software-models of industrial plants, which are simulated and visualized similar to its industrial originals and synchronized with them. The presented paper shows steps of design of digital twins upon some examples of the master degree program at the University of Applied Sciences Darmstadt. It will be described why and how it is possible to use the digital twins. The ways from an original device to its model will be drawn. The different stages of the development of software-models, which are equipped with different levels of source code, will be shown. The actual trends to include the hardware into a software-model, so called hardware-in-the-loop and rapid control prototyping, will be described. Finally the advantages and disadvantages of digital twins for engineering study including the economic considerations will be discussed.

Keywords: Simulation, Model-in-the-loop, Software-in-the-loop, Processor-in-the-loop, Hardware-in-the-loop

Introduction

The practical exercises on the real technological devices are supposed to be a necessary part by the study of engineering sciences at universities and by the training of operators for industry. But not all educational institutions are financially supported enough to implement a real industry process in own campus (see an example [Simons, 2012]) or to build their own well-equipped laboratories. For this purpose the educational institutions organize internship, factory tours and external lessons (see Figure 1) or build the physical models of the real systems, so called pilot plants, which also is associated with high costs.



Figure 1. Students of the University of Applied Sciences Darmstadt by external lessons

The development of the informatics and microcontrollers technology has made it possible to reduce laboratories costs through application of hardware-models of industrial processes. A hardware-model usually is a box with the microcontroller, located inside. On the front panel of such a box are LEDs, which pictured the simulated industrial process. In Figure 2 are shown examples of a pilot plant at one University of Applied Sciences (1998) and of the hardware-model at another University (2012). The relation between costs for this plant and the hardware-model is 20:1.



Figure 2. Pilot Plant of a Chemical Process (on the left) und hardware-model of a batch-reactor

Software-Models

The success of programming, software design and mathematical modelling of technological processes has opened a new way of costs reducing for laboratories by engineering study, namely: the software-models. An example of temperature- and level-control and its simulation are given in Figure 3.

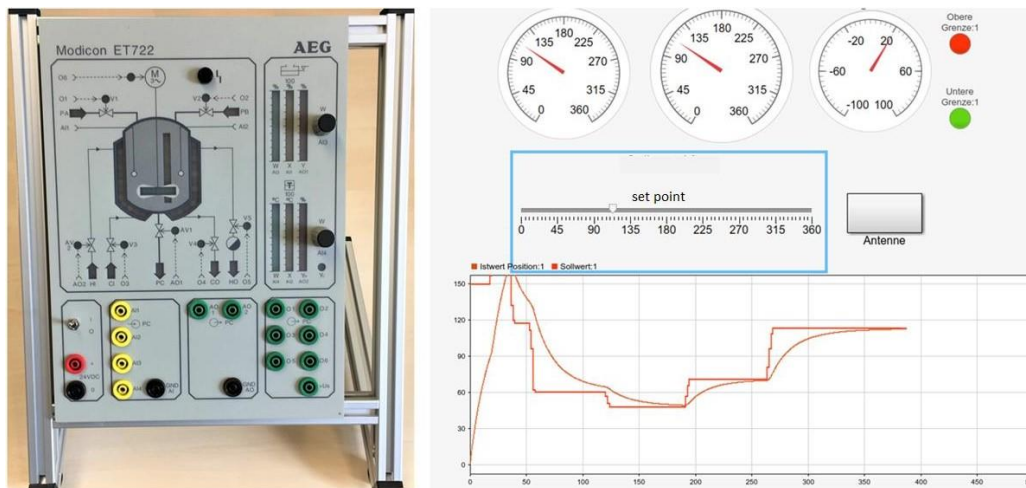


Figure 3. Hardware-model of a Batch-Reactor as a Laboratory Device (on the left) and its Software-model

But as well the simulation software has been developed and how low is its price, anyway it has a significant didactic disadvantage: it cannot replace the practical exercises on the real industrial devices.

Web Laboratories

The solution of the problem, how to use the favourable economic considerations of simulation software and at the same time to do practical exercises on the real devices is remote control. That was the reason why 1971 the NSF (National Science Foundation) began in the USA two big projects: TICCIT (Time-shared Interactive Computer Controlled Information Television) and PLATO (Programmed Logic for Automated Teaching Operations). The goal of both projects was similar: to show how efficient is the online education via Internet. Twenty-five years after first NSF projects, mentioned above, the international group of professors, including author of this paper [Henry, Zacher, 2010, June 29-July 02], proposed NSF a new project called EVO-ELE (Engineering Virtual Organization for Engineering Laboratory Education). The goal of the project was to extend development of online laboratories, further the sharing of human and engineering. As a result of the EVO-ELE project was 1995 the UTC-WebLab (the laboratory of University of Tennessee at Chattanooga) connected to the web to be operated remotely [Henry, 1998] and since that had shared over 10,000 laboratory experiments per year [Henry, Zacher, 2010, Feb]. The concept of UTC-WebLab is the combination of remote control of labor

equipment with the databank and virtual tools for data transfer from campus to the external user. In using the remote laboratory each student configures inputs of the plant (constant, step, sine, pulse, ramp or custom, its size and its duration) and sends the configuration via Internet to the UTC. The step responses come back from UTC-WebLab as a text- and csv-file, which students import to the EXCEL or MATLAB®/Simulink and evaluate (see Figure 4). Of course the simulation will be done by the user himself after the experimental results are transferred back. Farther students choose a kind of controller (P or PID), set parameters of controller, simulate and send the controller-configuration again to the real controller of the UTC-WebLab. In such a way the calculation results will be tested and corrected if necessary. Each student receives a personal task, so that the results cannot be exchanged between students and demands an own calculation.



Figure 4. Students of Master-courses at Universities of Applied Sciences Darmstadt (l.), ReinMain (m.) and Bingen by sharing UTC-WebLab experiments from University of Tennessee at Chattanooga

A web-station for a remote control of a temperature plant (see Figure 5), similar to the UTC-WebLab, was developed later also at the University of Applied Sciences RheinMain [Hankammer, Heil, Scherer, 2009].

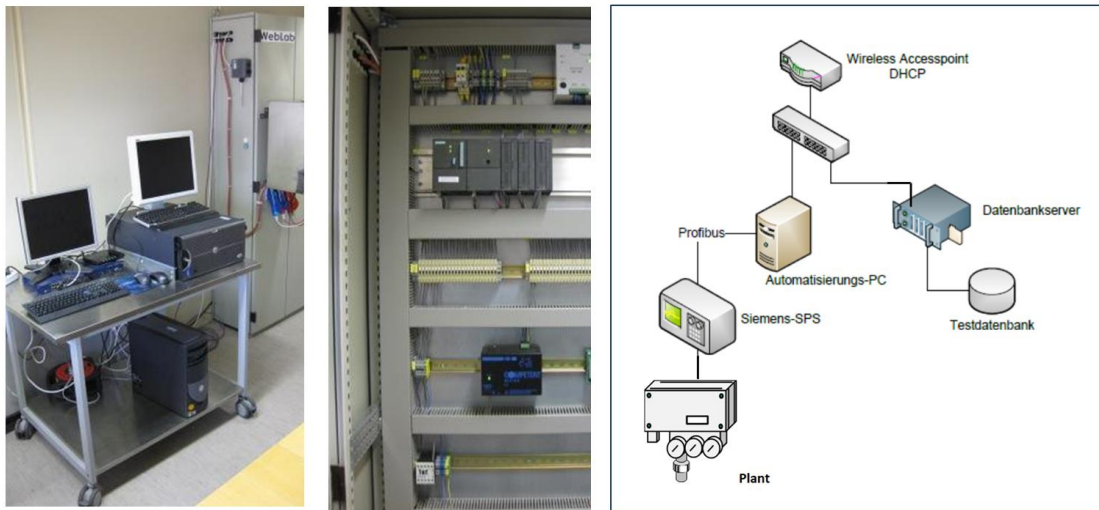


Figure 5. Remote Control System of the University of Applied Sciences RheinMain

Concluding we can generally classify the laboratories for engineering study as follows:

- Real industrial processes (“industry world”)
- Laboratories with pilot devices of the industrial processes (“pilot world”)
- Laboratories with hardware-models of real devices (“real world”)
- Virtualized laboratories with simulations instead of real devices (“virtual world”)
- Remote laboratories with real devices placed far from students (“WebLab”)

Fusion of the “Real World” and “Virtual World”

A concept of global digitalization of information, known as “Industry 4.0”, has led in the last five to six years in the industry and also by laboratories of engineering study to the fusion of the “real world” and “virtual world”, namely to so called “digital twins”.

Although the definition of digital twins was known since 1991 its concept for engineering was published only ten years ago in [Piascik, R., et al. (2010)]: “The digital twin concept consists of three distinct parts: the physical product, the digital/virtual product and connections between the two products. The connections between the physical product and the digital/virtual product are data that flows from the physical product to the

digital/virtual product and information that is available from the digital/virtual product to the physical environment.” Simplified this definition can be formulated as follows: A digital twin is a simulated and visualized software-model of an original system, which looks exactly like original system, operates in real time so, that its operations are synchronized with the original system (see Figure 6).

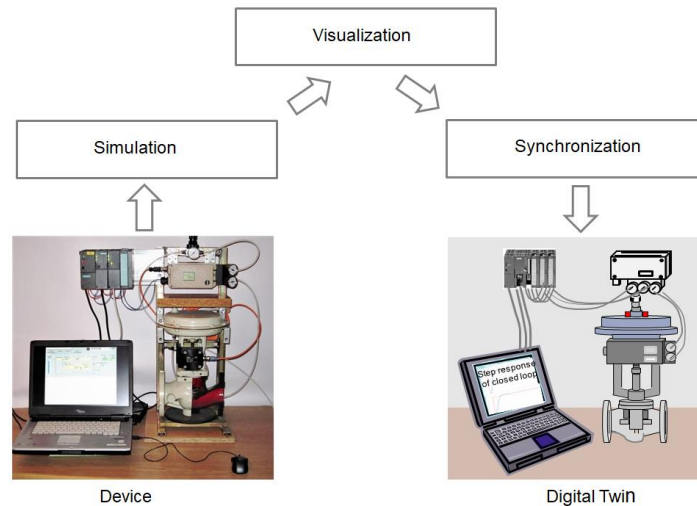


Figure 6. Developments Stages of Digital Twins for Study and Education

In the following the development and the use of digital twins by the study of engineering sciences will be shown on some examples. But first it will be described, what kinds of simulation of industrial automation systems have been developed in the last ten years.

Kinds of Simulation

It is widely known that a real automation system consists of two parts: a process to be controlled, called plant, and a controller device. The first mathematical models of industrial plants, which were proposed about 60 years ago, were mostly ordinary or partial differential equations. The solution of such equations with the IT-technology of those years was difficult. With the development of microprocessor technology, the mathematical models were programmed and implemented using CPUs. As mentioned above the signals were displayed with LEDs on panels, so that the hardware-models were offered instead of real systems, which were controlled with real PLCs. In the following, they are referred as controller-in-the-loop (CIL).

In the next stage of modelling both, real controllers and real plant or their hardware-models, were simulated with software. The interaction of two such software-models on a host PC is called model-in-the-loop (MIL). The software-models are visualized so, that the 2D or 3D images of real plants and controllers are displayed on the PC monitor. However, it is an offline simulation that does not run in the real time.

The modelling was thoroughly evolved over the past five to six years. The corresponding source code was created from the software-model of the controller, usually the C or PLC code.

Two options of application of such created code are possible:

- If the controller code is executed together with the software-model of the plant on the same host, then such application is called software-in-the-loop (SIL).
- If software model of the plant is executed on one host (usually on a PC) and the controller code is implemented on another host (usually on a microcontroller board), then such application is called processor-in-the-loop (PIL).

The classification of MIL, SIL, PIL is shown in Figure 7. There are two levels: an industry level and an education level. The difference between both levels is: while in the industry level the real technological devices are simulated, in the education level usually are simulated the hardware-models of real devices. The same difference concerns the digital twins. It could be explained by the different goals of both levels: each level simulated devices, which it has.

Further in Figure 7 is shown, as mentioned above, that digital twins are created by supplementing the software models with realistic images and allowing these models to run in real time (synchronizing them with the original

physical systems). The digital twins can be controlled via virtual interfaces similar to the real machines via real interfaces and fieldbus systems with the same control software. A “virtual world” is created corresponding to the concept “Industry 4.0”, exactly reproducing the “real world” in terms of time, space and function.

From Figure 7 are understandable goals and tasks of digital twins for university courses, namely: to analyse, to design and to test the real control on virtual plant models in such a way that the difference between real and virtual plants is hardly noticeable.

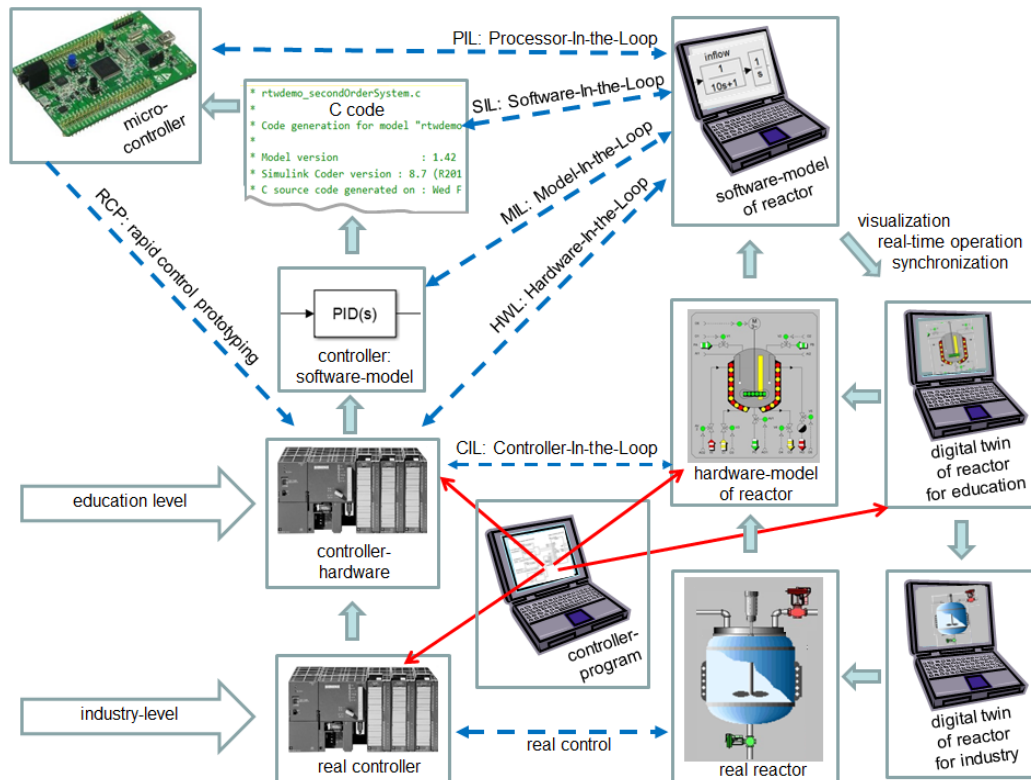


Figure 7. Classification of Hardware- and Software-models

From the user's point of view is another kind of classification possible (see Figure 8):

- Hardware-in-the-Loop (HWL). This is an interaction of a real hardware controller with the software-model of the plant. This option is e.g. used when an existing controller is to be tested in a laboratory instead of testing on the real plant.
- Rapid Control Prototyping (RCP). A controller for this option is only a software-model, the plant is a real physical device or hardware. This enables the algorithm of a controller to be checked on a real plant and if necessary to be improved (see [Zacher, 2019] and [Mille, 2017]).



Figure 8. Classification of Hardware- and Software-models from User's Point of View

Digital Twins by Engineering Study: Results

A digital twin should contain components which are important for illustrating the real industrial technology in the classroom. In addition, the digital twin should meet a certain technical standard to explain the scientific principles of real process. Of course, the number and type of process Inputs/Outputs is important. The main criterion for a digital twin is the operation time, namely the same reactions to the inputs as by the real system. The programming language plays only a subordinate role for the conception of digital twins and depends on the selection of the host-controller. An example of digital twin and videos can be found in [Szabo, Gurkasch, 2018].

Digital Twin for Process Automation

In the Figure 9 is shown the digital twin developed in [Lutz, Herbst, 2019] and the original hardware-model, consisting of an "OSLO 3" process simulator with the AC-700F controller from the ABB Freelance product line, served as the real process. The goal of this project was to create a digital twin for e-learning and practical exercises by university courses such as "Automation", "Feedback Control", "Bussystems" or "PLC".



Figure 9. Digital Twin (l.), simulated and controlled with MATLAB®/Simulink. The hardware-model OSLO-3 (r.), controlled with PLC Freelance and field controller AC-700 of ABB.

The real hardware-model (see Figure 9, on the right) runs on the PLC Freelance of ABB. The digital twin is simulated and visualized with the MATLAB®/Simulink. The SFC (sequential function chart) of the industrial PLC by hardware-model was implemented by digital twin with the "Stateflow"-tool of MATLAB®.

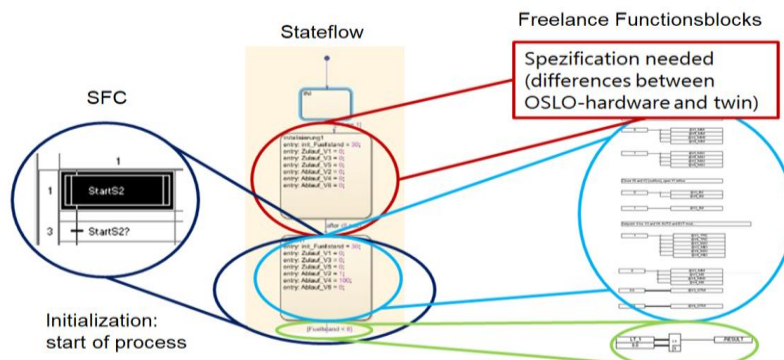


Figure 10. "Translation" of SFC from PLC in "Stateflow" of MATLAB®

However, it was not easy to apply the SFC to digital twin because of differences by programming in PLC and MATLAB®, so that the executing commands were inserted into the graphical user interface. The control of valves in MATLAB® and in PLC Freelance is also different, so that an adjustment was necessary. The problem

of synchronization or the adaptation of the model to the real time, however, was simply solved with MATLAB®. It was only necessary to use the menu command "Run (paced)" under the "Simulation" tab. The entire process was broken down into "paces" and "slowed down".

Finally, another important task was solved by creating a digital twin, namely, the visualization of the level indicator with a bar in such a way, that the digital twin matches the real application as closely as possible. With the help of 90 comparators, the level values of the model were displayed directly with the LEDs.

A video, given in [Lutz, Herbst, 2019], illustrates the function of the digital twin for level control, which has already been used successfully by university courses and projects. For example, in [Czwalina, et al, 2019] a security control based on the PLCopen security concept was first developed and tested on the digital twin and only after that implemented in the hardware OSLO-3.

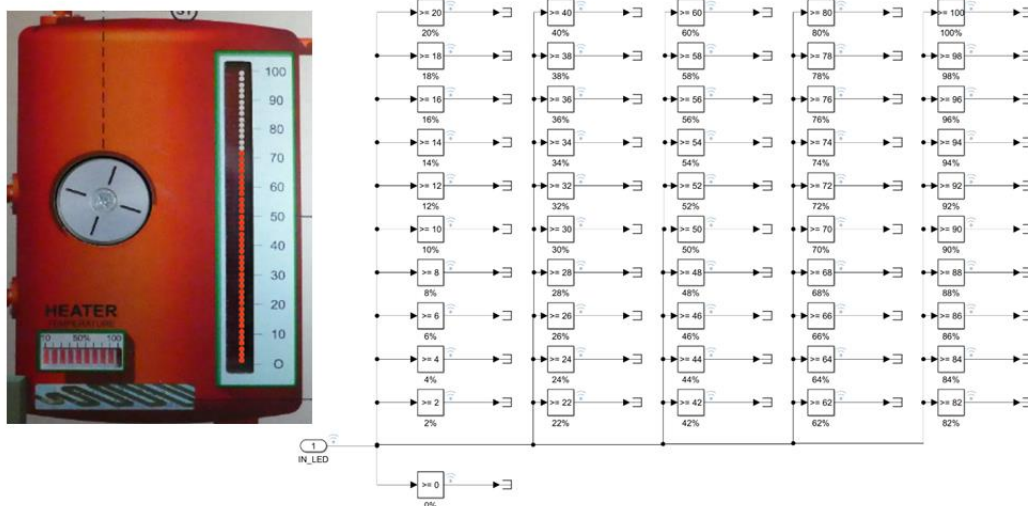


Figure 11. Visualization of level with LEDs of OSLO-hardware and "Dashboard" of MATLAB®

Working on this project a problem by using the digital twin was found. Namely, the real-time capability and also the synchronicity of the digital twin can be lost compared to the real process, if the system, on which the digital twin is running, does not offer the necessary performance. Another problem of digital twin is, that the simulation software is not platform-independent, i.e. when simulating on computers with MS Windows and on computers with MacOS there were differences in the visualization (arrangement of LEDs, buttons, animations).

Digital Twin for Mechatronics

In Figure 12 is shown the digital twin of a mechatronic device. While the digital twin of Figure 10 was developed for process automation based on an existing OSLO-3 hardware-model, the project team [Freitag et al, 2020] has first designed with the CAD SIEMENS NX the hardware-model of a conveyor system upon a real technological process, then build the hardware-model (see Figure 12).

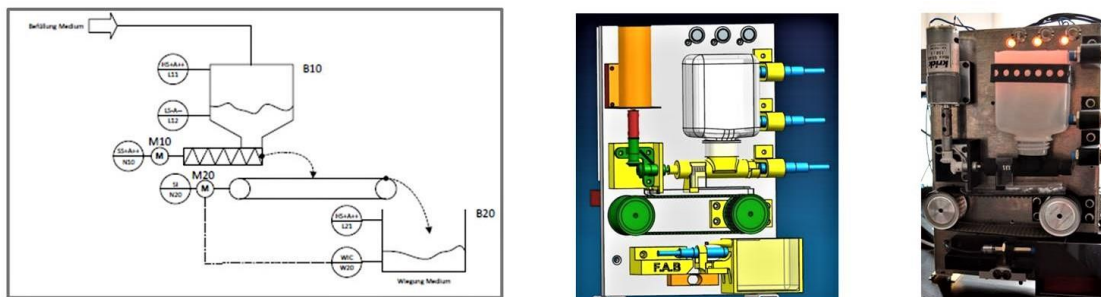


Figure 12. Steps of Development: Technology Concept (l.), CAD (m.) and real device (r.)

Finally a digital twin was developed using the same CAD SIEMENS NX completed with MCD software

(Mechatronics Concept Designer from SIEMENS) to animate the dynamic of conveyor (Figure 13). In opposite to the digital twin of OSLO-3 hardware-model (Figure 9) the process time of the digital twin of conveyor (Figure 11) was not important because the real processes (filling, weight control) are significantly slower as simulated processes.

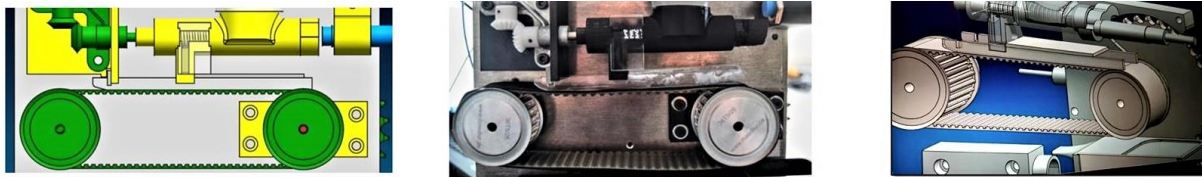


Figure 13. CAD of Conveyor (l.), real device (m.) and animated digital twin (r.)

The hardware-model of a mechatronic device shown above was special designed for learning purposes: it should be transportable and therefore also suitable for demonstration in the classroom. The focus by choosing the control device for hardware-model was on the number and type of process inputs and outputs as well as the question of whether the controller can provide the required control algorithm in the CPU. The decision was made in favour of an S7-1215C CPU of SIEMENS that offers 14 digital inputs, 10 digital outputs, 2 analog inputs and 2 analog outputs, as well as the possibility to implement a PID controller. This means that the passage of the medium to be transported can be simulated very close to reality, depending on the speeds of the conveyor belt. However, it was observed that the computing capacities of the used laptop were already insufficient. The considerable resources are required even for this relatively simple simulation task.

Digital Twins for Engineering Study: Discussion

Finally let us analyse under economic considerations what is better for an education laboratory, to virtualize one existing device to a digital twin or to build many real devices. It is supposed that an education laboratory consists of fifteen laboratory places. The calculation is given in the Table 1. The total costs of equipment for such laboratory are taken by the calculations to be 100%. First are the costs for the personnel to be brought up considered. The hourly rate is taken from an example of a price lists and reflects the costs that the university has to bear for one employee per hour. It is assumed that a trained technician, who is already familiar with the CAD-software, needs a week for the design, the dynamic determination and the following virtualization. The time needed to build fifteen real devices is estimated to fifteen working days.

Table 1. Costs for an education laboratory with 15 working places

<i>Position</i>	<i>Costs for real 15 devices</i>	<i>Costs for a digital twin</i>
Fee per our	0,206 %	0,206 %
Weeks	3	1
Days a week	5	5
Total fee	24,7 %	8,24 %
Price of a device	62,3 %	4,06 %
PLC costs (real control)	10 %	0,6 %
License for simulation software	--	20,9 %
License for CAD	--	18,8 %
License for PLC simulation	--	21,4 %
Installation costs	3 %	--
<i>Total Costs</i>	100 %	74 %

The cost savings of 26% by digital twins against real devices is clearly seen from the table 1. However, in addition to the economic analysis, the practical benefits of digital twins must also be analysed before a conclusion can be drawn. A distinction should also be made between the two cases, on the one hand digital twins for training purposes and on the other hand the general advantages and disadvantages.

The greatest problem by creating of a digital twin for training purposes is the design effort, which is, however, only one-off. Also the high costs for CAD-licences belong to a disadvantage. The problems by design and use of digital twins were mentioned above. They are:

- the adaptation of the model (Fig. 3.1) to the real time
- the implementation of real PLC algorithms into simulation software
- the visualization of the LED level indicators of real PLC with a bar of simulation software
- limited computing capacities even by relatively simple simulation tasks

As for the benefits of digital twins, there is on the one hand easier preparation for the training or experiments. The work with digital twins can be stated immediately, because no hardware is required to be installed or tested. On the other hand, an unlimited number of people can work with a digital twin and apply it.

Conclusion

The benefits of digital twins are:

- Easy preparation for training or experiments.
- No hardware is required to be installed or tested.
- An unlimited number of people can work with digital twins and apply them.
- The internship with digital twins can be made as comfortable as in classic laboratory rooms.

The greatest advantage of digital twins for training purposes is the adaptability and expandability depending on the training goal. The digital twins presented in this paper are perfect as illustrative material for learning basic knowledges in many courses of automation and PLC programming. This enables a teacher or a trainer to create extensions as needed according to specific questions of the training participants.

The disadvantage by creating a digital twin is the design effort, which is only one-off. Also the high costs for CAD-licences belong to a disadvantage. However the whole costs savings by the use of digital twins in an education laboratory consisting of fifteen laboratory places are 26% against fifteen real devices.

Although some points in the creation of digital twins should also be viewed critically, the numerous advantages make them an indispensable tool in today's technology and university world.

References

- Czwalina, D., Gilung, W., Jeckel, T., Heymann, J. (2019). PLCOpen für digitalen Zwilling des OSLO-3. *Hochschule Darmstadt, FB EIT, Fernmaster Studiengang*. Retrieved on the 29.01.2020 from https://www.zacher-international.com/C22_Team_Projekt/PLCOpenOSLO/PLCOpenDigiZwi.pdf
- Freitag, E., Alonso y Gonzalez, C., Buhl, J. (2020). Digitaler Zwilling einer Förderanlage. *Hochschule Darmstadt, FB EIT, Fernmaster Studiengang*. Retrieved on the 29.01.2020 from https://www.zacher-international.com/C22_Team_Projekt/DigiZwi/DigZwi_Foerderstrecke.mp4
- Hankammer, A. (2009). Entwicklung eines Moduls zur Bereitstellung und Verarbeitung von Daten als Teil eines Weblabors. *Diplomarbeit des FH ITE der Hochschule RheinMain*.
- Heil, A. (2009). Entwicklung eines mit LabVIEW gesteuerten, intelligenten Regelmoduls als Teil eines Weblabors an der Hochschule RheinMain, *Diplomarbeit des FH ITE der Hochschule RheinMain*.
- Henry, J., Zacher, S. (2010, June 29-July 02). Description of the remote UTC-WebLab for engineering education and interactive demonstration with online experiments. *International Conference "Remote Engineering & Virtual Instrumentation" REV, KTH, Stockholm, Sweden*.
- Henry, J. (1998). Laboratory teaching via the WWW. *ASEE Southeast U.S. Annual Meeting*,
- Henry, J., Zacher, S. (2010, Feb.). WebLabs in Control Engineering Education: status and trends. *7. AALE Angewandte Automatisierung in der Lehre und Forschung, FH Technikum Wien, 10./11.*
- Lutz, M., Herbst, D. (2019). Digitaler Zwilling der OSLO-Tafel. *Hochschule Darmstadt, FB EIT, Fernmaster Studiengang*. Retrieved on the 29.01.2020 from https://www.zacher-international.com/C22_Team_Projekt/Digitaler_Zwilling_L_H/DZLH.mp4
- Mille, R. (2017). Rapid Control Prototyping eines ASA-Controllers mit MATLAB PLC Coder. Verlag Dr. Zacher. Retrieved on the 29.01.2020 from <https://www.szacher.de/Verlag-Dr-Zacher/Mille-ASA/>
- Piasecik, R., et al. (2010). Technology Area 12: Materials, Structures, Mechanical Systems, and Manufacturing Road Map. *NASA Office of Chief Technologist*.
- Scherer, Ch. (2009). Erstellung eines Webseitenaufttritts für ein Internetlabor der Hochschule RheinMain. *Diplomarbeit des FH ITE der Hochschule RheinMain*.
- Simons, S. (2012). *Automated Assembly Line at CRA of University of Applied Sciences Darmstadt*. Retrieved on the 29.01.2020 from <https://www.youtube.com/watch?v=RyzDVajwliw>
- Szabo, A., Gurkasch, T.: (2018). Digitaler Zwilling der Simulationstafel ET722. *Seminararbeit DHBW Stuttgart, FB Technik*. Retrieved on the 20.01.2020 from https://www.zacher-international.com/Projekte/DHBW_Stuttgart/ET722_DigZwi/ET722_DigZwi_Sz_Gur.mp4
- Zacher, S. (2019) RCP. Rapid Control Prototyping. *Automation-Letter Nr. 31, Verlag Dr. Zacher*. Retrieved on the 29.01.2020 from <https://www.szacher.de/Automation-Letters/>