Realization of a Complex Control & Diagnosis System on Simplified Hardware

R Stetter, M Swamy Prasad

Hochschule Ravensburg-Weingarten, 88241 Weingarten, Germany

E-mail: stetter@hs-weingarten.de

Abstract. Energy is an important factor in today's industrial environment. Pump systems account for about 20% of the total industrial electrical energy consumption. Several studies show that with proper monitoring, control and maintenance, the efficiency of pump systems can be increased. Controlling pump systems with intelligent systems can help to reduce a pump's energy consumption by up to one third of its original consumption. The research in this paper was carried out in the scope of a research project which involves modelling and simulation of pump systems. This paper focuses on the future implementation of modelling capabilities in PLCs (programmable logic controllers). The whole project aims to use a pump itself as the sensor rather than introducing external sensors into the system, which would increase the cost considerably. One promising approach for an economic and robust industrial implementation of this intelligence is the use of PLCs. PLCs can be simulated in multiple ways; in this project, Codesys was chosen for several reasons which are explained in this paper. The first part of this paper explains the modelling of a pump itself, the process load of the asynchronous motor with a control system, and the simulation possibilities of the motor in Codesys. The second part describes the simulation and testing of a system realized. The third part elaborates the Codesys system structure and interfacing of the system with external files. The final part consists of comparing the result with an earlier Matlab/SIMULINK model and original test data.

1. Introduction and background

For several centuries, human kind has depended mainly on agriculture for its well-being. The growth of agriculture in main (dry) regions depends on the efficiency of irrigation systems. Irrigation efficiency experienced tremendous growth when pumps were introduced. Due to their versatility, pump systems have many applications in industrial environments, from small water pumps to large petroleum industries. Even though the designing and modeling of pump systems has evolved significantly, the efficiency of certain kinds of pumps, however, is not in accordance with the needs of the 21st century. The efficient monitoring and controlling of pump systems can reduce their power consumption by up to one third [1]. A promising and economic option for efficient monitoring and controlling of (complete) pump systems (including driving means and consumer system) is the implementation of programmable logic controllers (PLCs). PLCs are very robust, developed to industrial standards, and relatively cost effective. This paper describes an approach for the future implementation of control and monitoring systems for pumps in PLCs, within which the complete system is modeled and simulated in controller-based software. Similar attempts can be observed in the field of realization of fractional models and fractional regulators into PLC controllers [2].

Content from this work may be used under the terms of the Creative Commons Attribution 3.0 licence. Any further distribution of this work must maintain attribution to the author(s) and the title of the work, journal citation and DOI.
Published under licence by IOP Publishing Ltd

This research is part of a large project aiming to develop a well-founded prognosis of future control and diagnosis systems for pumps (compare e. g. Kleinmann et al. [3], [4]). For the development of smart and efficient pumps, the Hochschule Ravensburg-Weingarten, Germany, and the Hochschule für Technik Rapperswil, Switzerland, along with a leading German pump manufacturer formed a research team. The aim of the project team is to develop an advanced monitoring, diagnosis and control system for pumps. The whole project is mainly based on a burner application of a positive displacement pump – a spindle screw pump. Consequently, the statements in this paper are related to burner applications.

2. Modelling

In prior research, an elaborate model of the pump, the consumer system (burner) and the motor was developed in Matlab/SIMULINK (compare e. g. Kleinmann et al. [5]). The results are promising but the requirements for computer performance are rather large. Therefore, in ongoing research, the possibility of achieving similar results on less powerful computer hardware are investigated; this paper focuses on simulating PLCs with Codesys (an acronym for controller development system, previously stylised CoDeSys); Codesys is a development environment for programming controller applications according to the international industrial standard IEC 61131-3 [6].

Matlab/SIMULINK is software mainly designed for simulation, and hence provides great flexibility for modeling. Codesys is mainly designed for rather simple controlling tasks and has very limited built-in mathematical functions, and hence mathematical functions, such as differentiation, integration and matrix operations, are not possible.

The modeling of a pump can be divided into three major parts: the pump, consumer system or process load (burner application), and electrical motor. Such a model is built into separate blocks which are then integrated. This provides greater flexibility for analysis and testing. In this research, Codesys Control Win V3 is used as a PLC simulator. Figure 1 gives an overview of the blocks of the model and the sources for the formulae and parameters.

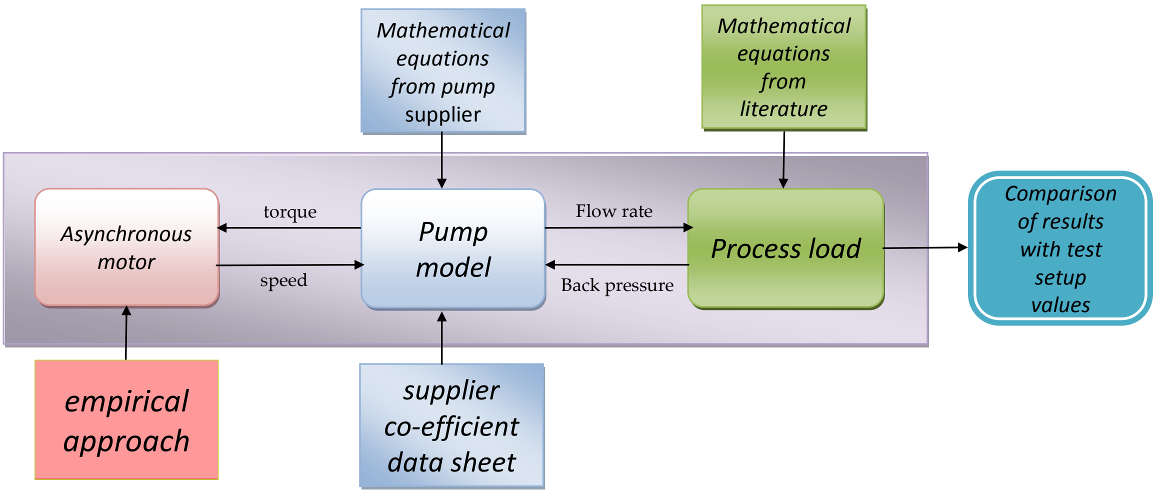
Mathematical equations from pump supplier
Mathematical equations from literature
Asynchronous motor
torque
speed
Pump model
Flow rate
Back pressure
Process load
Comparison of results with test setup values
empirical approach
supplier co-efficient data sheet

Figure 1. Complete system model.

2.1. *Pump model*

In this research, three screw spindle pumps (including different variants) are taken into account and modelled. The system is modelled in such a way that different variants can be tested and analyzed simply by changing the model number. The project aims to use devices with low computational power. Hence, the mathematical equations are kept as simple as possible. The mathematical equations are mostly based on prior research (compare Kleinmann et al. [4], [5]), but external references are also added.

For monitoring and controlling, certain parameters are measured and given as feedback to other blocks. The parameters include flow rate, speed, torque and differential pressure. As shown in Figure 2, torque (M) from the pump is given as a feedback to the motor in the on-line simulation so that the power of the motor can be controlled. For advanced control of whole system, the back pressure from the process load is also calculated. The inputs and outputs of the model are shown in Figure 2.

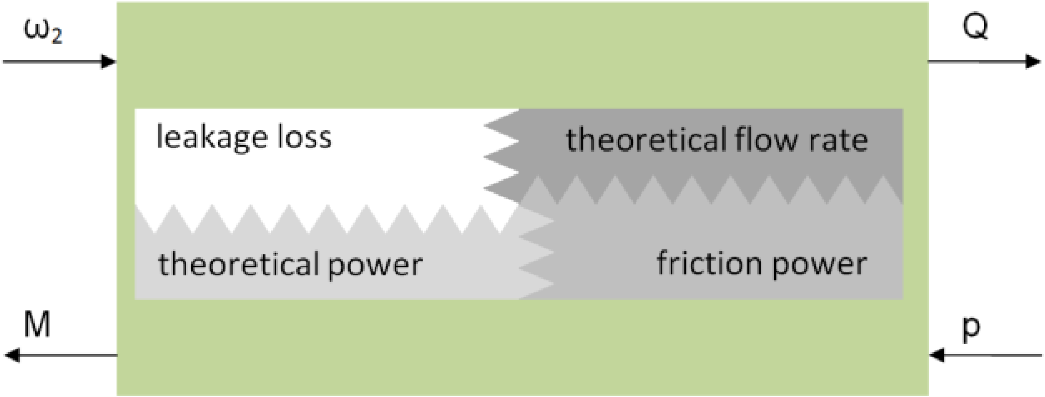

Figure 2. Pump model input and output.

In order to calculate the mentioned parameters, further parameters are needed, which include losses due to leakage and friction along with theoretical power and theoretical flow rate [5]. The following equations are used to model the pump. For the calculation of the flow rate, Equation 1 may be used:

$$Q_{th} = \left(\frac{3.D_a^{\ 3}.\pi.\tan\alpha.K_p}{2.10^7} \right).n_2 \tag{1}$$

Equation 2 may be used for the calculation of the theoretical power:

$$P_{th} = \frac{Q_{th2}.\Delta p}{600} \tag{2}$$

For the calculation of the pump torque load, Equation 3 may be used:

$$M_{load} = \frac{P_{act}}{\omega_2} \tag{3}$$

In these calculations, the value of the profile factor (K_p), diameter (D_a) of the pump and pitch angle (α) must be provided by the company which manufactures the pump. Figure 3 (left) shows a flowchart of the realization of this model in Codesys.

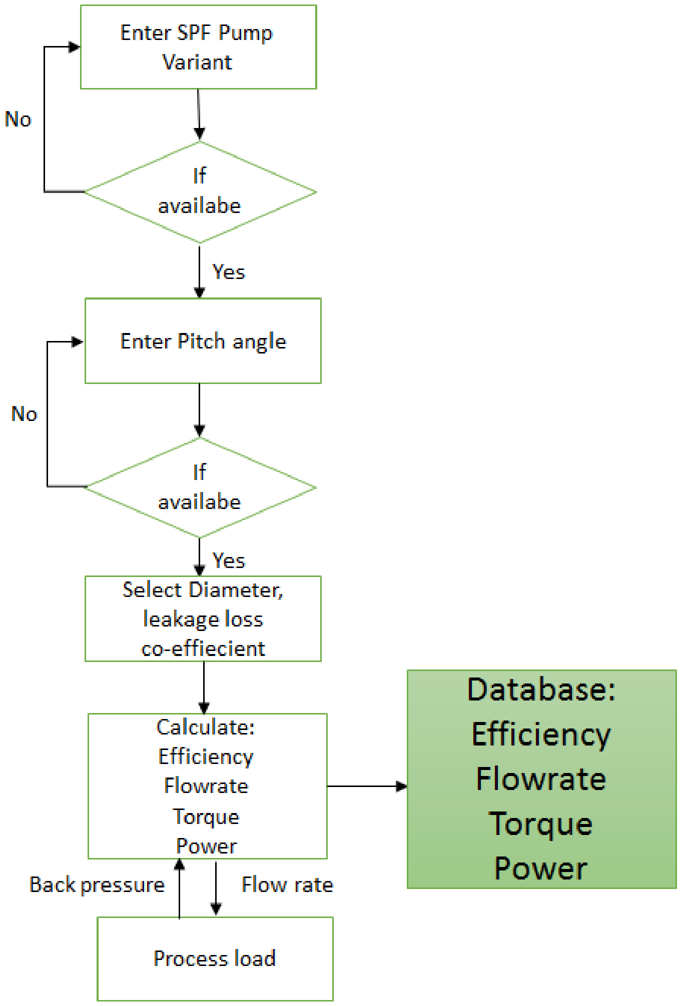
Enter SPF Pump Variant
No
If availabe
Yes
Enter Pitch angle
No
If availabe
Yes
Select Diameter, leakage loss co-effiecient
Calculate: Efficiency Flowrate Torque Power
Database: Efficiency Flowrate Torque Power
Back pressure
Flow rate
Process load

Figure 3. Pump model input and output.

2.2. Process load model

The process load is application specific. Since screw spindle pumps have a wide variety of applications, the requirements for each component can vary. The component includes different types of pipe, pipe bend, valve, nozzle, Y- pipe.

For modeling pipes, the mathematical formulae are taken from literature. For nozzle modelling, the direction of the nozzle was considered. Since nozzles can be both expanding and contracting, it was modelled in such a way that it will be decided whether it is expanding or contracting according to the user specifications. Additionally, pipe-bends were modelled similarly to prior research and the bend loss co-efficient values were refined.

The components are modeled into separate function blocks and put together to form a library. The advantage of creating a library for the process load is that different combination of components can be used. Every combination produces different results. Hence accuracy and efficiency can be improved. It also helps to avoid repetitive coding.

The output from each component is back pressure. The summation of pressure of each component will provide the total back pressure of the process load (Equation 4).

$$\Delta p_{tot} = \Delta p_n + \Delta p_{n+1} + \Delta p_{n+2} + ... \tag{4}$$

2.3. Electric Motor

The induction motor, or asynchronous motor, is generally used in industrial drive due to its robustness, reliability, and economical causes (in the test stand in the validation (compare section 5) a 2.2 KW nominal power asynchronous motor is used). As mentioned earlier, Codesys is controller based software and does not have the flexibility of using built-in library functions, and hence it was decided to model the system using an empirical approach. The advantage of an empirical approach is that simple mathematical equations, along with control systems, can be modelled.

Since the model runs continuously, the pump torque and speed of the motor, which can easily be gathered from a frequency converter for the asynchronous motor, are stored in a separate array. The current value of the pump torque is compared with the previous value of the pump torque and if the value exceeds defined limits (in both directions), then a change of the system state will be assumed and the speed of the motor will be increased or decreased proportionally according to the amount specified. Since the asynchronous motor is controlled by the frequency converter to a certain set (constant) speed motor, the speed will return to the set speed following a state change after a specified amount of time, which is necessary for the motor control to adapt to the new situation. In order to simulate that characteristic with rather simple calculations, a time delay function is used. If a defined delay time is over, the speed of motor will return to the set speed. The proportional factor and the time delay need to be found for each motor and each control system. The flowchart of this empirical motor model is shown in Figure 4.

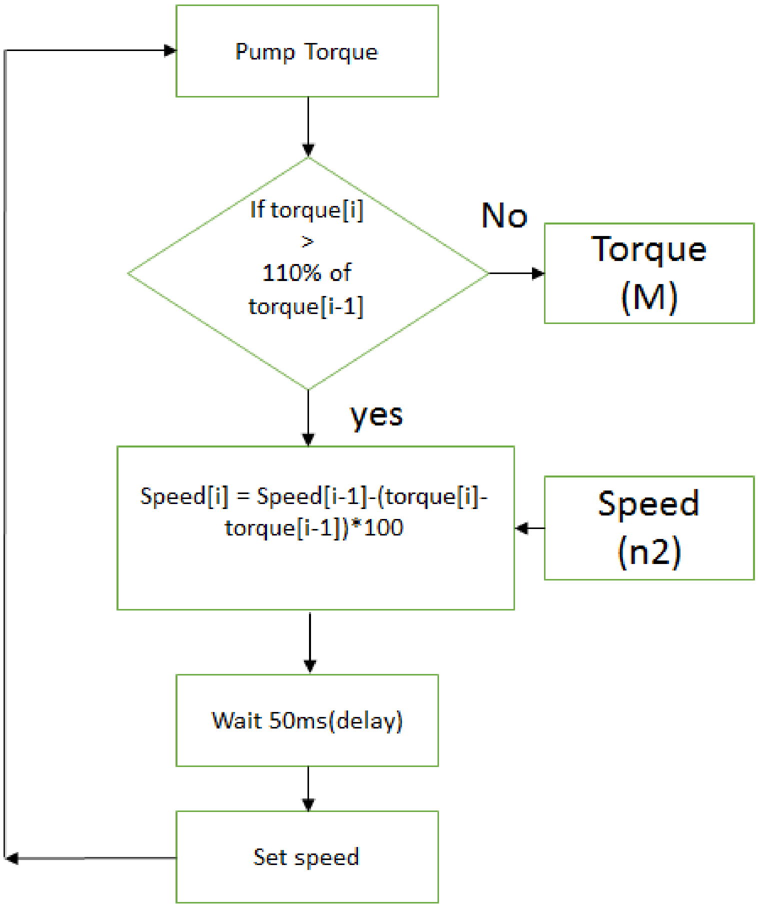

Figure 4. Flowchart of the motor model.

3. Programmable Logic Controller

Automation is the key word in today's industrial growth. In order to achieve maximum efficiency and reliability in production systems, industries are in need of automated production process. Programmable logic controllers (PLCs) help to achieve industrial requirements. The PLCs are available for small to large scale applications with varying input and output ports, as well as processing speed and built-in memory. PLCs are rigid and immune to electrical noise, which helps to achieve hard, real-time requirement. In this project, the advanced control system which is used in burner applications must be as rigid as possible. Therefore, it was decided to build an advanced control system for pumps with PLCs.

As the first step, the control system must be modelled in a simulated environment with hard, real-time requirements. Codesys helps to model such PLC-based systems in a simulated environment.

3.1. CoDeSys

CoDeSys stands for COntroller Development SYStem, which is a freeware provided by 3S-Smart Software Solutions. Codesys has an abundant library available to model a variety of systems. Codesys supports all five PLC programming languages defined by IEC 61131-3 to model and simulate the system. Here in this project, the system is modelled using "structure text".

The structure text (ST) is one of the PLC programming languages which is very similar to high level programming languages such as C, C++ or C#. Structure text has the advantage of modelling the system with conditional and looping statements.

Codesys allows the user to interface the model between different programming languages with the help of PLCopen XML. PLCopen XML is a special program which is only available in versions after V3.5 and created by PLCopen. PLCopen XML exports the program to a XML file, which is mostly hardware and software independent. The newly created XML file can be modified according to hardware requirements – this feature is quite unique.

3.2. Device

The output project file of Codesys is supported by most of the PLC hardware device providers. Codesys uses a device tree to know which devices are used in the project. For simulation, ~~e~~Codesys provides a variety of PLC simulators. The simulators include~~s~~ win V3, RTE V3, x64, HMI, and softmotion win V3. For simulating under windows OS, win V3 is suitable. Therefore, this project uses Codesys Control Win V3 as PLC simulator. The Win V3 device includes a API driver interface for customized I/O drivers and C functions, and also a SoftPLC-specific configuration. It supports standard field bus configurations, which are mentioned in IEC 61131-3, with no additional tool requirement.

3.3. Task configuration

The task configuration helps to manage the priority between applications. The type of process has to be mentioned in the task configuration. Since the model has to run in a loop for undetermined time, the type of process for this project is cyclic. The task configuration also helps to provide information about the maximum number of tasks and the maximum number of cyclic tasks.

3.4. Visualization

In order to realize a Human Machine Interface (HMI), Codesys provides visualization support. This support offers many possibilities for HMIs. In this project, the visualization part of Codesys helps to get values from the user without a lot of programming. For the customized monitoring of system, the visualization part of Codesys seems to be very helpful.

3.5. Advantages

Codesys is a free license software and can be installed legally without copy protection on further workstations. When compared to other PLC software, Codesys offers a wide variety of simulation possibilities. Codesys also provides object-oriented programming which is only available at higher programming levels. Object-oriented programming is widely used in modelling in this project. Codesys offers online as well as offline debugging capability.

3.6. Limitations

Since Codesys is automation software, mathematical functions are limited. While modelling the electrical motor, mathematical equations which involve differentiation and integration cannot be implemented.

4. Simulation

As mentioned earlier in Section 2, the project is divided into three subsystems: the pump, consumer system and electrical motor. The simulation of the subsystems is discussed in this section.

4.1. Program Organization Unit (POU)

The starting point of every project is PLC_PRG. From PLC_PRG, all POUs are executed.

The object-oriented programming part of Codesys can be realized by using POUs. The POUs are subcategorized into Program, Function and Function Block. The Program returns one or several output values. The values assigned to variables can be retained from the last run of the program. The Program can be called by other Programs, but not by Functions. The Function Block is also a POU with several output values. The variables in a Function Block can be used by other Functions or Function Blocks using the object-oriented concept of Codesys, which is only available in higher level programming languages. The major difference between a Program and a Function Block is that unlike a Program, the value of the last run cannot be retained in a Function Block.

As shown in Figure 5, the model of pump is subcategorized into efficiency, flow rate, power, speed and torque load.

The subcategory “intia” is the initialization of the system. All subcategories are defined under Program (PRG) since they return more than one value. The model is subcategorized in a way that essential parts of the model are built separately and then collectively form a folder in which all detailed information about the pump is given. The advantage of this separate modeling of subsystems is that they can be run and tested separately, which simplifies the debugging process.

In the process load, the system is subcategorized into pipe, pipe bend and nozzle. The major part of process load modeling is that the components should return the value of back pressure. Since back pressure is the only output value, the system will return only one value, and hence, they are defined under Function (FUN).

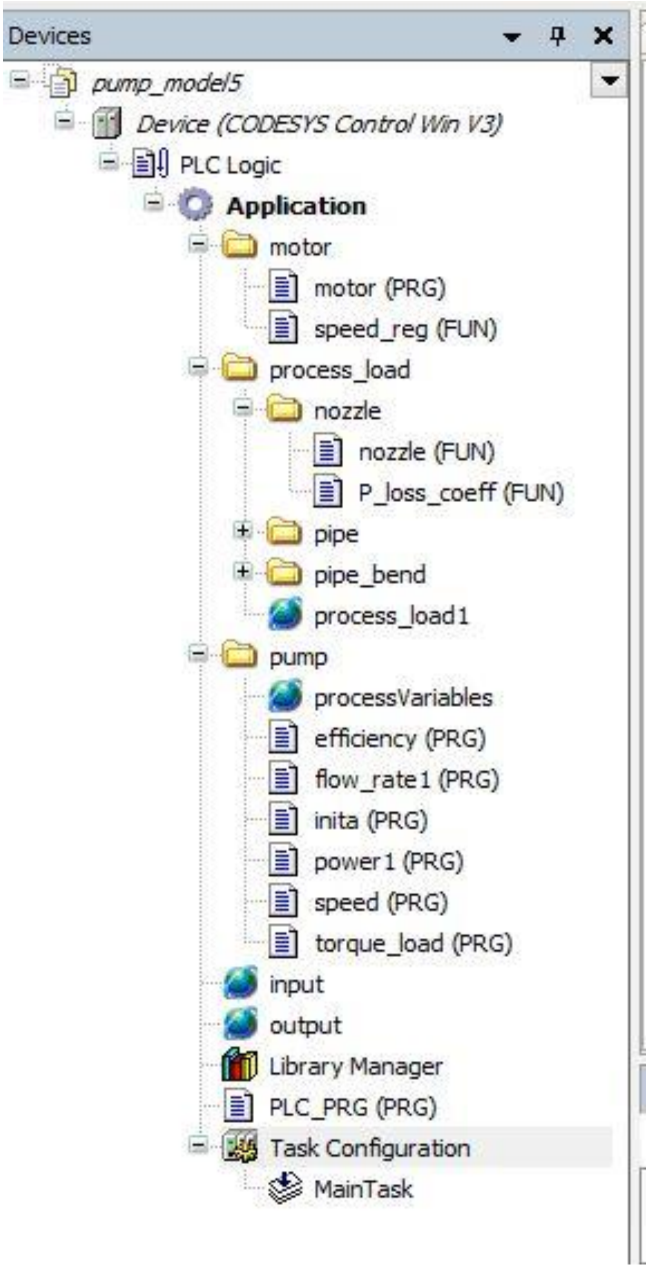

Figure 5. Structure of the project

4.2. Global Variables

Global variables provide a method to access variables throughout the system. A global variable (GVL) is marked with a separate symbol. There are three global variable lists used in this project.

There are separate global variable lists for input and output. The input global variable list consists of the pump model, the viscosity and the pitch angle. They are defined as global variables, since they are used all through the project. The output global variable list consists of the pump efficiency, the theoretical and the actual flow rate and of different losses in the system. There is another global variable list for process variables. The major disadvantage of GVL is that computation for GVL is much higher than normal variables, which are defined inside the POUs.

5. VALIDATION OF THE MODEL

To check the result obtained from simulation, results from the test setup used in prior research were compared (compare Kleinmann et al. [5] - Figure 6).

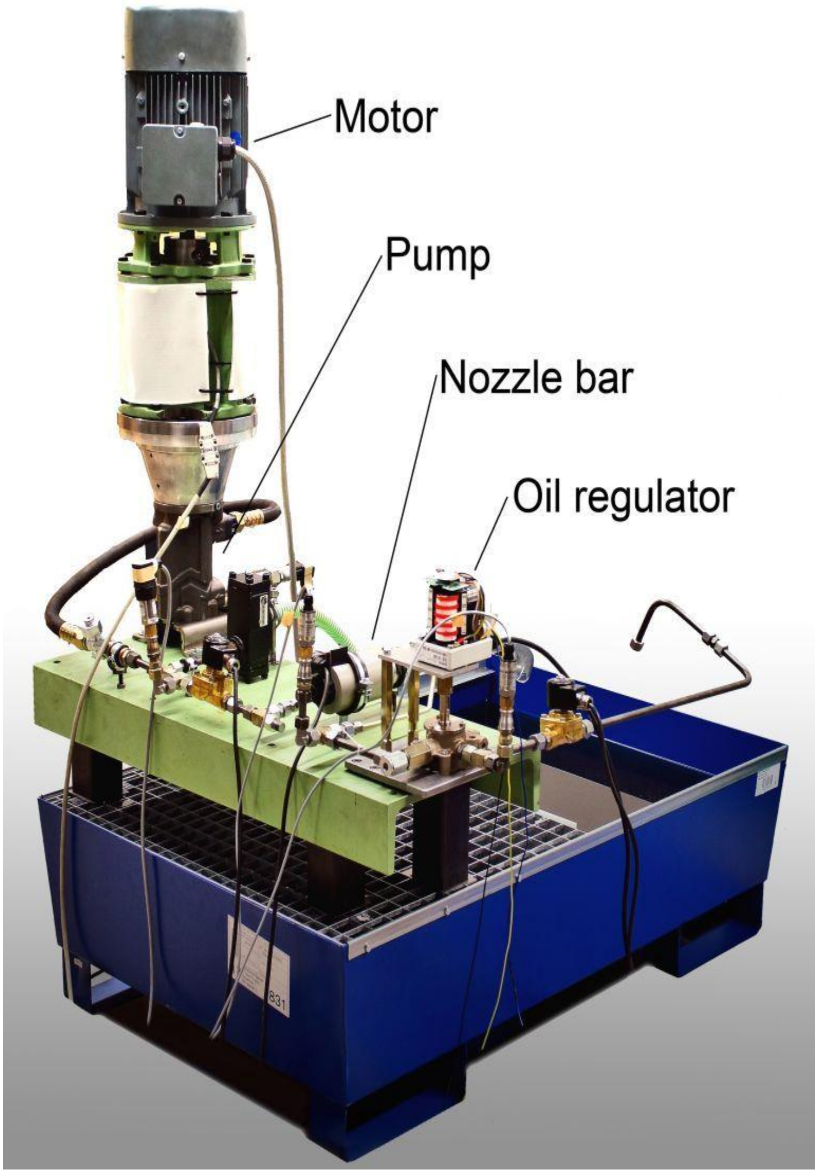

Figure 6. Test stand

The values obtained from the Codesys model were compared with the values from a prior model realized in Matlab/SIMULINK and also with test setup values. The results of a comparison of both models are plotted as graphs (see Figures 7 and 8).

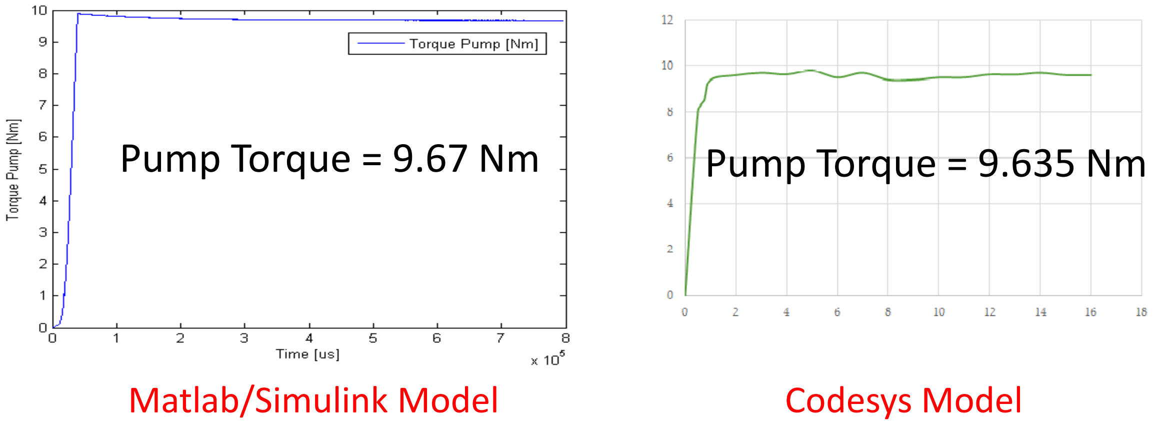

Figure 7. Comparison of models for torque: Matlab/SIMULINK (left) and Codesys (right)

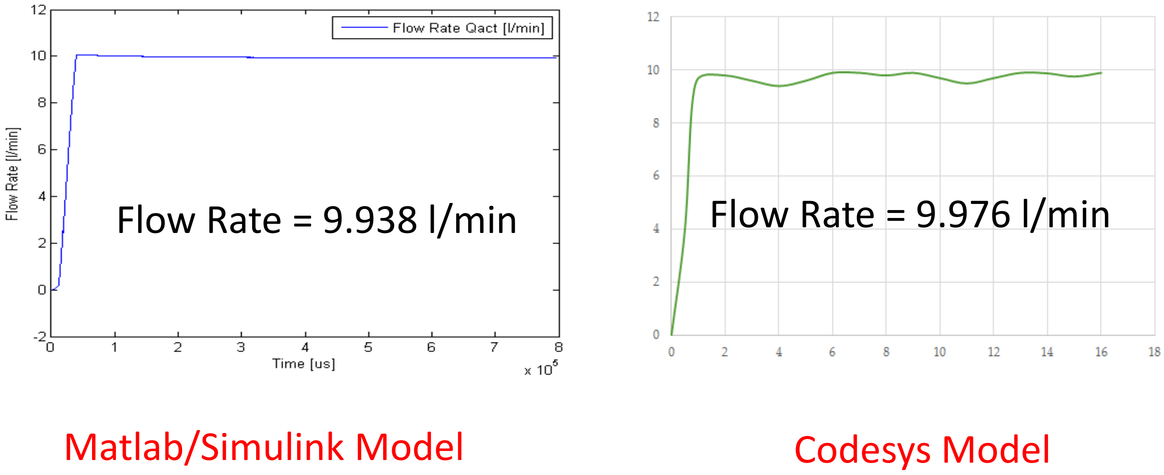

Figure 8. Comparison of models for flow rate: Matlab/SIMULINK (left) and Codesys (right)

6. Conclusion

Several survey show an astonishing contradiction between high a high energy saving potential of control systems and a relatively low application ratio in industrial practice. One root cause may be economic considerations – sometimes unstrung concentration on investment costs can be observed. Consequently, research projects which focus on the application of relatively cheap hardware for elaborate control activities such as the research of Kopka [2] are of paramount importance. This paper presents the modelling and simulation of smart pumps using the PLC simulation system Codesys. The results of this investigation show that PLC's can provide an economical and robust implementation possibility for industrial applications. It proved to be especially significant to model the asynchronous motor with its control system integrated into the frequency converter. For this purpose, a rather mundane empirical model was chosen, which relies on state changes and control delays. The simulation results obtained from the Codesys model correspond well with results from an elaborate Matlab/SIMULINK model from prior research which we require much more elaborate hardware. This comparison shows that the values obtained are within the tolerance limit. Since Codesys has general syntax defined by IEC 61131-3, the code can be implemented in any controller for further verification. Further steps will include this implementation and an expanded verification procedure.

References

[1] Friedl, C.: Energie-Optimierung von Pumpen rechnet sich. In: *Maschinenmarkt*, November 2007.

[2] Kopka, R.: Implementation of a Fractional Model–Based Fault Detection Algorithm into a PLC Controller. J. Phys.: Conf. Ser. 570 072005, 2014.

[3] Kleinmann, S., Dabrowska, A., Hoffmann, M., Kühn, H., Koller-Hodac, A., Stetter, R.: Advanced Control of Industrial Pump Systems. In: *Proceedings of the 7th Workshop on Advanced Control and Diagnosis*, ACD'2009, Zielona Góra, Poland.

[4] Kleinmann, S., Dabrowska, A., Zuniga Mera, A., Koller-Hodac, A., Stetter, R.: Diagnosis of Three Screw Spindle Pumps Based on Static and Dynamic Modelling. In: Bokor, J. (Ed.): *Proceedings of the 9th Workshop on Advanced Control and Diagnosis*, ACD'2011, 17. and 18. November 2011, Budapest, Ungarn.

[5] Kleinmann, S.; Stetter, R.; Abdul Jalal, M. F.: Praxisorientierte Modellierung von Pumpensystemen für Monitoring, Regelung und Diagnose. In *at – Automatisierungstechnik* 59 (2011) 5, pp. 309 – 322.

[6] Vogel-Heuser, B.; Wannagat A.: Modulares Engineering und Wiederverwendung mit CoDeSys V3, Oldenbourg Industrieverlag: 2008.

[7] Kleinmann, S., Leonardo, D., Koller-Hodac, A.: Intelligente Pumpenantriebe. Simulation, Condition Monitoring, Fehlerdiagnose und Energieeffizienz. *VDI-Konferenz Antriebssysteme 2011*, Stuttgart, September 13-14, 2011-08-10.

[8] Isermann, R.: Mechatronic Systems: Fundamentals. Berlin: Springer, 2005.

[9] Isermann, R.: Fault–Diagnosis Systems: An Introduction from Fault Detection to Fault Tolerance. Berlin: Springer 2005.

[10] Kallesöe, C.; Cocquempot, V.; Izadi-Zamanabadi, R.: Model Based Fault Detection in a Centrifugal Pump Application. In: *IEEE Transactions on Control Systems Technology*, Vol. 14, No. 2, 2006, p. 204-215.

[11] Miklosevic, K.; Spoljaric, Z.; Jerkovic, V.: Analysis of Electric DC Drive Using Matlab Simulink and Simpower System. J. J. Strossmayer University of Osijek. Osijek, Croatia

[12] Kleinmann, S., Dabrowska, Koller-Hodac, A., Leonardo, D., Stetter, R.: Model of a combined pump and drive system for advanced control and diagnosis. In: *The Archive of Mechanical Engineering*. Volume LVII, Number 4 / 2010, pp 405 - 414. 2010.

[13] Räbinger, K., Maksoud, T.M.A., Ward, J.: "Development of a finite volume model for the compressible gap flow inside a screw pump", University of Glamorgan School of Technology Wales, UK, 2005.

[14] Stetter, R.; Paczynski, A.; Dabrowska, A.: Industrial Application of Advanced Control and Diagnosis. In: Witczak, M.; Rozpra-Faygel, M. (Eds.): *Proceedings of the 1th European Workshop on Control Engineering for Industry, CEIND'2012*, 16. and 17. April 2012, Glogów, Polen.