

# TELEOPERATION OF SERVODRIVE SPEED CONTROL USING A MATLAB/SIMULINK

BERK, P.; LAKOTA, M.; STAJNKO, D.; VINDIS, P.;  
OCEPEK, M. & RAKUN, J.

**Abstract:** *The modern approach to the design of control systems requires the use of development tools for rapid development time (fast prototyping). One of such tools is the simulation tool (Matlab/Simulink 2008b), on which we have made a synthesis of the analog servodrive model speed control, where an application on the real process has been made by using teleoperation (XPC Target System). Our system was designed by tools Mathworks. The designing cycle was first started by building a mathematical model of the control plath. In the mathematical model of the control path the dynamic properties of the system were considered. By using the Matlab/Simulink subsystem a simulation model of the servodrive and controller was created. Connection of the real process and program Matlab was enabled through PC24 and PC26AT cards. If the simulation results were not satisfactory, we repeated the modeling-simulation-analysis process until desired response was reached.*

**Key words:** *teleoperation, synthesis, speed control, servodrive*



**Authors' data:** Bs. Elect. Eng. **Berk**, P[eter]; Assoc. Prof. **Lakota**, M[iran]; Assoc. Prof. **Stajniko**, D[enis]; Dr. **Vindis**, P[eter]; Msc. **Ocepek**, M[arko]; Dr. **Rakun**, J[urij], University of Maribor, Faculty of Agriculture and Life Sciences, Pivola 10, SI-2311, Hoce, Slovenia, peter.berk@uni-mb.si, miran.lakota@uni-mb.si, peter.vindis@uni-mb.si, denis.stajniko@uni-mb.si, marko.ocepek@uni-mb.si, jurij.rakun@uni-mb.si

**This Publication has to be referred as:** Berk, P[eter]; Lakota, M[iran]; Stajniko D[enis]; Vindis, P[eter]; Ocepek, M[arko] & Rakun, J[urij] (2012). Teleoperation of Servodrive Speed Control Using a Matlab/Simulink, Chapter 27 in DAAAM International Scientific Book 2012, pp. 315-326, B. Katalinic (Ed.), Published by DAAAM International, ISBN 978-3-901509-86-5, ISSN 1726-9687, Vienna, Austria DOI: 10.2507/daaam.scibook.2012.27

## 1. Introduction

Teleoperation is a technical engineering discipline (Muskinja et al., 2011) that remotely controls or regulates the mechatronic assemblies. Teleoperation can be implemented in local, private information network or through public information network. The Ethernet network is has very interesting as public information network. Of course the Ethernet network has good (cheap, widely practiced in virtually every home with a phone line) and bad properties (sometimes reduction of throughput of speeds of transmission due to congestion). Teleoperation has a major role in controlling the remote camera and observation of the room via internet connections (Bratina et al., 2011).

In the recent years the use of teleoperation via the Internet has spread to areas outside the universities, particularly to industrial applications of management, data collection and monitoring of real industrial processes or subprocesses. Today there are already the first serious industrial applications (Muskinja et al., 2011), where the operator of the manufacturing process monitors the performance of the industrial process via a real time video system, or where this is not possible due to darknes, heat, dust, pressure ..., through a virtual interface connected on one side with the real system measuring systems giving virtual image of the process.

The user may through this human-implemented process make simulations in the virtual world (training of operations) and when he is satisfied with them, he easily transfers all the necessary information to the real system and initiates the implementation process on the real system. In the field of the teleoperation techniques the following major areas of application, summarized in the next paragraphs are met.

Teleoperation of sphere in the magnetic field (Golob, 1999). The experiment itself is performed with a metal ball floating in the magnetic field. Thus, when the density of magnetic field flow on the device increases or decreases the metal ball can be moved up and down. Of course, the ball position measured and gives feedback to the positioner. The user provides the desired amount of the balls and also the parameters of regulators (Figure 1). On this principle the magnetic bearing, operate.

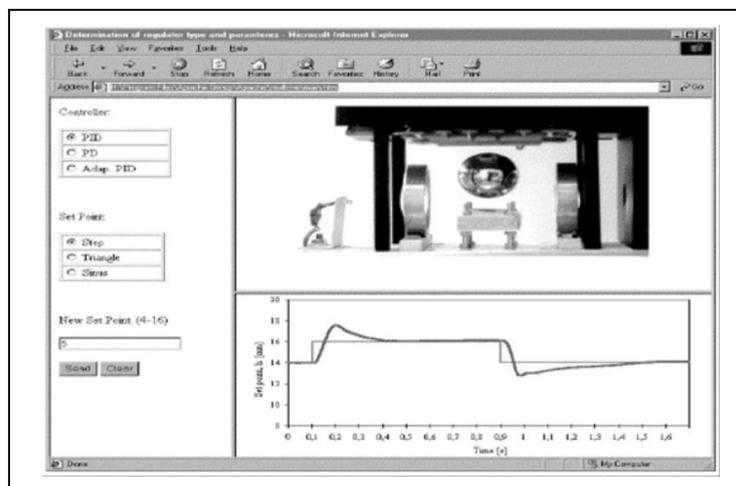


Fig. 1. An interactive Internet site for the teleoperation device of the magnetic sphere

The Internet teleoperation for common navigation and manipulation represents the general concept of the Internet teleoperation (Kamrani et al., 2005) which sets three level system architecture for teleoperation. The concept was used in developing the system for teleoperation that allows two human operators to safely manage the two cooperating mobile robots in unknown and dynamic environment from any two PCs which are connected to the Internet and on a specially developed program is installed - the client and which are connected to a simple joystick with a force feedback. The method of teleoperation based on events was used that synchronizes the actions of each robot with the operator's commands to eliminate instability as a result of unknown and varying time delays. Experimental testing confirms that the developed teleoperation system allows the operators successful joint navigation and manipulation in complex environments.

Random time delays can cause instability in a real time closed loop circuit which is based on the principle of telerobotic systems. Therefore is proposed (Wang et al., 2005) a new approach to modeling end-to-end delays on the dynamics on the Internet, where identification was used and it was made to control the system in real-time Internet telerobotic operations. In the case of a single model, the model must be adapted to the operating conditions, before the use of the appropriate mechanism for monitoring of the application, can start.

## **2. Design of the servodrive speed control using a software and hardware components**

The XPC Target system ensures the implementation of applications in the external mode. The idea is the application can take place on several host computers and one target computer that it is connected to the control system. In our case we had a host and a target computer. On the host computer a Matlab/Simulink 2008b is loaded with the addition of XPC Target, needed for our application. In the target computer a analogue/digital (AD) and digital/analogue (DA) cards were installed. Speed PI control was performed using the drivers that are represented by block DA1 (DA converter) and AD converter, in the Figure 2. In the driver a software algorithm was written that had the function of linking the AD, DA card and the target computer processor. For easier implementation a driver for the DA card in the Matlab/Simulink program was created. For a AD driver the C code must be written, because the blocks in Simulink block library do not support routines, needed to program a driver for AD card.

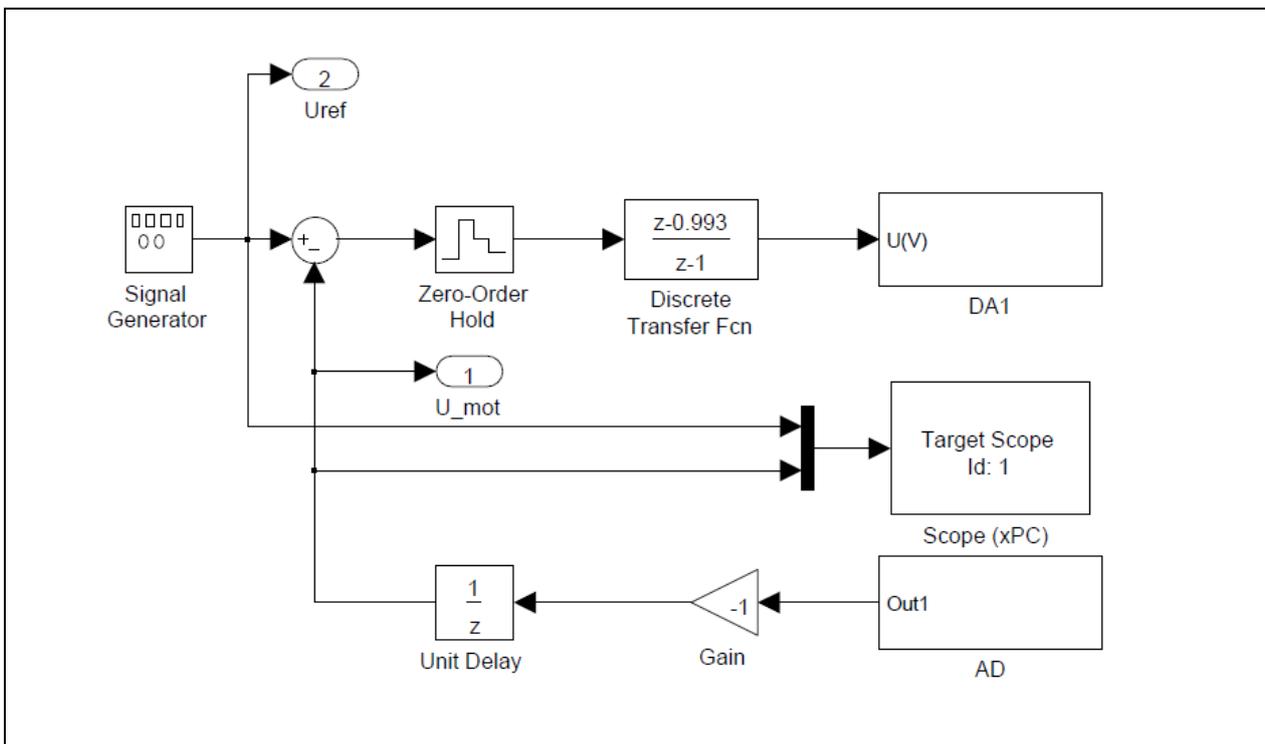


Fig. 2. Block diagram of the speed regulator in the servodrive teleoperation

Model which was set up in Matlab/Simulink program displays the servodrive speed control performed by means of teleoperation on the XPC target system. Speed PI-controller was performed using the drivers represented by the blocks DA1 on Figure 7 and the AD on Figure 8. The controller parameters were set by means of the simulation package Simulink. Onto the input DA converter was brought the real value of voltage obtained from the tachogenerator.

### 2.1 XPC-Target (addition to Matlab/Simulink 2008b)

XPC Target is a product of the company MathWorks for application of external mode operation. It is a target-host solution aimed at building prototypes, testing and developing of time systems. In this environment our desktop PC is used as a host computer with Matlab/Simulink 2008b, the Simulink program was used to create models, to drive the simulation and to generate the code. On the other PC which is a compatible system as a target PC the generated code is executed in real time. Options of XPC Target system are:

- To create and build the target applications on a host computer.
- Implementation and control of target applications on the target computer with signal acquisition and parameter setting.
- The use of inexpensive hardware for the computer on which the XPC target environment was loaded.

After completing of development and testing, the target application can be used as a timing system that runs on a specified target computer. This way of implementation of applications is possible with Matlab/Simulink 2008b with added XPC Target.

## 2.2 Software Matlab/Simulink 2008b

Matlab is a software package designed for numerical calculations, such as arithmetic operations of vectors, differential equations and presentation of results. For such extensive use the Matlab has available libraries through, which more knowledge about the regulation (continuous, discrete, fuzzy), designing of filters, XPC Target System can be reached. Entering commands into Matlab's workspace can be done directly through the command line in the command window or the commands are entered into a separate file with ending \*.m. Recording of the command in the Command window or in a file allows its repeated execution, without re-entering. Matlab's functions are easily completed by a user through the Matlab programm language. The basic program structure is the matrix. In the Matlab program it is not necessary to define variables and to reserve space in memory. Effective programming in Matlab is achieved by pre-booking of memory, the use of commands with a dot (they operate on the matrix components) and avoiding to use for loops.

Simulink is a graphical upgrade of Matlab 2008b. It is used for modeling and simulation of mathematical models. Models can be assembled with available blocks, so that detailed knowledge of Matlab commands is not required. Collections of blocks used for the synthesis of the fuzzy and classical regulator, can be found under the directory in Simulink. It allows simple graphical representation of simulation results. It can be run in Matlab environment by using Simulink or through the icons in Matlab Toolbar.

## 2.3 PC24 high-performance 4-channel digital/analogue converter card

The PC24 card is a 4-channel analog voltage output card with individual step selected output voltage levels in the range of 0 V to +2.5 V,  $\pm 2.5$  V, 0 V to +10 V,  $\pm 10$  V. For flexible addressing on the card it must be ensured that the base address is in the range of 000hex to FF0hex. Through a series of switches the selection of the interrupt area (IRQ2-IRQ7) can be determined. The card contains four 12-bit AD converters two ways of triggering, 1MHz frequency source, flexible addressing. The card also contains a programmable wait state generator for faster card functioning.

## 2.4 PC26AT high-performance 16-channel analogue/digital converter card

The PC26AT card contains 16-channels with 12-bit resolution. PC24 and PC26AT card were used in a compatible IBM XT, AT/386 computer. Components of PC26AT cards are:

- 16-channel analog input data channels with multiplexers that enable high-performance AD conversion.
- 14-input voltage measuring range with the values involved,  $\pm 1.0$  V  $\pm 2.0$  V  $\pm 5.0$  V and  $\pm 10$  V bipolar, and 2.0 V, 4.0 V and 10.0 V unipolar.
- Control 1 MHz crystal oscillator on the card for precise time measurement.
- Three 16-bit counters.
- High performance counter.

### 2.5 DC servodrive

In our application a DC servo motor type ESCAPE 28D11, with related data, was used as shown in Table 1 below.

Motor constants	Symbol	Value
Rated Voltage	$U_n$	12 V
Stream idle	$I_0$	0.044 A
Permissible maximum current	$I_{max}$	1.5 A
Inductance coil	$L_a$	0.0003 H
Winding resistance	$R_a$	2.5 $\Omega$
Rated engine speed	n	5800 rpm
Torque constant	$K_t$	0.0195 Nm/A
Constant voltage	$K_e$	0.01957 Vs/rad
Mechanical time constant	$T_m$	0.012 s
Moment of inertia of the engine	J	$17.6 \cdot 10^{-7} \text{ kgm}^2$
Viscous friction	B	$1 \cdot 10^{-6} \text{ Nms/rad}$

Tab. 1. Parameters of DC servo motor and tacho generator

The tacho generator gives us in its output the voltage that is proportional to the speed of rotation of the motor shaft. At the output of the tacho generator via signal adjustment (AD converter), the analog signal converted to digital signal with which the value of the rotation of the axis of the servo motor was presented. In our application the tacho generator DC motor was used, which was identical to a regulated direct current motor (ESCAPE 28D11). Connection between the regulated DC motor and tacho generator was made by means of the belt.

### 3. Modeling of control system in Simulink

The servodrive presented in Figure 3 is a speed controlled DC motor drive (Berk, 2001), adapted to the needs of the educational process. The device is designed to learn characteristics of the control loop and cascade control of a DC motor.

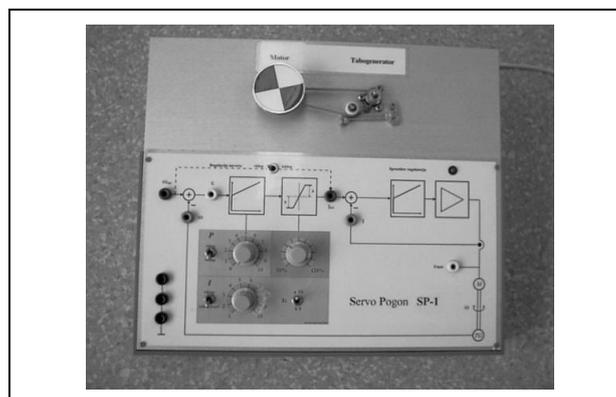


Fig. 3. Analog servodrive

Based on the mathematical model a model of DC servo type ESCAPE D-28 in the Simulink was created and represented by the equation below:

Equation of rotor of DC servo motor:

$$U_a = K_e \cdot d\varphi/dt + i_a \cdot R_a + L_a \cdot di_a/dt \quad (1)$$

Where:

- $U_a$  - supply voltage of dc servomotor
- $K_e$  - voltage constant
- $\varphi$  - swivel
- $i_a$  - the current through the coil of dc servomotor
- $R_a$  - winding resistance
- $L_a$  - coil inductance
- $t$  - time

DC servo motor torque equation:

$$T_e = J \cdot (d^2\varphi/dt^2) + B \cdot d\varphi/dt + T_L \quad (2)$$

Where:

- $T_e$  - electric torque of DC servomotor
- $J$  - moment of inertia of the engine
- $\varphi$  - swivel
- $B$  - viscous friction
- $T_L$  - torque

Then we have the equation of the rotor of DC servo motor and torque, were mapped in the s-area:

$$U_a(s) = K_e \cdot s \cdot \varphi(s) + I_a(s) \cdot R_a + L_a \cdot s \cdot I_a(s) \quad (3)$$

and

$$T_e(s) = J \cdot s^2 \cdot \varphi(s) + B \cdot s \cdot \varphi(s) + T_L(s) \quad (4)$$

Then the equation is translated in such form (Berk, 2001) that the block model of dc servo motor could be drawn as shown in Figure 4 below.

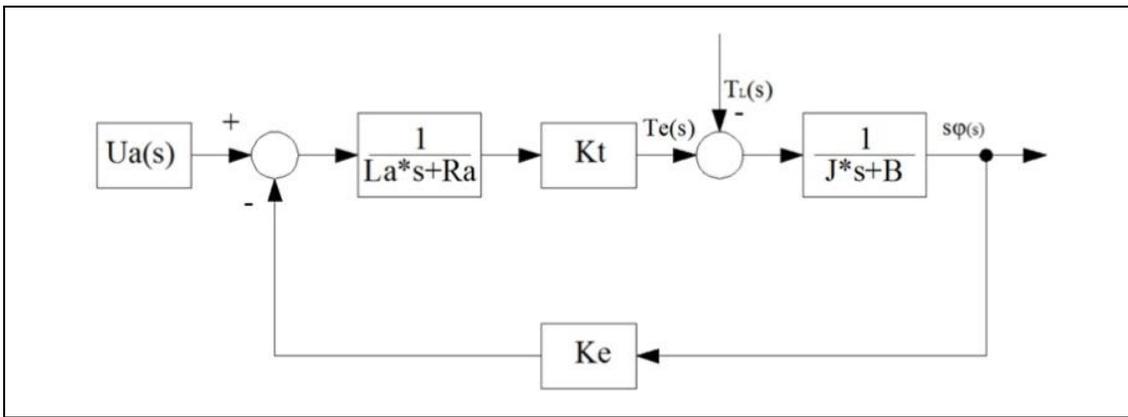


Fig. 4. Model of DC servomotor

From the model shown in Figure 4 it can be seen that the DC motor consists of electrical and mechanical parts. In the calculated total moment of inertia ( $J_S$ ) the moment of inertia of the motor rotor ( $J_M$ ), tacho generator ( $J_T$ ) and moment of inertia of the drive element ( $J_d$ ), attached to the motor shaft were considered (Berk, 2001) and then the final model of dc servo motor in the Matlab/Simulink 2008b was made as shown in Figure 5. After modeling of DC servodrive the model of the analog servodrive speed control was made as shown in Figure 6.

Synthesis of the speed PI controller of the analog servodrive were performed by Bode's method. Parameters obtained were not entirely satisfactory, because in our case we have to deal with the nonlinear system (in the control track, the limiter were we covered). Due to nonlinearity of the control loop the parameters had to be finely tuned by trial. Parameter setting procedure was repeated until the desired response was reached.

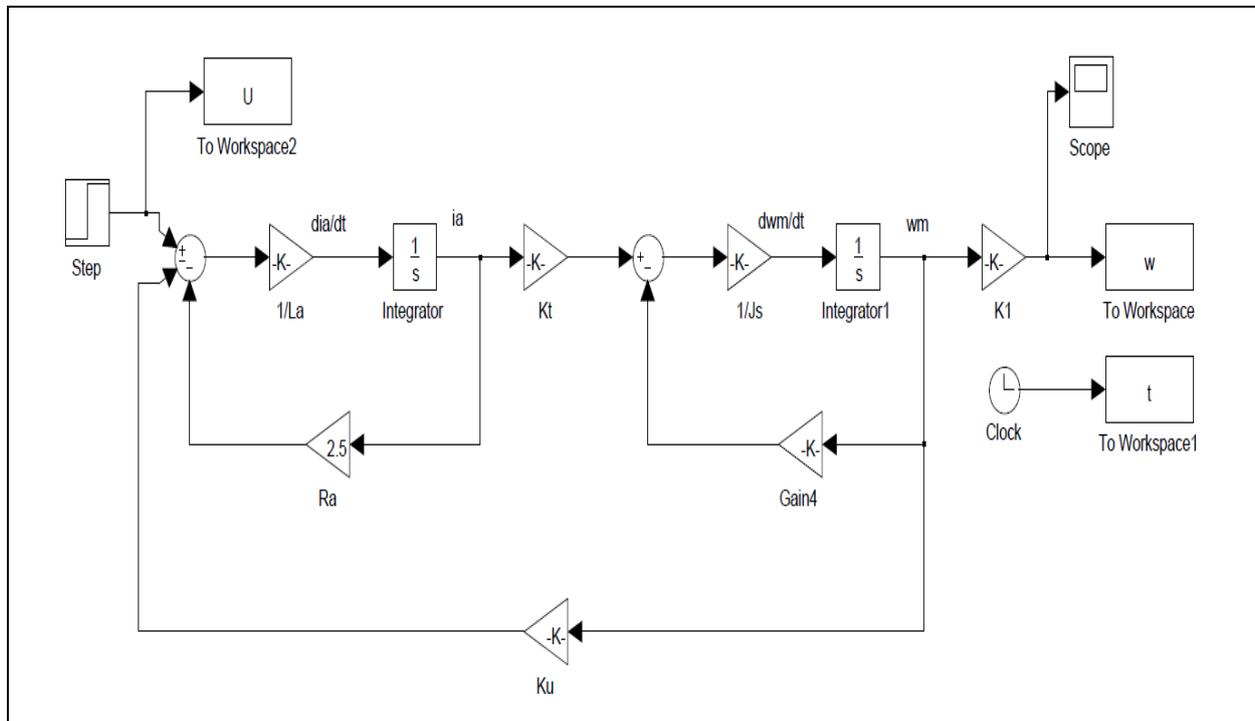


Fig. 5. Model of DC servo motor in Simulink



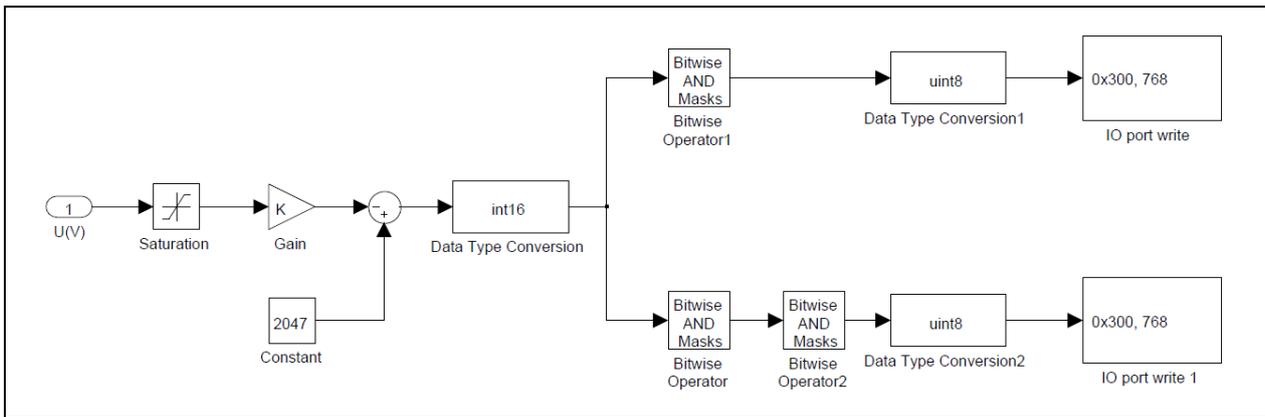


Fig. 7. Block diagram of DA1 block in Matlab/Simulink

### 3.2 Model of driver for the AD converter in the Matlab/Simulink

Content of the block of the AD converter are presented in Figure 8. With the help of block IO port read, a part of the program was written in C code. In C code, the initialization was covered. In the control PPI register, which is on the memory address of  $303_{16}$ , hexadecimal value 92 had to be entered, which means that the port A and port B are defined as inputs and the Port C as output. Of course, Port A, Port B and Port C are the PPI interface ports for the transfer of data words. It was necessary to write a function for AD conversion, for the specified channel.

Entering data word  $0x02$  at address  $302_{16}$ , means the choice of channel 0, bits PC4-PC7, which represent the multiplexer to choose the input channels are all set to zero. Also the bit PC1 is set, with which the program trigger was selected. Then the value  $0x03$  was entered in the register, which is located at  $302_{16}$ , which means the start of the positive front transformation. The end of the conversion was analyzed using bit B5 (ADC busy status). AD conversion data record is 12-bit and that data record is written into the registers that are on the memory address  $300_{16}$  and the  $301_{16}$ th. Into the register on the address  $300_{16}$ , lower eight bits are recorded, that are to be labeled A0-A7 and in the register on the address  $301_{16}$ , the upper four bits labeled B0-B3 are written. Data record is of type double, so in our case the double block was used, where also had to be performed scaling, using the gain block, in which the value  $10/32768$  was written down. At the output of the AD card the actual value of the voltage was obtained, which is then led to the adder of our speed control.

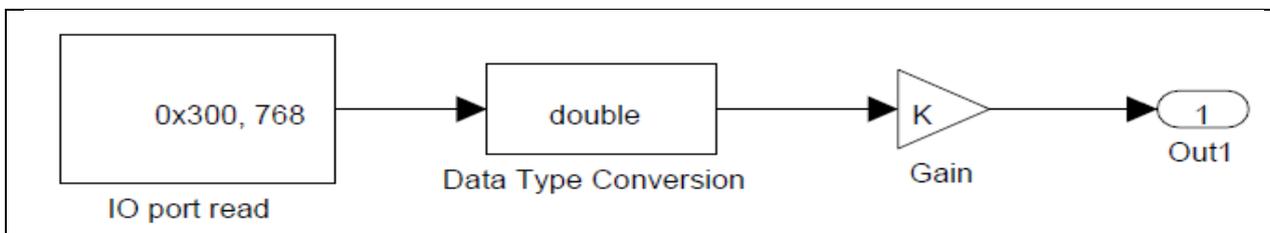


Fig. 8. Block diagram of AD block in Matlab/Simulink

## 4. Results

The simulation results first show the responses to the step in the DC servomotor model, made in the Matlab/Simulink, Figure 5. The graph in Figure 9

shows the reference value in the form of voltage, with the dc servomotor is excited. Figure 10 shows the rate of speed of DC servodrives excited by step voltage 12 V. Following successful test of the model, the DC analog servodrive speed control model (Figure 6) was tested, where constant values of  $K_p$  parameter (gain of PI controller) and parameter (integration time constant) were set until the desired response with 15% over swing was reached on in Figure 11. After finishing the settings the XPC Target environment with new parameters was updated, and the desired response of speed of servodrive with 15% over swing was reached, as shown in Figure 12 below.

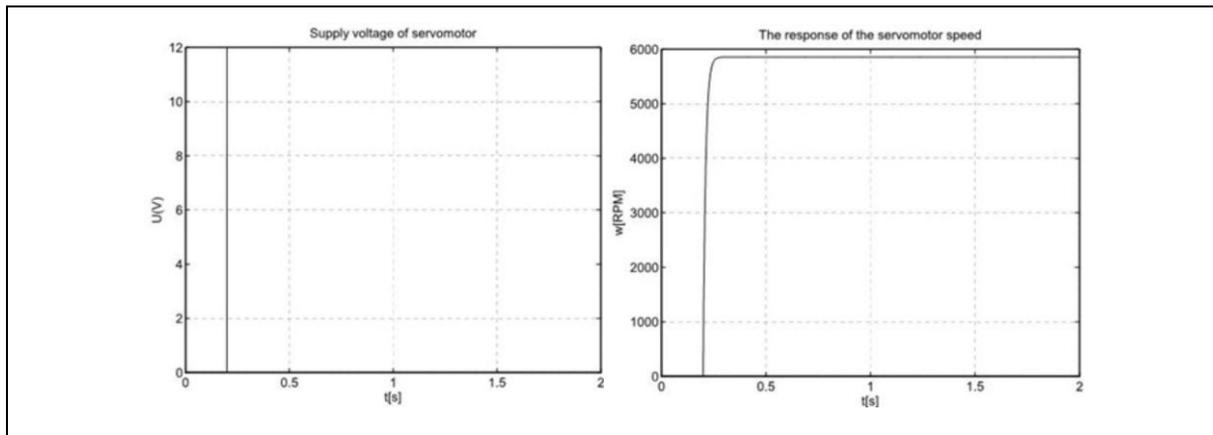


Fig. 9. Reference value

Fig. 10. Response to step

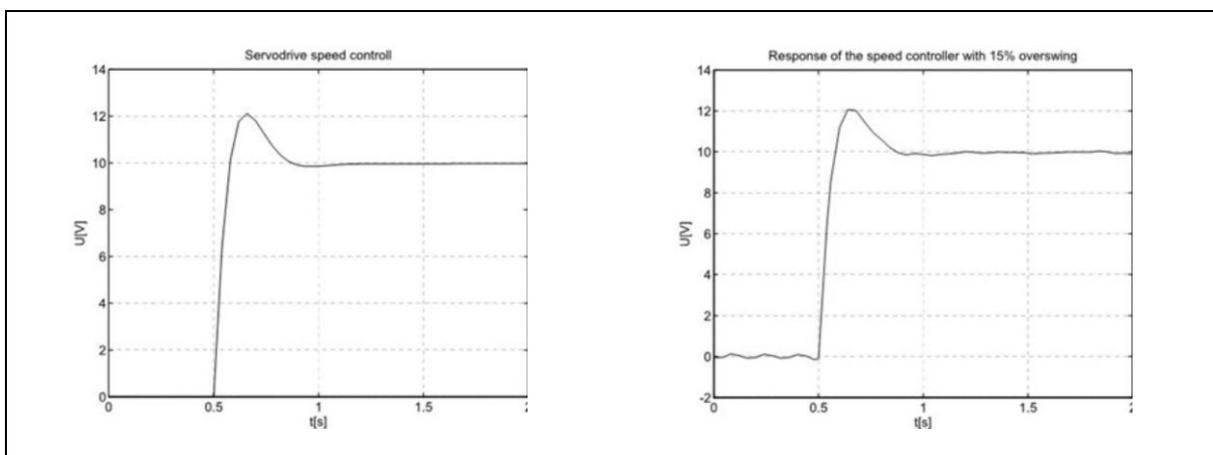


Fig. 11. Response to step of servodrive speed control

Fig. 12. Response rate with 15% over swing servodrive using the XPC Target System

## 5. Conclusions

In the application model, we have realized the speed control of servodrive speed control teleoperation on the principle of system, using the XPC target operating system. We met problems with the model of servodrive in Matlab/Simulink 2008b, for accurate determination of the parameters of speed PI controller, as the synthesis of PI-controller using Bode's method does not allow accurate adjustment of the parameters of speed PI-controller.

The problem occurred as a result of the real nonlinear system of model of servodrive due to the presence of limiters in the model. Responses on the principle of Simulink simulation model and real servodrive show that speed control is highly dependent on the constant of gain ( $K_p$ ) and on time constant  $T_i$ . Even in case small of changes in these values the response of the system is very much changed. If the constant value of constant  $K_p$  is increased, the gain is in the dynamics of the response of the control system, but the achievement of the desired position is aperiodic (overswing appears). Decreasing of the constant  $K_p$ , does away with the overswing, but the system becomes slower and the set time increases. It is very important to find an appropriate value of the  $K_p$ , which will compromise the two extremes. Using the time constant eliminates the static error in the control loop.

Servodrive speed control through teleoperation shows the advantages of this method. Possibility of remote management process is now increasingly interesting area (learning on the distance, control remote on the industrial processes ...). The idea of our application is that multiple users on the host computers communicate with one target computer. Appropriate area for further research can be seen in the design and implementation of applications on the same computer.

## 6. References

- Muskinja, N.; Riznar, M.; Tovornik, B.; Bolf, N. & Mohler, I. (2011). *Remote lab for process control education using OPC and web based technology*, V: Remote & virtual labs: exp.at'11, 1st Experiment@ International Conference, Lisbon, Portugal, pp 1-5
- Bratina, B.; Muskinja, N.; Golob, M. & Tovornik, B. (2011). *Fault detection and isolation remote laboratory concept and experiments*, V: Remote & virtual labs: exp.at'11, 1st Experiment@ International Conference, Lisbon, Portugal, pp 1-6
- Berk, P. (2001). *Teleoperating speed control of servo-drive with Simulink*, UDK: 621.313.2:004.7(043.2), University of Maribor, Faculty of Electrical engineering and Computer Science, Maribor, Slovenia
- Kamrani, E.; Momeni, H. R. and Sharafat, A. R. (2005). *A Novel Adaptive Control System for Stable Teleoperation via Internet*, ISBN: 0-7803-9354-6, 2005 IEEE Conference on Control Applications, Toronto, Canada, pp 1164-1169
- Wang, M. & James N.K. (2005). *Interactive control for Internet-based mobile robot teleoperation*, Robotics and Autonomous Systems, vol. 52, pp 160-179
- Golob, M. (1999). *Decomposition of a fuzzy controller based on the inference break-up method*, Intelligent data analysis, [Print ed.], vol. 3, pp 127-137
- \*\*\* (2001), [http://courses.washington.edu/css457/matlab/learning\\_matlab.pdf](http://courses.washington.edu/css457/matlab/learning_matlab.pdf)-The Mathworks, MATLAB student version, Accessed: 2012-01-06
- \*\*\* (2010), [http://www.mathworks.com/tagteam/65537\\_xpc\\_target\\_driver\\_ug.pdf](http://www.mathworks.com/tagteam/65537_xpc_target_driver_ug.pdf)-The Mathworks, xPC Target™ 4, Accessed: 2012-02-06