

Rapid prototyping approach for dynamic load emulation

Lazar Sladojević, Milutin Petronijević, Miodrag Stojanović

Abstract – In this paper the new solution to the problem concerning dynamic load emulation is presented. This paper represents the improvement of a compensator method application with linear PID controller. Detailed simulation results provide good starting point for controller tuning and verification of the main idea. New technique, based on rapid prototyping approach is described and the experimental results are presented. Simulation and experimental results are compared for emulation of linear and nonlinear loads. Some limitations in use of proposed controller are gained and future improvements are noticed.

Keywords – Dynamic load emulation, Mechanical load, Rapid prototyping, Matlab simulation.

I. INTRODUCTION

In order to anticipate the performance of a designed system, engineers regularly use some of the well-known simulation tools. Today, the most of the simulations are carried out on digital computers. Mathematical models of systems are created and their behavior is inspected.

However, in order to overcome a number of uncertainties, sometimes the laboratory prototyping is recommended. Unfortunately, this is not always possible to achieve. A good example of this would be an electrical drive that is designed to work with some kind of nonlinear load. This is why the dynamic load emulation was developed. It is a mean of “hardware simulation” of any drives load condition that can be mathematically described. The dynamic load emulation procedure consists of two drives rotating on the same shaft with rigid coupling. One of them is the drive under test (DUT), the one whose behavior under certain conditions is inspected. The other one is the drive “acting” as the load, often called dynamometer (DYN). It doesn’t have to be the same kind of drive as the drive under test, but it has to be torque-controllable.

There are four most commonly used methods for achieving the dynamic emulation:

1. The compensation method [1]
2. The feedforward regulation method [2]
3. The feedback regulation method [3]
4. The PI estimation method [4]

Lazar Sladojević, Milutin Petronijević and Miodrag Stojanović are with the Department of Power Engineering, Faculty of Electronic Engineering, University of Niš, Aleksandra Medvedeva 14, Niš
 E-mail: lazar.sladojevic@elfak.rs, milutin.petronijevic@elfak.ni.ac.rs, miodrag.stojanovic@elfak.ni.ac.rs

This paper will address the compensation method given in [1] and a way of its improvement. The theoretical hypothesis will be simulated using the Matlab/Simulink simulation software and experimentally proven using the experimental set-up described in section IV. The algorithm for dynamic emulation is developed in Simulink and transferred to a dSpace development platform which controls both machines. The DUT is a vector controlled, squirrel cage induction motor. The DYN is also a squirrel cage induction motor, but its torque is controlled directly, via a DTC drive. The proposed emulator will later be used for testing of speed controller robustness and performances.

II. Traditional compensation method

A. The inverse dynamics model

The most basic idea of a dynamic emulation algorithm is shown in Fig 1. This is so called inverse dynamics approach. The idea is very simple. The shaft speed is estimated from position encoder signal and used for calculation of a necessary load torque which is to be applied by a dynamometer. The flaw of this approach is the fact that it needs the inverse mechanical characteristic, i.e. the load torque as a function of speed:

$$T_l = f(\omega_r) \tag{1}$$

where T_l is the load torque and ω_r is the angular speed of the rotor.

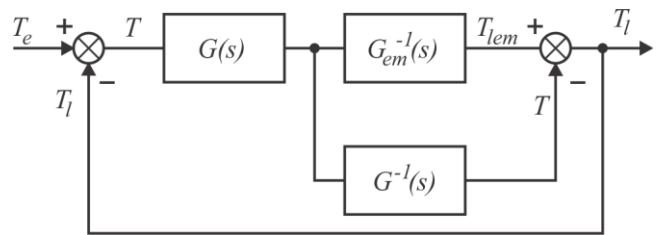


Fig. 1. The inverse dynamics approach to dynamic load emulation

On Fig. 1, $G_{em}(s)$ is the transfer function of an emulated load, while $G(s)$ is the transfer function of a mechanism - two rotors and a shaft.

$$G(s) = \frac{1}{J_ms + B_m} \quad (2)$$

The characteristic from Eq. (1) is practically unachievable since it requires the derivative of speed over time. For example, a simple linear load inverse mechanical characteristic is given in Eq. (3):

$$T_l = J \frac{d\omega_r}{dt} + B\omega_r \quad (3)$$

where J is the moment of inertia and B is the viscous friction coefficient of a linear load.

Such an approach would be impractical because measurement of acceleration is very submissive to noise. Further, it can also exhibit stability problems when the algorithm is implemented digitally. This is why the different approach was developed.

B. Regulating the speed instead of torque

The different approach is depicted in Fig 2. Consider for the beginning that $G_{comp}(s) = I$. The transfer function of speed with respect to the driving torque is as follows:

$$\frac{\omega_r(s)}{T_e(s)} = G_{em}(s) \frac{G(s)G_t(s)}{1 + G(s)G_t(s)} \quad (4)$$

where $T_e(s)$ is the driving motor torque, $G_{em}(s)$ is the emulated load transfer function, $G(s)$ is a transfer function of a mechanism and $G_t(s)$ is an emulated speed PI controller as shown in Fig. 2. The desired speed-torque characteristic is, however, given with Eq. (5).

$$\frac{\omega_r(s)}{T_e(s)} = G_{em}(s) \quad (5)$$

Eqs. (4) and (5) have different expressions on their right-hand sides, so the compensator is introduced to neglect these differences. Its transfer function is obviously:

$$G_{comp}(s) = \frac{1 + G(s)G_t(s)}{G(s)G_t(s)} \quad (6)$$

By evaluating this transfer function, however, it can be seen that that it is improper. The degree of numerator is higher than degree of denominator. One possible solution of this problem is shown in this paper.

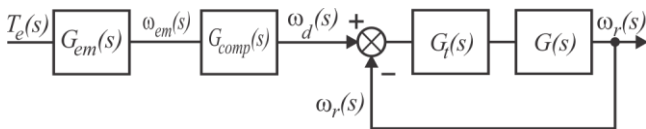


Fig. 2. The traditional compensation principle

C. Existing solution of a compensator problem

Before the suggested solution is explained, the existing solution has to be reviewed. It is adopted from [1] and consists of a modification of a compensator transfer function in discrete domain. Instead of continuous, the discrete compensator is used, with its discrete transfer function being:

$$G_{comp}(z) = \frac{1 + G(z)G_t(z)}{z G(z)G_t(z)} \quad (7)$$

In order to compensate for the introduced delay, the transfer function of an emulated load has to be modified too. It now becomes:

$$G_{em}(z) = z \cdot Z\{G_{ho}(s)G_{em}(s)\} \quad (8)$$

where $G_{ho}(s)$ is the transfer function of a zero-order hold function used to discretize $G_{em}(s)$. The newly formed compensator can be realized in the digital processor which is normally used to execute the algorithm.

III. THE SUGGESTED DYNAMIC LOAD EMULATION APPROACH

This chapter covers the suggested approach to solving the compensator problem. The simplest way of designing the emulated speed controller, $G_t(s)$ is also shown.

By introducing a differential effect into a PI controller, $G_t(s)$ i.e. creating a PID controller, the degree of denominator in the compensator transfer function raises by one, and is now equal to the degree of the numerator. Such a compensator can be realized and simulated, even in continuous time domain. The differential effect also enhances dynamics of the system, especially when sudden changes in reference occur. The new scheme is very similar to the one from Fig. 2. The main difference is in the new transfer function of a controller, which is given in Eq. (9). Also, one more block is added to the regulation loop which is shown on the right side of Fig. 2. This block is mainly added for a purpose of calculating the parameters of the PID controller, but it also brings the whole model closer to reality. It accounts for any delay caused by the time necessary to execute the control algorithm, time needed for acquisition of data, calculating the necessary variables, delay caused by the inner current loops etc. This delay is described by the term T_{ek} . The new loop is shown in Fig. 3. The transfer function of the added block is simply:

$$G_{alg}(s) = \frac{1}{1 + sT_{ek}} \quad (9)$$

The transfer function of an ideal PID controller is given with:

$$G_t(s) = G_{PID}(s) = K_p + K_i \frac{1}{s} + K_d s \quad (10)$$

where K_p is the proportional gain, K_i is the integral gain, and K_d is the differential gain of a PID controller.

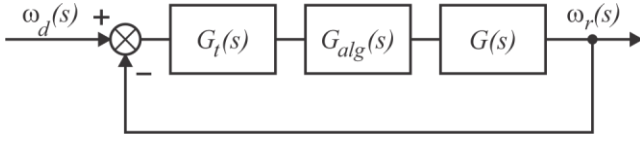


Fig. 3. The control loop of a suggested method

Although an ideal controller can't be made, it is possible to use Eq. (10) to calculate parameters of a real controller. The real PID controller's transfer function is given in Eq. (11):

$$G_{PID-real} = K_p + K_i \frac{1}{s} + \frac{K_d}{K_p} \cdot \frac{s}{1 + sT_D} \quad (11)$$

where T_D is filtering time constant of a differential term.

In order to calculate the parameters, the scheme on the Fig. 3 is used. Since the controller regulates the emulated speed, ω_{em} , that is, tries to make the real rotor speed equal to the emulated one, a simple feedback loop on the Fig. 3 provides for a characteristic equation of a controlled system. It is derived from Eq. (12):

$$1 + W_0 = 0 \quad (12)$$

where $W_0(s)$ is the open loop transfer function of a loop from Fig. 3. The characteristic equation in its extended form is given in Eq. (13):

$$s^3 + \frac{J_m + T_{ek}B_m + K_d}{J_m T_{ek}} s^2 + \frac{B_m + K_p}{J_m T_{ek}} s + \frac{K_i}{J_m T_{ek}} = 0 \quad (13)$$

In the previous equation, J_m is the moment of inertia and B_m is a viscous friction of a motor drive mechanism, respectively.

The previous equation is a characteristic equation of a third order system. It can be mathematically represented as a juncture of two systems: one of the order of two, and one of the first order. The goal is to have the poles of an order two system complex and dominant and the third pole should be negative, non-dominant and as far from imaginary axis as possible. Having in mind previous, the Eq. (13) can also be written in the following form:

$$(s - s_3) \cdot (s - s_2) \cdot (s - s_1) = 0 \quad (14)$$

where s_1 , s_2 and s_3 , are the roots of the Eq. (13) and are given in their well-known forms:

$$s_{1/2} = -\zeta\omega_n \pm j\omega_n\sqrt{1-\zeta^2} \quad (15)$$

for the second order system and:

$$s_3 = -k_3\omega_n \quad (16)$$

The ζ and ω_n are the dumping coefficient and a natural frequency of a second order system, respectively. The value $k_3=40$ in the Eq. (16) is selected based on detailed simulation of whole system.

The recommended values of ζ and ω_n are given in [5]. They are:

$$\xi > 0.6 \cdot (1 - M_p) \quad (17)$$

and:

$$\xi\omega_n \geq \frac{4.6}{t_s} \quad (18)$$

where M_p is the overshoot in pu and t_s is the settling time in seconds. Those two are the common parameters that are optimized in the process of adjusting the parameters of a controller.

With ζ and ω_n determined, the only thing left to do is to replace those values in Eqs. (15) and (16), calculate the poles of the system, and replace those values into Eq. (14). By comparing the Eqs. (13) and (14), the parameters of PID controller are obtained.

IV. SIMULATION RESULTS

The simulations were carried out for the torque regulation of a linear load and a two-mass system with backlash. The machines that were used for simulations are the ones that were used in real experiments. Linear load is simulated simply, by using its Torque - speed characteristic given with Eq. (19):

$$G_{em}(s) = \frac{1}{J_{em}s + B_{em}} \quad (19)$$

The reference q-axis current of vector-controlled DUT is a ± 1 A square signal with the period of 0.8 seconds and duty cycle of 50%. It is assumed that there is nominal magnetic flux in the machine.

Simulation results for a linear load of different inertias and friction coefficients are given in figure 4. Fig. 4 a) represents linear load with the parameters $J_{em} = J_m$ and $B_{em} = B_m$, Fig. 4 b) load with parameters $J_{em} = 2J_m$ and $B_{em} = 20B_m$, while Fig. 4 c) shows the response of a linear load with parameters $J_{em} = 5J_m$ and $B_{em} = 50B_m$.

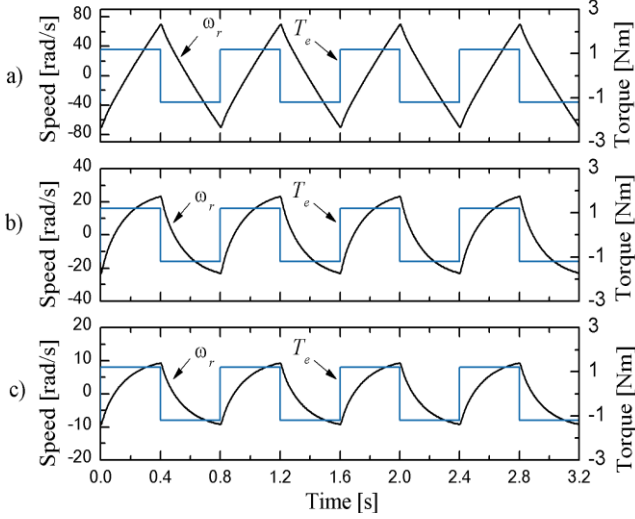


Fig. 4. Speed response of an emulated linear load

The next system for emulation is the simplest two-mass system with backlash as described in [6]. Mathematical model of a two mass-system is given by the following set of equations:

$$\frac{d\omega_{em}}{dt} = \frac{1}{J_m}(T_e - T_s - B_m\omega_{em}) \quad (20)$$

$$\frac{d\omega_{em2}}{dt} = \frac{1}{J_{em}}(T_s - T_{ext} - B_{em}\omega_{em2}) \quad (21)$$

$$\omega_{diff} = \omega_{em} - \omega_{em2} \quad (22)$$

where ω_{em} is the goal, emulated speed, T_e is the driving motor (DUT) torque, ω_{em2} is the dynamometer (DYN) rotor speed, T_{ext} is the external torque applied to the system (if it exists) and J_{em} and B_{em} are the emulated moment of inertia and viscous friction, respectively. The following equations describe the backlash in the system:

$$T_s = k_s D_a(\varphi_{diff}) \quad (23)$$

and:

$$D_a(\varphi_{diff}) = \begin{cases} \varphi_{diff} - \alpha, & \varphi_{diff} > \alpha \\ 0, & |\varphi_{diff}| < \alpha \\ \varphi_{diff} + \alpha, & \varphi_{diff} < -\alpha \end{cases} \quad (24)$$

where k_s is the elasticity of shaft, $D_a(\varphi_{diff})$ is a dead zone function, with α being the width of a dead zone and φ_{diff} is a difference in shaft angles of the two machines. The term φ_{diff} is obtained by integrating ω_{diff} , Eq. (22). Simulink model of a two-mass system is given in Fig. 5.

Simulation of a two-mass system was carried out with the following parameters:

- Fig. 6 a): $J_{em} = J_m$, $B_{em} = B_m$, $k_s = 2$, backlash of 15 degrees
- Fig. 6 b): $J_{em} = J_m$, $B_{em} = 20B_m$, $k_s = 2$, backlash of 15 degrees

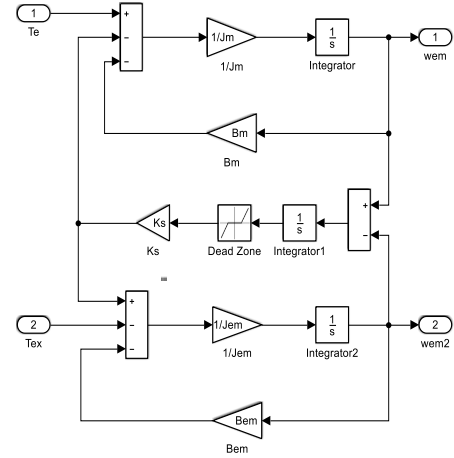


Fig. 5. Simulink model of a two-mass system with backlash

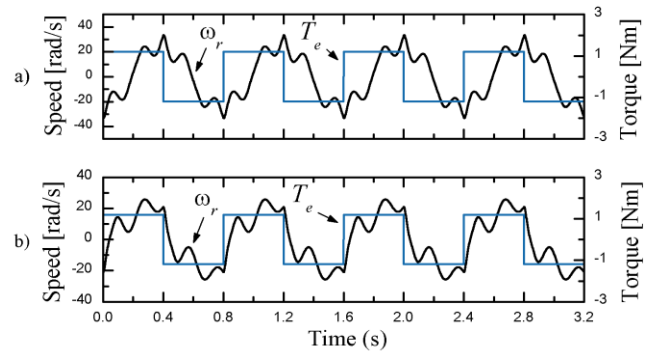


Fig. 6. Speed response of an emulated two-mass load with backlash.

It is to be observed from Figs. 5 and 6 that the system is acting as expected, which means it is well modeled. But this is yet to be confirmed with the experiments. Experimental setup and obtained results are shown in the following chapter.

V. Experimental results

The experiments were carried out according to the setup from Fig. 7. The emulation algorithm was developed in Simulink and transferred to a dSpace platform [7] which is used to execute it. This is so called rapid prototyping approach. Rapid prototyping is a great way of developing all kinds of control algorithms and bridging the gap to the practical application. After building a block scheme in Simulink, it can easily be translated into execution code and transferred to controller. Once loaded into the DSP, the

algorithm is ready for execution. The experimental software – Control Desk, also allows creating a custom control panel for the experiment monitoring from which the important variables can be observed and acquired. DSP system controls both drives.

The drive under test is a vector controlled induction motor, while the drive emulating the load is a Direct Torque Controlled (DTC) drive. Same emulated load conditions as in simulations were used.

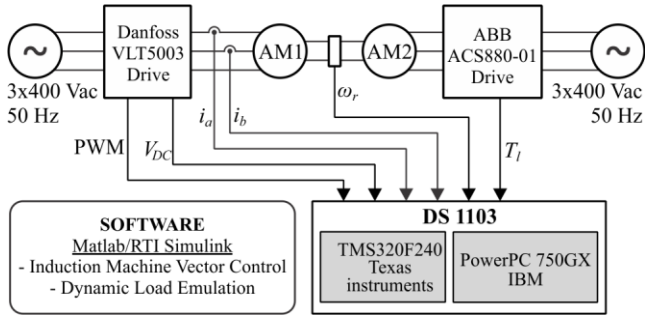


Fig. 7. Laboratory setup for the experiments

The machines that were used are standard squirrel-cage induction machines. Their most important parameters are given in Table I.

TABLE I
PARAMETERS OF THE MACHINES USED IN EXPERIMENTS

Rated power - P_n [W]	1500 W
Rated speed - n_n [o/min]	2860 o/min
Pole pairs - p	1
Rated phase voltage - U_{sn} [V]	230 V
Rated phase current - I_{sn} [A]	3.4 A
Stator resistance - R_s [Ω]	5.45 Ω
Stator leakage inductance - L_{ls} [mH]	11.8 mH
Rotor resistance - R_r [Ω]	3.18 Ω
Rotor leakage inductance - L_{lr} [mH]	11.8 mH
Magnetizing inductance - L_m [mH]	294.2 mH
Drive total inertia - J_m [kgm ²]	0.0035 kgm ²
Friction coefficient - B_m [Nms]	0.0022 Nms

Fig. 8 shows experimental results for linear load with the following parameters:

- Fig. 8 a) - $J_{em} = J_m$ and $B_{em} = B_m$
- Fig. 8 b) - $J_{em} = 2J_m$ and $B_{em} = 20B_m$
- Fig. 8 c) - $J_{em} = 5J_m$ and $B_{em} = 50B_m$

Experimental results for a two-mass system with backlash are shown in Fig. 9 in the following order:

- Fig. 9 a): $J_{em} = J_m$, $B_{em} = B_m$, $k_s = 2$, 15 degrees backlash

- Fig. 9 b): $J_{em} = J_m$, $B_{em} = 20B_m$, $k_s = 2$, 15 degrees backlash.

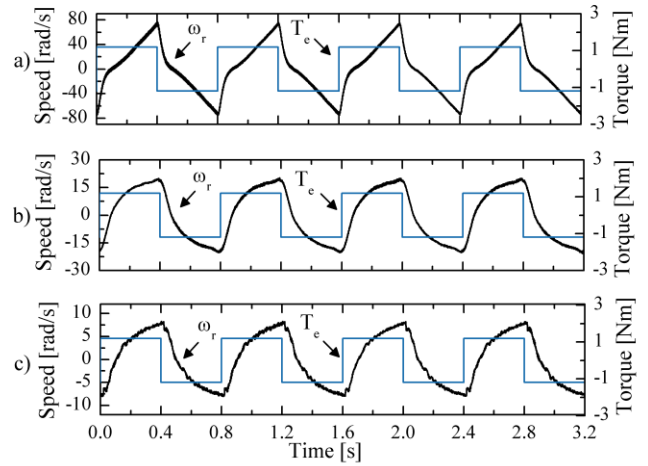


Fig. 8. Experimental speed response of an emulated linear load

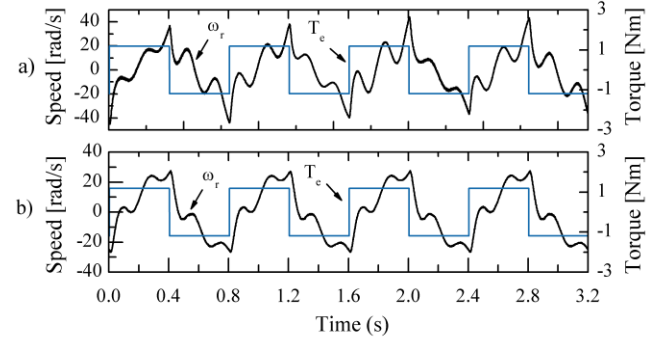


Fig. 9. Experimental Speed response of an emulated two-mass load

From Figs. 4 a) and 9 a) it can be seen that the experimental results do not fully correspond to the simulation results for the low values of inertia and friction. That reason for that lies in the fact that the parameters of PID controller do not provide exact tracking capabilities. The approach used to calculate those parameters utilizes the transfer function of an ideal PID controller, which doesn't fully relate to the real one. In practice, the controller is realized digitally, with low pass filtration applied to differential effect of the controller. This part has high influence on the dynamics of the system and must be designed with great care. If this is not the case, the impact of unmodeled dynamics is higher, and the results obtained do not match the simulation results.

For higher values of inertia and friction, conducted experiments give much better results. Both shape and maximal values of the speed response from Fig. 4 b) and c) are very similar to ones from Fig. 9 b) and c). Unmodeled dynamics do not have as high impact as before because the

whole system is slower.

Similar conclusion can be made for two-mass system with backlash. Speed responses from Figs. 6 and 10 are of similar shape and values, but there is some unmodeled dynamics that is not compensated. Therefore, the parameters of PID controller should be optimized or the application of nonlinear controller can be a promising solution for better tracking.

VI. Conclusion

In this paper, the rapid prototyping approach was used for dynamic load emulation. A short review of a design process is given in order to give initial parameters of the controller. The used algorithm represents a modified compensation method. Instead of PI, a PID controller was designed.

First stage in design are simulations in the Matlab/Simulink software. These simulations contain all elements that are described in this paper and later used in practical implementation. Linear load with different parameters was simulated first. Three different sets of parameters were adopted and the simulation results were shown. After that, the two-mass system with backlash was simulated. Two different sets of parameters were chosen and the corresponding results are presented. The results show that the system is acting as expected.

Experimental verification of results is depicted in Chapter V. The same load models with the identical parameters as in simulations were used. The experimental results show good agreement with the simulation results for higher values of emulated inertia and friction. For lower values of inertia and friction, however, the differences between simulations and experiment are bigger. This implies that there is a certain amount of unmodeled dynamics in the simulation model that the provided PID controller is not able to compensate. Further research in

this field is required in order to improve the performance of a tracking controller. The experimental results for two-mass system with backlash show the similar performance.

Nevertheless, the whole research shows a great potential of the dynamic load emulation with rapid prototyping approach. Many different types of load can be emulated and tested. Switching from one type to another is very easy and intuitive, and only takes several minutes. This technique will later be used for testing of more advanced speed and torque controllers for motors.

REFERENCES

- [1] Z. Hakan Akpolat, Greg M. Asher, Jon C. Clare "Dynamic Emulation of Mechanical Loads Using a Vector-Controlled Induction Motor-Generator Set", IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS, VOL. 46, NO. 2, APRIL 1999.
- [2] M. Rodič, K. Jezemik, M. Trlep, „A Feedforward Approach to the Dynamic Emulation of Mechanical Loads“, 35th Annual IEEE Power Electronics Specialists Conference, 2004
- [3] M. Rodič, K. Jezemik, M. Trlep, „Dinamična emulacija mehanskih bremen“
- [4] M. Rodič, K. Jezemik, M. Trlep, “Dynamic emulation of mechanical loads - approach based on nonlinear control“
- [5] B. Veselić, „Metode digitalnog upravljanja i estimacije“, Elektronski fakultet Niš, 2017 godina
- [6] K. Kyslan, E. Kušnir, V. Fedák, M. Lacko, F. Ďurovský, „Dynamic Emulation of Mechanical Loads with Backlash Based on Rapid Control Prototyping“, 16th International Power Electronics and Motion Control Conference and Exposition, Antalya, Turkey 21-24 Sept 2014
- [7] DS1103 Hardware Installation and Configuration for release 5.2, dSpace GmbH, 2006