Maciej SAJKOWSKI, Krzysztof KRYKOWSKI

Silesian University of Technology, Faculty of Electrical Engineering, Department of Power Electronics, Electrical Drives and Robotics
doi:10.15199/48.2015.06.23

Control system design using rapid prototyping method on the examples of voice controlled mobile robot and 6dof parallel manipulator

Abstract. *The subject of the paper is a description of a process of a control system development using a rapid prototyping approach. The purpose of this article is to present a control system design process on two examples of electromechanical devices. The first example is a mobile wheeled robot, guided by operator's voice commands and the second example is a parallel manipulator based on permanent magnet brushless DC (PM BLDC) motors.*

Streszczenie. *Przedmiotem artykułu jest opis procesu projektowania systemu sterowania z wykorzystaniem metody szybkiego prototypowania. Celem publikacji jest przedstawienie tego procesu w oparciu o dwa przykłady urządzeń elektromechanicznych. Pierwszym przykładem jest kołowy robot mobilny, kierowany za pośrednictwem głosowych poleceń operatora, drugi przykład dotyczy manipulatora równoległego napędzanego sześcioma silnikami bezszczotkowymi prądu stałego z magnesami trwałymi.* ***(Opracowanie systemu sterowania przy użyciu metody szybkiego prototypowania na przykładach robota mobilnego kierowanego poleceniami głosowymi i manipulatora równoległego o sześciu stopniach swobody)***

Keywords: rapid prototyping, control system, parallel manipulator, mobile robot, voice control
Słowa kluczowe: szybkie prototypowanie, system sterowania, manipulator równoległy, robot mobilny,sterowanie głosem

Introduction

The subject of the paper is a development of control systems using rapid prototyping method. We present the approach on two examples from area of electric drives and robotics. First of them concerns voice controlled, wheeled, semi-autonomous robot and the other example is a part of 6DOF (six-degree-of-freedom) parallel manipulator drive based on six PM BLDC (permanent magnet brushless DC) motors. The applied methods of control are called "Software in the Loop" and "Hardware in the Loop" [7], it is one of modern ways of control system design, development and testing. We have applied MATLAB-SIMULINK software for the development of the mentioned control systems. We used a special toolbox called xPC target, available within that environment. That toolbox gives possibility of turning typical PC (personal computer) into real time control device. That method is an alternative to applications of the other target systems, like digital signal processors (DSP), Field Programmable Gate Arrays (FPGA) or graphical processing units (GPU), what is proposed in case of SIMULINK model design in paper [4]. Here we shall present a case study of process of selecting such minimizer for an example CAD system. The stochastic optimizer will be compared to the deterministic one in terms of the simplicity, robustness, quality of the results and speed. Hardware in the loop method allows simulating real world conditions using typical personal computer as real time computational unit. There are some literature examples, where that approach is described in various applications. Electric vehicle model implementing vehicle dynamic and changing driving conditions as well as power battery model are described in the paper [20]. Complex environment is modelled in order to develop and test battery management system designed for electric car. Another similar example of hardware in the loop methodology is a model of hybrid power system described in the paper [19]. Designing control system for electric drives could be supported by described rapid prototyping methodology. An example of hardware in the loop application in electric drives is described in paper [15]. The paper refers to development of simulation platform for induction motor operating in precision servo drive. This example is illustrative of how control system architectures could be reconfigured in a short time depending on research demands [8]. Research described in article [18] can be categorized as application of "software in the loop" approach, where the controller software code is executed under control of real time operation system in order to replace the hardware controller of the device. This method is very useful in development of control systems. As it comes from the publication [10] this approach has not only been successfully used for more than ten years, but also gives possibility of building complex control system consisting of a few controllers based, for example, on neural network. Development of such control software without the support of simulation model is a complex process, difficult to formalize, which may lead to control system errors. Moreover, described methodology is applicable for plants based on various physical phenomena like electromagnetic levitation described in cited article. On the basis of publication [16], it could be stated, that rapid prototyping methodologies are also applicable in robot development. It seems that the boundary between "hardware in the loop" and "software in the loop" approaches is not strictly defined. The publication [1] can be used as an example, where control of the quadrotor aerial vehicle is described. In this case control system consists of two controllers: low level for motors control and high level based on xPC real time environment. The other mentioned controller is responsible for data acquisition, processing using flight dynamics model, and generating the output control information for the motors. This is consequent with "software in the loop" approach, but in flight dynamics subsystem there should be some elements of real world environment, which is typical for "hardware in the loop" approach. In our research we have mainly designed the control software, which controls in real time real devices operating in real world environment. Another publication [6] in that area concerns simulation of a unmanned aerial vehicle in real time. As it comes from the mentioned papers "hardware in the loop" approach is especially useful in case of objects, which require large area or space for operation, so it is difficult to do research in limited space of laboratory. The other papers [2], [3], [5], [9] and [14] are evidence, that rapid prototyping approach, especially based on xPC target technology is widely used in area of aircrafts research and development, where high reliability is essential. As it comes from the literature analysis, development of the control systems using rapid prototyping approach reflects current trends in systems engineering. The paper presents a step by step presentation of control system design, what could be useful for the other researchers.

The control system for mobile robot guided by voice commands

We have developed a mobile robot as an experimental platform, designed for the development of control algorithms on the level of drive control as well as on the level of master control based, for example, on voice processing in connection with robot surroundings analysis. A view of the described platform is presented in Fig. 1.

Fig. 1. View of the wheeled robot used in research

The robot is driven by differential drive, which consists of two low power dc motors. This machine is powered from DC lead acid 12 Volts battery. Two drive wheels are propelled by separate motors through integrated reducing gears. There are also incremental encoders mounted on the motor shafts. Stability of the robot framework is achieved by using additional passive wheel mounted on swivel. The DC motors are controlled by an microprocessor based on board. The controller is equipped with a dual H-bridge circuit, which converts PWM and control signals from microprocessor into appropriate rotation direction and velocity of each motor. The other function of the controller is operation of four ultrasonic range sensors, which are mounted in front of the robot. Basic control of that machine type comes from the kinematics, which is presented in the Fig. 2.

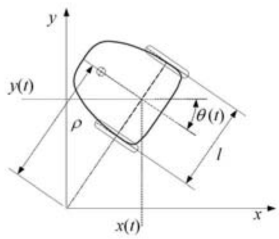

Fig. 2. Kinematic structure of a differential drive wheeled mobile robot

Motion of the robot is based on the kinematics presented in fig Fig. 2. It can be described with the generalized coordinates **q** and controlled by two control values **u**, corresponding to right and left velocities of the wheels.

(1)
$$\mathbf{q} = (x, y, \theta)^T$$
$$\mathbf{u} = (v_\mathbf{r}, v_\mathbf{l})^T$$

where:

(x, y) the pair is Cartesian coordinates of the mobile robot main wheel axis central point,

θ is an angle of orientation of mobile robot and is subtended between the longitudinal axis of symmetry of the vehicle and the ordinate of a base coordinate system X,

l is a robot wheel axis length,

v_r is a linear velocity of the robot right wheel,

v_l is a linear velocity of the robot right wheel,

ρ is a radius of turn of the vehicle.

The relationship (2) determines the radius of turn of the described vehicle.

(2)
$$\rho = \frac{l(v_r + v_l)}{2(v_r - v_l)}$$

From the velocities of the vehicle wheels the rotation ω and translation v velocities of the whole vehicle can be obtained (3)

(3)
$$v = \frac{1}{2}(v_r + v_l)$$
$$\omega = \frac{1}{l}(v_r - v_l)$$

The formulas above are direct kinematics task, which should be solved for example in case of trajectory visualization. Finally, relationship (4) defines generalized coordinates change coming from previously defined velocities of the vehicle.

(4)
$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{pmatrix} = \begin{pmatrix} \cos\theta \\ \sin\theta \\ 0 \end{pmatrix} v + \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \omega$$

The inverse kinematics, which is essential in case of control, comes from converting the formula (2) into parametric equation, where radius of turn is specified, and velocities of the wheels are calculated. There is an important implication for differential drive, that turn is possible when velocities of the wheel are different, but in case of equal velocities the vehicle moves straight ahead or straight backward. We have implemented the formulas listed above in MATLAB-SIMULINK environment in order to visualize and record the trajectory of the controlled vehicle. Determination of the vehicle turn radius was based on current linear velocity and setpoint of the wheels velocities difference, specified on discrete level. We have determined a set of phrases in Polish, due to the control of the robot by operator's voice commands. There are, among others, commands for accelerating, decelerating and changing the wheels velocities difference, thus manoeuvring with various radius turns is possible. We have proven the above during the laboratory research.

Assuming that the control algorithm for the described robot could be quite complicated and should also interpret signals coming from voice recognition unit, it has been decided, that master part of the control system should be based on the x86 computer board. This type of master controller was connected by serial interface to the previously mentioned low level controller based on microprocessor. An operating system for the computer board was xPC target coming from MATLAB package. The model of the control system was developed in MATLAB-SIMULINK environment and automatically converted into C language program, which was compiled and run on the target system controller. Setting actual drive parameters is performed through hardware voice recognition unit, whose output in form of serial interface was connected to the computer board. Thus there were two sources of data for master control system, one was voice recognition unit and the other was microprocessor board, which was operating motors encoders and ultrasonic sensors. The other computer was used for model design under MATLAB-SIMULINK environment. After model compilation and running on the target system, the abovementioned computer unit was also used for registration and visualisation of selected signals waveforms and trajectory of the robot.

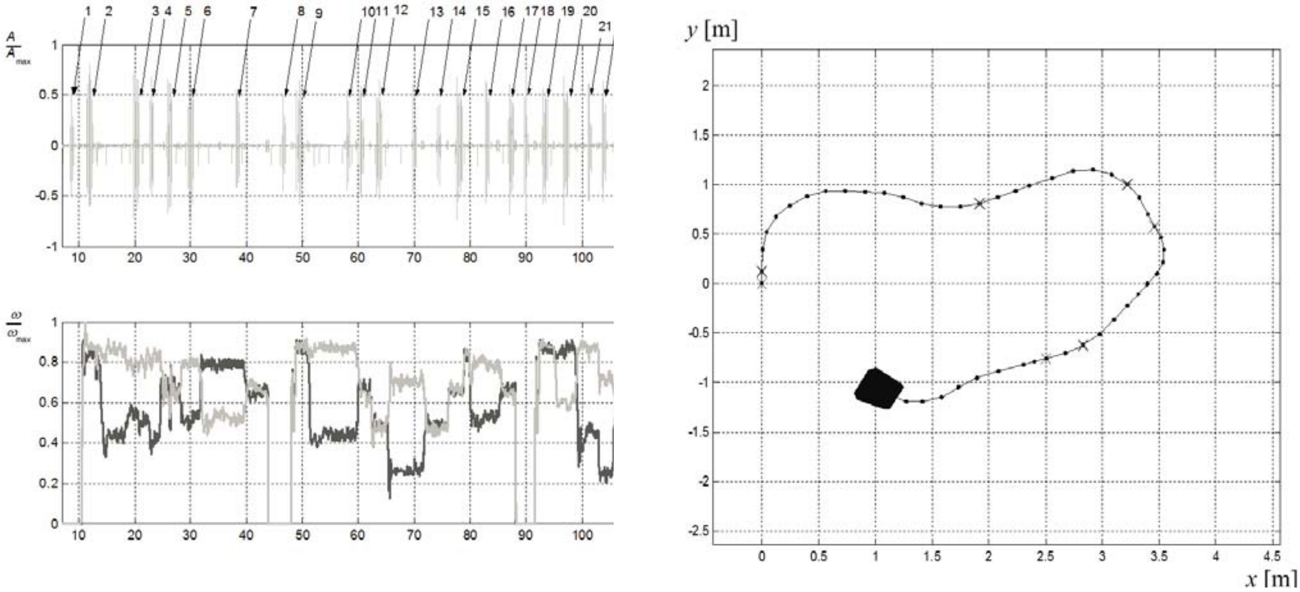

Fig. 3. The waveforms of voice commands signal amplitude (upper graph) and corresponding reaction of robot motors velocities (lower graph) on the left side, The trajectory of voice guided mobile robot obtained during laboratory tests on the right side

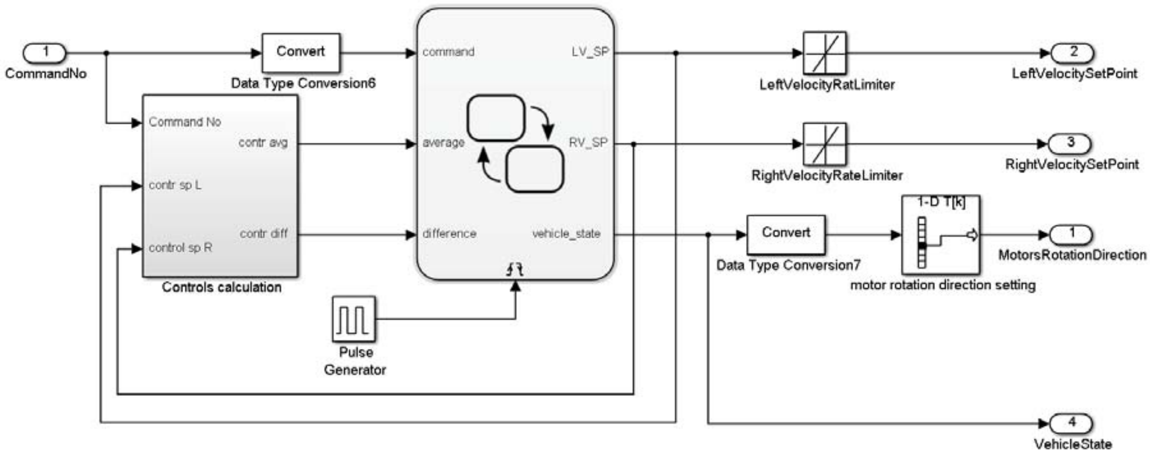

Fig. 4. Part of the control system model including the state machine

An example of obtained result is presented in Fig. 3 on the left side. There are shown waveforms of voice commands presented on the upper graph, and corresponding reactions of the robot as waveforms of motor velocities. The obtained trajectory was depicted in Fig. 3 on the right side. Moments of selected voice commands are marked as a cross on the trajectory. Within the structure of controller software there were dual PI controllers applied, one for each motor. Both controllers were operating in additional loop with proportional controller responsible for holding desired difference between motors rotational velocities. At the first stage of the research a complete model of the robot was developed for the purpose of simulation research. Also the direct kinematic formulas were implemented in order to visualize the trajectory. Next stage of the research involved operation in real time on the basis of the described “software in the loop” approach. During this stage models of the motors were replaced by communication structure, which was responsible for passing the control values to microprocessor controller of real motors. The feedback from incremental encoders, which periodically generated numbers of pulses was received by appropriate block developed in the MATLAB-SIMULINK environment.

In specified period the numbers were transformed into input signals for the described velocity controller. The set points of motors velocities were worked out on the basis of specific voice commands through deterministic part of the controller implemented in form of state machine. The other state machine was utilized in order to complete data through serial interface by hardware units like microprocessor board and voice recognition module. The finite state machines were implemented using STATEFLOW toolbox of MATLAB software. Part of the control system, which consists of the state machine intended to generate motors velocities setpoints is presented in the Fig. 4. The input signals of the machine are: the number of voice commands, setpoints of the average and difference of motors velocities, as well as square-wave signal for triggering the machine periodically. The machine generates left and right wheel velocity setpoints and gives the signal of current state, which is used for determination of each motor rotation direction.

Control sysem of the parallel manipulator

The second example of advanced control system implementation using rapid prototyping paradigm was the controller for 6DOF parallel manipulator. The modelling and design process of that device were described in papers by [13], [12]. As it comes from publications [17] and [11], an analysis and control of the 6DOF manipulators with open or closed kinematic chain is a complex task and it is demanding in respect of a computational power. First stage of our research involved design of the parallel manipulator kinematic structure. A result of that stage is presented in Fig. 5, where kinematic relationships of the manipulator are visualized by tool available in the MATLAB-SIMULINK environment. The idea of that device control is based on the inverse kinematics calculation which could be quite a complex task for computational unit.

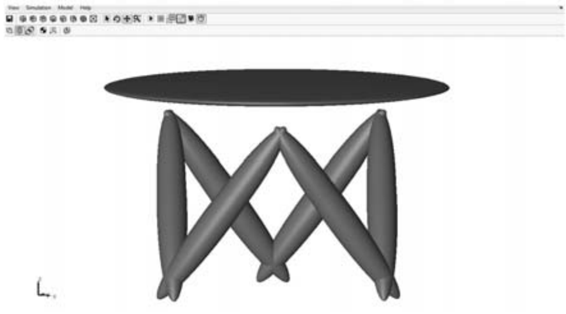

Fig. 5. Visualization of 6DOF parallel manipulator in MATLAB-SIMULINK environment

The next stage of the research was the implementation of the abovementioned inverse kinematics. The complete model of manipulator platform is shown in Fig. 6. There is a part used for reference trajectory setting, which is given as three Cartesian coordinates and three angles of manipulator upper platform. These values are passed to inverse kinematics calculation module, and values of each of six manipulator actuators set lengths are obtained as a result.

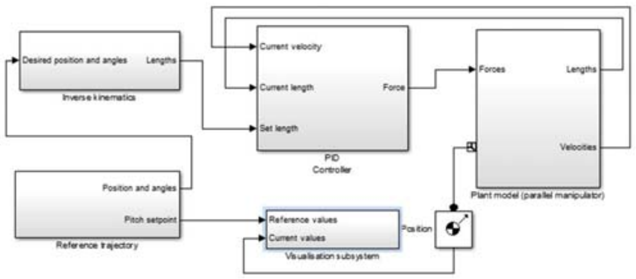

Fig. 6. The structure of the inverse kinematics based controller for 6DOF parallel manipulator implemented in MATLAB-SIMULINK environment

During the experiment the computer model of parallel manipulator was used in order to verify implemented inverse kinematics controller. The example result of the movement of the manipulator platform single coordinate was shown in Fig. 7. There is a screen from xPC target controller visible in the picture. The idea of model based controller software implementation is the same as in the example of voice guided robot described in the previous part of the paper. The reference trajectory and reaction of manipulator model were depicted in form of step response for set point value change.

We have designed the described control model system not only in order to verify if it is possible to control the manipulator in real time, but also for control of operation of real device. Assuming, that in real manipulator described in paper [12] PM BLDC motors controlled by CAN bus was used, it was necessary to design appropriate structure for control and data exchange. A controller built in the motor is controlled by CANopen protocol, so we have implemented a set of commands of that protocol. A useful tool for that implementation was STATEFLOW toolbox, mentioned in the part referring to

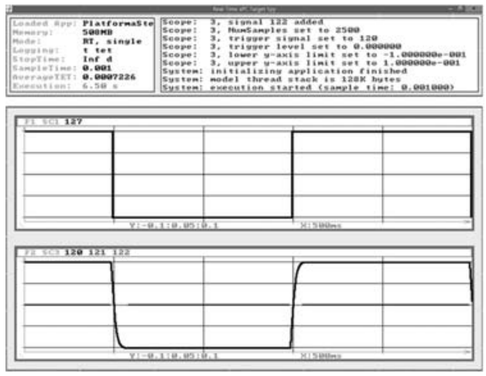

Fig. 7. The waveforms of step response for single coordinate of 6DOF parallel manipulator (upper waveform - the setpoint of actuator position, lower waveform the actuator response

voice control mobile robot control system design. That was due to the sequential nature of the abovementioned protocol. There should be sent certain commands for initialization of the PM BLDC motor drive, and then it is possible to control the setpoints for torque, velocity or position of the motor. Additionally with the CANopen protocol we have obtained the motor velocity and position. As it came from our considerations, the best mode of linear actuator control was position controller mode, based on an incremental encoder built in the motor. State machine with indicated input and output signals for control motor of single actuator is presented in Fig. 8. The values transmitted through CANopen protocol are involved in the control loop, thus received values should be delayed before model calculation. There are two blocks, one for receiving data and the other for sending data. These blocks handle model data exchange with the CAN bus hardware.

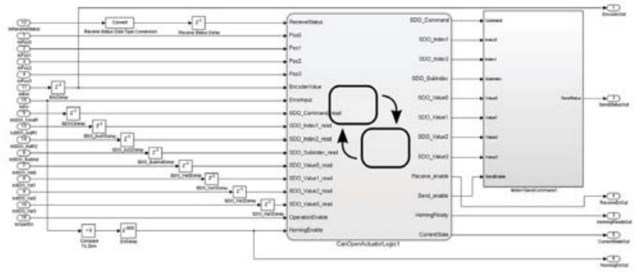

Fig. 8. Part of the control system including the state machine and CAN bus frame sending

All operations within the control software are executed in real time. For single actuator control loop calculation time is equal to 6 milliseconds. In that case a limitation is the complexity of the model. The reaction of real actuator recorded through encoder of the motor is presented in the Fig. 9.

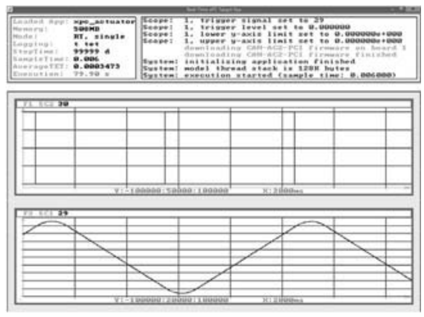

Fig. 9. Linear actuator drive response for step of position setpoint

Summary and conclusions

Subject of the research was the development of two kinds of control systems. One of the systems was designed for electric differential drive mobile robot, guided by voice commands. The other system was designed for control of parallel manipulator based on six PM BLDC motors. The common element of the two research paths concerning the systems was implementation based on "Software in the loop" design paradigm. In both cases the development platform of the systems was xPC target MATLAB toolbox. We have performed the following design stages:

- An analysis of the requirements as well as input and output signals for both types of control systems;
- Design of models in MATLAB-SIMULINK environment;
- Determination of controller parameters e.g. PI controller in case of two motor drive and PID controller for model of parallel manipulator;
- Parameterization of the designed controllers and verification of the models in simulation environment;
- Development of communication framework designed for operation with hardware controllers of the motors;
- Implementation of serial interface binary exchange frames in case of two motor drives and voice commands microcontroller unit;
- Implementation of CAN bus data exchange in case of parallel manipulator drive system;
- Design of Finite State Machines logic for voice control command interpretation and for CANopen protocol operation;
- Design of the detailed models subsystems for implementing complex models of whole controllers structures;
- Determination of sample time for the designed controllers models and appropriate parameters of the controllers tuning;
- Research using designed models in typical control scenarios.

The steps listed above are not only a description of the conducted research, but are simultaneously an algorithm of control system design using rapid prototyping approach. Previously described methodology is applicable especially for electromechanical and mechatronic systems, which was confirmed in literature analysis. From our point of view the developed control systems are foundations for further research, both in area of implementation of developed control algorithms in microcontrollers, and in applying our experiences in design of other control systems. On the basis of performed work in range of computer control, it is possible to state, that rapid prototyping of the controller method gave appropriate results in case of voice guided mobile robot, as well as of 6DOF parallel manipulator. Obtained simulation results have proven that it is possible to control complex real objects in real time and simultaneously record selected data utilized in control system.

REFERENCES

[1] Bo Z., Bin X., Yao Z., Wei Z., Hardware-in-loop simulation testbed for quadrotor aerial vehicles”, Control Conference (CCC), 2012 31st Chinese, pp.5008-5013, 2012;

[2] Budiyono A., Putro I.E., Yoon K., Raharja G.B., Kim G.B., Real-time hardware simulation of a small-scale helicopter dynamics, Aircraft Engineering and Aerospace Technology, Vol. 82 iss. 6, pp.360 – 371, 2010;

[3] Dołęga B., Rogalski T., The new conception of the laboratory testing of the FBW control system for small aircraft, Aircraft Engineering and Aerospace Technology, Vol. 76 iss. 3, pp.293 – 298, 2004;

[4] Chedjou J.C., Kyamakya K., CNN-based ultrafast solver of stiff ODEs and PDEs for enabling realtime Computational Engineering, COMPEL: The International Journal for Computation and Mathematics in Electrical and Electronic Engineering, Vol. 30 iss. 4, pp.1333–1349, 2011;

[5] Karam W., Mare J.C., Modelling and simulation of mechanical transmission in roller-screw electromechanical actuators, Aircraft Engineering and Aerospace Technology, Vol. 81 iss. 4, pp.288 – 298, 2009;

[6] Lu P., Geng Q., Real-time simulation system for UAV based on Matlab/Simulink, Computing, Control and Industrial Engineering (CCIE), IEEE 2nd International Conference on , vol.1, no., pp.399 - 404, 2011;

[7] Osama M.A., Abed N.Y., Real-time simulation of electric machine drives with hardware-in-the-loop, COMPEL: The International Journal for Computation and Mathematics in Electrical and Electronic Engineering, Vol. 27 iss. 4 pp.929-938, 2008;

[8] Pinzon-Ardila O., Angel L., Useche M., xPC Target an option for position control of robotic manipulators, Robotics Symposium 2011 IEEE IX Latin American and IEEE Colombian Conference on Automatic Control and Industry Applications (LARC), pp.1-6, 2011;

[9] Rydło K., Rzucidło P., Chudy P., Simulation and prototyping of FCS for sport aircraft, Aircraft Engineering and Aerospace Technology, Vol. 85 iss. 6, pp.475 – 486, 2013;

[10] Shiakolas P.S., Piyabongkarn D., On the development of a real-time digital control system using xPC-Target and a magnetic levitation device, Decision and Control, 2001. Proceedings of the 40th IEEE Conference on, vol.2, pp.1348-1353, 2001;

[11] Srinivasan A., Nigam M.J., Fuzzy PD+I control of a six DOF robot manipulator, Industrial Robot: An International Journal, Vol. 35 iss. 2, pp.125–132, 2008;

[12] Stenzel T., Hetmańczyk J., Grzesik B., Selected aspects of modeling of linear actuator based on pm bldc motor, The Symposium on Electromagnetic Phenomena in Nonlinear Circuits - EPNC 2014 pp. 125-126, 2014;

[13] Stenzel T., Sajkowski M., Grzesik B., Design and Implementation of 6-DOF Parallel Manipulator Driven by Permanent Magnet Brushless DC Motors, 18th International Conference on Methods and Models in Automation and Robotics (MMAR) 2013, pp. 372-377, 2013;

[14] Trilaksono B.R., Triadhitama R., Adiprawita W., Wibowo A., Sreenatha A., Hardware-in-the-loop simulation for visual target tracking of octorotor UAV, Aircraft Engineering and Aerospace Technology, Vol. 83 iss. 6, pp.407 – 419, 2011;

[15] Truong N.V., Vu D.L., Hardware-in-the-Loop approach to the development and validation of precision induction motor servo drive using xPC Target, Computer Science and Software Engineering (JCSSE), International Joint Conference on, pp.159 - 163, 2012;

[16] Verdú J.M., Sabater Navarro J.M., González Penella V.J., García Aracil N.M., López Buendía Á.M., Rapid design and synthesis of robots using standard computer tools, Industrial Robot: An International Journal, Vol.40 iss.4, pp.363 – 381, 2013;

[17] Wang Z., He J., Shang H., Gu H., Forward kinematics analysis of a six-DOF Stewart platform using PCA and NM algorithm, Industrial Robot: An International Journal, Vol. 36 iss. 5, pp.448 – 460, 2009;

[18] Wei X.; Jihong C., Jin Y., Design of servo system for 3-axis CNC drilling machine based on XPC target, Computing, Communication, Control, and Management, 2009. CCCM 2009. ISECS International Colloquium on, vol.1, pp.447-450, 2009;

[19] Xiao-Kun Y., Hong-Wen H., Lian-Yun P., Xiaolin Z., Hardware-in-the-loop simulation on a hybrid power system, Power Electronics Systems and Applications (PESA) 2011 4th International Conference on, vol. 1, no. 5, pp. 8-10, 2011.

[20] Yuheng L., Zechang S., Jiayuan W., Design for battery management system hardware-in-loop test platform, Electronic Measurement & Instruments,ICEMI ’09. 9th International Conference on, pp.3-399,3-402, 2009

***Authors**: Ph.D., Eng. Maciej Sajkowski,*
Prof., Ph.D., Eng. Krzysztof Krykowski,
Department of Power Electronics, Electrical Drives and Robotics, Faculty of Electrical Engineering, Silesian University of Technology, ul. Boleslawa Krzywoustego 2, 44-100 Gliwice, Poland, email: maciej.sajkowski@polsl.pl,