ISSN 1068-3712, Russian Electrical Engineering, 2015, Vol. 86, No. 6, pp. 309–313. © Allerton Press, Inc., 2015.
Original Russian Text © A.M. Kostygov, A.M. Zyuzev, E.M. Solodkii, A.V. Kukharchuk, M.V. Mudrov, K.E. Nesterov, 2015, published in Elektrotekhnika, 2015, No. 6, pp. 8–13.

The State of and Prospects for Using Hardware–Software Simulators of Electrotechnical Complexes

A. M. Kostygov[a], A. M. Zyuzev[b], E. M. Solodkii[a], A. V. Kukharchuk[a],
M. V. Mudrov[b], and K. E. Nesterov[b]
[a] *Perm National Research Polytechnic University, Komsomol'skii pr. 29, Perm, 614990 Russia*
[b] *Ural Federal University, Yekaterinburg, Russia*
e-mail: journal-elektrotechnika@mail.ru

Received May 15, 2015

Abstract—A review of publications on simulation of electromechanical devices and electric power systems is given in the present article. Examples of structures are considered. A technique for creation and design features of software–hardware simulators for various purposes are discussed. It is noted that the practical interest in the simulator n electricity and electrical engineering is caused by the complexity of experimental research and commissioning, as well as their cost and possible risks of damage to equipment during setting up and testing. It is shown that modern information processing equipment, including programmable logic devices (PLDs), allow one to solve the equations of mathematical models of complicated electrotechnical complexes and a real time system, which makes it possible to construct software and hardware simulators that can be effectively used in design, commissioning, and training students and staff of enterprises.

Keywords: electric power systems, model, simulator of electric drives, real time

DOI: 10.3103/S1068371215060073

Recently, questions regarding simulation of the operation of electric drives and electrotechnical complexes in real time have been raised by electrical engineers and scientists with increasing frequency. Such problems are due to the complexity of experimental research and commissioning in electrical and power engineering, their high price, and risks of damage to equipment, as well as the appearance of new software–hardware facilities allowing one to create simulators for modeling the operation of electrotechnical complexes and systems with a high degree of reliability.Analysis of publications shows that simulators of electrical equipment have been developed to model the dynamic characteristics of electric drives [1–3], converters [4], converter–motor systems [5], or converter–motor–device systems [6] as concerns particular problems. For instance, to adjust the control system with the electric drive, a simulator of the power circuit of the electric drive that imitates the operation of a converter, electric motor, and load, as well as receiving real signals for controlling the converter's keys and generating the feedback signals to the control system, can be used. The simulator operates in accordance with the following cycle: reading of input signals, processing of them, and generation of output signals. The duration of this cycle defines the area of possible application of the simulator: long cycles are generally sufficient for the thyristor converters. Transistor converters with PWN require a minimum cyclic time.

Constructing the intelligent electricity networks, which are being high-efficiency and controllable new-generation electric power systems, is one of the main problems in modern electric power engineering. Modeling of the objects is necessary for prediction of operation regimes under various conditions with the development of the means of adapting generators and consumers to the current situation.

The efficient test run of the whole range of the required tasks and algorithms under continuous and high-precision real-time simulation of the united range of normal and emergency regimes and processes on the equipment and electrical systems is possible on specially developed complexes.

The software–hardware simulator can be realized on a personal computer under the control of real-time operating systems [7].

The presence of the rapid processing unit allows one to implement the complicated algorithms of processing the input signals with the application of double precision variables, which have a floating point. This guarantees high precision of calculation of the equipment model and the minimization of rounding errors, which is especially important at integrators in the object model. Although the frequency of the input signals' digitizing can reach hundreds of kilohertz, the reading–processing–generation cycle frequency does not generally exceed several hundreds of hertz. In this case, the input signals are read by blocks (oscillogram

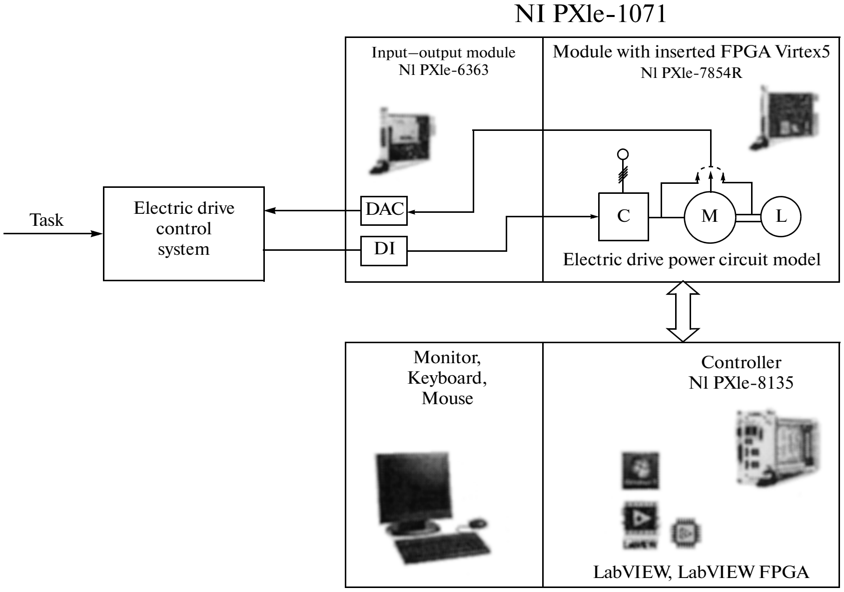

Fig. 1. Structure of simulator based on a PWN.

fragments) containing hundreds or thousands of values. The low speed of output signal generation is caused by the structure of the personal computer and limits the area of application of similar simulators by modeling systems with the thyristor converter.

Imitation of the operation of a transistor converter requires a higher rate of response, which can be provided only on the basis of a digital signal processor (DSP) or a programmable logic device (PLD). We not have data on using DSPs for construction of such simulators. The variant of implementing simulators based on PLDs is widely used due to its high response rate and convenient facilities for programming. Figure 1 shows the structure of such a simulator.

The data from [1–4] and our experience show that it is possible to solve the systems of equations of, e.g., a asynchronous motor with a frequency converter containing PWN on with a period of 1 μs, which is enough to reveal the responses of the electric drive to real controlling influences.

A TECHNIQUE FOR CREATING ELECTRIC DRIVE SIMULATORS

The algorithm of generating the program code of the simulator is as follows. The model of the imitated system is designed and tested in MatLab/Simulink. After adjusting, it is compiled by MatLab in the HDL code and loaded into the PWN via a special software. This scheme allows one to obtain the result after a short time, but the cost of MatLab and the required expansions can be too high for small companies. Another way to create the simulator software is to program the model in LabVIEW, adjust it on a personal computer, and perform further manual conversion—optimization for use on the PWN [6]. Here, all the operation stages, including the code loading into the PWN, are implemented by the LabVIEW function, with the simulator having a high rate of response.

In the design of electric drive simulators [8–12], the concept of development of the program code based on serial generation of the code of the model in MatLab/Simulink with further manual transfer to LabVIEW was used. The described technology allows rapid creation of the reference model (in MatLab/Simulink) and highly efficient code for the PWN (in LabVIEW). The model generated in LabVIEW is checked by comparing its results with the results from the model created in MatLab/Simulink. To obtain the code implemented by the PWN with the maximum rate of response, instructions from HighThroughput-Math are used. Moreover, the division operations may change to multiplicative ones because of the higher speed.

In applying the described method, mathematical models operating in real time are used, such as the thyristor voltage converter model [8], ac electric drive [9], three-phase transistor voltage inverter [10], asynchronous electric motor in a two-phase coordinate system [11], and dc motor with independent excitation [12].

An example of the code of the latest model is shown in Fig. 2.

One of the PWN's features is that it is able to run many operations concurrently. Therefore, to obtain a high calculation speed, a code needs to be created that is optimized for the parallel implementation. The use of the classical serial construction of the calculation algorithm in Simulink or the LabVIEW models leads to unsatisfactory (in terms of calculation speed) results. The code optimized for parallel implementa-

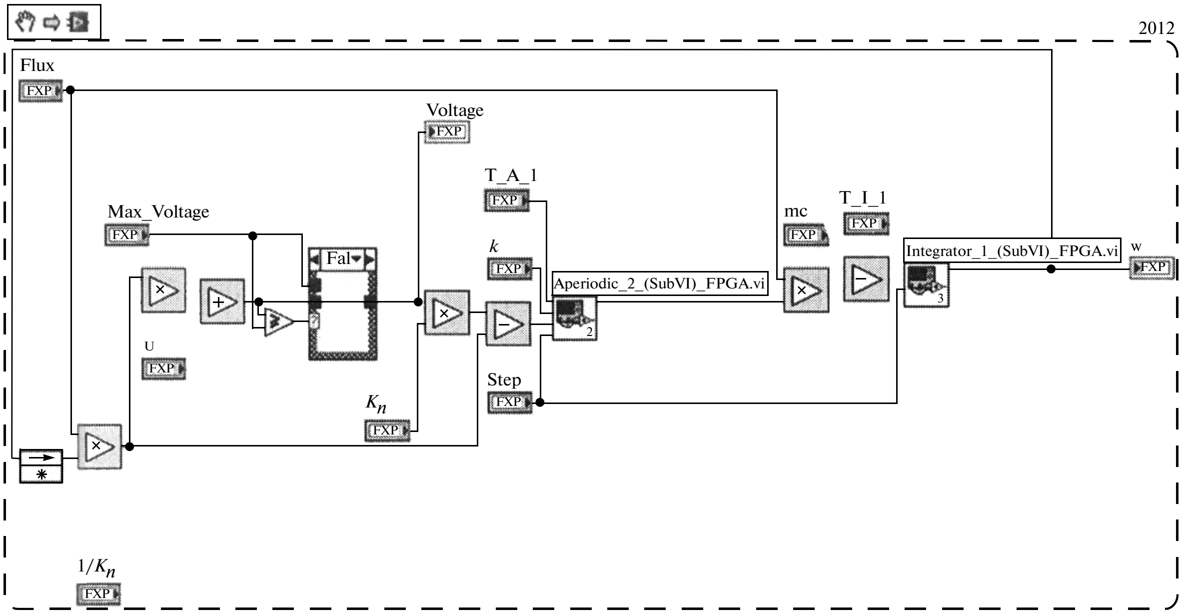

Fig. 2. Example of the program code of the model of a dc motor with independent excitation in LabVIEW FPGA.

tion can be obtained if the initial equations are written in normal Cauchy form. The code providing calculation of the corresponding variable on the basis of the values of others taken from the previous cycle is then generated in each equation.

Upon a rather small (in comparison with the time constants of the simulation object) step of calculation, such a concept gives enough satisfactory result. For instance, the difference between the serial and parallel algorithms did not exceed 1% (the comparison was implemented over the electromagnetic moment of the motor) on modeling a asynchronous electric motor with the calculation step of 1 μs. The mean-square deviation was 0.1%.

Another feature of the PWN is that it does not support mathematical commands over variables with a floating point, only types with a fixed one and integers are accessible. Integer mathematics makes the calculation of models more complicated, since it requires using a special system of relative units. Application of the variable with a fixed point with a specific place to store the integer and fractional parts seems simpler. The bit capacity should be selected considering the range of possible variations of variable values. An increase in the bit capacity of variables decreases the calculation speed and raises the program volume. Therefore, the model can turn out to be from the chosen PWN, which leads to inadmissibly slow calculation. The reduction of the bit capacity results in decreasing the simulation accuracy owing to the accumulation of round errors. Specifically, it is characteristic for models with a large number of differential equations.

Based on the model of an asynchronous electric motor, we investigated the calculation error from the bit capacity of the used variables. Figure 3 shows the relationship between the mean-square error emerging at the calculation of the electromagnetic moment of the motor and the error of the variables of the model (the minimum value, which can be stored by the variable). The relationship between delta and the bit capacity is given in the table. As a result, it was found that variables with a capacity of 36 bits, wherein the fractional part involves 32 bits, are enough to calculate the model of an asynchronous electric motor with a step of 10^{-6}. Here, the mean-square error of calculating the electromagnetic moment of the motor does not exceed 0.01%.

Table

Delta	9.09E-13	1.82E-12	3.64E-12	7.28E-12	1.46E-11	2.91E-11	5.82E-11	1.16E-10	2.33E-10	4.66E-10	9.31E-10
Bit capacity	40	39	38	37	36	35	34	33	32	31	30

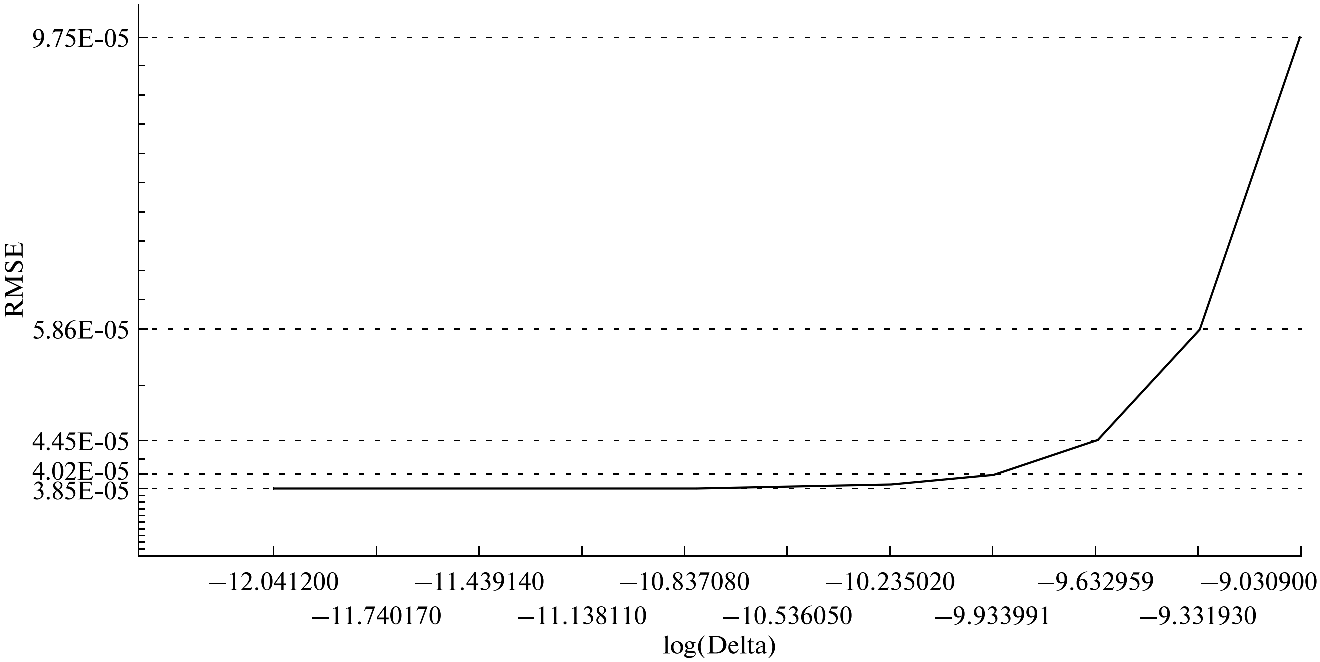

Fig. 3. Relationship between the calculation error and the error of model variables.

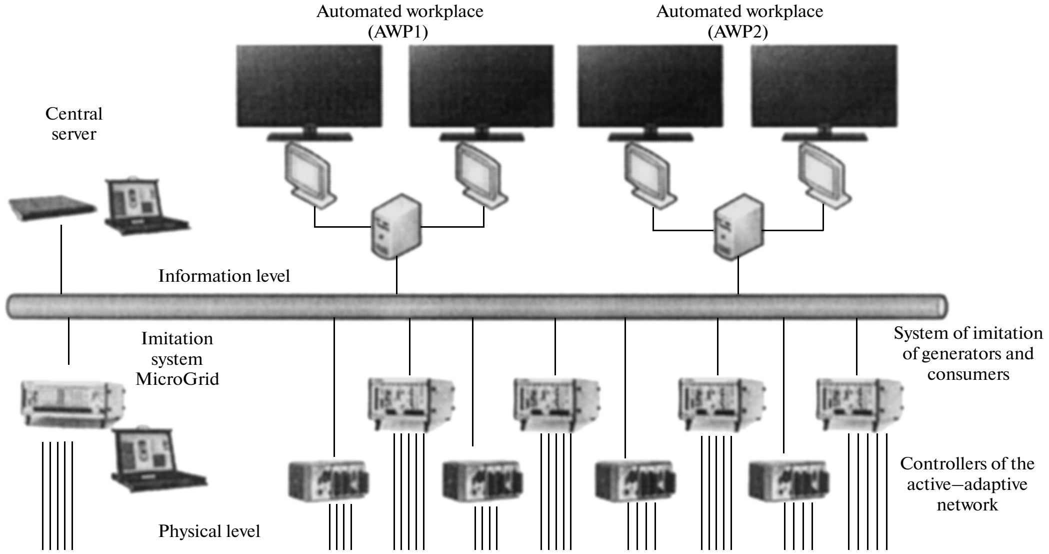

Fig. 4. Architecture of the seminatural simulation complex.

TECHNIQUE FOR CREATION OF ELECTRICITY SYSTEM SIMULATORS

As a whole, the technical implementation of the concept of the software–hardware simulator of the active-adaptive network is based on the following principles [13]:

—hardware-in-the-loop, HIL; and

—using the PWN technologies for the hardware implementation of the mathematical models and input–output of the signals in real time.The NI PXI and NI CompactRIO platforms, which allow transparent transfer from the seminatural simulation to the prototype development, as well as using already existing automated means of measuring and control (Fig. 4), act as the hardware base.

The program part is implemented in LabVIEW. The example of constructing the program part of the simulator, this being the bond in the active adaptive network, is shown in Fig. 5. A set of subprograms (a framework) describing the model of relationship

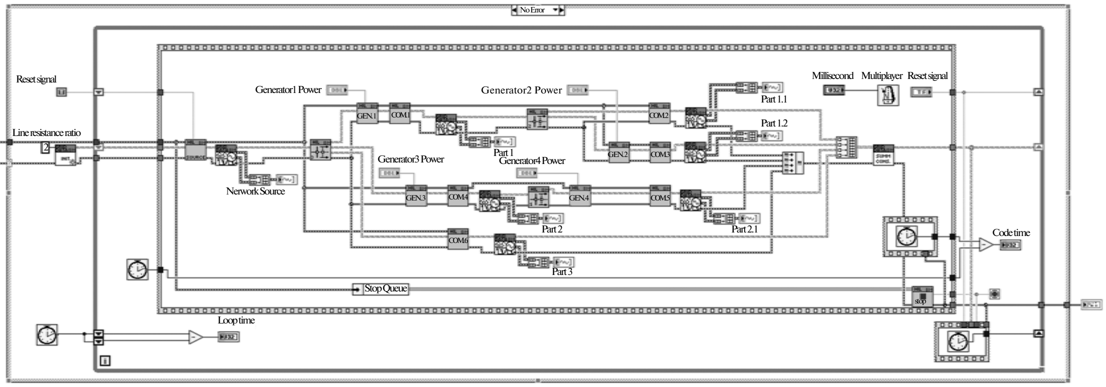

Fig. 5. Block diagram of the model of the relationship between elements.

between the elements of the electrical power supply system allows one to specify the arbitrary structures of the network and the parameters of the elements [14].

The present hardware–software simulator allowed one to test and work out the algorithms and techniques of using the following.

—A neural network and connectionist systems of controlling the modes, prediction of electrical energy consumption, and expert systems of supporting decision-making.

—Adaptive automatic control for the renewable sources of energy.

—Adaptive control of the flexible systems of the alternate current.

—Transmission of data on operation and control of operation regimes of the active-adaptive network between its elements.

REFERENCES

1. Dufour, C., Bilanger, J., Abourida, S., and Lapointe, V., FPGA-based real-time simulation of finite-element analysis permanent magnet synchronous machine drives, *Proc. IEEE Power Electronics Specialists Conf.*, Orlando, 2007.
2. Duman, E., Can, H., and Akin, E., FPGA-based hardware-in-the-loop (HIL) simulation of induction machine model, *Proc. Power Electronics and Motion Control Conf. and Exposition (PEMC)*, Antalya, 2014. DOI 10.1109/EPEPEMC.2014.6908564.
3. Gregor, R., Valenzano, G., Rodriguez-Pineiro, J., and Rodas, J., FPGA-based real-time simulation of a dual three-phase induction machine, *Proc. 16th European Conf. on Power Electronics and Applications (EPE'14-ECCE Europe)*, Lappeenranta, 2014.
4. Wei, Li., Gregoire, L.-A., Souvanlasy, S., and Belanger, J., An FGPA-based real-time simulator for HIL testing of modular multilevel converter controller, *Proc. IEEE Energy Conversion Congress and Exposition (ECCE)*, Vienna, 2014. DOI: 10.1109/ECCE.2014.6953678
5. Ziuzev, A.M., Nesterov, K.E., and Mudrov, M.V., The software-hardware simulator of the electric drive, *Proc. 16th European Conf. Power Electronics and Applications (EPE'14-ECCE Europe)*, Lappeenranta, 2014. DOI: 10.1109/EPE.2014.6911018.
6. Zyuzev, A.M., Nesterov, K.E., and Mudrov, M.V., A hardware-software complex for real-time modeling of electric drives, *Russ. Electr. Eng.*, 2014, vol. 85, no. 9, p. 591.
7. Ahmadeev, E., Beliaev, D., Ilijin, E., and Weinger, A., The virtual test bench of medium voltage controlled AC drives, *Proc. 15th IASTED Int. Conf. on Applied Simulation and Modelling*, Rhodes, June 26–28, 2006, pp. 340–345.
8. Zyuzev, A.M., Nesterov, K.E., and Mudrov, M.V., PC Software Certificate no. 2014660944, 2014.
9. Zyuzev, A.M., Nesterov, K.E., and Mudrov, M.V., PC Software Certificate no. 2014660942, 2014.
10. Zyuzev, A.M., Nesterov, K.E., Mudrov, M.V., and Kostylev, A.V., PC Software Certificate no. 2014661060, 2014.
11. Zyuzev, A.M., Nesterov, K.E., Mudrov, M.V., and Kostylev, A.V., PC Software Certificate no. 201461267, 2014.
12. Zyuzev, A.M., Nesterov, K.E., and Mudrov, M.V., PC Software Certificate no. 2014660946, 2014.
13. Petrochenkov, A.B., Romodin, A.V., and Kychkin, A.V., Information-measuring and control system for local intelligent electric energy network, *Inf.-Izmerit. Upr. Sist.*, 2014, no. 9, pp. 4–11.
14. Petrochenkov, A.B., Frank, T., Romodin, A.V., and Kychkin, A.V., Hardware-in-the-loop simulation of an active-adaptive power grid, *Russ. Electr. Eng.*, 2013, vol. 84, no. 11, p. 652.

Translated by A. Evseeva