ISSN:2320-0790

An International Journal of Advanced Computer Technology

Measurement, analysis and data collection in the programming LabVIEW

Stefan Koprda, Zoltan Balogh

Department of Informatics, Faculty of Natural Sciences, Constantine the Philosopher University in Nitra, Slovakia

Abstract: The work deals with handling of systems modelling, simulation knowledge and the processor and particular processor core overload issue. Programming environment LabVIEW was designed for this field of measuring, analysis and data acquisition. It is a highly productive development environment that engineers and scientists use for graphical programming and unprecedented hardware integration. The environment is predominantly designed to simulation, managing and parameter and other factor measuring. By this graphical program we are able to analyses and consequently display the results on the output interconnection. The environment LabVIEW is applied to the computer processor overload issue in this work. The result is direct technological virtual instrumentation of overload representation in the real time. Finally this instrumentation is more simple, transparent and especially more effective than a foregoing hardware solutions. By the block diagram design it is possible to monitor the processor and its core overload. Consequently these measurements are recorded into the text file and display writer in the percentage value in certain time interval.

Keywords: LabVIEW; processor; monitoring

I. INTRODUCTION

The main aim of modelling, the meaning of models creation, is to describe the content, structure and behavior of the real system representing a certain defined part of reality. The term model is closely connected with simulation, by means of which it is possible to represent the modeled system and its behavior also in a real time, for example on a computer, and to monitor or modify it [1], [2], [3].

A system can be defined as an object, already existing, or understood in an abstract meaning, which we intend to explore, while during its existence the system can evolve and cooperate with other systems, which form its surroundings [4].

Under the term "model" we generally understand a system, which is a certain simplification of the original of the modelled system. Between the original and its model there exists a homomorphous relationship of the display, while we differentiate between abstract models, which we can logically ponder over, and simulation models, on which we can perform simulation experiments [5], [6].

Modelling, in terms of research technology, is a replacement for the explored system by its model; its aim is to obtain information on the originally explored system by means of an experiment with the model [7].

Modelling is a multidisciplinary activity, since the knowledge of mathematics and physics, theory of systems, probability theory, informatics, cybernetics or cognitive sciences, operation research, and others, can take a share in. Modelling serves not only to solve practical problems, but it is also designed for the realization of certain research and experiments [8], [9], or to simulate phenomena and processes.

Modelling language is every artificial language, which can be employed for the expression of information or knowledge on the systems in the structure, which is defined by a consistent set of rules. In general, they are used for the interpretation of the meaning of components in the structure. Modelling language can be graphic or textual [10].

Simulation is a method of acquisition of new knowledge on the system by means of experimenting with its model [11].
Verification of the model occurs in the moment when the model creator designedly tests seemingly correct

version of the model in order to find and correct the mistakes, which could have originated during the modelling phase.

Validization occurs, when the model creator and professionals assess to what degree is the created model satisfactory and suits the original [12], [13].

It is inevitable to state that no model can be verified or validated for 100%, since neither validation nor verification is absolute. Every model is a certain representation of the system and its behaviour is even in the most ideal case only an approaching to the behaviour of the real system. If we affirm that the model was verified or validated, it means that we have carried out a sufficient number of tasks, tests and analyses. The process of verification and validation always remains to a great degree a matter of subjectivity [5].

The word "virtually" more and more frequently appears in context with new technologies connected with the fast development of information technology. We can meet virtual reality, as part of the show-business (film, computer games), or as the means facilitating coping with complex situations in certain fields of human activity. Our lives are entered by an offer of goods in virtual department stores displaying their goods by means of Internet in the so-called e-shops. Another area, connected with the word „virtual" and the beginning of new technologies, in connection with measurement and measurement technology is called virtual setting (see Fig. 1.) [14].

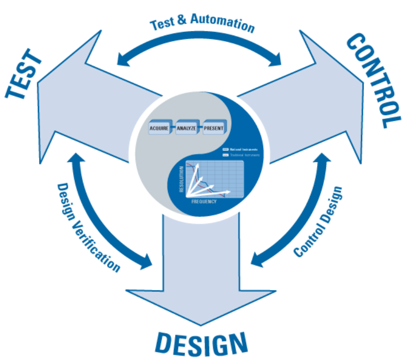

Fig. 1 Virtual setting

From among standard software tools for the area of measurement can be used for example table processors, which cover the phase of presentation and partially the one of analysis of the measured data.

According to the other point of view, in the market of software tools we can find closed systems, which provide the user with a limited variety of functions, programmed by their creator, and which cannot be extended further in a simple way.

Besides these, there exist also open systems, which provide the user with a whole range of functions, without limiting him, since they can be extended in a simple way, as to the needs of the user. These are the so-called development environments.

We shall deal with the work in the program environment LabView by the firm National Instruments. LabView is a development environment for the creation of applications, the so-called virtual instruments, oriented to the spheres of measurement, processing and utilization of the measured data [15].

II. Planning and Desingn Process

Planning and design tips for developing a LabVIEW application (see Fig. 2.).

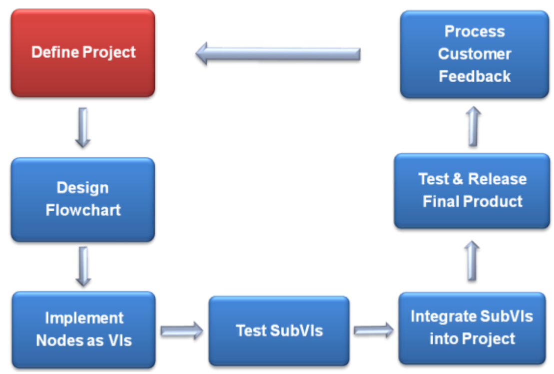

Fig. 2 Planning and Design Process in LabVIEW

This is one possible design approach, and is not intended as a general solution.
Discuss the steps in this design process:
– Clearly define project goals and system requirements.
– Design a flowchart for the application.
– Implement nodes in the flowchart as subVIs where possible.
– By creating a hierarchical set of VIs, you can find and fix bugs more quickly during testing.
– After the individual components work, begin integration into the larger project.
– Test the final product and release it.
– Use customer feedback and updated design goals to improve the product.

Remote control of processes

The process management task represents natural extension or completion of the process monitoring task [16]. The substantial difference between the two tasks rests in the fact that the monitoring task (access of type read only) can be utilized by more users at once on several network nodes, or on several Internet clients, while the management task (access of type write) is unique from the point of view of the process [17]. It is necessary to provide this uniqueness with an appropriate synchronization of access of individual users to the process management.

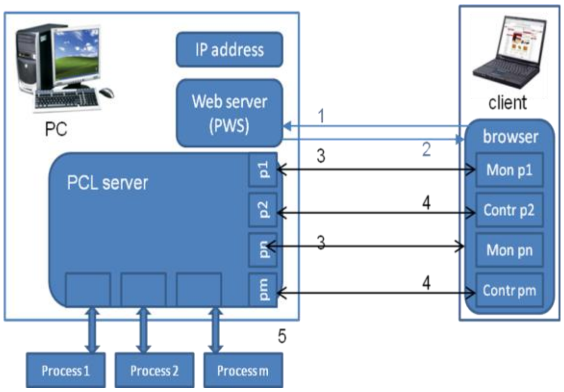

Fig. 3 Object remote management principle

For the remote control of the controlled object through Internet it is possible to make use of the principle of communication between the client and the server. It is depicted in Fig. 4, where Mon p1 through Mon pn indicate application – monitoring through ports p1 and pn, Contr p2 through Contr pm indicate application – control through ports p2 through pm. In fig. 1, numeral values designate: 1 – Client´s page requirement, 2 – Server respond (page sending), 3 – communication: PCL server - applet monitor through an odd port, 4 – communication: PCL server - applet control through an even port, 5 – communication: PCL server - process through PCL card (PC LabCard). Unlike the monitoring process, where connection between the client and the server is executed through one channel (through one port), in case of control it is necessary to create a two-channel communication by means of two ports, while the other communication channel is designed for the transfer of command variable from the client to the server and through it to the process.

III. A Design of the Virtual Instrument for the Processor Loading Monitoring

In the program environment LabVIEW we created a virtual instrument, which allows us to monitor processor loading at various activated processes and the subsequent archiving of processor loading into the text file.

If we want to measure dates in real network it is important to define measurement conditions so we can collect more accurate data. Svec and Munk recommend the elimination of gratuitous network traffic and the clock synchronization between all nodes [18].

The front panel (see Fig. 4.) is formed by two displays indicating actual capacity utilization of the processor and the time flow of the processor loading history. In the bottom part of the virtual instrument is the block, which allows us to enter the file path to the file, into which we intend the data to be recorded. For the correct registration of data into the file it is inevitable to enter the file, into which the recording will be executed first, and only then we can activate the virtual instrument.

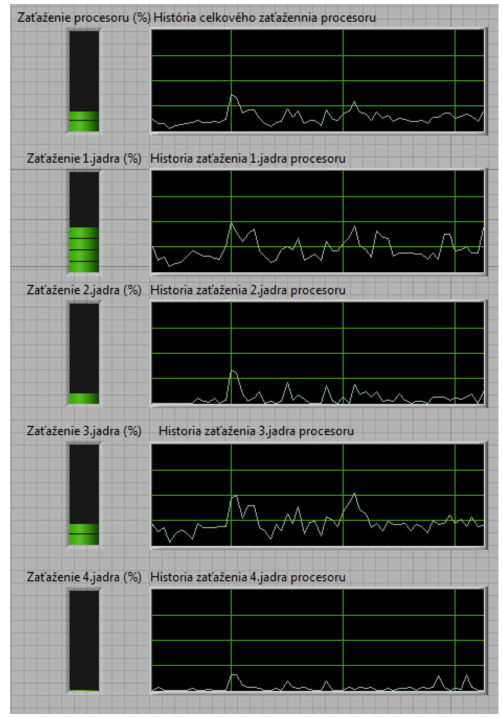

Fig. 4 Virtual instrument front panel in the program environment LabVIEW

An inseparable part of the front panel (of the virtual instrument) is the block diagram, which is interconnected with the front panel, and includes individual blocks ensuring the instrument run. You can see the block diagram of the instrument for measuring and monitoring processor loading in Fig. 5.

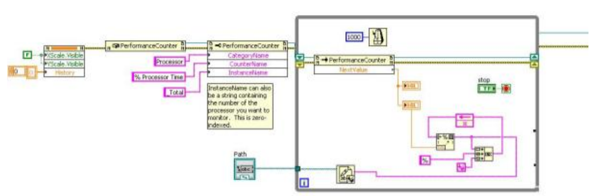

Fig. 5 Block diagram of the instrument for measuring and monitoring processor loading

For the creation of the block diagram a While loop was used. It ensures repetition of the algorithm during the validity period of the set stop condition. The cycle will be terminated only under condition that the state of the stop condition fed into the input of the condition terminal according to the previous setting will be True (Stop If True), or False (for the setting Continue If True).

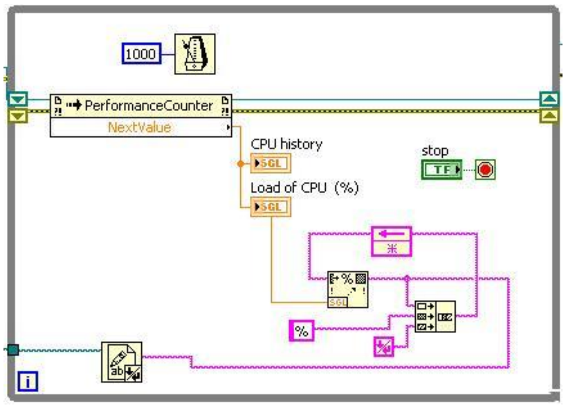

Fig. 6 Block diagram of the instrument – While loop

Inside the loop While are situated blocks, which allow us to draw processor loading into the graph. An important part of the loop is the function, which records the obtained data on processor loading in percentages into the text file. This is the Write to Text file Function. This function enters the chain or field of chains into the file as lines. In order to get the obtained data in percentages, we used in While loop the function (1), which shall allow us to enter certain inputs with values at each passing through the loop. In our case we used percentage and the following line spacing.

Outside the While loop there is the block, which is able to find out the input values on loading from the processor. Besides this, there is also the function Path, which serves for entering the file path.

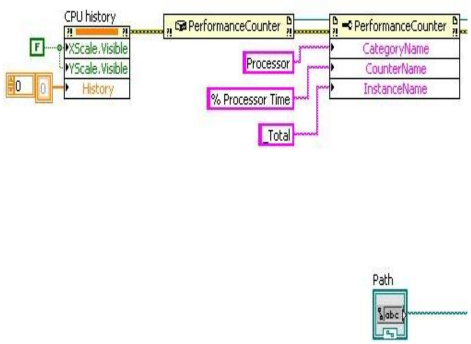

Fig. 7 Block PerformanceCounter

Blok Performance Counter (see Fig. 7) is a chain allowing us to monitor a certain number of processes. In this case we used it for the monitoring of processor loading.

IV. Discussion

In operating systems several processes can be under way and in the system several instances of one program can exist. The run of one process has thus an influence on the run of the other and vice versa. Operating systems, which support parallel running of several processes, normally contain synchronizing instruments, by means of which it is possible to solve synchronizing tasks.

The created application/program in the LabVIEW environment monitors the processor loading and records individual statuses of processes, which are recorded in log files, and can be subsequently evaluated by means of various statistic methods. The created application can be used as a diagnostic instrument for the tuning of processes demanding as to calculation and memory. The application can also serve for fast detection of processor temperature. By setting of appropriate levels of maximum and critical temperature, the application can protect the computer processor from the devastating effects of heat by the error tone and timely disconnection of the system.

Having finished programming of the block diagram, it is possible to remote monitor the processor loading, and in case of need, it will be possible to terminate individual processes excessively loading the processor. By means of the remote control it is possible to monitor and subsequently control the overall loading of CPU. Controlling the processes from the remote computer by means of WWW technologies provides us with numerous assets:
Access to the controlled object from any Internet node, Starting this application from any platform supporting Java - browsers,

Controlling and monitoring the processes – they can be practically verified by the use of services provided by the virtual lab. Demanding and costly installation of technological objects (or their models) is not inevitable.

V. Conclusion

The article presents the issues of modelling and simulation in the graphic environment LabVIEW from the firm National Instrument. The possibility to simulate real processes offers many advantages to designers and advance designers from various spheres, such as time saving and costs minimization. Prototype production is preceded, or fully replaced by the phase, in which using virtual instruments and simulation of real processes shall reveal plenty of errors leading to frequent and costly interference with the physical prototype, or total abjection of the method of solution of the given problem. Characteristic feature of the visualizing instrument - LabVIEW is utilization of the graphic environment, or G language. The designed virtual instrument is able to monitor and subsequently record processor loading in percentage. The created application can be set on various types of computers in a real time. The application was transformed to the ending exe, which means that it is not necessary to have installed the environment LabVIEW.

ACKNOWLEDGMENT

This publication is supported thanks to the project UGA VI/35/2013 Simulation and modeling of control algorithms in static systems, Faculty of Natural Sciences, CPU Nitra, Slovakia.

VI. REFERENCES

[1] Z. Balogh, and C. Klimeš, "Modelling of education process in LMS using Petri nets structure," *Proceedings of the IADIS International Conference e-Learning 2010, Part of the IADIS Multi Conference on Computer Science and Information Systems 2010*, MCCSIS 2010, pp.289-291.

[2] Z. Balogh, and M. Turčáni, 2011, "Possibilities of modelling web-based education using IF-THEN rules and fuzzy petri nets in LMS," *Communications in Computer and Information Science 251 CCIS (PART 1)* , pp. 93-106.

[3] S. Koprda, Z. Balogh and M. Turčáni, "Fuzzy Control Rules Base Design," *2011 5th International Conference on Application of Information and Communication Technologies,* AICT 2011 , art. no. 6110962.

[4] A. Backlund, "The definition of system," *Kybernetes*, Vol. 29, No. 4, 2000, pp.444–451.

[5] P. Peringer, "Modelování a simulace," Fakulta informačních technologií: Vysoké učení technické v Brně, 2006.

[6] J. Fejfar, J. Šťastný and M. Cepl, "Time series classification using k-Nearest neighbours, Multilayer Perceptronand Learning Vector Quantization algorithms," *Acta Universitatis Agriculturae et Silviculturae Mendelianae Brunensis* 60 (2), 2012, pp.69-72.

[7] I. Křivý and E. Kindler, "Simulace a modelování," *Učební texty Ostravské university*, Přírodovědecká fakulta: Ostravská univerzita, 2001

[8] D. Klocoková, "Integration of heuristics elements in the web-based learning environment: Experimental evaluation and usage analysis," *Procedia - Social and Behavioral Sciences,* ISSN 1877-0428, 2011, vol. 15, pp.1010-1014.

[9] D. Klocoková, and Munk, M. "Usage analysis in the web-based distance learning environment in a foreign language education: Case study," *Procedia - Social and Behavioral Sciences*, ISSN 1877-0428, 2011, vol. 15, p. 993-997.

[10] H. Xiao, "A metamodel for the notation of graphical modeling languages," *Computer Software and Applications Conference*, COMPSAC, Vol.1, 2007, pp.219-224.

[11] CH. Chung, "Simulation Modeling Handbook, A Practical Approach," *CRC Press*, Boca Raton, 2003.

[12] R. G. Sargent, "Verification and Validation of Simulation Models," *Proceedings of the 2003 Winter Simulation Conference*, 2003, pp.37-48.

[13] K. Jensen, "Coloured Petri Nets," *Basic Concepts, Analysis Methods and Practical Use*, Vol.2, Springer-Verlag, 1997.

[14] J. Žídek, "Grafické programování ve vývojovém prostředí LabVIEW," Ostrava: *Vysoká škola báňská v Ostrave*, 2002.

[15] Z. Balogh "Modelovanie, simulácia a riadenie procesov s využitím grafického programovacieho balíka LabVIEW," *Inovácia výskumu katedier informatiky Nitra* : UKF, 2009, ISBN 978-80-8094-579-4, pp. 11-17.

[16] P. Horovčák, and M. Roškanin, "Metódy monitorovania technologických procesov s využitím www technológií," *Zborník referátov konferencie Automatizácia a počítače v riadení procesov*, TU Zvolen, 1998, pp.15-29.

[17] I. Leššo, D. Baluch, P. Horovčák, J. Futó, and J. Budiš, "Monitorovací systém vŕtacej súpravy pre účely riadenia," *Zborník referátov 9. medzinárodnej baníckej konferencie,* Riadenie procesov získavania a spracovania surovín, FBERG TU Košice 1997, pp.91-96.

[18] P. Švec, and M. Munk, "IPv4/IPv6 performance analysis: Transport layer protocol impact to transmission time," *2011 International Conference on Internet Technology and Applications*, iTAP 2011 - Proceedings, art. no. 6006335.