

# Romanian Strategy on Virtual Instrumentation Training in High Schools

T. Savu<sup>1</sup>, D. Sporea<sup>2</sup>

<sup>1</sup> Centre for Advanced Technologies - CTANM, POLITEHNICA University, 313 Spl. Independentei, 060042, sector 6, Bucharest, Romania

<sup>2</sup> National Institute for Lasers, Plasma and Radiation Physics, Atomistilor St. 409, Magurele, RO-76900, Romania  
savu@ctanm.pub.ro; sporea@fin.nipne.ro

**Abstract.** In the beginning, the paper explains briefly the principles of graphical programming, based on the industrial standard programming environment LabVIEW. Past efforts to introduced graphical programming in high schools in Romania are reviewed. Future plans to support this strategy within the European Union funded Comenius "Hands-On Science" project are presented.

**Keywords.** Data acquisition, graphical programming, teachers' training, virtual instrumentation.

## 1. Introduction to graphical programming

Programming in a graphical environment may be described as defining an algorithm by drawing its related diagram and not by writing text based instructions and commands. In this way the programming act does not require any more command names and complicated syntactic rules to be memorized, so the risk for programming errors is drastically reduced [1], [2].

Two main windows are available for the user: a front panel in which objects are inserted for defining the future user interface and a diagram in which the data flow is described.

The objects to be inserted in the front panel are picked-up by the user from a controls palette (Fig. 1) in which several groups for different data types (i.e. numerical, Boolean, strings and tables, lists, arrays, clusters) and various other programming aids (i.e. paths, graphs, dialog boxes) are available. As an example the group for numeric objects is represented in Fig. 2. The user simply drag-and-drops an object from the palette to the front panel in order to design the function blocks of his interface.

When an object is inserted into the front panel, a corresponding "terminal" automatically appears in the application's diagram.



Figure 1. Controls palette (partial)

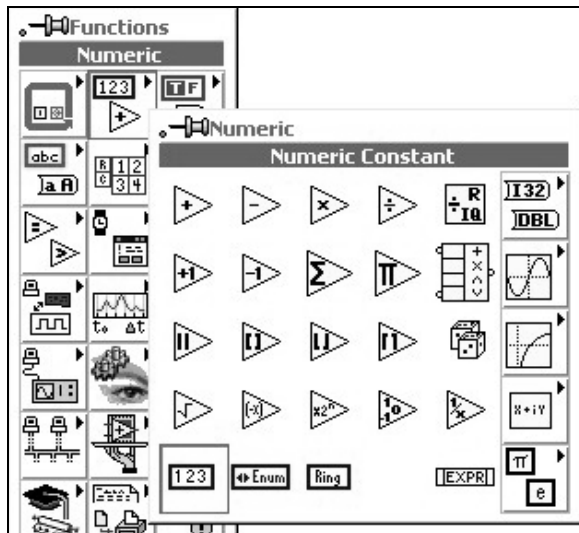
In the diagram the user may insert different function symbols for describing the applications's algorithm.



Figure 2. Numeric objects group (partial) opened in the controls palette

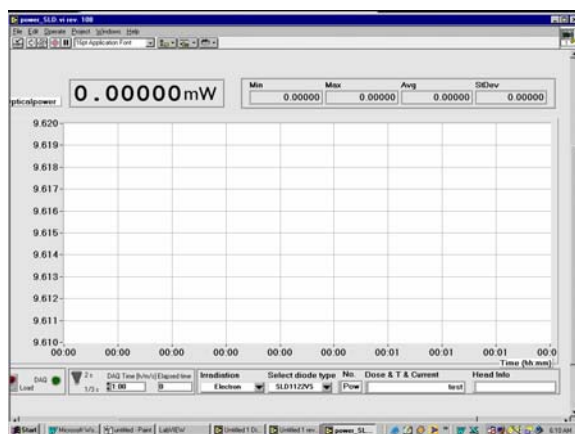
The symbols are picked-up from a function palette (Fig. 3) in which different groups are available, each one for a certain scalar data type (numeric, Boolean, string) or for a certain data structure (arrays, clusters etc.). Functions groups are also available for file operations, data acquisition, advanced mathematical functions, application control etc.

For defining the algorithm, the user may also insert different programming structures: *For* and *While* loops, case and sequence structures, event programming structures.



**Figure 3. Numeric functions group opened in the functions palette**

The control functions available to the user on the front panel can be tailored according to custom needs through their defined properties. Input and output data can be displayed in numerical and graphical form, with scalable frames. A complete color palette helps the functionality design of the user's interface. An example of such a front panel is given in Fig. 4, where data acquired from a remote instrument are represented in a graphical form. The operator has also the possibility to input some parameters or to select/define files management path for saving and recalling data. For the case of remote control of instrumentation the user's interface can be designed to mimic the real interface front panel.



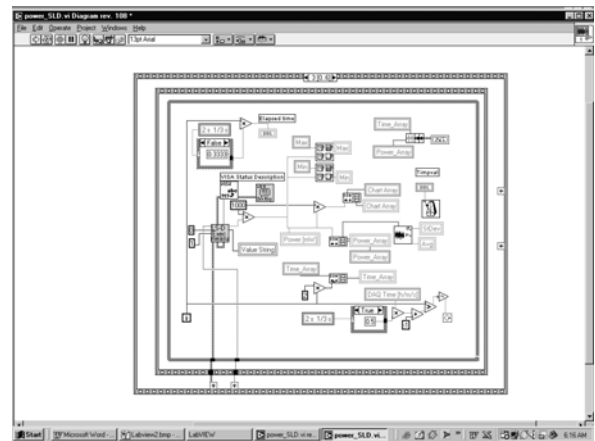
**Figure 4. The graphical user's interface built with LabVIEW for the remote control of an instrument**

Each function symbol has one or more input and output terminals. By connecting the

terminals representing the objects in the user interface and the functions terminals, the programmer is defining the application's data flow (algorithm). The resulting diagram is more intuitive and it can more easily be understood by other programmers. Sets of functions can be grouped into subroutines called virtual instruments (VI). Virtual instruments can be further organized according to a user defined hierarchy. By working with a virtual instrument layout the developer can build his own "functions" libraries, based on already tested programmes.

LabVIEW programming enables data acquisition from various PC-based or external acquisition boards, image acquisition and processing, instrumentation control through different connecting links (serial, parallel), motion control, distributed processing, etc. Additionally, the programmer can control different signal conditioning blocks.

The concept of the application diagram is illustrated in Fig. 5, where programme sequences and conditioning branches are marked as the frames which include other programming functions having a graphical representation. All symbols associated to functions are very intuitive and easy to remember and understand.



**Figure 5. The diagram for the application on the remote control**

In order to ease the programming task, the environment includes some debugging tools such as: break point, step-by-step execution, skip over, highlighted execution, pause, etc. The programmer can change parameters during the testing process or can monitor data flow.

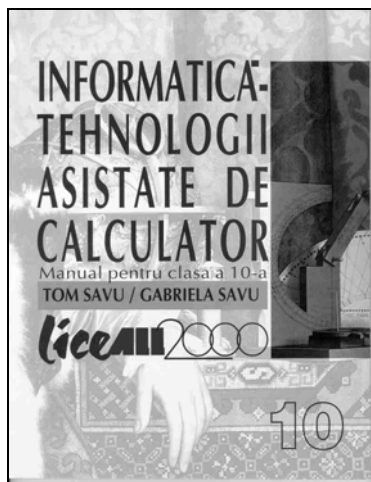
Being the most advanced and the most widely spread graphical programming environment, LabVIEW, the industry-standard programming

environment developed by National Instruments, may be described as a new step in computer programming, allowing a non-professional user, with a minimum operating experience, to realize a complex, user-friendly, professional looking interface, while taking full advantage of the big number of specialized functions and procedures.

## 2. Training in graphical programming

In Romania, considering that graphical programming competencies are, for pupils in technological education, as useful as office software packages, graphical programming topics were introduced in the 10<sup>th</sup> year syllabus and a dedicated manual was printed (Fig. 6).

The Romanian Ministry of Education acquired, in 2002, with a great support from National Instruments Corp. U.S.A., LabVIEW full development licenses for several hundreds high-schools, distributing the first licenses especially in the units where the environment was already known, evaluation versions were previously used or teachers had some LabVIEW training or prerequisites.

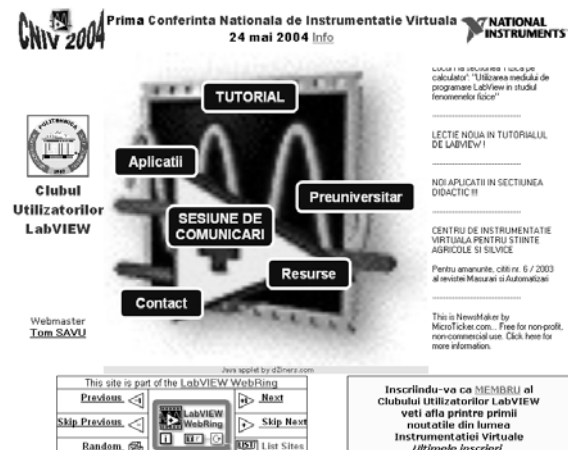


**Figure 6. High-school manual for graphical programming, introduced in 2000**

After a short period during which teachers' LabVIEW training was organized in independent training centers, two of the main Romanian universities (POLITEHNICA University of Bucharest and the Bucharest University) are now joining into a consortium for organizing the teachers training at new and higher quality standards. LabVIEW training is intended to become compulsory for computer science teachers and to be considered as part of the continuing education programmes for the others.

Training will be recognized only in centers which will be accredited by the university consortium, an accreditation condition being that for the tutors to be trained in one of these universities.

Using LabVIEW as a programming tool is encouraged not only in computer science classes, but also for developing useful applications for physics, chemistry, mechanics and other classes. Applications developed by high-school teachers and pupils are advertised through LabVIEW User Club (Fig. 7), a set of Web pages listed also in the LabVIEW Web Ring.



**Figure 7. LabVIEW Users' Club main Web page**

After several years during which an annual seminar was organized, allowing the high-school teachers and pupils to present their developments, this year a National Conference of Virtual Instrumentation will host together presentations from industry, research, academia and high-schools.

A national project is now under preparation [3], for implementing computerized measurements into high-school laboratories. Several high-schools had already received free data acquisition boards from National Instruments Corp. U.S.A., teachers and pupils being encouraged to develop their own set-ups and software applications (Fig. 8).

In the frame of the Comenius "Hands-on Science" project, the institutions of the two authors are co-operating to further support the spread of LabVIEW programming in high schools. Within this frame they signed a protocol for strategic collaboration to support among other objectives: the assistance to high school teachers in using LabVIEW, the coordination of funding efforts for the acquisition of needed

equipment; the preparation of a special section for high school teachers at the Annual National Conference on Virtual Instrumentation. This year Conference already exposed the notable results obtained by the Romanian programming community from high schools. The section dedicated to teachers included more than 15 papers dedicated to the use of LabVIEW in both simulation and data acquisition. Romanian authors were encouraged to participate with their contribution to the International Conference in Slovenia.



**Figure 8. Data acquisition application developed in a high-school physics laboratory (courtesy of “Grigore Moisil” Theoretical High-School, Bucharest, Romania)**

The Comenius project team will consider also the possible integration of its activities with other training institutions in Romania (i.e. the CREDIS Center [4]) in order to exploit in the best conditions the available resources.

Some funds from the project budget will be dedicated to the acquisition of hardware required for the preparation of some experiments in high schools laboratories (physics, chemistry, etc.). The purchasing will be directed towards

sensing components and systems and some training kits which can be converted into laboratory autonomous experiments.

Additionally, demo set-ups with complex instruments from the National Institute for Lasers, Plasma and Radiation Physics will be available to high school teachers during dedicated visits at the Institute.

The project will promote also:

- a closer cooperation with network members interested into the subject;
- the financial support to teachers interested into this type of training to participate to international events.

### 3. Acknowledgements

The authors would like to acknowledge the financial support from the EU within the frame of the Comenius network “Hands-on Science”, project No.

### 4. References

- [1] National Instruments Corp. LabVIEW User Manual. U.S.A., Austin; 2003.
- [2] Savu T., Savu G. Informatics – Computer Aided Technologies (Original title: Informatica – Tehnologii Asistate de Calculator). Romania, Bucharest: ALL Publishing House; 2000.
- [3] Savu T., LabVIEW Virtual Laboratories (Original title: Laboratoare Virtuale LabVIEW). Proceedings of the National e-Learning Conference; 2003 Jul 3-5; Romania, Bucharest: University of Bucharest.
- [4] M. Munteanu, B. Logofatu, R. Lincke, „Instrumentatie virtuala, LabVIEW”, Universitatea din Bucuresti Editura CREDIS 2001, 466 pag.