Using virtual instrumentation in the remotely controlled experiment

Nicolae Patrascoiu*

University of Petrosani, Automation, Computers, Electrical and Energetics Engineering Department, 20 Universitatii street, Petrosani, 332007, Romania

Abstract. The virtual instruments, built in LabVIEW graphical language, are the basis of those applications in terms of input signals generation, output signals acquisition, data processing, and results display in a form that is explicit and easy to understand. Because through these applications, all the commands needed to control power supplies or programmable devices for signal commutation or data acquisition equipment are available via the front panel, offering the possibility to control many experiments. This makes it possible to access them via the desktop, which makes remote control possible through remote control applications.

1 Introduction

It is known that, in the study of electronic devices and circuits, performance characteristics are used to understand their behavior. Among these, static characteristics are employed to measure specific electrical quantities that vary slowly or remain largely constant over time, and for diodes, in the example analyzed in this work, the current-voltage (I-V) characteristic is typically used.

In this paper, the case of measuring static characteristics remotely is exemplified for semiconductor diodes.

For a rectifier diode, the I-V characteristic in the first quadrant, obtained under forward bias, is of interest because, under reverse bias, the diode is practically blocked. For a Zener diode, the I-V characteristic in the third quadrant, obtained under reverse bias, is of interest as it exhibits the voltage stabilization effect [1-2].

Based on this aspect, in the operation of the laboratory stand controlled by software, it is necessary to adjust the applied voltage value on the diode, as well as to select its polarity. It is also required that, simultaneously with setting the input voltage, the voltage drop across the diode and the current flowing through it are measured.

2 Hardware structure

The necessary hardware structure for visualizing and interpreting these measurements was designed to control all input variables and read the outputs.

* Corresponding author: nicolaepatrascoiu@upet.ro

In this regard, equipment and devices have been used that enable software control via the computer's communication ports, and all control elements required for capturing static characteristics are arranged on the front panel of the virtual instrument created for this purpose.

The hardware structure of the entire experimental study system of semiconductor diodes is shown in Figure 1.

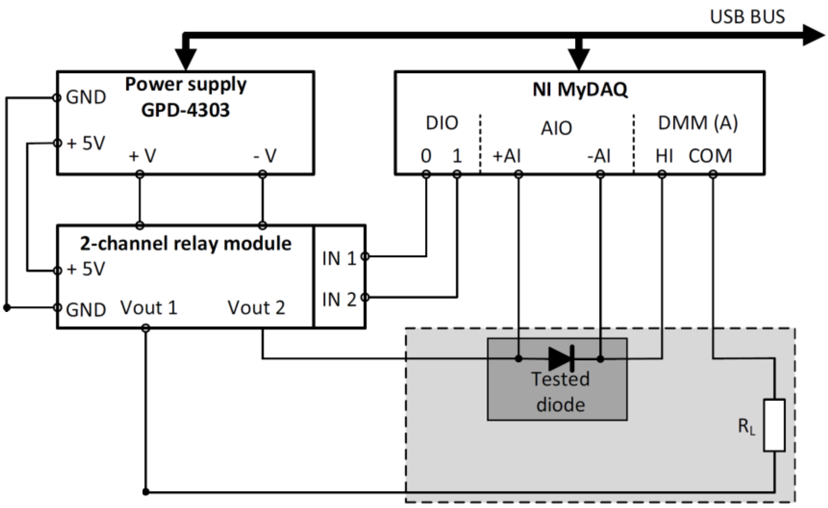


Fig. 1. The hardware structure of the experimental system.

This diagram includes the following components:

1. GPD–4303S, a programmable voltage power supply that offers facilities for both front panel programming and software programming, with USB drivers and LabVIEW support;
2. A 2-channel relay module, allowing the user to build the connection scheme for controlling the polarity of the power supply outputs. Control is achieved via TTL-compatible signals applied to control inputs IN1 and IN2, respectively;
3. NI myDAQ, a portable low-cost data acquisition device (DAQ) that uses NI LabVIEW-based software tools, enabling users to generate signals or measure and analyze acquired signals [3].

The GPD-4303S power supply provides the user with three adjustable DC voltage channels: two of them (CH1 and CH2) in the range of 0–30V, and one (CH3) in the range of 0–5V, each with a current capacity of 3A, along with a fixed 5V channel (CH4), also with a capacity of 3A.

Through the standard USB interface, the power supply can be connected to a PC, and via the included USB drivers and LabVIEW software, the three output channels can be controlled through LabVIEW applications. The output voltage adjustments can be made with a resolution of 1 mV, and current adjustments with 1 mA, either via the digital controls on the front panel of the power supply or through software applications. In this case, these adjustments are performed using the "Static diode characteristics" software application, which will be presented later in this paper [4].

The 2-channel relay module allows the user to build the connection scheme for controlling the polarity of the power supply output. The NC (normally closed) contact of one relay and the NO (normally open) contact of the other relay are used to apply the output voltage (+V and -V) of the power supply, as shown in Figure 2. By generating logical combinations 01 or 10 at the control inputs IN1 and IN2 of the module, the COM contacts of the two relays produce the two polarities of the output voltage: +Vout1 and -Vout2 or -Vout1 and +Vout2.

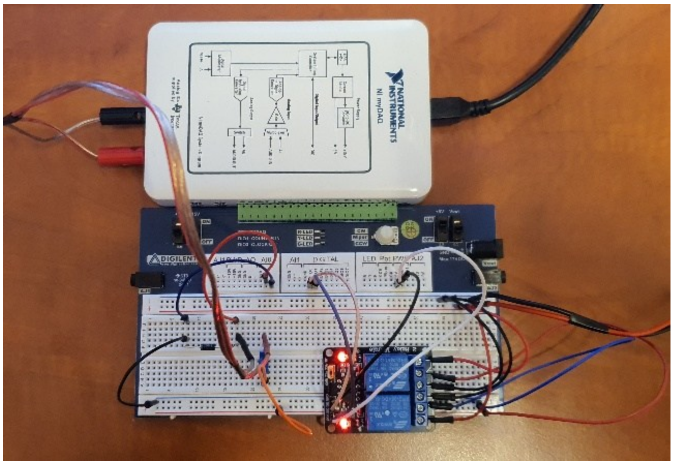


Fig. 2. The connections of the experimental system

The resulting output voltage will be used to plot the characteristics corresponding to the forward or reverse bias of the tested diodes.

3 Software structure

The software required for generating input signals, acquiring the responses, and processing them in order to obtain data as well as their graphical representation is developed in the LabVIEW graphical programming environment and is referred to as a virtual instrument. This virtual instrument is composed of the *front panel*, which represents the user interface of the application, and the *block diagram*, which constitutes the actual program for controlling the laboratory setup [5].

3.1 The front panel of the virtual instrument

The front panel, shown in Figure 3, represents the user interface and includes controls and indicators. The *controls* are input elements through which the user can enter data or commands, while the *indicators* are output elements used to display or visualize data in the format defined by the user.

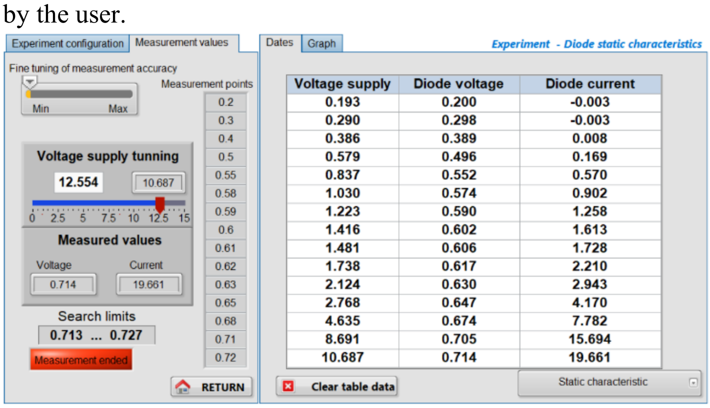


Fig. 3. The front panel of the virtual instrument

The front panel of this virtual instrument includes several windows used during the setup stages of the experiment, as well as for data visualization or graphical representation of characteristics.

These windows are used for:

1. **Experiment configuration**, through which the diode polarization (forward or reverse) and the communication USB port are selected. Additionally, the number of measurement points, which will define the shape of the static characteristic, is set, along with the value range around these points within which the parameters are measured.
2. **The measurement procedure**, which follows the structure of the general algorithm presented in Figure 4.

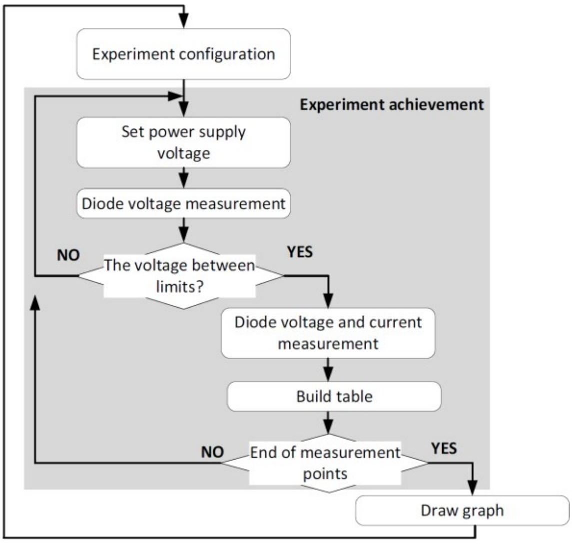


Fig. 4. The general algorithm of the virtual instrument

The user has access to a slider-type control through which they can modify the supply voltage value. As a result, both the voltage drop across the diode and the current through the diode will change. If the diode voltage falls within the previously set limits relative to the set value of the measurement point, the two quantities will be recorded in the constructed table. The procedure continues until all values corresponding to the number of measurement points set earlier are recorded.

3. **The window corresponding to recorded data and graph visualization** offers the user the option to switch between them. From the data window, one of the possible graphical representations can be selected using a selection button. Once the graphical view is selected by choosing the “Graphic” window, the user has access to a graph whose appearance can be adjusted in terms of line thickness and interpolation method used for rendering. The user can choose between the following interpolation methods via a selection button: *linear*, *polynomial*, *spline*, and *rational*, with the possibility to adjust the interpolation factor for each of them.

3.2 The block diagram of the virtual instrument

The block diagram, shown in Figure 5 includes operations, functions, and programming structures required to run the application.

The program is designed to run sequentially, following the algorithm presented earlier, using the *Sequence* structure.

The main structure includes two sequences:

- The **first** is dedicated to the settings necessary for conducting the experiment.
- The **second** is dedicated to measurements, data acquisition, and data processing.

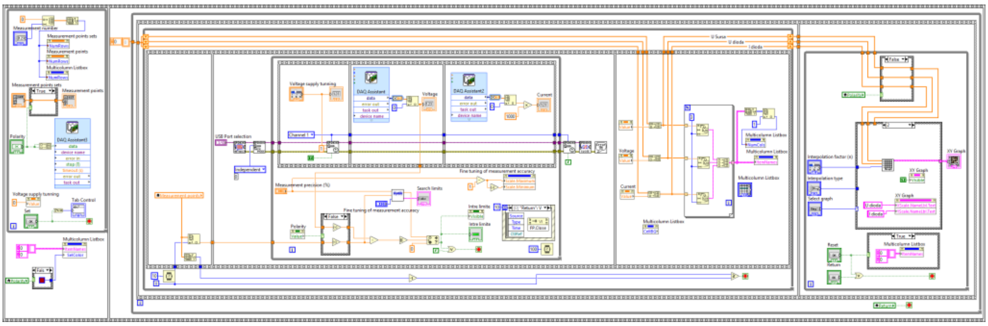


Fig. 5. Block diagram of the virtual instrument

Transition between the first and second sequence occurs by stopping a *While loop* in which the program remains until the Next button is pressed. Pressing this button signals the completion of all setup procedures.

The second sequence is the actual experimental phase and includes all the necessary programming structures. The program remains in this sequence until the experiment ends, triggered by activating the Return command.

This second sequence includes two sub-sequences:

- One for data acquisition (Figure 6)
- One for processing the measured data (Figure 7)

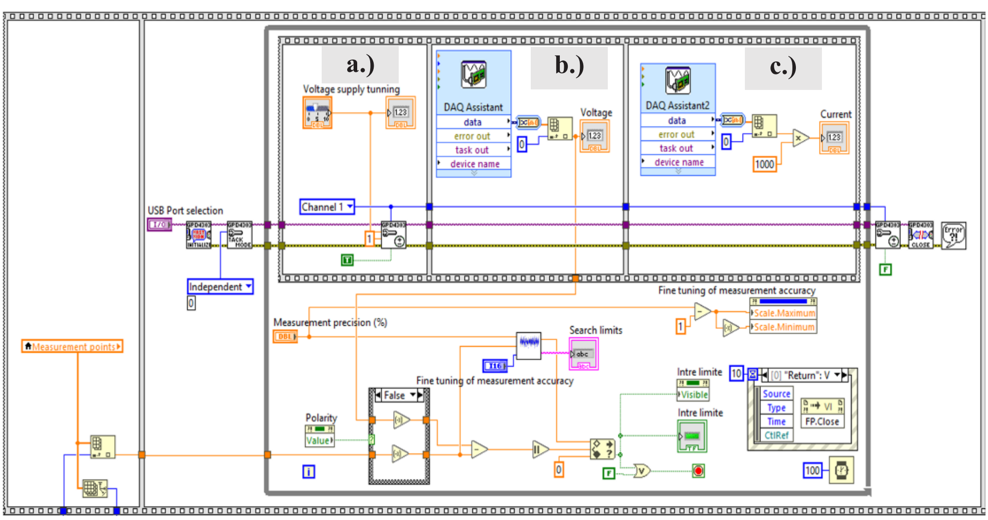


Fig. 6. Data acquisition sequence

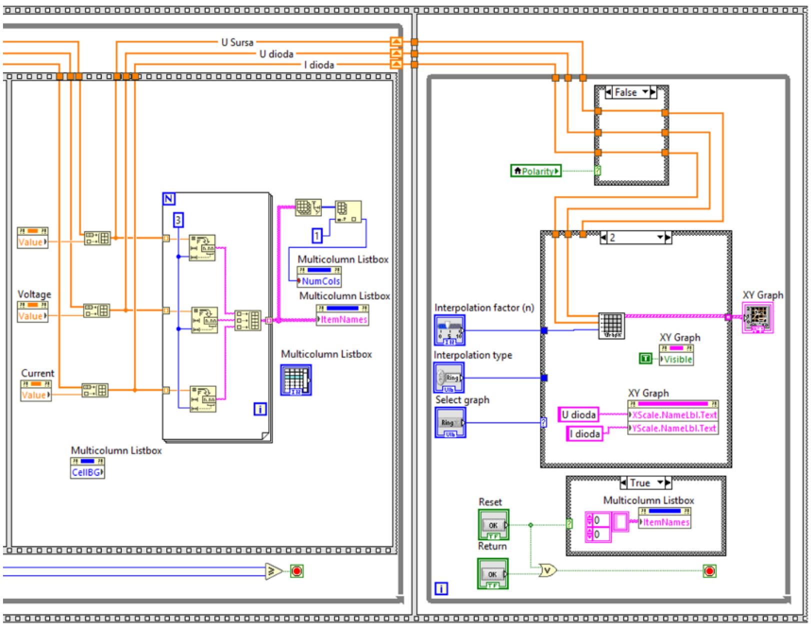
The data acquisition sub-sequence is used for:

1. Controlling the supply voltage from the programmable source *GPT4303,* (Figure 6.a)
2. Measuring the voltage drop across the diode, (Figure 6.b)

3. Measuring the current through the diode (Figure 6.c)

These operations are performed using driver functions for interfacing the power supply with the LabVIEW programming environment, as well as *DAQ Assistant* functions.

Fig.7. Data processing sequence



The sequence used for data processing includes the operations necessary to build the measured data tables as well as those necessary for their graphic representation.

4 Remote desktop applications study

Remote desktop applications are software tools that allow remote access and control of a computing system via a communications network [6-7].

In the context of virtual instrumentation, these applications facilitate remote access to laboratory equipment such as that presented above, allowing the control and monitoring of experiments with sequential circuits from a distance. In search of a useful application for this purpose, in this paper, the following applications are considered:

1. **TeamViewer** which is a multi-platform application commonly used for technical support, online collaboration, IT administration, and remote work, allowing full access to a device located in another location. It has the advantage of an easy-to-use interface and a free version. However, frequent use in the free version may lead to restrictions, and commercial licenses can be expensive.
2. **AnyDesk** which is an application highly regarded for its speed and efficiency, making it a popular choice for users seeking a simple and reliable solution for remote device control. It is commonly used for technical support, IT administration, or accessing files and applications remotely. It is also multi-platform, offering low latency and fast data transmission, which is particularly useful for real-time connections. AnyDesk provides free features for personal use, boasts high responsiveness even on slower connections, and offers flexible configuration options for individual users or companies.

3. **Microsoft Remote Desktop** is a free application developed by Microsoft that allows users to access and control computers remotely. It is integrated into Windows operating systems and is best suited for users of Microsoft OS. It allows connecting to a Windows PC from anywhere, over a local network or the internet, providing support for fast connections and efficient resource management. It has the disadvantage of being limited mainly to Windows devices (the server must be Windows) as well as the interface and configurations can be less intuitive for beginners.
4. **RustDesk** is a free and open-source remote desktop software designed to provide a secure, self-hosted alternative to commercial solutions like TeamViewer or AnyDesk. Built in the Rust programming language, RustDesk emphasizes performance, privacy, and full user control. Unlike many proprietary tools, RustDesk allows users to host their own servers, ensuring that no third-party servers handle your data. This is particularly valuable for individuals and organizations with strict data protection policies or those who need to comply with privacy regulations (e.g., GDPR). His key features include:
   - Cross-platform support **(**Windows, macOS, Linux, iOS, Android**)**
   - End-to-end encryption
   - Clipboard synchronization
   - File transfer capabilities
   - No configuration required for NAT traversal (by default)
   - Lightweight and fast interface

   RustDesk is ideal for tech-savvy users, sysadmins, or companies looking to maintain full ownership of their remote access infrastructure, without sacrificing usability or speed.

For an objective evaluation of remote desktop applications in the context of virtual instrumentation, Table 1 presents a comparative analysis based on the specific criteria of such applications:

Table 1. Comparative analysis for remote desktop applications.

Criterion	RustDesk	AnyDesk	TeamViewer	Chrome Remote Desktop
Cost	Free	Medium	Hight (comercial)	Free
Performance	Excellency	Verry Good	Good	Satisfactory
Security	Very hight	Hight	Hight	Average
Cross-platform	Complete	Complete	Complete	Limited
Self-hosting	Yes	No	No	No
Hight Configurability	Hight	Average	Average	Very limited

In the context of remote control of applications via virtual instrumentation, specific evaluation criteria include:

- **Latency and real-time response**: RustDesk demonstrates high performance and speed, being comparable to AnyDesk in terms of the low latency required for real-time control of sequential circuits.
- **Connection stability**: TeamViewer offers the most stable connection for extended sessions, but AnyDesk presents frequent disconnection issues, a critical aspect for long-term experiments.

- **Data transfer**: For virtual instrumentation applications that require frequent transfer of experimental data, all analyzed applications support file transfer, except Chrome Remote Desktop which has limited functionality.

The choice of RustDesk for implementing the remote control system of virtual instrumentation is based on several key factors specific to academic and research-oriented applications [8].

- **Economic Considerations:** Its open-source nature completely eliminates licensing costs—an essential factor for academic projects operating under limited budgets. Unlike TeamViewer, which can cost up to €1500 annually for commercial licenses, RustDesk provides free access to all features.
- **Security and Privacy:** End-to-end encryption ensures the confidentiality of transmitted data—critical in experiments involving sequential circuits where data may contain sensitive research information. The self-hosting capability eliminates the risks associated with centralized servers provided by commercial vendors.
- **Transparency and Auditability:** The open-source code allows for security auditing and adaptation to specific application requirements—an essential advantage in the academic environment where methodological transparency is fundamental.
- **Real-Time Performance:** Its modern architecture, built on Rust, ensures high performance and low latency—critical aspects for the real-time control of virtual instruments developed in LabVIEW.
- **Connection Stability:** The connection spread is more stable compared to AnyDesk, which frequently suffers from disconnection and video sync issues—detrimental for experiments that require continuous monitoring.
- **Architectural Flexibility:** The ability to configure a dedicated relay server offers complete autonomy over the communication infrastructure, eliminating dependency on third-party services and ensuring system availability for critical virtual instrumentation applications.

5 Conclusions

This paper presents one of the applications developed using virtual instrumentation in LabVIEW, which enables the performance of several experiments related to various topics from the educational or research curriculum in the field of electrical engineering.

The experiment is operated solely through the front panel of the virtual instrument, without requiring any direct user interaction with the laboratory equipment or devices.
This opens up the possibility of remotely controlling the entire experiment in a remote laboratory-type configuration.

By choosing the RustDesk remote desktop application, an optimal balance is ensured between performance, security, flexibility, and cost for academic and research-oriented applications, making it a viable alternative to expensive commercial solutions. The self-hosting capability and open-source nature are decisive advantages for applications in these fields, where transparency and data control are top priorities.

References

1. E. L. Frenzel, Contemporary Electronics: Fundamentals, Devices, Circuits, and Systems, (McGraw-Hill, 2014)
2. Horowitz, Paul & Hill, Winfield. The Art of Electronics (3rd ed.). (Cambridge University Press 2015)

3. https://mil.ufl.edu/3111/docs/myDAQ/myDAQ_user_guide_and_specs.pdf [Accessed 29.08.2025]
4. Rana K.P.S, Vineet K. and J. Mendiratta, An educational laboratory virtual instrumentation suite assisted experiment for studying fundamentals of series resistance–inductance–capacitance circuit, European Journal of Engineering Education 42.6, pp.1220-1239, DOI:10.1080/03043797.2017.1284764 , 2017
5. https://www.ni.com/pdf/training/us/core-1-sample-course-manual.pdf, [Accessed 01.07.2025]
6. G. Blokdyk, Remote Desktop Software A Complete Guide – 2020 Edition, ISBN 978-1867346197 (5STARCooks, 2021)
7. Kumar, S., & Sood, S. K.. Remote Desktop Services: Technologies and Security Challenges. Journal of Network and Computer Applications, 133, 45–60. https://doi.org/10.1016/j.jnca.2019.01.014, 2019
8. RustDesk: Open-source Remote Desktop Software. Disponibil la: https://rustdesk.com (RustDesk Documentation 2024).