

ADVANCED SOLUTIONS FOR PERFORMING REAL EXPERIMENTS OVER THE INTERNET

Raymond Berntzen¹, Jan Olav Strandman², Tor A. Fjeldly³ and Michael S. Shur⁴

Abstract *¾ We describe several advanced solutions for performing laboratory experiments over the Internet. These include the use of the commercial software LabVIEW 6i, and solutions based on the functionalities of modern web browsers. The former is relatively easily implemented by means of the powerful graphical programming environment with which so-called virtual instruments are created. The browser-based solutions are more flexible, and enable the server to respond in several different formats, giving the client great flexibility in storing, processing and presenting the data received, and to handle security and queuing issues. To enhance the learning experience, we have developed a user-friendly interface on the client-side, with circuit and instrument-like images in which the experimental configuration and parameters can be controlled. Our remote lab is applied to characterization of semiconductor devices and circuits as part of senior level courses in electrical engineering.*

Index Terms *¾ Remote laboratory, World Wide Web, Internet, remote education, LabVIEW, COM+, semiconductor devices, device characterization.*

INTRODUCTION

The Internet offers interesting possibilities for disseminating educational material to students, both locally and as part of remote education. Laboratory experiments are a vital part of engineering education, which have so far been considered impractical for distance learning. However, recent advances in Internet/web technologies and computer-controlled instrumentation presently permit net-based techniques to be utilized for setting up remote laboratory access [1]. Also, the use of Internet and studio classrooms is an emerging trend for promoting "individual discovery" as a strategy for enhancing engineering education. Here we describe how these techniques can be combined with traditional laboratory exercises to create an efficient environment for interactive, on-line operation of lab experiments over the Internet, to be used either in a studio setting, or from a remote location as part of a distance learning strategy.

Our system is based on a server/client architecture, where the clients communicate with the server and the experimental setup using modern web browsers [1], or in

some cases using separate software that can be downloaded from the Internet for free.

Previous versions of our remote lab were based on a TCP/IP (Transmission Control Protocol/Internet Protocol) solution, which used a Java Applet on the client (i.e., student) side [1-3]. This was achieved by means of a JVM (Java Virtual Machine) in the web browser that can download and execute Java code. The client sees a pop-up window that provides interaction and communication directly with the server. Java Applets provide good control. However, unsigned Applets make it awkward for the client to store and present the measurement data, and to transfer them to other applications (except by "cut-and-paste") because of Java's security structure. A further problem with Java is that the functionality of an Applet may vary between different browsers.

The LabVIEW 6i software from National Instruments [4] offers an interesting alternative for providing the desired Internet access to the lab [5,6]. In this solution, the lab-side server runs a full version of LabVIEW 6i, which incorporates Internet communication capabilities and functionalities to access/control instruments and to acquire/output data. The client can communicate with the server and the experimental setup in two ways: either by means of a web browser, which runs a dedicated CGI (Common Gateway Interface) script in the server, or using the LabVIEW Player, which can be downloaded and installed for free. In both cases, the client will be capable of executing VIs (Virtual Instruments) specifically developed for the experiment in question, providing the client with great ability to control the remote instrument and to receive and present the desired experimental data.

Other solutions seek to utilize more fully the functionalities of the most recent browsers, enabling the server system to respond in many different formats, such as JavaScript, HTML or XML (eXtensible Markup Language), which gives the client great flexibility in storing, processing and presenting the data received [6,7]. This is achieved by creating Web-solutions based on either the ISAPI (Information Server Application Information Interface) server extensions, or on a COM+ (Component Object Model with extensions) solution at the lab side.

The interfacing between the server and the experimental setup is realized either with a GPIB card and instruments, or by means of inexpensive PCI/ISA cards,

¹ Raymond Berntzen, UniK – Center for Technology, N-2027 Kjeller, Norway, raymond@unik.no

² Jan Olav Strandman, UniK – Center for Technology, N-2027 Kjeller, Norway, janolav@unik.no

³ Tor A. Fjeldly, UniK – Center for Technology, Norwegian University of Science and Technology, N-2027 Kjeller, Norway, torfj@unik.no

⁴ Michael S. Shur, ECSE Department, Rensselaer Polytechnic Institute, Troy, NY 12180-3590, shurm@rpi.edu

which can be installed directly in the expansion slots of the server [8].

The present version of the lab is dedicated to semiconductor device characterization. It includes several experiments that are performed on a microelectronic test chip, and is used as a lab module in courses on semiconductor devices and circuits at the senior or first year graduate level at the Norwegian University of Science and Technology and at Rensselaer Polytechnic Institute. Both remote and local students are enrolled in these courses, and can conduct the lab exercises on an equal footing. By analyzing experimental characteristics obtained from individual devices, the students can extract SPICE parameters, which are used for simulating the behavior of circuits, for example, by means of our circuit simulator AIM-Spice [9]. Such simulations can then be compared with experiments on simple circuits in the test chip. For the theoretical part of such a course, we use our textbook "Introduction to Device Modeling and Circuit Simulation" [10]. We have also good experience with on-line classroom lab demonstrations for more in-depth illustrations of the course material.

Based on this concept, we envision that lab courses and course modules within many disciplines of engineering and science can be offered inexpensively to students world-wide.

LABORATORY SETUP

An overview of the remote lab measurement setup is shown in Fig. 1. The server is a Dell Power Edge 4300 with a 500 MHz Pentium III processor, Windows 2000 Server operating system, and MIIS (Microsoft Internet Information Server) version 5.0. The main instrument is an HP 4142B Modular DC Source/Monitor, a high-speed, accurate, and computer controlled DC parametric measurement instrument for characterizing semiconductor devices. It can force or measure voltages and currents within 4 ms, and store up to 1023 data samples in the internal memory. Up to eight different plug-in modules can be used with this instrument, allowing us to tailor the instrument to suit our needs.

In our setup, we installed two HP 41421B SMUs (Source Measurement Units), in addition to the built-in 0 V source GNDU (GrouND Unit). The HP 34970A Data Acquisition/Switch unit (with three HP 34904A Matrix Switch plugin modules) is used as a matrix switch in the remote lab, allowing the client to configure the experimental set-up. This is done by activating the matrix to make desired connections to terminals/devices of the test chip for a specific measurement series. Figure 1 also shows a Tektronix TDS 3052, 500 MHz oscilloscope. This instrument is used mostly for testing and servicing the setup, but it can also be used as part of the lab for measuring transient events or waveforms. As indicated in Fig. 1, the server communicates with and controls all the instruments via the GPIB bus. The layout of a CMOS test chip used is indicated in the middle right of Fig. 2.

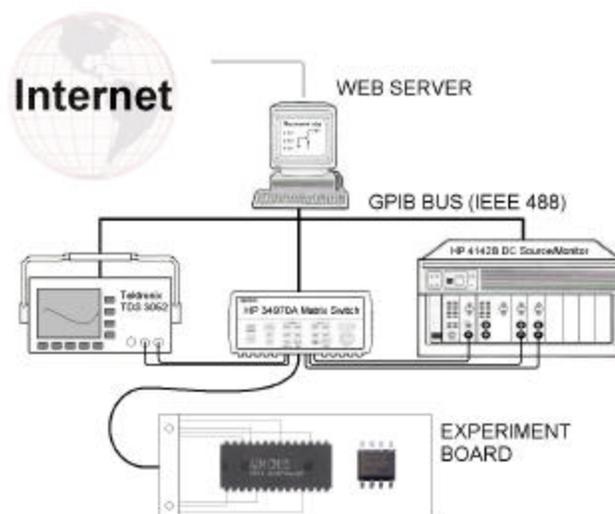


FIGURE 1.

OVERVIEW OF THE REMOTE LABORATORY SETUP. THE CMOS TEST CHIPS ON THE EXPERIMENT BOARD INCLUDE DIODES, *N*- AND *P*-CHANNEL MOSFETS OF DIFFERENT GEOMETRIES, AND CMOS TEST CIRCUITS.

The HP VISA (Virtual Instrument Software Architecture) for Windows was used for developing I/O (input/output) applications in C++ based on the VISA I/O library. In VISA, applications can open multiple sessions to a resource simultaneously. However, when it is necessary for a session to execute without interference, a lock function can be invoked. In our implementation, the *Exclusive Lock* is used for granting so-called atomic access for a given thread to resources that are otherwise shared between many users.

CLIENT INTERFACE

An ambition of our remote lab is to provide a good learning experience. For this purpose, we have developed a user-friendly client-side interface with circuit- and instrument-like images to allow the experimental configuration and parameters to be controlled interactively by the user. This interface was realized by means of the SVG (Scalable Vector Graphics) file format [11]. SVG is a new vector-based, open-standard file format developed by the World Wide Web Consortium, which represents a new generation of dynamic, data-driven and interactive graphics. An Adobe SVG Viewer plugin may be downloaded from [11].

Figure 2 shows an interactive web page created with SVG and JavaScript, where the client may configure the measured to be performed. Several setups are prepared for direct selection, as indicated in the table at the lower left-hand side. An arrow indicates the selection made, and the corresponding circuit diagram pops up in the middle of the page. However, by clicking on the various elements in the circuit diagram, a given setup may be further customized and the external wiring of the test chip is displayed in the upper right of Fig. 2. The experimental settings are entered

in the tables in the lower right. The external connections are physically implemented via the matrix switch once the "submit" button is activated.

This client interface is compatible with several of the technical solutions described below.

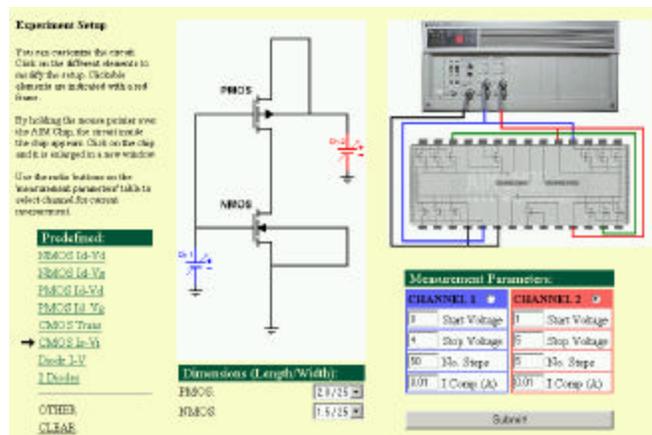


FIGURE. 2

INTERACTIVE SVG CLIENT WINDOW FOR CONFIGURING THE EXPERIMENTAL SETUP. A CMOS TEST CHIP AND ITS EXTERNAL WIRING ARE SHOWN ON THE RIGHT-HAND SIDE. THE CIRCUIT SHOWN MEASURES THE SHORT-CIRCUIT CURRENT DURING SWITCHING.

LABVIEW SOLUTIONS

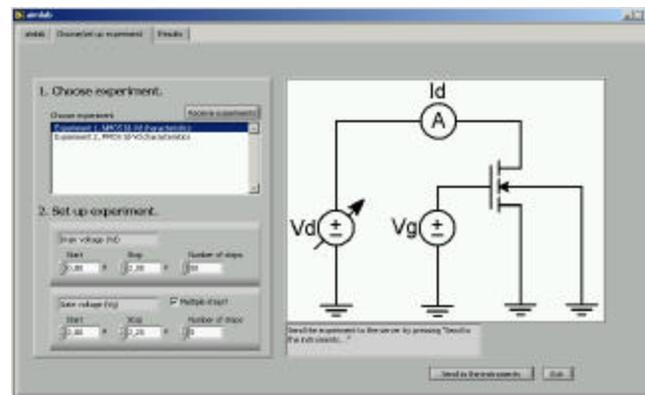
LabVIEW version 6i from National Instruments is a powerful graphical programming development environment for data acquisition and control, data analysis, and data presentation. It comes with advanced Internet-ready capabilities and the concept of measurement intelligence, which includes automatic measurement hardware configuration for fast application development. Instead of writing program code, we build Virtual Instruments (VIs) with front panel user interfaces that may contain numeric displays, meters, charts, advanced graphs, etc. Functionality is specified in block diagrams. The VIs allow control of any GPIB instrument.

It is possible for the client to run the lab experiments over the Internet in two different modes, either with the LabVIEW Player or using a Web browser.

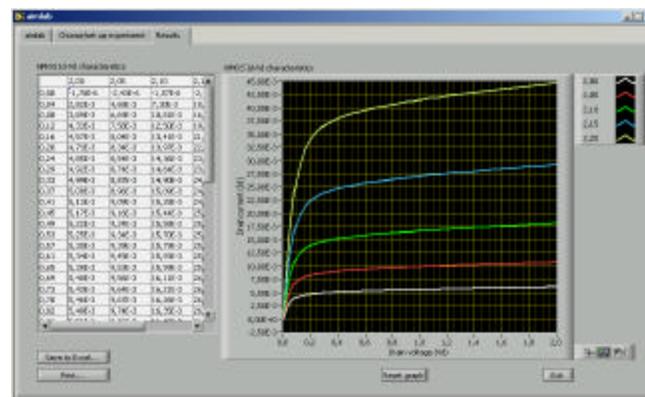
Using LabVIEW Player

The LabVIEW Player is an executable that is available free of charge from National Instruments. It is designed specifically for sharing measurement and automation knowledge across the web in the form of secure LabVIEW VIs. By downloading and installing the LabVIEW Player, anyone can open and run LabVIEW VIs. Hence, clients obtain test results and measurement data from remote locations by means of the built-in Internet tools. The LabVIEW Player is richly equipped with functionalities for manipulating the data received and the graphs generated to fit the users need.

Figure 3a) shows the panel of a VI created for a set of experiments on current-voltage characteristics of *n*-channel MOSFETs. Each entry in the list of experiment contains instructions for the matrix switch on how to connect the measurement instrument to the appropriate terminals of the



A)



B)

FIGURE. 3

CLIENT WINDOWS GENERATED USING THE LABVIEW PLAYER. A) REMOTE LAB VIRTUAL INSTRUMENT PANEL FOR SELECTING EXPERIMENTS AND SPECIFYING EXPERIMENTAL CONDITIONS; B) MEASURED MOSFET CURRENT-VOLTAGE CHARACTERISTICS FOR A SELECTION MADE IN A).

test chip. The ranges and steps for the gate and drain voltages applied to the transistor are specified in the table. The corresponding data received by the client is shown in Fig. 3b). Note that the experimental data can be forwarded to an Excel document for further analysis with the push of a button.

A drawback of the LabVIEW Player is its considerable size, about 17.2 MB. This makes it inconvenient for downloading via a telephone modem for home use.

Using CGI Scripts

By means of the LabVIEW Internet Developers Toolkit, it is also possible to use a built-in HTTP (Hypertext Transport Protocol) server to make the VI front panels viewable from regular web browsers. The Toolkit also includes VIs for

building CGI scripts for use with the LabVIEW server. Hence, with this solution, there is no need to download the large LabVIEW Player executable. Moreover, this solution provides additional flexibility in the transmission of measurement data, permitting different formats such as JavaScript, HTML or XML. Hence, the interactive SVG client window shown in Fig. 2 can be used for configuring the experimental setup. A simpler solution with predetermined setups is shown in Fig. 4.

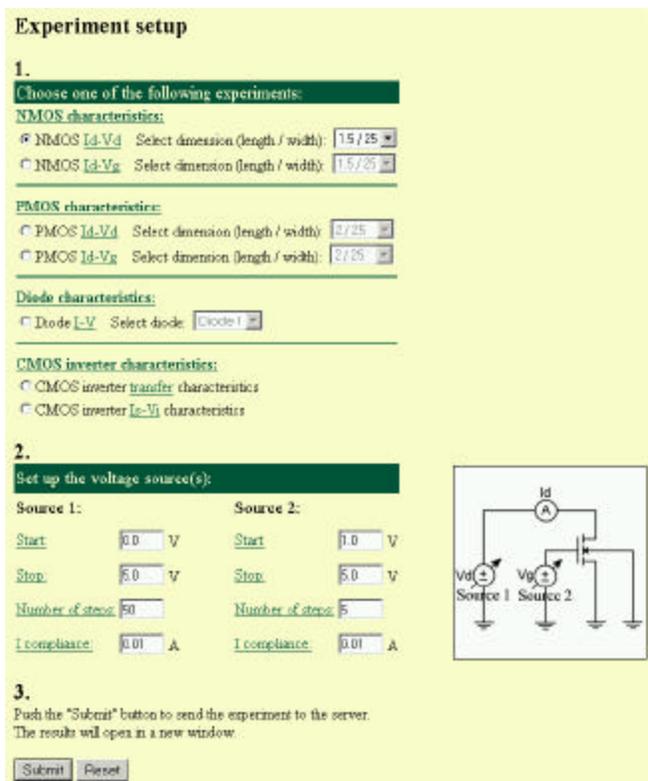


FIGURE 4

CLIENT WINDOW GENERATED USING THE LABVIEW CGI SCRIPT FOR SELECTION OF PREDEFINED EXPERIMENTS AND SETUP PARAMETERS.

The corresponding VI, which includes building blocks for communication with the client browser and the instruments is shown in Fig. 5. Figure 6 shows a presentation of experimental results at the client side. Figures 4 and 6 are the CGI equivalents of Fig. 3 a) and b) obtained by using the LabVIEW Player.

Note that from built-in servers it is possible to respond to multiple clients and continuously update their displays. This is an example of how the Toolkit libraries can be programmed to convert VIs into image files for display within HTML pages. Security levels can also be incorporated into the server to limit access to front panels and data. Likewise, access to the virtual instruments can be password controlled based on the user's IP address.

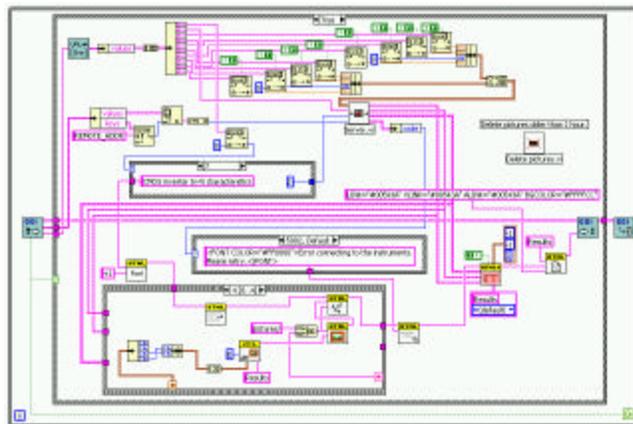


FIGURE 5

BLOCK DIAGRAM OF LABVIEW VIRTUAL INSTRUMENT WITH BUILDING BLOCKS FOR THE CGI SOLUTION.

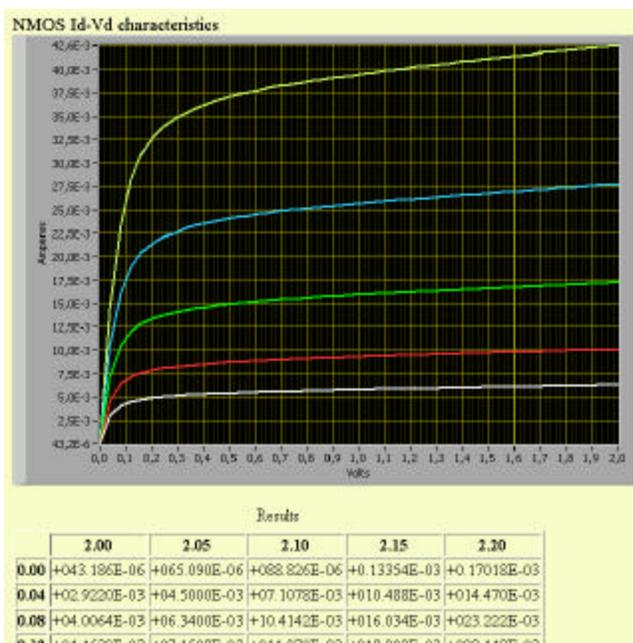


FIGURE 6

CLIENT SIDE DISPLAY OF EXPERIMENTAL RESULTS USING THE LABVIEW CGI SOLUTION.

ISAPI AND COM+ SOLUTIONS

Using ISAPI

ISAPI server extensions belong to the Microsoft ActiveX technology, and act to extend the capabilities of the HTTP (Hypertext Transport Protocol) server included in the IIS. Such extensions are an alternative to the CGI standard incorporated into HTTP, allowing browser programs to interact with scripts or separate executable programs running on the server. However, the ISAPI extensions are faster, more versatile, and have added functionality compared to

CGI. For specific tasks, such as our remote lab, ISAPI extensions are programmed in visual C++ by means of a 'wizard' that comes with Microsoft Visual Studio.

ISAPI server extensions are implemented as a DLL (Dynamic Link Library) file on the Web server. They load when called, and act as a programming interface between the server and the server-side applications. In response to requests from the client, they typically return information as HTML/XML code that may include JavaScripts or Applets. If a stylesheet created in the XSL is applied to the XML code, the data can be directly presented as graphs or tables in the client terminal [6]. Different kinds of ISAPI filter functions can be implemented to perform various useful tasks for our remote lab. These include port management, authentication and access restriction, page translation, logging and traffic analysis, and encryption. Since the filters execute with every HTTP request, they can provide IP address based access control. Port management and encryption filters contain useful security features. The log filter can trap client information such as date, time and path, from which an activity log can be created.

The page translation filter can be used to transform XML to, for example, HTML or WML using XSL stylesheets. For clients using up-to-date browsers, the XSL transformation can take place on the client-side, which requires that a stylesheet be transferred to the client along with the XML data. If the client browser does not support an XML parser, the filter executes the XSL stylesheet on the server, in which case the stylesheet is selected based on the browser used. In fact, an ISAPI module running on MIIS can hook onto the XML data stream and transform the code into HTML on the fly.

Using COM+

COM+ is the second generation of COM, which is a set of services that allows us to create object-oriented, customizable and upgradeable, component-based distributed applications. One of the primary goals of COM is to make it easy to write multi-tier applications that can be reused from any programming language. COM+ includes new features that make COM easier to use and simplifies the development. Some of these features are: component load-balancing, just-in-time activation, asynchronous method invocation, in-memory database, queued components, improved administrative services.

COM+ applications are typically middle-tier applications; that is, they move information between clients and back-end resources such as databases (in our case, measurement data from instruments).

The COM+ component developed for the remote laboratory is a library application, which is compiled to a DLL, and is loaded into the process of the client. The component is called from an HTML-form in the experiment configuration page, as shown in Fig. 7, via an ASP (Active Server Page). This page also receives the measurement results from the component and presents them to the client.

Figure 8 is an example of data presented in the client browser using SVG graphics.

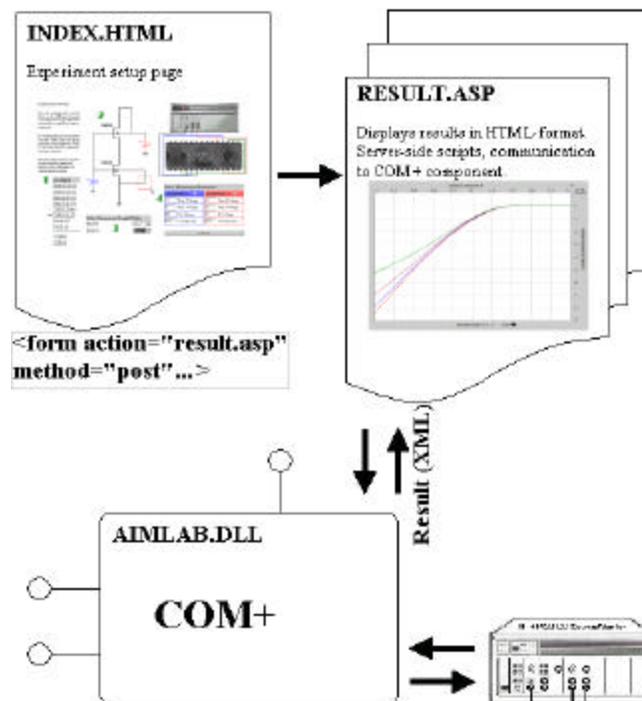


FIGURE 7

REMOTE LAB SYSTEM BASED A COM+ SOLUTION. THE HTML SUBMISSION PAGE FROM THE CLIENT CALLS THE COM+ COMPONENT VIA AN ACTIVE SERVER PAGE. MEASURED DATA ARE RETURNED ALONG THE SAME PATH.

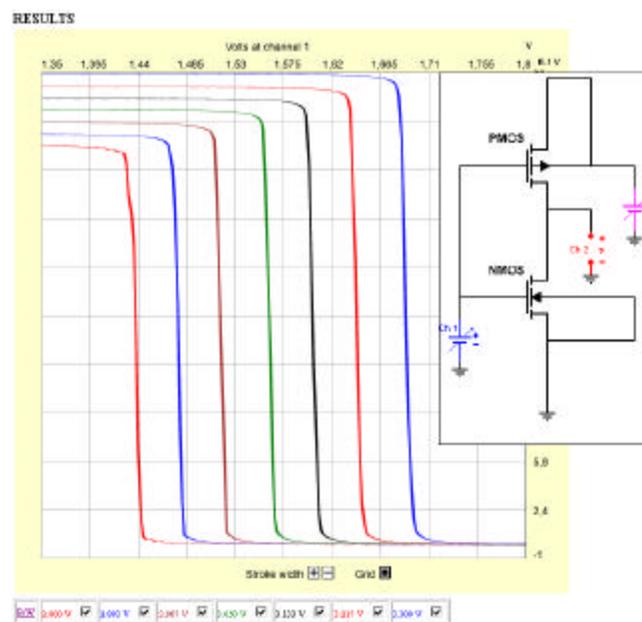


FIGURE 8

SVG PRESENTATION OF CMOS TRANSFER CHARACTERISTICS IN THE COM+ SOLUTION. SVG'S INTERACTIVITY ALLOWS ZOOMING, PANNING, ETC. OF THE GRAPHICS IN THE CLIENT BROWSER.

ASP is a relatively new technology from Microsoft, widely used for creating dynamic web sites and web-distributed applications. Complex pages and applications can be created with ease, and it has the ability to interact with COM/COM+ components. VBScript contained in the result.asp document performs server-side scripting to build the HTML-code that displays the measurement results. JavaScript can also be used, but VBScript was chosen because of its powerful features in string and table manipulation.

CONCLUSION

We have investigated several possible realizations of a system for conducting interactive, on-line experiments via the Internet. Specifically, we considered a lab for characterization and analysis of microelectronic devices, which constitutes a module in courses on semiconductor technology at our institutions.

Two of the solutions are based on LabVIEW 6i from National Instruments, using either the LabVIEW Player executable on the client side, or by running a CGI script via a regular web browser. The former solution requires the relatively large Player executable to be downloaded and installed. The CGI solution gives more freedom in creating a tailor made system, for example by utilizing the XML format and SVG graphics. Both of these solutions are relatively easily implemented as a result of the powerful graphical programming development system that comes with LabVIEW.

Two other solutions are based on the ActiveX technology from Microsoft. They utilize either the ISAPI server extensions or a COM+ solution. Both take advantage of the advanced functionalities of modern web browsers and servers, permitting the server to transmit data in several suitable formats for presentation and analysis at the server side. Besides, both offer a number of useful tasks related to security, queuing, logging, etc. The COM+ solution is probably the most viable of the two, since it offers a well-structured development environment that has been widely adopted by the web community.

At our remote lab site in Norway, we have presently included the two LabVIEW solutions in addition to the COM+ solution [12]. From our alternative site at Rensselaer Polytechnic Institute, we offer a similar remote lab based on a TCP/IP solution [3].

ACKNOWLEDGMENT

This work was supported by a grant from the governments in the Nordic countries through Nordunet2.

REFERENCES

- [1] Shen, H., Xu, Z., Dalager, B., Kristiansen, V., Strøm, Ø., Shur, M. S., Fjeldly, T. A., Lu, J., Ytterdal, T., "Conducting Laboratory Experiments over the Internet", *IEEE Trans. on Education*, Vol. 42, No. 3, 1999, pp. 180-185.
- [2] Fjeldly, T. A., Shur, M. S., Shen, H., Ytterdal, T., "AIM-Lab: A System for Remote Characterization of Electronic Devices and Circuits over the Internet", *Proc. 3rd IEEE Int. Caracas Conf. on Devices, Circuits and Systems (ICDCS-2000)*, Cancun, Mexico, IEEE Catalog No. 00TH8474C, pp. 143.1-143.6 (2000).
- [3] URL: [Http://nina.ecse.rpi.edu/shur/remote/](http://nina.ecse.rpi.edu/shur/remote/).
- [4] URL: [Http://www.ni.com/labview/](http://www.ni.com/labview/).
- [5] Berntzen, R., M.Sc. Thesis, Norwegian University of Science and Technology (2001).
- [6] Smith, K., Strandman, J. O., Berntzen, R., Fjeldly, T. A., Shur, M. S., Shen, H., "Advanced Internet Technology in Laboratory Modules for Distance Learning", *Proc. ASEE 2001 Annual Meeting*, Albuquerque, NM, accepted for publication.
- [7] Strandman, J. O., M.Sc. Thesis, Norwegian University of Science and Technology (2000).
- [8] Shen, H., Shur, M. S., Fjeldly, T. A., Smith, K., "Low-Cost Modules for Remote Engineering Education: Performing Laboratory Experiments over the Internet", *Proc. 29th ASEE/IEEE Frontiers in Education Conference (FIE'00)*, Kansas City, Missouri, ISBN 0-7803-6424-4, TID-7 (2000).
- [9] URL: [Http://www.aimspice.com/](http://www.aimspice.com/).
- [10] Fjeldly, T. A., Ytterdal, T., Shur, M. S., *Introduction to Device Modeling and Circuit Simulation*, John Wiley & Sons, New York, NY (1997).
- [11] URL: [Http://www.adobe.com:82/svg/](http://www.adobe.com:82/svg/).
- [12] URL: [Http://aimlab.unik.no/](http://aimlab.unik.no/).