AC 2010-1866: A WEB SERVICE AND INTERFACE FOR ELECTRONIC DEVICE CHARACTERIZATION

Sumit Dutta, University of Illinois at Urbana-Champaign

Sumit Dutta is currently pursuing a B.S. in electrical engineering at the University of Illinois, expecting to graduate in 2011. His research interests lie broadly in physical electronics. He is a recipient of the NASA Aeronautics Scholarship and Raytheon US FIRST Robotics Scholarship, and is an Edmund J. James Scholar. Sumit is a member of Tau Beta Pi, Engineers without Borders, and Eta Kappa Nu.

Shreya Prakash, University of Illinois at Urbana-Champaign

Shreya Prakash is currently pursuing a B.S. degree in electrical engineering from the University of Illinois, expecting to graduate in 2012. Her research interests lie in physical electronics and circuits. She is a member of WECE.

David Estrada, University of Illinois at Urbana-Champaign

David Estrada received his B.S. in EE from Boise State (2007) and M.S. in ECE from UIUC (2009). He is now pursuing his Ph.D. in ECE, working on thermal characterization of CNT arrays. He is also interested in developing sub-nanosecond electrical thermometry techniques. David is the recipient of the SURGE Fellowship, Micron Fellowship, and NDSEG and NSF Fellowships.

Eric Pop, University of Illinois at Urbana-Champaign

Eric Pop received the B.S. and M.Eng. degrees in EE and the B.S. degree in physics, in 1999, from the Massachusetts Institute of Technology, Cambridge, and the Ph.D. degree in EE from Stanford University, Stanford, CA, in 2005. Since 2007, he has been with the Department of Electrical and Computer Engineering, University of Illinois, Urbana–Champaign. He is also with the Micro and Nanotechnology Lab and the Beckman Institute at the University of Illinois. His group studies nanoscale energy transport, power dissipation in integrated circuits, and novel nonvolatile memory. Prior to Illinois, he spent 16 months at Intel, working on phase-change memory, and completed a postdoc at Stanford, investigating thermal properties of carbon nanotubes. Dr. Pop is also a member of MRS and HKN, and the Faculty Advisor for the HKN Alpha chapter at the University of Illinois. He received the Arnold O. Beckman Research Award (2007), DARPA Young Faculty Award (2008), the SRC Fellowship, and SRC Best Paper and Best Poster awards.

A Web Service and Interface for Electronic Device Characterization

Abstract

User access to lab instruments is often accomplished with heavyweight program extensions such as LabView applets and plug-ins, which do not provide quick and simple remote instrument control. Instead, growing support for Web Services (WS) has recently enabled lightweight XML messaging for instrument control and measurement readings. Here we report the development of a WS and website interface designed for remote measurements of electronic devices, with classroom and Internet-wide applications. Using standard browsers (e.g. Firefox, Safari even on an iPhone) without additional plug-ins, remote users control a Keithley 2612 source-measurement unit and monitor results in real time from any computer connected to the network. This interface presently allows users to perform a variety of typical transistor measurements, but can be expanded to future applications in micromechanical or bio/nano device testing.

Introduction

Instruments for electrical testing may be connected and controlled by a computer for data capture via the IEEE-488 general purpose interface bus (GPIB) or through Ethernet ports utilizing the VXI-11 protocol.[1] This has ushered in an era of remote instrument control for virtual laboratories as the World-Wide-Web (WWW) and WWW-capable instruments have become ubiquitous. While web-based remote instrumentation control has been investigated for over a decade, these environments have generally centered on Java server software, PHP Hypertext Processor (PHP) server scripts, or direct control via LabView.[1-4] Previous LabView implementations of remote electronic measurements require users to download either a ~100 megabyte LabView plug-in or a specific Java Runtime Engine,[5] in addition to having a compliant browser.[6] Other remote laboratories required LabView or XWindows to be downloaded.[7]

The advent of Web Services (WS) enables the connectivity between vastly different computing services by combining the Hypertext Transfer Protocol (HTTP) for universal communication that does not depend on opening other network ports,[8] and the Extensible Markup Language (XML) for universally parsable text data transfer.[9] A WS is a standardized means to expose the inputs and outputs of a process to a variety of other remote systems using standardized messaging protocols. The service-oriented model presented by WS has been successfully used for remote instrument control middleware with many possible interfaces.[9-10] Representational State Transfer-based (ReSTful) WS are a class of WS in which messaging is accomplished by having the client send requests using HTTP (e.g. GET, PUT, POST, and DELETE) while receiving XML re-

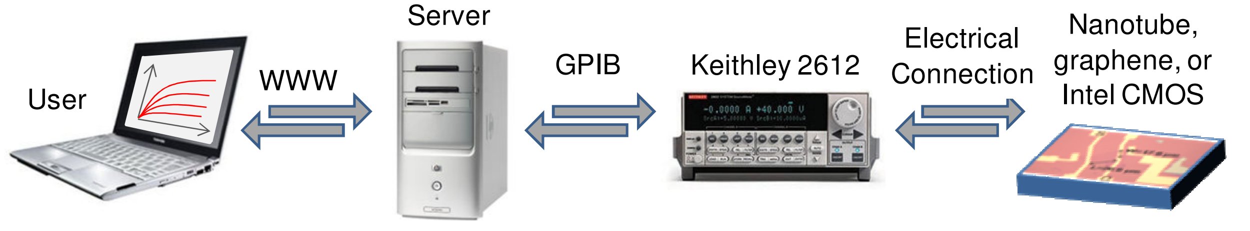


Figure 1. Schematic of user interaction with remote instrument and device via WWW interface.

sponses. By making use of the recently added Web Services capabilities in LabView, virtual instrument (VI) control panels can be operated via external computers by exposing the inputs and outputs of instrument controls as a WS. This WS allows external users to request certain measurements to be performed using the LabView-controlled instrument by sending an HTTP request to the LabView-exposed service URL.

In this work we report the development of a WS and website interface designed for remotely conducting measurement tests on electronic devices, with classroom and Internet-wide applications. This interface presently allows users to perform a variety of typical transistor measurements, and is appropriately abstracted to enable micromechanical or bio/nano device testing applications in the near future. In addition, remote users of our WS can control a Keithley 2612 measurement unit with *standard browsers* without additional plug-ins (even Safari on an iPhone), and monitor their results in *real time* from any connected computer or web-browsing mobile device (Fig. 1).

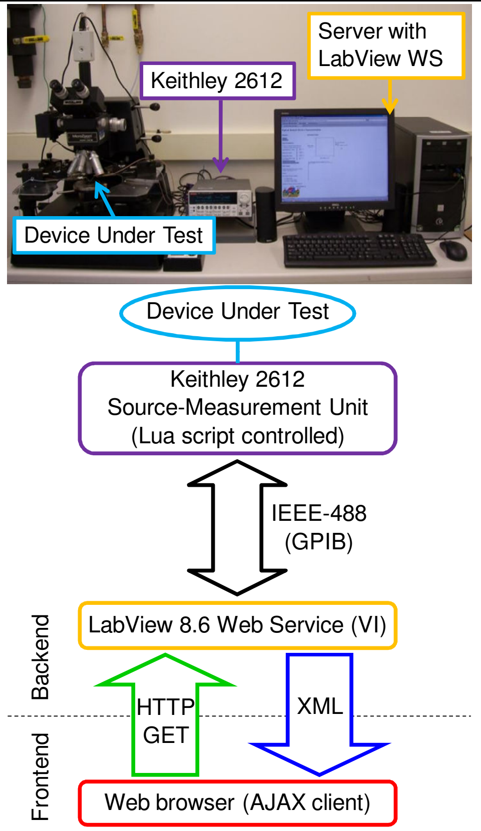
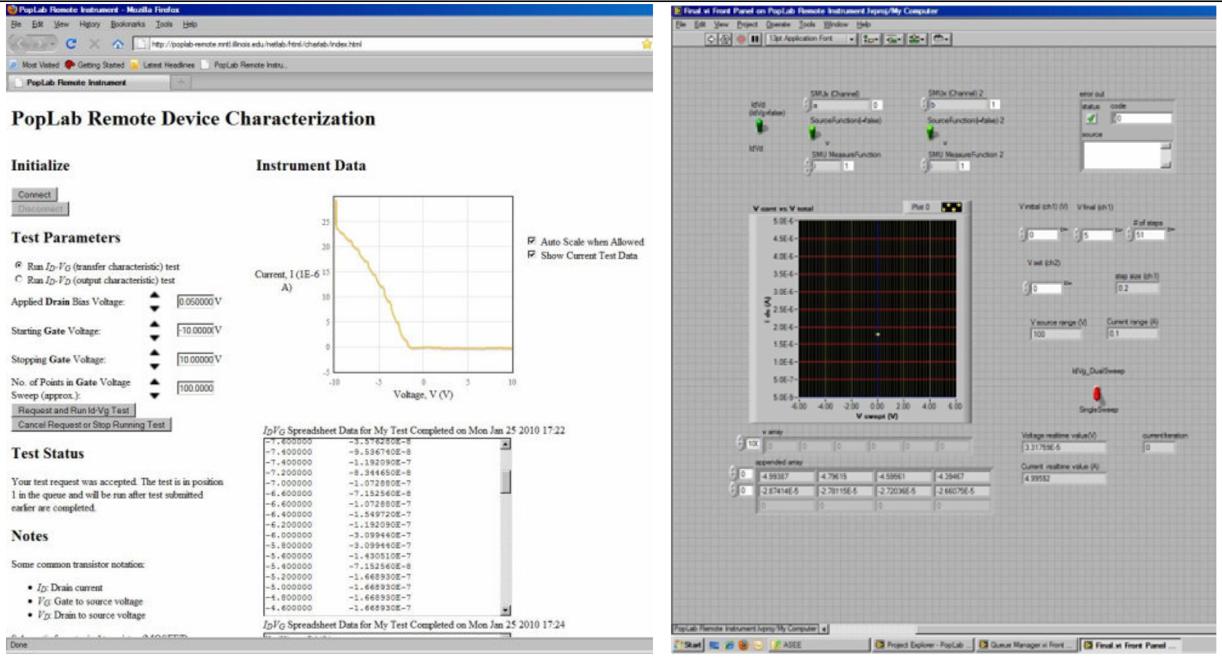

Server with LabView WS
Keithley 2612
Device Under Test
Device Under Test
Keithley 2612 Source-Measurement Unit (Lua script controlled)
IEEE-488 (GPIB)
Backend
LabView 8.6 Web Service (VI)
HTTP GET
XML
Frontend
Web browser (AJAX client)

Figure 2. Photograph of hardware used for our remote laboratory (top) and schematic of the remote instrument WS and Web interface architecture (bottom).

The initial WS design is aimed as part of a large undergraduate electronic device course (~150 students), where individual lab access is prohibitive. With the WS access, students benefit from doing real-time measurements, and can perform subsequent data analysis. Currently, the WS enables measurements of typical silicon transistors fabricated at the University of Illinois (Fig. 2 and Fig. 5), state of the art nanoscale transistors provided by Intel, and cutting edge research devices (e.g. carbon nanotubes) fabricated in the Micro and Nanotechnology Laboratory at the University of Illinois. The WS backend software successfully places student test requests into a queue, conducts the tests in order, and provides ongoing measurements to all connected users. We find that representational state transfer, asynchronous JavaScript, and XML messaging (ReSTful AJAX) are sufficiently quick to allow an adjustable real-time data transfer rate tuned to the desired measurement sensitivity.

Web Service (WS) and Remote Instrumentation

The WS developed for the remote controlled instrument is a ReSTful WS that accepts simple HTTP requests and returns XML files with results. This enables any type of client, e.g. a JavaScript enabled Web browser (with no other plug-ins) or other independent software, to run electrical characteristic measurements on a remote server. The server is hosted by the Pop Lab[11] in the Micro and Nanotechnology Laboratory at the University of Illinois in our case.[11]



PopLab Remote Instrument - Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://poplab-remote.mntl.illinois.edu/netlab/html/chartlab/index.html
Most Visited Getting Started Latest Headlines PopLab Remote Instru...
PopLab Remote Instrument

PopLab Remote Device Characterization

Initialize

Connect
Disconnect

Test Parameters

● Run I_D-V_G (transfer characteristic) test
○ Run I_D-V_D (output characteristic) test

Applied Drain Bias Voltage: 0.050000 V

Starting Gate Voltage: -10.0000 V

Stopping Gate Voltage: 10.00000 V

No. of Points in Gate Voltage Sweep (approx.): 100.0000

Request and Run Id-Vg Test
Cancel Request or Stop Running Test

Test Status

Your test request was accepted. The test is in position 1 in the queue and will be run after test submitted earlier are completed.

Notes

Some common transistor notation:

- I_D: Drain current
- V_G: Gate to source voltage
- V_D: Drain to source voltage

Instrument Data

☑ Auto Scale when Allowed
☑ Show Current Test Data

I_DV_G Spreadsheet Data for My Test Completed on Mon Jan 25 2010 17:22

```
-7.600000      -3.576280E-8
-7.400000      -9.536740E-8
-7.400000      -1.192090E-7
-7.200000      -8.344650E-8
-7.000000      -1.072880E-7
-6.600000      -7.152560E-8
-6.600000      -1.072880E-7
-6.400000      -1.549720E-7
-6.200000      -1.192090E-7
-6.000000      -3.099440E-7
-5.800000      -3.099440E-7
-5.600000      -1.430510E-7
-5.400000      -7.152560E-8
-5.200000      -1.668930E-7
-5.000000      -1.668930E-7
-4.800000      -1.668930E-7
-4.600000      -1.668930E-7
```

I_DV_G Spreadsheet Data for My Test Completed on Mon Jan 25 2010 17:24

Done

SMUx (Channel): a 0

SMUx (Channel) 2: b 1

SMU MeasureFunction: 1

SMU MeasureFunction 2: 1

code: 0

V initial (ch1) (V): 0

V final (ch1): 5

# of steps: 51

V set (ch2): 0

step size (ch1): 0.2

V source range (V): 100

Current range (A): 0.1

v array: 10X 0 0 0 0 0 0

appended array:
0 | -4.99387 | -4.79615 | -4.59561 | -4.39467
0 | -2.87414E-5 | -2.78115E-5 | -2.72036E-5 | -2.66075E-5
 | 0 | 0 | 0 | 0

Voltage realtime value(V): 3.31759E-5

current iteration: 0

Current realtime value (A): 4.99582

Figure 3. Screenshots illustrating the web interface of the remote instrument seen by the user (left) and the LabView VI backend which runs on the web server (right).

To demonstrate connectivity with the Pop Lab service endpoint, a user-friendly AJAX client was developed for use by students in a large undergraduate class. The AJAX client works on any Web browser with support for a JavaScript XML HTTP Request, a common feature in most Web browsers on all operating systems including Internet Explorer, Mozilla Firefox, and even Safari running on the iPhone. The client does not require any page refreshes or new page loads because all controls and displays are on a single page, as shown in Fig. 3. Real-time HTML plotting is accomplished using Flot, an open source JavaScript library that uses the HTML `<canvas>` tag or emulates it when not available.[12] The XML HTTP requests that must be implemented by the AJAX client are described in Tables 1 and 2.

Currently, the user may select either a drain current vs. gate voltage (I_D-V_{GS}) or drain current vs. drain voltage (I_D-V_{DS}) measurement to be performed on transistors using a remotely accessed Keithley 2612. Appropriate sourcing parameters are pre-programmed in the LabView VIs, which also set the voltage and current compliances. Information about the requested measurement and associated compliance limits is determined in the `initialRequest`, an initiating request made by the client (Table 1). While determining the information for requesting a test, data currently being collected by the remote lab instrument is displayed to all active sessions. This feature allows students to collaborate as well as view the results of others to develop a better intuition. Users can choose whether to acquire and display data for the ongoing test, whether it is their own, or the test of another user. Nevertheless, the test requests made by all connected users execute in the chronological order in which they were received.

As the test data is acquired by the instrument, it is made available to the client in real time. Every second, a `newDataRequest` with no data other than the standard HTTP headers is sent, for which an XML data response is provided with the newest data points recently acquired by the lab instrument. This data is about 15 kB in size, highlighting the low bandwidth requests owed to sending data in small packets rather than in a high-overhead HTML page. The delay in transporting data over the HTTP protocol is typically ~100 ms, which is usually fast enough for getting data in real time, at a rate below ~10 Hz.

Users view data in a separate text box for each test run while connected. The text box provides a mechanism for the data to be easily copied and pasted into a spreadsheet, plotted, and analyzed. The AJAX client data plot is also updated in real time. A screen capture (`Alt+PrintScreen`) can copy the plot for use elsewhere. Only a mouse is needed to operate the AJAX client because the controls are all buttons and numerical input fields that are controlled by arrow buttons.

Table 1: User Session and Data Acquisition Events

Event	HTTP Request	XML Response
initialRequest Establishes identification and session	Browser request: • `HTTP GET /dataprovider` • `formName=initialRequest` • User-identifying HTTP headers	XML text response contains: • Available names of tests • Numerical voltage control limits • Plot/axes options (name, range) Browser interpretation: • Provide list of tests and inputs • Apply limits to input controls • Set plot/axes name and range
newDataRequest Retrieves latest unseen data points from server	Browser request: • `HTTP GET /dataprovider` • `formName=newDataRequest` • User-identifying HTTP headers	XML text response contains: • Ongoing test's name • Whether new test • New data points Browser interpretation: • Update data displayed (plot, text box table) with new points, refreshing if data from a new test • Hold/identify current/past tests

Table 2: Test Queue Manager Events

Event	HTTP Request	XML Response
testRequest Submits request for a user-defined test to be validated and entered into server queue	Browser request: • `HTTP GET /queuehandler` • `formName=testRequest` • User-identifying HTTP headers • Test name • Voltage range/bias • Number of points	XML text response contains: • Test name • Corrected voltage range/bias • Number of points Browser interpretation: • Notify user of successful test request, showing modifications
stopRequest Cancels running/queued test	Browser request: • `HTTP GET /queuehandler` • `formName=stopRequest` • User-identifying HTTP headers	XML text response contains: • Whether test stopped, why Browser interpretation: • Notify if error

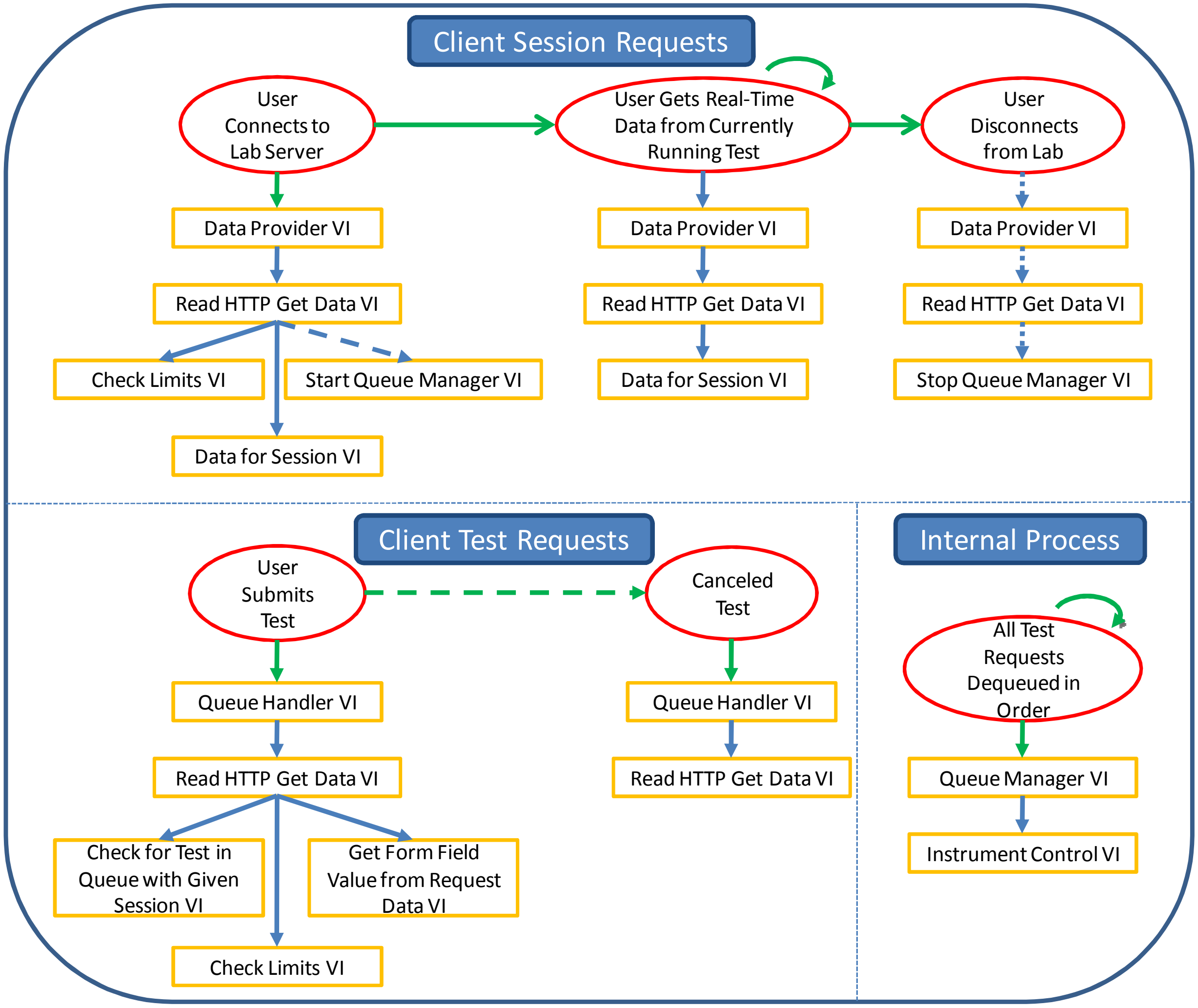


Figure 4. Simultaneous processes in client interaction with LabView server virtual instruments.

The WS backend service is written with eleven VIs in LabView (Fig. 4). The three categories of VIs are (1) software and hardware control options (2) VIs invoked by client requests such as the queue handler, and (3) internal process VIs which consist of the instrument control VI and queue manager VI. When a request is made by a client, that request is made to a particular VI in category (2) which has a unique URL. The VI checks the request inputs against the options in category (1) and accordingly takes action by verifying or correcting the test inputs and then enqueuing the test using the queue manager VI in category (3). The queue manager VI runs the instrument control VI for every dequeued test, dequeuing tests in the order they were enqueued.

The technologies used for this remote laboratory are remarkably stable and secure. An analysis of previous technologies used for remote laboratories shows that AJAX has the highest ranking in security, second-highest ranking next to HTML in universality, the third highest ranking in power and flexibility, and the highest ranking in development facilities.[10] AJAX development lends itself to interfacing with Web Services,[10] e.g. LabView ReSTful WS, due to the compatibility of (1) AJAX HTTP requests with ReSTful WS HTTP requests and (2) AJAX XML response actions with ReSTful XML responses. Hence AJAX is established as a readily available, crash-free, and commonly usable tool for remote laboratories, making it a paradigm of choice in developing a user interface. The use of WS is also well-warranted because it provides a highly flexible interface and permits dynamic use of different resources.[9]

Example Application for Engineering Education

Students in the undergraduate "Solid-State Electronic Devices" course, ECE 440,[13] will be invited to use the Remote Lab interface to run tests on transistors fabricated as part of the University of Illinois "Theory and Fabrication of Integrated Circuits" course, ECE 444, as shown in Fig. 5. At the end of the Spring 2010 semester, surveys completed by all students in ECE 440 will be compiled to assess the effectiveness of teaching using a remote instrument. With the support of Intel, several modern transistors (65 nm technology) will also be made available for remote testing using the Remote Lab. Students can then observe and compare the short-channel and long-channel device characteristics using modern technology. The interface can be modified for use by students at all experience levels ranging from elementary school to postgraduate studies.

One example analysis that students perform is the extraction of the threshold voltage (V_T) parameter of the MOSFET connected in the lab. This post-measurement analysis is often performed using the linear extrapolation method: taking the derivative of the drain current (I_D) with respect to the gate voltage (V_G) to find the inflection point, and then identifying the x-intercept of the tangent line at that point.[14] As shown in Fig. 6, this method is applied on data measured through the AJAX client as well as on data taken by the instrument locally in the lab, and shows close agreement between the locally and remotely acquired data. This can then be compared to the calculated V_T using reasonable doping concentrations and the drawn device dimensions.

Although complete survey results from ECE 440 students are not yet available (the course will cover MOSFETs in April 2010, after the submission of this manuscript), some preliminary feedback is available from undergraduates involved in nanotechnology research. Mr. Chun Ming Chin found the tool to be intuitive and simple. He and two other undergraduates suggested that relevant MOSFET device parameters (e.g. dimensions) could also be provided for additional post-measurement calculations. Another undergraduate, Mr. Gautam Shine, suggested the ability to hold multiple data sets on the

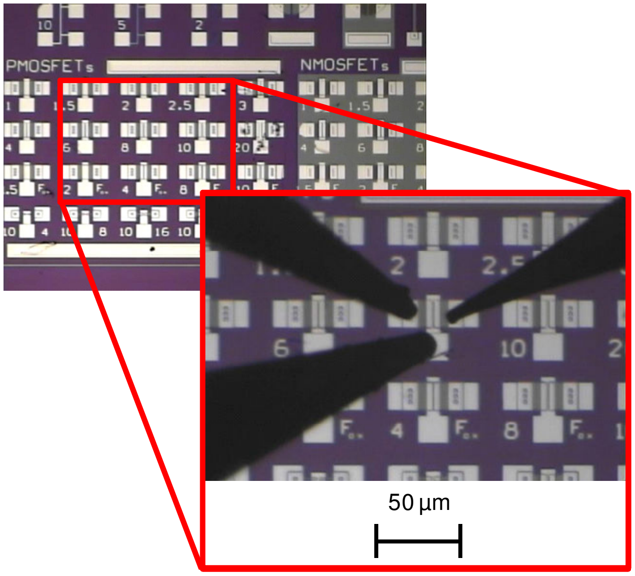


Figure 5. Optical image of transistors fabricated in the University of Illinois "Theory and Fabrication of Integrated Circuits" course. In the enlarged image, an individual transistor is contacted by the probe station as part of the remote lab setup.

AJAX client for a family of curves, which is possible with different client software but not currently with the Canvas plotting tool used in the AJAX client.

It may also become likely that students accustomed to using lab equipment would still prefer using that over a remote laboratory, but the improved accessibility of such an environment over a flexible Web interface is bound to be preferred by those farther from the university lab setting.[15] The remote lab is likely to gain popularity once it is made publicly available online, and different testing instruments are connected to the master server.[16]

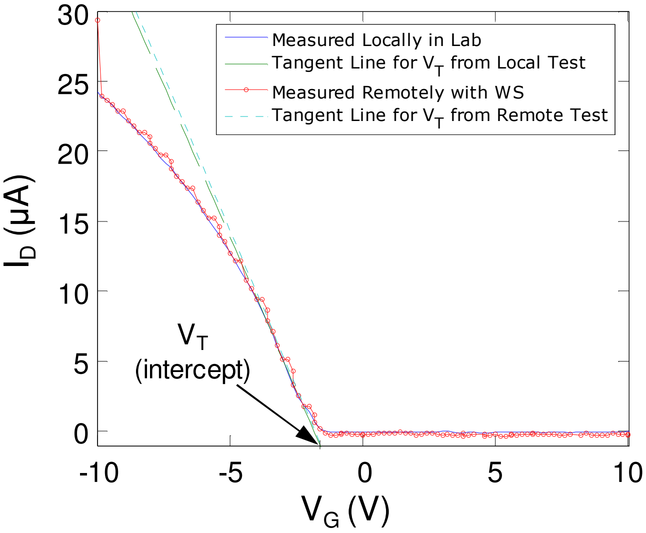


Figure 6. Comparison of local and remote measurements on a pMOSFET and subsequent threshold voltage extraction (V_T = -1.82 V from local measurement and V_T = -1.77 V from remote measurement).

Conclusion

We have developed a remote electronic device measurement setup which can run on any modern web browser without requiring additional plug-ins or downloads. The backend software places test requests into a queue, conducts the tests in order, and displays ongoing measurements to all connected users. We find that RESTful AJAX messaging is sufficiently quick for an adjustable real-time data transfer rate tuned to the desired measurement sensitivity.

The remote instrument setup is being applied in a large undergraduate classroom environment, and eventually released Internet-wide. Due to the modular design of the software layers, other instruments with a LabView driver or GPIB interface can be connected to conduct a variety of measurements remotely. In addition to the currently available DC current vs. voltage measurements, ongoing extensions include using the same Keithley 2612 for pulsed current-voltage characterization or connecting an additional instrument via GPIB to perform capacitance-voltage characterization.[17] Further standardization is possible by using the Web Services Resource Framework and other well-recognized data interchange methods.[9] Finally, we note that the flexibility of this interface allows for continued growth of the Remote Lab in available instrument backends as well as with the user interface frontends.

Acknowledgments

We would like to thank Mark Bohr and Intel Corporation for providing the Keithley 2612 and other Intel test devices for use in the remote laboratory. We also appreciate undergraduate feedback by Mr. Jen-Chieh Liu and Mr. Shengzhao Wu. Additional funding was made possible by the National Science Foundation grants CCF-0829907 and CAREER ECCS-0954423, the nanoHUB.org, the Micron Technology Foundation (D.E), the NSF and NDSEG Graduate Fellowships (D.E.), and the NASA Aeronautics Scholarship (S.D.).

Bibliography

1. J. Santos, *et al.*, "Instrumentation remote control through Internet with PHP," in *IEEE International Conference on VECIMS*, Istanbul, Turkey, 2008, pp. 41-4.
2. H. A. Basher and S. A. Isa, "On-campus and online virtual laboratory experiments with LabVIEW," in *IEEE SoutheastCon*, Memphis, TN, 2006, pp. 325-30.
3. D. Grimaldi, *et al.*, "Java-based distributed measurement systems," *IEEE Transactions on Instrumentation and Measurement,* vol. 47, pp. 100-103, Feb 1998.
4. W. Winiecki and M. Karkowski, "A new Java-based software environment for distributed measuring systems design," *IEEE Transactions on Instrumentation and Measurement,* vol. 51, pp. 1340-1346, Dec 2002.
5. L. Costas-Pérez, *et al.*, "Optimization of an industrial sensor and data acquisition laboratory through time sharing and remote access," *IEEE Transactions on Industrial Electronics,* vol. 55, pp. 2397-2404, Jun 2008.
6. S. Rapuano and F. Zoino, "A learning management system including laboratory experiments on measurement instrumentation," *IEEE Transactions on Instrumentation and Measurement,* vol. 55, pp. 1757-1766, Oct 2006.
7. M. A. Stegawaki and R. Schaumann, "A new virtual-instrumentation-based experimenting environment for undergraduate laboratories with application in research and manufacturing," *IEEE Transactions on Instrumentation and Measurement,* vol. 47, pp. 1503-1506, Dec 1998.
8. H. Vargas, *et al.*, "Web-Enabled Remote Scientific Environments," *Computing in Science & Engineering,* vol. 11, pp. 36-46, May-Jun 2009.
9. A. Bagnasco, *et al.*, "A Service-Oriented Educational Laboratory for Electronics," *IEEE Transactions on Industrial Electronics,* vol. 56, pp. 4768-4775, Dec 2009.
10. J. García-Zubia, *et al.*, "Addressing Software Impact in the Design of Remote Laboratories," *IEEE Transactions on Industrial Electronics,* vol. 56, pp. 4757-4767, Dec 2009.
11. *PopLab at Illinois*. Available: http://poplab.ece.illinois.edu/
12. *Flot*. Available: http://code.google.com/p/flot/
13. *ECE 440*. Available: http://courses.ece.illinois.edu/ece440/
14. D. K. Schroder, *Semiconductor Material and Device Characterization*: Wiley-Interscience, 2006.
15. M. C. Plummer, *et al.*, "A circuits II laboratory accessible by Internet," in *ASEE Annual Conference and Exposition*, Montréal, Quebec, 2002, pp. 767-774.
16. P. Arpaia, *et al.*, "A measurement laboratory on geographic network for remote test experiments," *IEEE Transactions on Instrumentation and Measurement,* vol. 49, pp. 992-997, Oct 2000.
17. M. Billaud, *et al.*, "Real measures, virtual instruments," in *IEEE International Caracas Conference on Devices, Circuits and Systems*, Oranjestad, Aruba, Netherlands, 2002, pp. I023-1.