

Different Platforms for Remote Laboratories in Mobile Devices

Ananda Maiti

School of Computing Science and Engineering, VIT University, Vellore, India
Email: anandamaiti@live.com

Balakrushna Tripathy

School of Computing Science and Engineering, VIT University, Vellore, India
Email: tripathybk@vit.ac.in

Abstract— Remote laboratory is an innovative approach to create and provide laboratory experience to geographically dispersed students from anywhere at any time. One of the most important aspects of remote laboratories is to provide the user maximum mobility and freedom to perform experiments. Apart from the PC-based remotely triggered laboratories to enhance technical education, mobile devices can play a major role in wider implementation of the laboratory for hardware-based remote experimentation. In this paper, different techniques, such as, Adobe Flash Lite, HTML5 and SMS for developing platforms for mobile devices are studied and compared.

Index Terms— m-learning, laboratory education, flash lite, online laboratory, HTML5, Java ME

I. INTRODUCTION

Laboratory education is compulsory for an engineering and science education program. Many of the laboratory courses include a set of hardware-based experiments. However, with fast increase in number of students and limited laboratory facilities, it is being difficult to conduct experiment in many cases in conventional laboratories.

Remote laboratory [1-7] can provide a solution to this. It is basically an information communication technology based approach to perform experiments using the same instruments as in a laboratory environment, but without having to be physically in the laboratory. A computerized interface is built to provide access to the laboratory via internet or LAN. The remote laboratories give the users mobility by allowing them to perform experiments beyond class hours when the laboratory is not available for physical access. This also increases the instrument usage efficiency and allows users to perform experiments from anywhere if they have internet or network access. This allows students to perform and learn more.

Remote laboratory development has been studied in detail. Several architectures for development of remote laboratory have been reported. Most laboratories focus on the use of remote laboratories from a laptop/desktop computer. This is because, the desktops can provide a more vivid and desirable alternative environment to the

users. However, it still reduces mobility, since the users have to carry around their laptops to be able to access the laboratories. An alternative to the desktop environment is the mobile devices such as smart phones, small tablets and cell phones. With the rise in use of such devices, several vendors such as adobe and Java have launched suitable platforms for development of applications on them. The platforms are unique and can provide almost the same experience as in the laptop/desktops.

In this paper, several platforms or methods are discussed that can be used to develop the laboratory environment in a mobile device. Section 2 describes the remote laboratory basic concepts. In Section 3, the modified architecture for remote laboratory from mobile devices is presented. In section 4, 5 and 6 the client is described and the different platforms used are compared. Section 7 presents a result and analysis of the system.

II. REMOTE LABORATORY FUNDAMENTALS

A remote laboratory is usually built using a complex architecture that helps eliminate any physical intervention required during an experiment. An experiment starts when a request is received with proper inputs. After that the corresponding Instruments are run and the output (data) is generated and the output is formatted into XML data such that it is understood by all interfaces. The output is then sent back to the users' client application which requests it. A remote laboratory architecture must ensure the following components (see Fig. 1):

- a. Scheduling: Since one instrument cannot be used by multiple users at the same time, i.e. one instrument cannot run multiple requests simultaneously, there is a scheduling scheme that maintains an order in which the users' requests are processed. The scheme should be such that all users get equal opportunities. Some typical methods are time slot, queuing and hybrid system [8].
- b. Automation: Experiments are carried on or with particular components. The instrument setting and components may be altered to perform the experiments. Hence switching is required to change between component settings. There may be other

kinds of automation necessary for performing tasks that otherwise require human interaction [7].

- c. Performance evaluation: The student’s laboratory is built to provide education. Hence the users must be evaluated. This procedure involves data analysis and the students need to upload their laboratory report and appear in online viva voce [9].

III. MOBILE LABORATORY ARCHITECTURE

The mobile devices laboratory architecture is based on the same system for any desktop system using a browser. The same set of web services are called for performing the experiment. Only the client side interface is changed to accommodate the needs of a mobile device. The components of the system are:

A. The Main NETLab Server

The NETLab Server [7] takes care of managing the user credentials and process the User Request for an experiment. It provides a scheduling scheme as multiple users cannot use the instrument simultaneously and Service Management connects the different instruments servers and collects data. It gets involved in measuring data from the equipment, control the different functions of a device, control the power control unit and collect the output data after an experiment request is successfully processed. It also maintains the user manuals that contain detailed information about the experiment, videos and images, equipment detail, interactive animation, question box and video chatting that would provide ways to communicate with the instructor and also other students. Thus the interface design for mobile devices needs special attention and is different from the usual interface for PCs. The client or the user interface is described later.

B. The Local Instrument Servers

The local servers are situated at the site of the Instruments and responsible for actual control of the experimental parameters. The Local server receives requests from the NETLab Server as web services, runs the instruments and returns the data in XML format.

C. Instrument Power Control Unit

Equipment is very expensive and cannot be kept on for long duration of time. The server powers the instruments through Web Services as and when required.

D. Intermediate Server for interpreting input (Optional)

For certain cases, a separate module for interpreting input from the users request is required. This is needed particularly in case of SMS based control. The module may be implemented in the main NETLab Server or on a separate server and is optional. But either way it requires a GSM modem to send and receive SMS to and from the user.

E. Web Services

A web service is a block of code that resides on the server and can be remotely invoked by the client. The client invokes s the web service by calling an HTTP link.

F. Security

Security of the system is mainly concerned with the safety of the instrument. As the instruments are very expensive, validation of inputs is performed to check whether the input may cause any damage to the instrument or not. This is done by checking the given parameters against all limits for any parameter. Data security is not important because instruments generate huge amount of numeric data for laboratory students which are not sensitive.

There are two ways to communicate to the server:

- Connection oriented services: In this case the

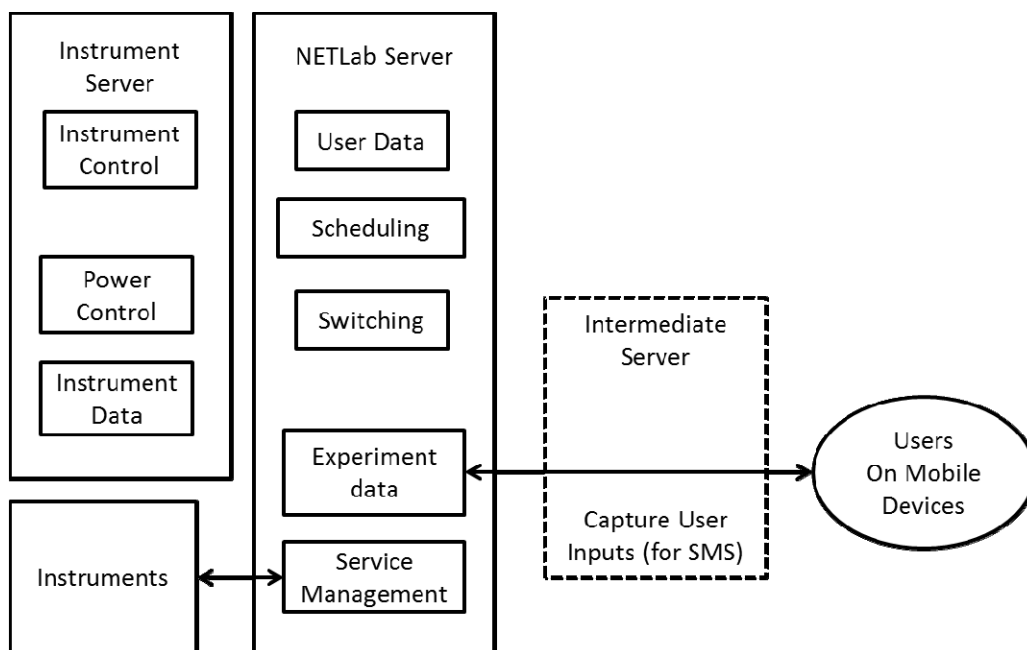


Figure 1. Remote Laboratory Architecture

client, usually a browser or a flash program, calls a web service and get the reply as data. The server keeps listening on a specified port (HTTP 80) for any incoming requests. A RESTful or SOAP web service performs the experiment using signal generation and acquisition operations and a response in XML format is sent back and the client converts the data and displays it to the user.

- Connection less services: In this case, a message such as a SMS is sent to the server and upon receiving the request, the server proceeds with the experiment run and storing the results. The results are then sent back as a message to the users mobile.

IV. THE CLIENT

The client side of the application is supposed to perform the following functions and characteristics:

- a) Logging into the systems:
This interface takes user credentials from the user. The server side NETLab consists of java servlets and java server pages with oracle as database. If the login is valid the next phase will begin. After that the user may book the time for an experiment which is then stored in the data base. Additionally, an 'immediate access' facility is present to allot the current time slot to the user, if it is free.
- b) Time slot booking for the experiment (if necessary): This step is necessary if a time slot scheme is followed. The student chooses a date and time and books an available time slot.
- c) Run the experiment: The interface must receive proper input parameters from the user regarding the experiment. Validation is necessary for correctness of input data to be transmitted to the server. However, the validation should be minimal since the client must be as thin as possible.
- d) View results: The results returned are always in XML format or in a picture format. The interface must plot graphs, charts or simply show the data.
- e) It must be a small program that can run fast
- f) It must fit within the small display area
- g) The mobile virtual laboratory client application needs to parse XML files.

The client interface can be built in different platforms. Each platform has different characteristics, advantages and disadvantages.

V. DIFFERENT PLATFORMS

A. Adobe Flash Lite

Adobe Flash Lite is an optimized light weight implementation of the Flash runtime for mobile devices [10]. Flash Lite 3.0 is based on Flash Player 8, supports Flash 8's newer ActionScript 3.0. According to Strategy Analytics, the number of Flash Lite enabled devices will reach more than 2.5 billion by the end of 2010. Thus mobile devices will provide greater flexibility in students' laboratory education in future.

The Flash Lite is designed for mobile devices and other portable devices that generates and displays multimedia content and applications developed with Adobe Flash tools to the users.

B. SMS and WAP

The SMS or short message service is a small piece of information typically n bytes that contains up to 160 characters in one single SMS. Every mobile device with a SIM card can send and receive SMS. It is a connection less method of sending information from mobile devices.

In order to run an experiment a SMS is sent to the server in a particular format. The format is dependent on the experiments. The SMS is received through a GSM modem and the server collects all the input parameters from the message and calls the web services with them. Upon receiving the results it is stored in the server (see Fig. 2 and 3).

Displaying the output to the user is the main area of concern since SMS cannot display graphs or charts. If a graph or chart is to be shown to the users, an image is generated from the data and stored on the server. The name is send and SMS to the users, who can view the image through the WAP browsers. The SMS and WAP features are wide available and very cheap [11].

C. HTML5

HTML5 is a new set of standards for the HTML hypertext markup language. With the new features added, such as the canvas and more efficient Scripting, the HTML5 is now a considerable option to present multimedia information to the remote laboratory users.

The HTML5 performs in a similar way as the Adobe Flash in data exchange and interface design. The HTML5 however is a script rather than being an executable program. Hence it requires a program usually a web browser to run the HTML5 scripts. The mobile device nowadays contains advanced browsers that support HTML5 specifications (see Fig. 4 and 5).

HTML5 is dynamic, meaning it can be used to create a wide variety of interfaces since it can contain a large number of different components. Also the new canvas element can be used to draw virtually anything from graphs to charts and thus it is perfectly able to depict the inputs and outputs of an experiment [12].

D. Java ME

Java Platform, Micro Edition, or Java ME, is a Java platform used for mobile devices. Like Adobe Flash Lite it is a scaled down version of Java SE that runs on such devices. Java ME supports a wider range of activity such as data storage, web services, GUI, advanced graphics etc.

According to surveys, there are more than 2 billion Java-enabled devices in use around the world. Hence it is widely used and can reach many users.

VI. COMPARISON OF THE PLATFORMS

Different platforms have their merits and demerits:

A. Efficiency or speed of execution on mobile device

Adobe flash uses vector graphics. This makes rendering new images and scenes a slower process. The HTML performs much faster since it follows the script

written. However, a very complex JavaScript code can easily bring down the performance speed. The Java ME is quick in generating images than both of the above because it uses scalar graphics. The SMS and WAP methods act differently, since there is no multimedia interaction from the user's side. The input must be given in text format only. The output is either in text format or in the form of an image that is simply downloaded and shown by the mobile phones softwares.

B. Flexibility to control different parameters

Every experiment is governed by a set of input parameters. The inputs are taken from the users and validated. The validation phase is required to ensure that the instruments are given proper input setting such that they are not damaged by invalid inputs. The validation phase mostly takes place in the server side, but may be required on the client side also. For this Flash, Java and HTML5 are equally capable of client side scripting. Also, a variety of controls ensures that the parameters can be given accurately. If any error occurs, it is immediately notified. For SMS and WAP, however, there is no client side checking. All inputs are parsed at the server side and if invalid a notification SMS is sent back.

C. Attractiveness or ability to design rich interface

All of the 3 platforms Flash, HTML5 and Java ME are capable of generating high end graphics and hence can be used for rich interface. The interface must be rich in the sense that it must replicate the real laboratory environment as much as possible. All 3 platforms support a wide range of component that can be used to design the interface to emulate the real instruments and component setting.

However, the SMS and WAP method lags far behind in this area since, the input and output are usually in text or image. Hence the interface cannot be made rich and thus are not suitable for educational purposes.

D. Scalability

It is the property of the platform that ensure scalability i.e., whether the platform is capable of implementing a wide range of experiments. Experiments are different and may vary from one another based on the type and amount of input and output, interactivity and processing time. For each case, the interface must deploy different strategies of scheduling, switching etc. Hence, the platforms used should be flexible enough for higher scalability.

The Adobe Flash, Java ME and HTML are all very scalable due to their high flexibility and ability for designing rich interactive interfaces. Also, complex algorithm can be implemented to deploy the scheduling or switching mechanisms.

As for the SMS and WAP method, large input experiments cannot be run using this method. In this case, the experiments are triggered with a limited number of parameters, as much as possible, to be put in an SMS.

E. Support

Adobe Flash is from Adobe, HTML5 is under w3c and Java is from Oracle Inc. SMS and WAP are not supported

by any organization, but they are standardized in the market. Hence, there is a good support base for all the platforms. The support is necessary because, if any bug or shortcomings are found in the platform then they must be immediately addressed. Otherwise, the developer has to change the entire system. The platform if supported by an organization will receive regular update and thus be very reliable to use.

F. Development and Operational Cost and Expertise

Development costs include the cost to build and deploy the system interface. For high end softwares like Adobe Flash and Java ME, considerable effort and time are required to develop high end interfaces. Hence, they are likely to be of high cost. HTML5 on the other hand relies on the more common HTML tags which are relatively easy to learn and use. Hence the HTML5 can be cheaper in development and implementation. Operational cost is the cost required to run the server and the instruments. This cost is same for all the technologies, since all the platforms run some code or script on the server and run the instrument for an experiment.

For SMS, however, an additional server and/or GMS modem must be maintained for interpreting the SMS for getting the inputs. Hence the operating cost increases a little bit.

G. Usage Cost

All technologies like Adobe Flash, HTML and Java ME uses internet to connect to the laboratory. Hence the usage cost is basically the cost of the internet in the mobile system and how much each platform uses the internet. Adobe Flash and Java ME are less users of the internet since they call the web services and only a few KB of data is actually transferred. HTML5, however, transfers much more data since each web page may be downloaded multiple times. Even if AJAX is used the overall data transfer is always higher. Hence the cost of using is lower in Adobe flash and Java ME compared to HTML5.

The SMS and WAP method work differently. Sending an SMS is very cheap and almost free with most mobile communication vendors. However, viewing the results may be more costly if the output is downloaded as an image instead of receiving an SMS in response.

H. Portability

Technologies like Adobe Flash and Java ME are dependents on specific environments, developed by their distributing organizations. The Flash id run only on Flash enable devices, and Java ME runs on Java enabled devices. These interfaces cannot operate outside their environments. This is a big drawback for the Flash and Java ME. HTML5 on the other hand can run on any mobile devices that support web browsers. Still, it requires the browser to support HTML5.

The SMS and WAP methods are however very portable since these can be sent and received on any mobile devices without the requirement of any special software. WAP is also standardized and is available on all phones.

I. Speed of Connectivity

The connectivity is dependent on the architecture of the system. Since the Flash and Java ME works by calling web services only, it reduces the amount of data transfer. Hence, the speed of getting the output is faster compared to HTML5. However, the speed difference is not significant.

The cases of SMS and WAP are different. In this case, the SMS is sent by the user and it is entirely dependent on the vendor when the SMS will reach the server, although it usually reaches within a minute. Upon receiving the SMS the server proceeds with the request. When the output is viewed, the WAP has to download the entire image, which is definitely slower. However, if the output is presented in an SMS, then the process becomes easier and usually faster.

J. Security

Each platform, Flash, Java and HTML5 provides its own safety measures and have been proven to be successful in commercial implementation. The SMS and WAP are secured because the communication takes place between two points via the telephone system.

provide learning contents. Hence, it must be of high educational value.

A. Educational Values:

Educational value is determined by learning outcomes which is directly dependent on the presentation of the contents. On mobile devices presentation of learning contents along with running the experiment is the major challenge due to both the space and processing power constraints. Hence, a suitable platform should be chosen that can produce rich interfaces. In this regard the SMS and WAP scheme fails miserably while the other 3 are almost at par with each other.

The HTML5 is obviously the best choice for creating learning environment since, the html standards are universal and easy to program compared to Adobe Flash or Java ME. HTML5 does not require any special platform to run and performs almost equally in speed and efficiency to both without compromising interface flexibility. This makes it run on most devices, which is a big advantage.

To simply trigger, control or monitor, the SMS is the best method since it is cheap. It can be used where the users have technical expertise.



Figure 2. SMS experiment format request on Nokia 2630.

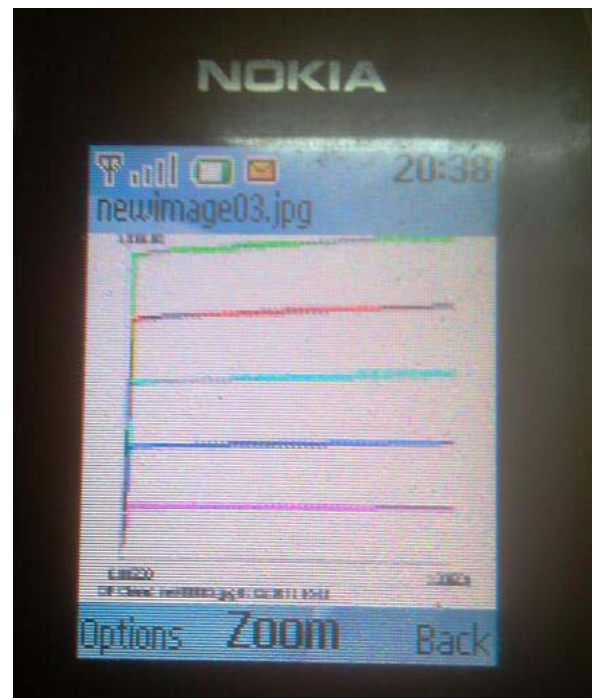


Figure 3. Output as viewed on WAP in Nokia 2630.

VII. EFFECTIVENESS OF THE PLATFORMS

The system of using mobile devices to access remotely triggered laboratory which allows a greater mobility and flexibility than using a Desktop PC. The major aim is to

TABLE I.
SUMMARIZED COMPARISON OF THE DIFFERENT PLATFORMS

| Characteristics | Adobe Flex | HTML5 | SMS and WAP | Java ME |
|---|--|--|--|--|
| <i>Efficiency or speed of execution on mobile device</i> | Not good due to vector graphics | Good, since it is interpreted by the browser | The processing depends on the reliability of client-server connection | Good, due to emphasis on scalar graphics |
| <i>Flexibility to control different parameters</i> | High | High | Medium | High |
| <i>Attractiveness or ability to design rich interface</i> | Very High, Graphics generated is of high quality | High | Very poor, since all message are sent in text and received in either text or images only | High |
| <i>Scalability</i> | Very high | High | Poor | Very High |
| <i>Support</i> | Yes, Adobe | Yes, W3C | No, but SMS and WAP are standardized | Yes , Java |
| <i>Development Cost and Expertise</i> | High | Low | Low | Medium |
| <i>Usage Cost</i> | High | Medium | Very low | High |
| <i>Speed of Connectivity</i> | High | Medium | Unstable | High |
| <i>Portability</i> | Not portable, only usable on devices with Flash Lite enabled | Highly portable, usable on any browser | Highly portable, SMS and WAP or browser is available on almost all mobile devices | Not portable, only usable on devices with Java enabled |

B. Case Study: Microelectronics and VLSI Engineering Laboratory

The students use their mobile phones to access the website. The home page of the Microelectronics and VLSI Engineering laboratory as seen in the mobile device (HTC HD7) is shown in Fig. 2. The webpage looks differently when accessed through the mobile devices compared to the one viewed on a PC. This can be attributed to small screen and other limitations in the mobile devices. Hence, to accommodate the information is kept the same, only changing the arrangement, size and orientation of the components.

In this case study, the BJT output characteristics experiment is discussed. To perform the BJT experiment, the voltage across VCE is changed and I_c is measured, while I_C -VCE is plotted.

The experiment has the following components: a laboratory manual, description of the experiment, experimental procedure etc. The laboratory management system applies a slot-based time scheduling scheme for this experiment. A user may book the time slot by selecting a particular date and time. However, the other components such as experiment manuals etc. are available all the time.

As a first the student books a time slot for performing the experiment and log on during booked time period. The hybrid time scheduling scheme is also used in the NETLab. The client server checks the user credentials and checks whether, the user is allowed to perform experiment or not. If the user is permitted, the user can enter parameters and initiate the experiment. The RGraph library is used to generate HTML5 code for generating graphs. CSS and AJAX have been used to create an

accurate operating environment. After the measurements are made, the data are transferred by eth web services. The html pages also displays users information and the time left during time slot. Once the time is over, a JavaScript function displays a warning and then closes the window. As the time slot or the duration of experiment is over, the link is broken with a warning. The user has limited number of chances to book and perform the experiment. This is done to optimize the use of the resources.

VIII. CONCLUSIONS

Different platforms for remote laboratory deployment are discussed. As the students are increasingly using mobile devices, the availability of remote laboratories on mobile devices will enhance the educational opportunities as the contents are now brought closer to them. With mobile devices, the users get more flexibility and mobility. As the standards of learning experience may not be compromised, a suitable platform such as Adobe Flash, HTML5 or Java ME must be chosen to create the contents as accurately as possible, so that students are not only able to perform experiments but also understand them.

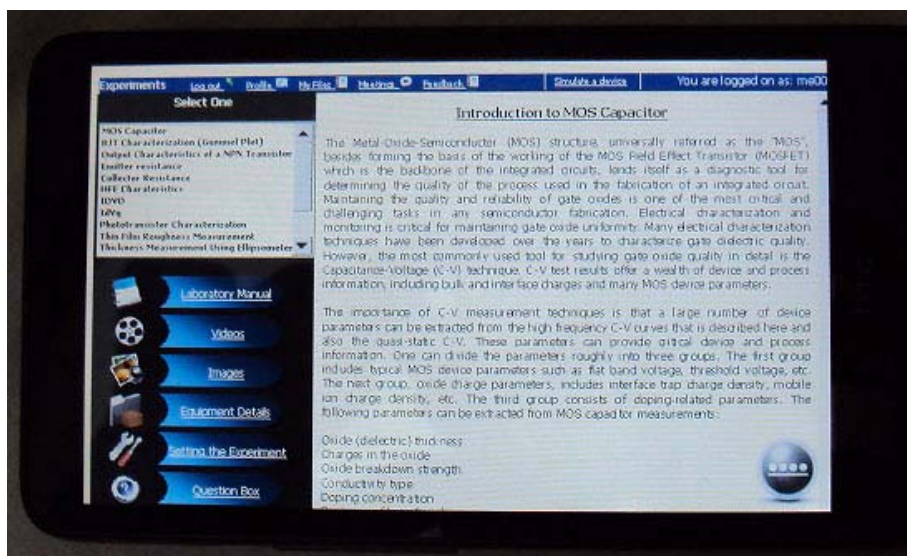


Figure 4. Manual page for the experiment in HTC HD.

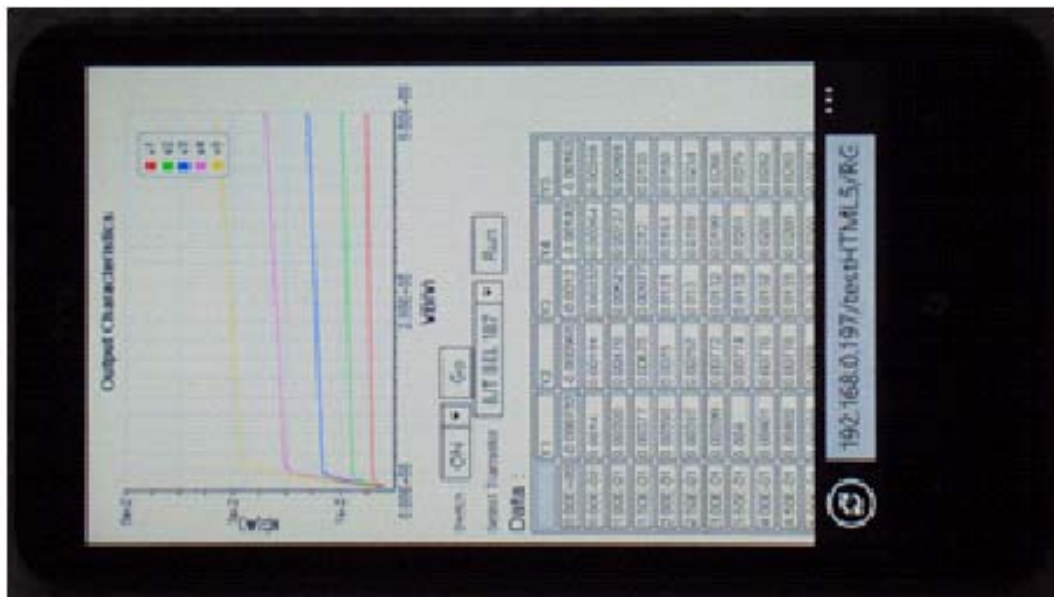


Figure 5. BJT output characteristics with HTML5 in a browser in HTC HD7.

REFERENCES

- [1] N. Ertugrul, Towards virtual laboratories: a survey of LabView based teaching/learning tools and future trends. *Int. Journal of Engineering Education*, 16:3, 2000.
- [2] Colton, M. Q. Knight, R.-A. Khan, and R. West, A web-accessible heat exchanger experiment, in *Proc. World Innov. Eng. Educ. Res. (INNOVATIONS 2004)*, W. Aung, R. Altenkirch, T. Cermak, R. W. King, and L. M. S. Ruiz, Eds. Arlington, VA: iNEER, pp. 93–106, 2004.
- [3] Amaratunga and R. Sudarshan, A virtual laboratory for real-time monitoring of civil engineering infrastructure, presented at the *Int. Conf. Eng. Educ. 2002*, Manchester, U.K., Aug. 18–22, 2002.
- [4] V. Ramakrishnan, Y. Zhuang, S. Y. Hu, J. P. Chen, and K. C. Tan, Development of a Web-Based Control Experiment for a Coupled Tank Apparatus, In: *Proceedings of the American Control Conference*, Chicago, 2000.
- [5] Schmid and A. Ali, A Web-Based System for Control Engineering Education, In: *Proceedings of the American Control Conference*, Chicago, vol. 5, pp. 3463-3467, 2000.
- [6] M. Auer, A. Pester, D. Ursutiu, and C. Samoila, Distributed Virtual and Remote Labs In Engineering, In: *Proceedings of the IEEE International Conference on Industrial Technology (ICIT)*, Maribor, Slovenia, vol. 2, pp. 1208-1213, 2003.
- [7] A. Maiti, NETLab: An Online Laboratory Management System, *iJOE*, vol. 6, pp. 31-36, 2010.
- [8] A. Maiti, Time Scheduling Schemes for Online Laboratory Management Systems, *iJOE*, vol. 6, Iss. 4, pp. 41-45, 2010.
- [9] A. Maiti, Automatic Evaluation of Student's Performance in Online Laboratories, *iJOE*, vol. 6, Iss. 3 pp. 44-49, 2010.
- [10] A. Maiti and A. Maiti, ADOBE Flash Lite–Based Online Laboratory for Mobile Phones *iJIM*, vol. 4, Iss. 4, pp. 45-48, 2010.
- [11] A. Maiti and B. K. Tripathy, Applications of Short Message Services in Remotely Triggered Laboratories, *iJOE*, vol. 7, Iss. 4, pp., 2011.
- [12] A. Maiti, A. Maiti, and C. K. Maiti, HTML5 Based Remote Laboratory Platform for Mobile Devices, accepted in *International Conference on Engineering Education*, Turku, Finland, 2012.

AUTHORS

Ananda Maiti is with the School of Computing Science and Engineering, Vellore Institute of Technology, Vellore 632 014, India.

B. K. Tripathy is a senior professor at the School of Computing Science and Engineering, Vellore Institute of Technology, Vellore 632 014, India.