

Development of a Distant Laboratory using LabVIEW*

K. K. TAN, T. H. LEE and F. M. LEU

Department of Electrical Engineering, National University of Singapore, 10, Kent Ridge Crescent, Singapore 119260. E-mail: kktan@nus.edu.sg

This paper presents the development of a distant laboratory experiment for engineering students at the National University of Singapore. The experiment is concerned with real-time modeling and control of a pilot-scale DC servo motor. The World-Wide-Web (WWW), with its ability to deliver multimedia and graphics across the world, is used as the medium to link the student to a remote experimental setup located in the distant laboratory. This paper describes the hardware and software constructs which facilitate students to perform the experiment remotely and efficiently, using only a commonly available, user-friendly web browser.

INTRODUCTION

MASS PRODUCTION of books has made education available to many, and provided the fundamental resources for the system of mass education today. At the dawn of the information technology (IT) age, available infrastructure and technologies are well poised to bring education to even more by allowing individuals to carry more responsibility for their own learning and achieve independence of thought and action by means of distant education.

In the present era, distant education is becoming important and necessary for several reasons. With the advent in IT, the exponential rate in the accumulation of information and rapid changes in information and technology necessitates an ongoing education and professional development. Distant education, with the tremendous potential in terms of time and space flexibility, has much to offer in this respect. This is particularly true for the engineering profession where rapid advancement in technology and innovation necessitates more efficient and flexible means of accumulating latest know-how and updated information. This is also probably a reason for the changing student demographics as life-long continuous learning has become the sole means necessary for engineers to remain relevant and competitive. A large percentage of university students, especially those pursuing higher degrees, are enrolled on a part-time basis with full family and work commitments. Convenience and flexibility in the curriculum in a continuing education system is thus of utmost importance.

Distant education has been through several stages since its inception in the mid to late 1800s [1]. Development of inexpensive and faster postal services in Europe and across the United States

gave birth then to the correspondence or home study form of education. Readings, study guides and other print materials are sent to students by mail. Students gain credit by completing specific assignments. This is the first generation of distant education. In the late 1960s and early 1970s, the second generation of distant education surfaced through the use of recorded media such as radios, televisions, audiotapes and telephones. These new ways of communication complement the study guides and local library resources available.

Distant education, in its present generation, is turning towards the powerful integration of computers and telecommunications to bring on real time interaction. The Internet is the world's largest, most powerful computer network connecting personal computers, sophisticated mainframes, and supercomputers around the globe. The World-Wide-Web (WWW), as the fastest growing Internet service, is seen as an effective tool for distant learning. The WWW project, started by Tim Berners-Lee while at CERN (the European Laboratory for Particle Physics) [2], seeks to build a 'distributed hypermedia system'. The WWW provides Internet users with a uniform and convenient means of accessing the wide variety of resources (pictures, text, data, sound, video) available on the Internet. Popular software interfaces, such as Internet Explorer (IE) and Netscape, facilitate navigation and use of the WWW. In effect, the web is a distributed multimedia environment, which potentially can provide a truly integrated teaching and learning environment.

However, as it is now, the WWW as an educational media is still far from perfect with certain pressing limitations. The most serious one associated with using the WWW as a media of distant learning is the amount of bandwidth consumed by multimedia files. The lack of broadband network infrastructure cripples the usefulness of WWW as a

* Accepted 9 September 1999.

true multimedia learning environment. Computer literacy and availability of network connections are additional factors to consider when designing a distant education system based on the WWW. Furthermore, a new distant education is not just an old class with new technology. Effective design and delivery requires considerable and synergetic effort by a team of people of different backgrounds and with different expertise—instructors, course designers, technicians, administrators, and students. Many of these hindering factors are becoming less important with time, due to widespread proliferation of computer and communication systems and the knowledge coming along with them. It is envisaged that in time to come, WWW-based educational systems will become a more important part of many of our distant learning programs.

Most distant learning facilities, based on the Internet, developed so far are focused on on-line lectures and tutorials [3]. Most of these can be presented as text-based materials and sent across the Internet easily. Apart from lectures and tutorials, laboratory sessions and hands-on practice are an indispensable part of any engineering education. However, the difficulty of manipulating laboratory apparatus over the Internet limits the development of on-line laboratory sessions. This paper explored the wider territories of executing a laboratory session via the user-friendly web browser environment. To this end, it is the desired objective that the user is able to control laboratory equipment at a remote location and recording real-time experimental data via the web browser.

The work reported in this paper is mainly concerned with the setup of a systematic control experiment on a pilot-scale DC servo motor system, although the approach may be easily extended to many other plants as well. Through a web browser, the user will have to carry out a series of procedures to extract important characteristics of the physical system and tune a PID

controller for motion control of the motor. The web address for the laboratory experiment is at <http://wwwlab.ee.nus.edu.sg>. Since a password is needed, users will need to be registered with the web administrator to gain access.

HARDWARE AND SOFTWARE

The hardware necessary and used in the development of the distant laboratory at the server's end, consists of a National Instrument's data acquisition card PCI-MIO-16E-4 installed onto a Pentium II 350 MHz personal computer. The acquisition card supplies two channels of analog output and up to 8 channels of analog input configured to non-referenced single-ended (NRSE) mode. The personal computer was used primarily for data acquisition and control of the servomotor. The hardware connection is depicted in Fig. 1.

The DC servomotor used is the LJ MS15 DC motor control module from L. J. Electronics. The module consists of a DC motor that is capable of being driven at speeds of up to 2500 rpm in either direction. A second DC motor driven directly by the first provides an analog voltage feedback proportional to the speed and direction of rotation. A continuous rotation potentiometer provides an analog output proportional to the angular position of the output shaft. A photograph and a schematic of the motor is given in Fig. 2.

LabVIEW with its intuitive graphical interface provides the development platform for the control programs (the virtual instruments, VIs) for the laboratory session. These VIs cannot be accessed directly across the Internet and viewed directly by a web browser. The main idea behind the implementation of the distant laboratory session is to deliver the image of the control panel for the VIs to the client. On the client's end, it allows the client to control the experimental apparatus through the web browser via the CGI interface. Essentially,

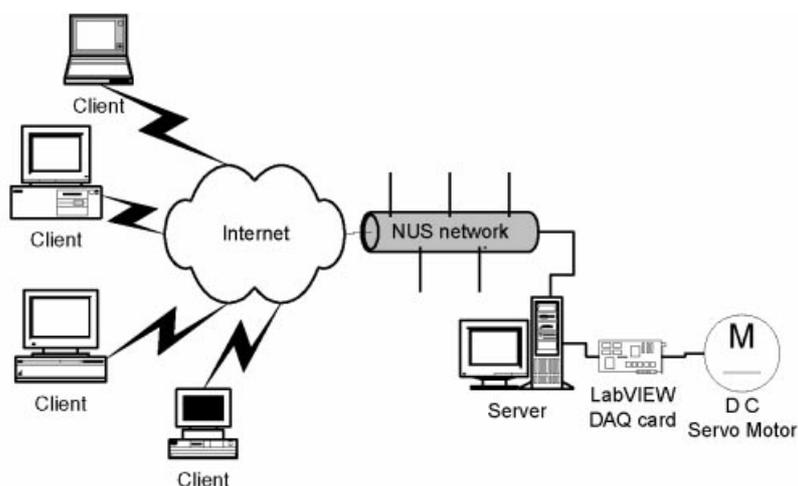


Fig. 1. Overview of the hardware connections.

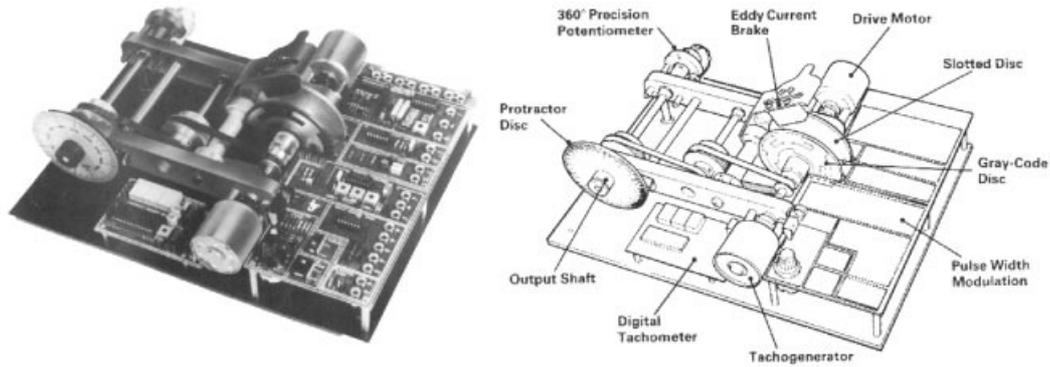


Fig. 2. Photograph and a schematic of the DC servo motor.

the distant laboratory application, as with other network applications, consists of mainly two parts: the client side and the server side. The client side is a simple web browser. At the server side, LabVIEW 5.0 supplemented with the Internet Developers Toolkit package are the only essential software needed. The software requirements and interaction is depicted in Fig. 3.

OPERATIONAL PRINCIPLES

The server—delivering front panel image

The Internet Developers Toolkit package is an add-on component to LabVIEW. It is essentially a collection of libraries for converting VIs into Internet-enabled applications, such as electronic mails and FTP [4]. First and foremost for the WWW, the toolkit consists of the G-Web Server as shown in Fig. 4. It is able to serve up the front panels of the VI as a picture to be viewed from the client's web browser. There are basically two modes of service. The first is the 'snapshot' mode where a snapshot of the front panel image will be

served. The second mode is the 'monitor' mode. In this mode, the effect served to the user is an embedded animation of the panel image. This is achieved with Server-Push technology. Successive 'frames' of the panel's image will be pushed from the server to the client one after another without waiting for subsequent requests from the client. The rate at which the frames are pushed over can be set at the G-Web Server configuration file. Presently, only Netscape's browser supports Server-Push technology.

With other web browsers, particularly Microsoft's Internet Explorer, it is also possible to create pseudo animation. This can be done by adding the following code to the header section of the HTML document.

```
<META HTTP-EQUIV = 'REFRESH' CONTENT = '10' >
```

The effect of this code is that the client's browser will reload the page every 10 seconds. However, the visual effect is not as good, and hence it is not supported in this development.

On the client side, the front panel image of the VI is referenced by the tag on the HTML

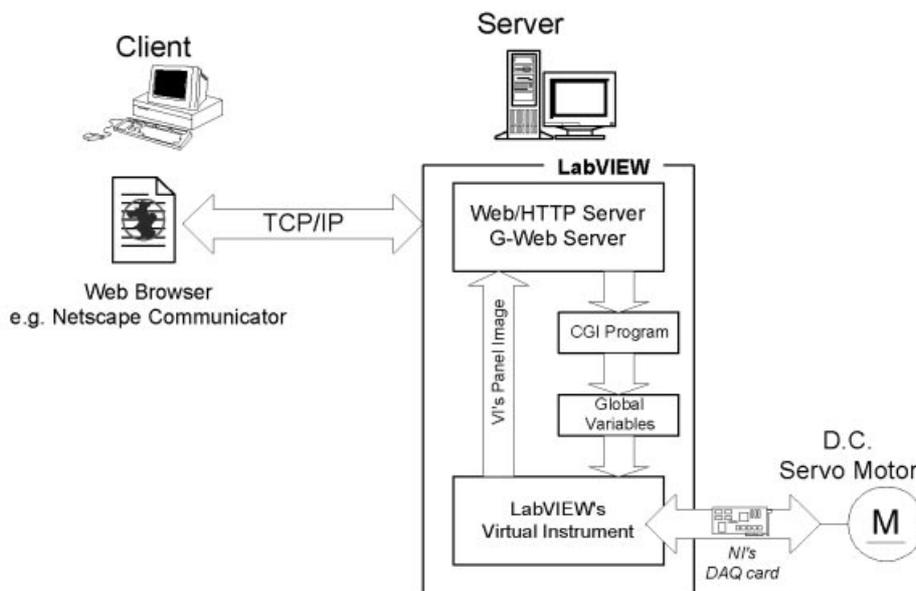


Fig. 3. Overall software interaction.



Fig. 4. G-Web Server.

document. The following shows the code to create an embedded animation on the HTML document.

```
<IMG SRC='.monitor?step1.vi' WIDTH='462'
  HEIGHT='384' ALT='step1.gif
  (24765 bytes)' BORDER='0'
  USEMAP='#step1'>
```

The source attribute (*src*) of the image tag points to '.monitor?step1.vi'. This will cause the G-Web Server to serve up the front panel image in the 'monitor' mode, animating the image to the client. Thus the G-Web Server provides a convenient means of delivering the VI's front panel image across the Internet.

Common gateway interface (CGI)

The package also includes a library for writing CGI (common gateway interface) programs for use with the G-Web Server. The CGI is the standard for interfacing external applications with information servers, such as HTTP or Web servers [5]. Whenever the user requests, for example by clicking on the hyperlink or image map, the URL (Uniform Resource Locator) corresponding to the CGI program, the CGI program will be executed in real-time on the server. Figure 5 shows a typical CGI program shell written with LabVIEW. CGI programs are able to perform virtually any task on the server. In this development, the CGI programs are used to load and unload the VIs and to set the control

parameters. In this way, the user will be able to gain control over the physical DC motor through the web browser.

There are two main ways of passing parameters to the CGI program, through the *PATH_INFO* or *QUERY_STRING* environmental variables. In this application, parameters are sent to the CGI programs via the *QUERY_STRING* environmental variables. *QUERY_STRING* is any character that follows the first '?' character in the URL. This string is encoded in the standard URL format of changing spaces to +, and encoding special characters with %xx hexadecimal encoding (in accordance with RFC 1738). Internet Developers Toolkit provides a collection of functions for getting *QUERY_STRING* from the environmental variables and decoding the encoded string. These functions are useful for developing CGI programs at the server side.

User interface

On the client side, there are two interfaces where the user may send commands over the Internet to control the laboratory apparatus. The first interface is the standard HTML form and the second, is through an image map. Figure 6 shows the HTML form as it appeared to the user. The user only needs to key in the desired control value, for example 2.1 V, and submit the form by clicking the 'OK' button. The CGI program as specified by

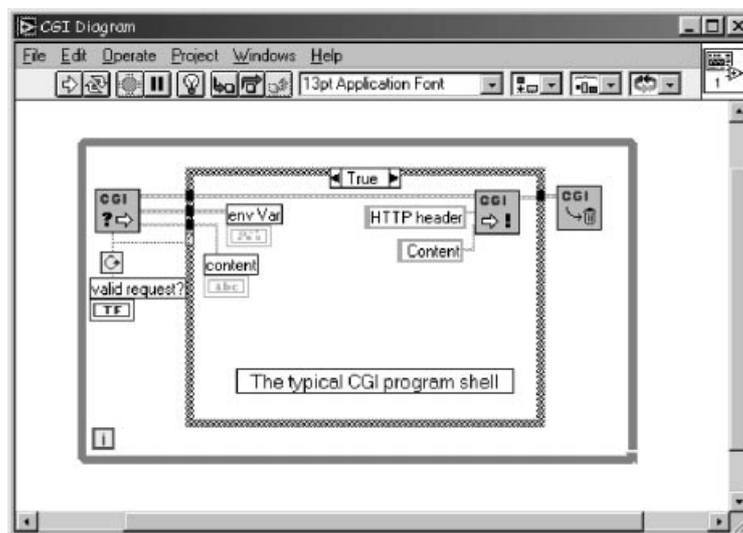


Fig. 5. A typical CGI program shell in LabVIEW.

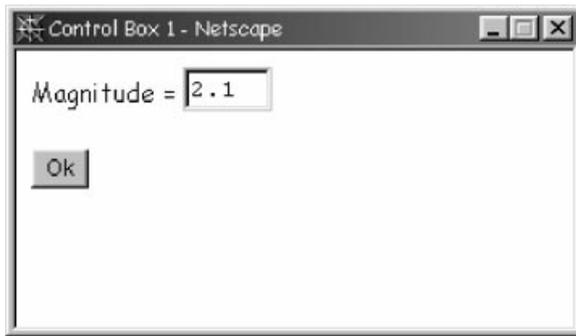


Fig. 6. The HTML form.

the 'ACTION' attribute (see the HTML code that follows) of the form will be executed at the server.

With the form interface, JavaScript is used at the client side to validate the user input. This ensures that invalid user input will not reach the CGI programs residing on the server. In addition, since the JavaScript is executed at the client side, the user will be notified almost instantaneously of the invalid input. The following shows the HTML code for the form. The checkMag() is a JavaScript function that validates the user's input.

```
<FORM NAME='submitForm' ACTION='/cgi-bin/
lab/vary1.vi' METHOD='GET' onSubmit=
'return checkMag(mag.value)'\>
<p>Magnitude = <input TYPE='text' SIZE='5'
MAXLENGTH='5' NAME='mag'\></p>
<p><INPUT TYPE='submit' NAME='submitButton'
VALUE='Ok'\></p>
</FORM>
```

In the second type of interface, the front panel image displayed on the client's browser is actually an image map, with hotspots defined strategically, as shown in Fig. 7. The picture on the left shows the front panel of the VI overlaid with the image map and the one on the right show the actual map only. Each hotspot on the image map has an associated HREF attribute whose value is the URL pointing to the respective CGI programs residing at the server. Thus the user may control

the lab apparatus by clicking on the image, thereby giving the user the virtual feel of directly controlling an instrument.

The following code shows the HTML code for the image map for Fig. 7.

```
<MAP NAME='step1' >
<AREA SHAPE='CIRCLE' COORDS='86, 96, 12'
HREF='/cgi-bin/lab/vary1.vi?mag=5'
onMouseOver='pointOn('mag5'); return
true' >
<AREA SHAPE='CIRCLE' COORDS='96, 64, 12'
HREF='/cgi-bin/lab/vary1.vi?mag=4'
onMouseOver='pointOn('mag4');
return true' >
<AREA SHAPE='CIRCLE' COORDS='78, 36, 12'
HREF='/cgi-bin/lab/vary1.vi?mag=3'
onMouseOver='pointOn('mag3');
return true' >
<AREA SHAPE='CIRCLE' COORDS='43, 36, 12'
HREF='/cgi-bin/lab/vary1.vi?mag=2'
onMouseOver='pointOn('mag2');
return true' >
<AREA SHAPE='CIRCLE' COORDS='23, 65, 12'
HREF='/cgi-bin/lab/vary1.vi?mag=1'
onMouseOver='pointOn('mag1');return
true' >
<AREA SHAPE='CIRCLE' COORDS='33, 97, 12'
HREF='/cgi-bin/lab/vary1.vi?mag=0'
onMouseOver='pointOn('mag0');return
true' >
<AREA SHAPE='RECT' COORDS='251, 36, 298, 69'
HREF='/cgi-bin/lab/vary1.vi?power=on'
onMouseOver='pointOn('powerOn');return
true' >
<AREA SHAPE='RECT' COORDS='251, 70, 298,
104' HREF='/cgi-bin/lab/vary1.vi?power=
off'
onMouseOver='pointOn('powerOff ');return
true' >
<AREA SHAPE='RECT' COORDS='152, 63, 195,
109' HREF='/cgi-bin/lab/vary1.vi?pause=
true'
onMouseOver='pointOn('pause');return
true' >
```

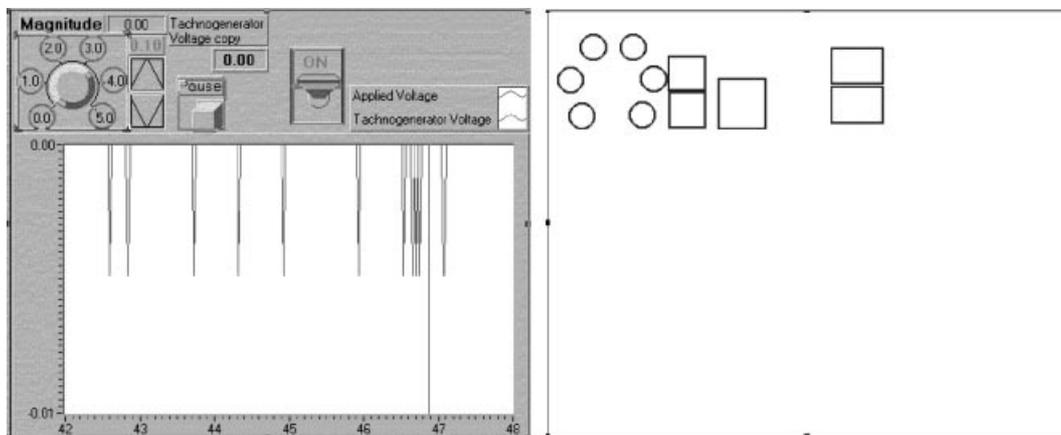


Fig. 7. Illustration of the image map displayed on the web browser.

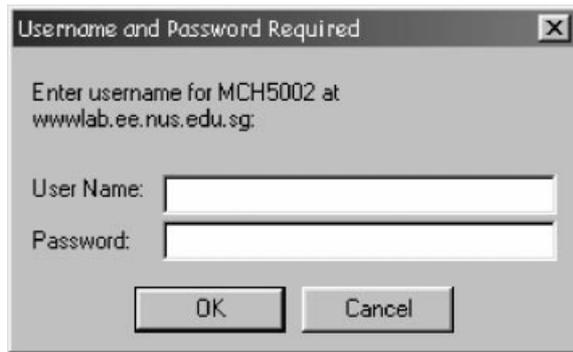


Fig. 8. The authentication dialog box of the Basic Authentication Scheme.

```

<AREA SHAPE='RECT' COORDS='109, 75, 142,
  108' HREF='/cgi-bin/lab/vary1.vi?mag=
  down'
onMouseOver='pointOn('magdn');return
  true' )
<AREA SHAPE='RECT' COORDS='108, 43, 142, 75'
  HREF='/cgi-bin/lab/vary1.vi?mag=up'
onMouseOver='pointOn('magup');return
  true' )
</MAP>

```

Security

Access security is incorporated into the G-Web Server. It is achieved via the Basic Access Authentication scheme as specified in HTTP/1.0 [6]. It is a simple challenge-response authentication mechanism that is used by a server to challenge a client request and by a client to provide authentication information. It is based on the model that the client must authenticate itself with a user-ID and a password for each realm. The server will service

the request only if it can validate the user-ID and password for the protection space of the Request-URL. With the G-Web Server, it is also possible to control access based on user name, password and user's IP address. Figure 8 shows the authentication dialog box.

Operating sequences

A typical session begins from the browser at the client end. The user will need to authenticate to the server. Upon successful authentication, the home page as shown in Fig. 9 will appear. The user may then initiate any stage of the experiment by clicking on any of the hyperlinks, Step1 to Step4. These hyperlinks actually refer to a CGI program. Whenever the server receives any request, this CGI program will verify that the power supply to the DC motor is enabled. Next, it will check for any ongoing session with the DC motor, since there is only one set connected to the server at the moment and therefore only one session may proceed at any point in time. Following these checks, the VI requested will be loaded into memory at the server. The front panel image will be served up and the experiment may begin.

Clicking of the image maps or submission of standard HTML forms will send commands from the client to the server. These commands will be sent to the server via the Internet with the TCP/IP set of protocol. At the server side, the G-Web Server will receive the client request through the designated socket and port number, usually 80 for HTTP. The server will process the client's request and serves up the front panel image to the client's browser. (A socket is one endpoint of a two-way communication link between two programs running on the network. A socket is bound to a

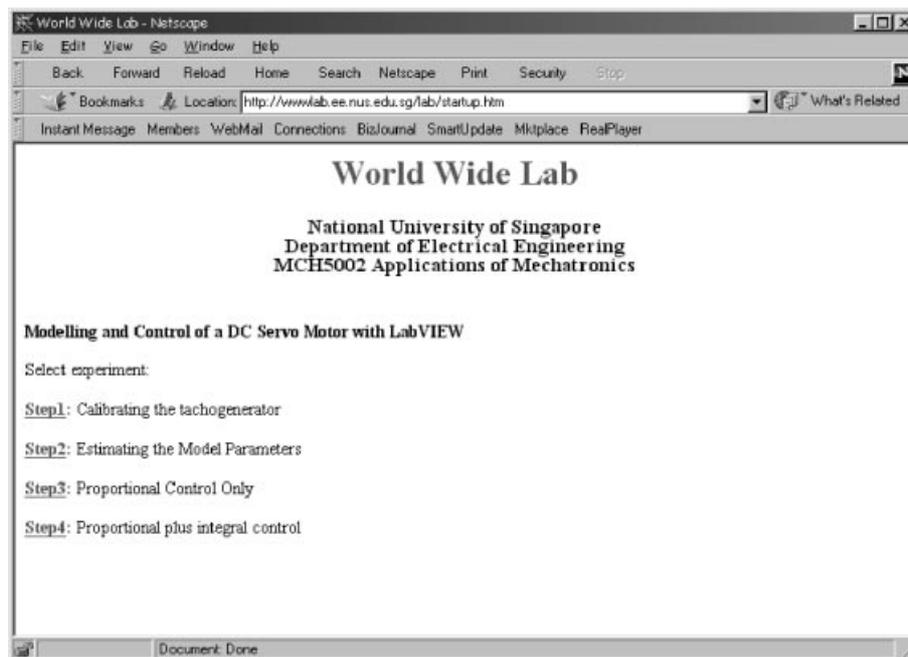


Fig. 9. The home page.

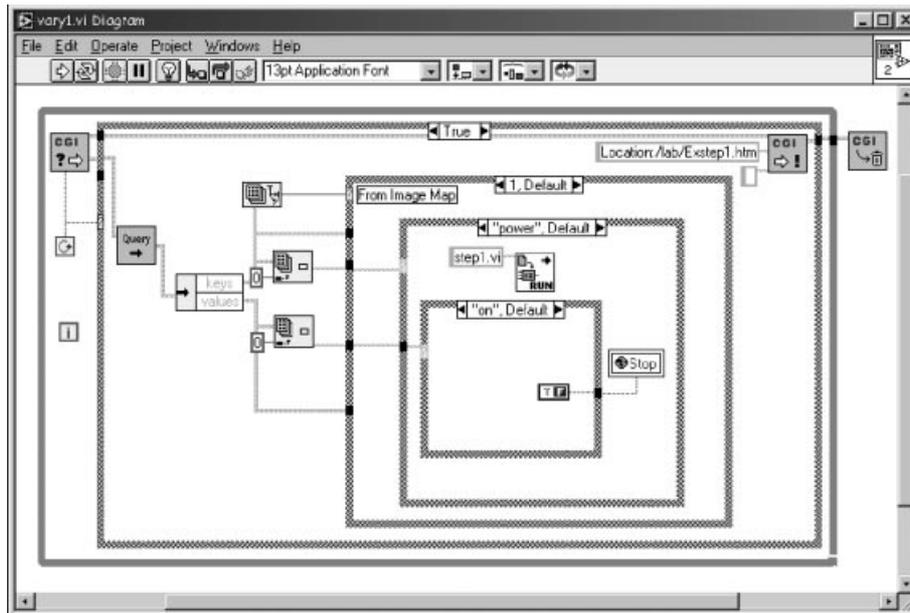


Fig. 10. A CGI program.

port number so that the TCP layer can identify the application that data is destined to be sent.)

When the user clicks on a hotspot on the image map associated with the HREF that points to `/cgi-bin/lab/vary1.vi?power=on`, the CGI program, `Vary1.vi`, that is going to be executed is shown in Fig. 10. The parameter `power=on` will be passed to the CGI program as `QUERY_STRING`. The CGI program will then decide the appropriate code to execute. In this case, it will change the global variable `'stop'` to false and run the VI `'step1.vi'`. The new value of the global variable will then turn on the VI that was loaded into memory. Thus the CGI programs in fact only changes the global variables. These global variables will then affect the VI that is controlling the DC motor. This is also shown graphically in Fig. 3.

THE EXPERIMENT

The experiment is a hands-on session on the application of computer-based control to a voltage-controllable electromechanical system—the DC motor. It is mainly concerned with the modeling and control of a pilot-scale DC servo motor system, fully instrumented with position and velocity measurements. The control software, or VI, is written in National Instrument's LabVIEW.

The main objective of the experiment is to introduce to the students:

- simple static and dynamic modeling of the DC motor system;
- manual and feedback control of the system for velocity set-point trajectory tracking;
- feedback control for position tracking.

The experiment consists of four major steps to be followed in sequence. In Step 1, the tachogenerator is calibrated. The motor speed as given by the tachogenerator output is the plant output and the applied voltage to the motor is the plant input. This part of the experiment enables the student to determine the important parameters of the system pertaining to the steady-state and dynamic characteristics. This is done by calibrating the relationship between the steady-state motor speed measured in volts by the tachogenerator and the input voltage.

In Step 2, the model parameters of the motor are estimated. To this end, a square wave bias with 1 V DC voltage is applied to the motor. The student observes a few sets of the step response by varying the magnitude of the square wave and estimates the plant parameters from the measurements logged. Next, the student performs open-loop manual control of the motor by loading the VI in Step 1. With the estimated plant parameters, the student estimates the necessary input voltage to track the setpoint both dynamically and at steady-state. The student shall observe the steady-state error associated with such manual control.

In Step 3, the angular velocity of the motor is controlled with a proportional (P) gain. The purpose is to examine the performance of a simple proportional feedback controller. The measured speed is compared with a desired setpoint. The applied voltage is proportional to the resulting error signal. The student shall note the relationship between the proportional gain to the steady-state error, the effects on transient response, stability of the system and relate these observations to theory.

In the last part of the experiment (Step 4), proportional plus integral (PI) control is used to control the velocity of the DC motor. From the previous stage, the student should have observed

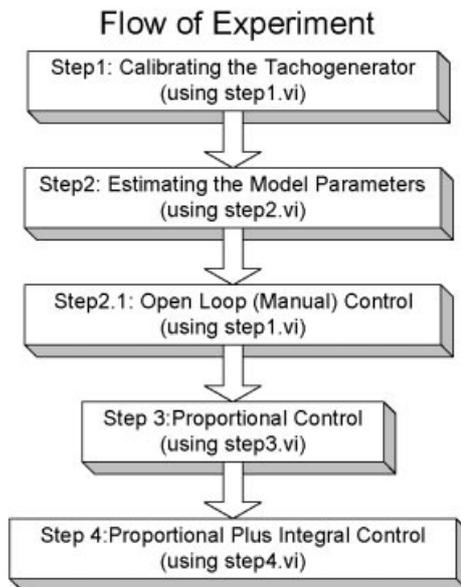


Fig. 11. Flow of experiment.

the steady-state error associated with manual or P control. Under PI control, the applied voltage is proportional to the sum of the error and its integral with respect to time. This means that the error signal will increase or decrease until the error is zero. Therefore the student is able to achieve zero steady-state error. The student is expected to

link the transient response to the reset time of the integral. Derivative gain is included in the VI for the more inquisitive student to achieve enhanced control performance. Figure 11 shows the flow of the experiment in block diagram form.

A snapshot of the browser executing the PI control is shown in Fig. 12. The student is able to tune the PID parameters and change the setpoint by clicking on the knobs. The square wave setpoint and the response are shown on the top graph while the control effort is depicted on the bottom graph. In this particular snapshot, the setpoint is 2 V and the PID parameters are set to 3, 0.1 and 0 respectively.

Additional considerations

The experiment offers a 3-hours hands-on session for a graduate course comprising of mainly part-time mechanical engineering students. It provides for the time and space factor flexibility necessary for these students in executing the exercises while still retaining an essentially real-time operational environment.

The main inherent difficulty, at the moment, is the connection speed. For a smooth animation of the front panel image, the connection speed must maintain at a rate of 5–10 kbps. Since TCP/IP is a 'best effort' network protocol, it does not guarantee any quality of service. Thus, the quality of animation may vary and the responsiveness of

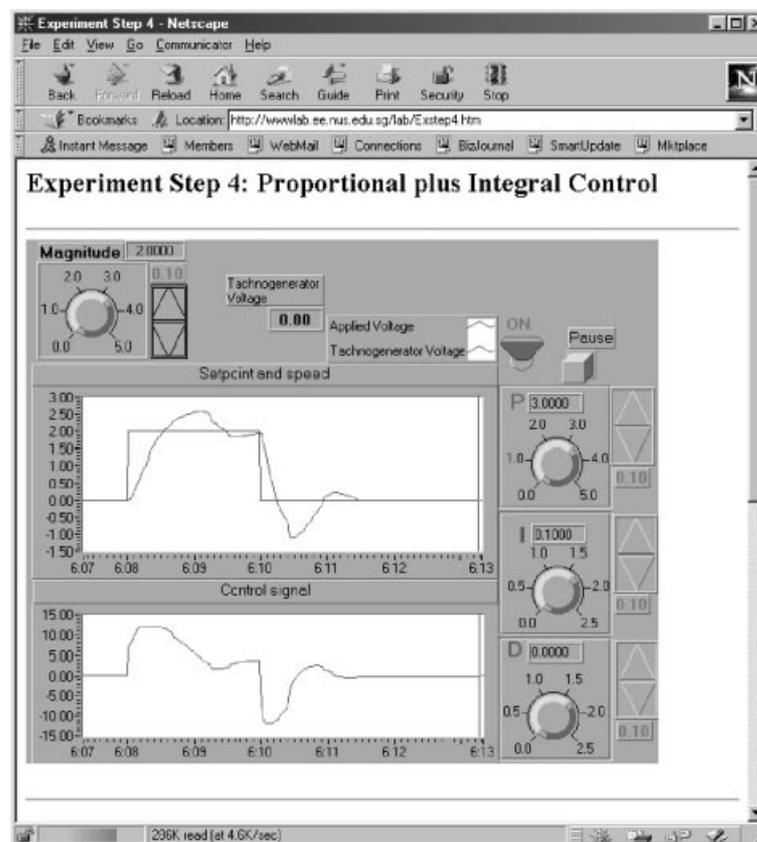


Fig. 12. A snapshot of the browser executing PI control.

the motor may not be as desirable in view of unpredictable network traffic.

The hardware cost of the setup consists mainly of the computer acting as the server, the DC motor, LabVIEW with Internet Toolkit and its data acquisition card, in total around US\$3000 for a single-station setup.

FUTURE TRENDS AND DEVELOPMENTS

Distant education via the Internet is poised to change the way educational institutions traditionally operate. With fast emerging developments in IT and the infrastructure in place, more institutions and organizations are expected to embark on distant education programs. There will be more cooperation as well as competition among institutions to promote and provide distant education. Thus, both students and instructors need to make adjustments to adapt to the new learning environment.

Education systems have generally accepted and are generally ready for the transition to one which is based on IT as the backbone. The main limitation in the use of IT in distant education has been mainly with the lack of infrastructure and computer literacy. However, activities in these areas have blossomed over the last decade and will continue across the next millennium. The technology for an interactive multimedia form of distant education is already in place. In Singapore, for example, with the advent of the IT2000 Masterplan, where a national broadband network would be available to every home and office, bandwidth of 5 to 10 megabits per seconds will be available to every common user. The core broadband network infrastructure is based on Asynchronous Transfer Mode (ATM) switching technology which will run at speeds of up to 622 Mbps. This network will be equipped with switched virtual circuits capability which allows ATM subscribers to initiate a connection to the choice destination without the need for pre-configuration by the network operator [7]. This will open up exciting new

opportunities for distant teaching and learning. The development work presented in this paper represents only a small but significant step in the evolution of distant education. Current and future work in this area will focus on more complete setups and expand the work to remote monitoring and control of automation processes in factories.

CONCLUSIONS

Distant education is evolving to play a more significant role since its inception in the last century. It has been driven by the need of life-long learning, especially for engineering professionals. This has created the need for a learner-centered, case-based, contextualized and democratic education system which can be facilitated by the use of IT to provide a fully interactive and multimedia learning environment.

This paper has presented a particular piece of work dedicated to the development of an Internet experiment to carry out modeling and control of a pilot-scale DC, motor using only a web-server on the client's side. The experiment thus developed can be attended remotely and performed at the student's own pace. Internet connection and freely downloadable web browser are the only tools needed by the student. The system is based on the G-Web Server which is provided by the Internet Developers Toolkit package, an add-on component to LabVIEW. The control of the laboratory equipment from the browser is achieved through CGI, the standard interface between a HTTP server and other resources of the server's host computer. In this case, the other resource, apart from those mentioned above, is the DC servo motor VIs written in LabVIEW stored in the memory of the host computer.

Acknowledgment—The authors would like to express their deepest gratitude to National Instruments for their support and advice during the course of development.

REFERENCES

1. Moore and Kearsley (1996), Study Guide for Distance Education: A Systems View. Available via web at <http://www.gwu.edu/~etl/deguide.html>
2. Tim Berners-Lee Biography, <http://www.decus.org/decus/people/berners-lee.html>
3. The Globewide Network Academy, <http://uu-gna.mit.edu:8001/uu-gna/index.html>
4. National Instruments LabVIEW Internet Developers Toolkit Webpage <http://www.natinst.com/labview/internet/>
5. National Center for Supercomputing Applications, NCSA HTTPd Home Page, <http://hoohoo.ncsa.uiuc.edu/cgi/>
6. World Wide Web Consortium, W3C Security Resources, <http://www.w3.org/Security/>
7. Singapore ONE home page, <http://www.s-one.gov.sg/html/mainmenu.html>

Tan Kok Kiong is currently an Assistant Professor at the Department of Electrical Engineering, National University of Singapore. His research interests are in the areas of industrial process control, automation and motion control.

Lee Tong Heng is currently an Associate Professor with the Department of Electrical Engineering, National University of Singapore. He is also the Vice-Dean (Research) with the Faculty of Engineering.

Leu Fook Meng received the B. Eng. degree from the Department of Electrical Engineering, National University of Singapore in 1998. He is currently a Research Engineer in the Department of Electrical Engineering, Control Division, National University of Singapore.