

INTERNET-BASED REMOTE CONTROL OF THE OSCILLOSCOPE BY A THIN CLIENT

Domen Hudoklin

University of Ljubljana, Faculty of Electrical Engineering,
Ljubljana, Slovenia, domen.hudoklin@fe.uni-lj.si

Abstract – Measuring equipment is often complex and expensive. So training on a simulation in virtual experiments is in many cases a good idea, especially if it is conducted over the internet. On the other hand, mathematical models, needed for the simulation of a complex measuring device and complex measuring phenomena have limited efficiency. For this reason a remote control of the real equipment, in this particular case the oscilloscope, is presented in this article. The training efficiency is increased by time sharing the instrument over the internet. Another goal was also to develop a realistic web-based interface that could be used by thin clients, such as an ordinary internet browser without the need for installation of supporting applications. Article discusses the benefits and drawbacks of such approach.

Keywords: remote oscilloscope, e-training

1. INTRODUCTION

Simulation of the complex measurement equipment has many advantages in training process. Among others, the hardware requirements for such virtual instrumentation are very generic, time responses can be controlled, and it enables simultaneous training of more trainees, all even more so by bringing it online [1]. On the other hand, complex functionality of the measuring device and real-world complex signals require also a complex mathematical modelling of the simulation, which follows the real situation only up to a certain degree.

For that reason a remote instead of a virtual control [2-4] of the true digital storage oscilloscope (DSO) was developed, which was offered broadly for training purposes. The main idea was to control the DSO over the internet by a thin client. This means that no more than ordinary internet browser is needed without having the need to install lengthy supporting applications. It should be run on a public computer without administrator rights needed for such installation, and also on systems with different (even low-level) operating systems, such as 3rd generation cellular phones.

Another requirement for the DSO was to follow the real interface as realistically as possible. As it will be presented later on, the control panel was photographed in order to closely simulate key presses by a mouse click, and the DSO screen is displayed pixel by pixel to gain its true response.

Each user of the DSO is granted a certain time for its operation. A real time interaction between the user and the server/DSO can be followed for different purposes.

2. SYSTEM ARCHITECTURE

Different thin clients access the system through a simple internet browser using only HTML and basic JavaScript routines. After an authentication, clients start to operate the DSO remotely. If there are more users that want to operate the DSO at the same time, they are put in a queue and wait for the others to finish. Each user is granted certain predefined session time according to first-in-first-out (FIFO) rule. Based on the number of users (currently approx. 300) and their experience, the session duration is set to 30 minutes. If the queue is empty, they can continue to work.

After a successful authentication user starts to receive live image from the DSO in 1 second intervals. For every image the user's interface sends a separate request automatically by the user-side JavaScript routine. As shown in Fig. 1, the request is sent through the internet to server-side user session control module and then to the DSO driver, which instructs the DSO over the USB to return the screen image.

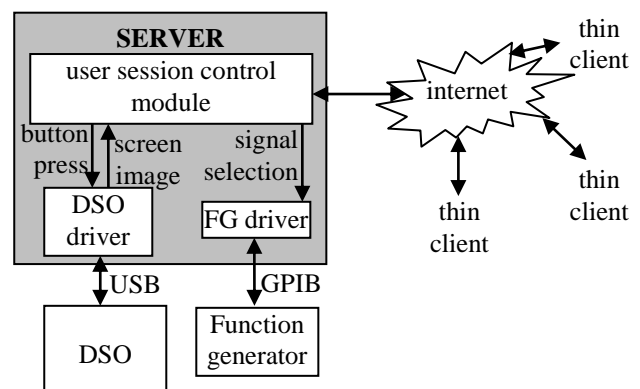


Fig. 1. Block diagram of the internet-based remote control of the DSO

While the screen image is received continuously, the user can operate the DSO by mouse clicking on the button in the virtual panel on the client side. Each button-click request is sent through the internet to the server-side user session control module. If the session is still active, the request is

again sent to the DSO driver and then to the DSO. The DSO in use, currently the Agilent DSO3102A, allows simulated button clicks programmatically, as if they were pressed by an operator through the physical control panel. This way, users can get hands on the complete oscilloscope features, including going through the different menus, by which they can test the triggering settings, waveform storage possibilities, display settings, filtering, probe and other channel settings. In this respect the trainee gets better insight to the full DSO possibilities.

The screen image of the DSO has resolution 320 x 240 pixels, which makes it possible to transfer it in different ways due to fairly small file size (on average 3 KB). At first, an attempt was made to capture the image by a simple web camera. A good quality picture was obtained only with an older DSO with monochromatic display and not with our newer DSO. The second approach by obtaining the image over the USB (not a standard feature for a particular DSO) gave far better results, as the image is copied pixel by pixel from the DSO memory. The downside of the latter approach is the time needed to read the picture and make it ready for the internet transfer. Typical time delay amounts to approximately 2 to 3 seconds, which is acceptable compromise.

To train on a set of different signals, a function generator (FG) is connected to both of the channels of the DSO. The generator can either be set to output one of the built-in functions or other test signals that were recorded previously. The client instructions to the generator are done again through the user session control module, preceded by the FG driver, which communicates with the FG over the GPIB.

3. USER INTERFACE

One of the key elements in e-training is also a user interface. It is generated by a server and passed to the client. The server together with both drivers, for DSO and FG, is developed in Labview 8.6. This version brings a major change in web application development. It includes a third-party web server, which is integrated in the Labview, which makes it easier to operate the hardware. Moreover, the user interface is now to be developed by other means (not Labview), using all the existing web technology. By this we get more control over the development of the user interface; however it requires more knowledge of other web-based technology. On top of that, it also gives a possibility to use the technology that is already incorporated in different

internet browsers. This means that the user interface can be a very simple program and also that the client does not need to download and install lengthy supporting applications of some 100 MB, just to provide an environment for the interface application to run.

One of the goals of this project was to make the user interface as realistic as possible and that it would be able to access it with a thin client. For this reason ordinary hypertext mark-up language (HTML) was used with basic JavaScript routines to automatically refresh the screen image from the DSO. The combination of the two enables the user interface to be run even on some cellular phones with simple internet browsers. Realistic look of the user interface was made by taking a high resolution image of the control panel of the DSO. The parts of the image with knobs were hyperlinked to the server. Fig. 2 shows the user interface in an internet browser.

4. DISCUSSION AND CONCLUSIONS

Training on a presented system has many advantages and also some drawbacks. The main advantage is that it is accessible over the internet by a thin client with generic software requirements (internet browser). This is very helpful, when the training is taking place on a computer, where the user does not have administrative privileges and installation of supporting applications is not allowed. Another advantage is that the trainee is controlling a real physical device, which gives him a deeper insight into the complex measurement features by allowing him to use the installed menus. This also eliminates the need for complex modelling of each function that is otherwise available in virtual instrumentation. The modelling is avoided also for preparing the test signals, since they are physically coupled with the oscilloscope, and they don't need to be obtained by prior data acquisition. The user interface is very realistic and imitates the real control panel. The interaction between the trainee and the DSO is immediate and the training is not done only on the user's computer locally. This gives a possibility to control each trainee activity, which is helpful for getting his indirect opinion for system improvement and also to assess his activity if in fact needed due to course requirements. Last but not least, the system program architecture allows easily replacing the type of device, because it does not need to mathematically remodel its functions, but simply reprogram the button clicks.

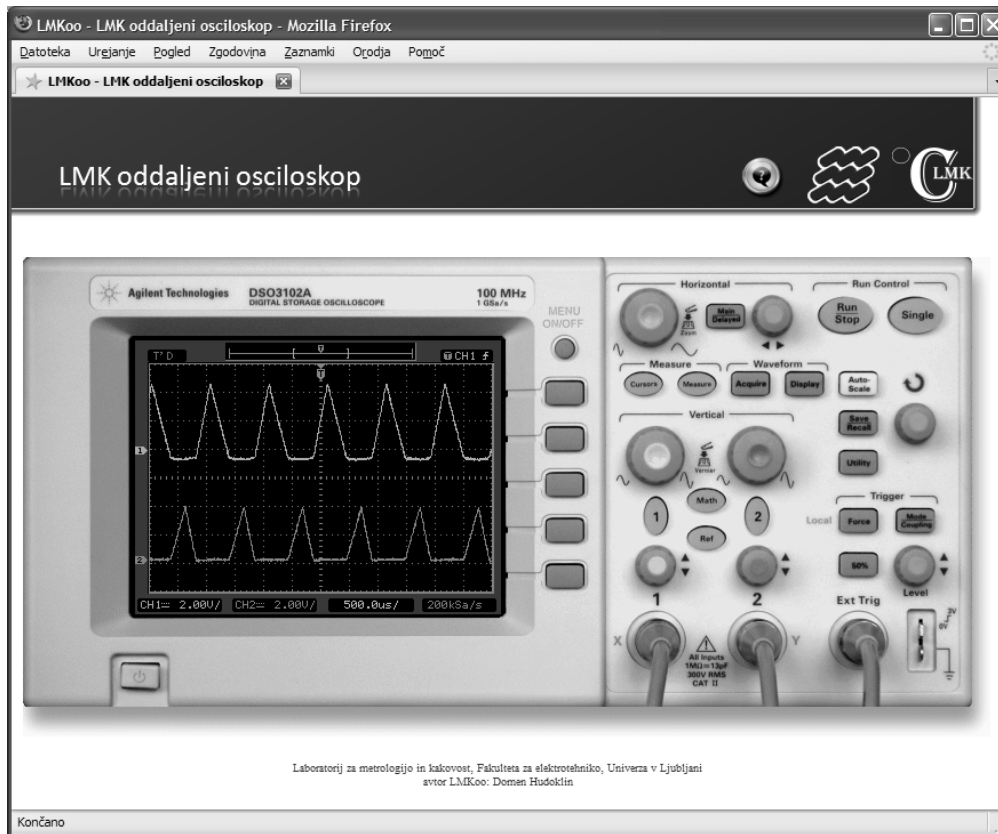


Fig. 2. User interface of the remote DSO in an internet browser

The main disadvantage is that the system does not allow more users to train at the same time as it is the case in virtual systems. Another disadvantage is that the system is a compromise between quality of a screen image and the response time, which introduces delays of approximately 2 to 3 seconds. The button-click requests are, however, without this kind of delay. In order to make a screen image to refresh over the internet faster, alternative use of web camera is possible, although at expense of the image quality. The drawback is also that the simple test signals can not be generated just by writing mathematical function, but need to be generated physically by a function generator. And finally, as oppose to the simulation, a true physical device needs to be purchased and maintained in order for system to work properly.

The presented system for remote control of the DSO over the internet has been successfully introduced in the beginning of October 2008 to approximately 300 students at the Faculty of Electrical Engineering, University of Ljubljana. Unfortunately, the system for tracking the web statistics of the students' response was developed only after a month since the introduction. However, judging on the visual activity of the oscilloscope, their response was encouraging for further development. Fig. 3 shows the number of browser hits (requests) from mid November on.

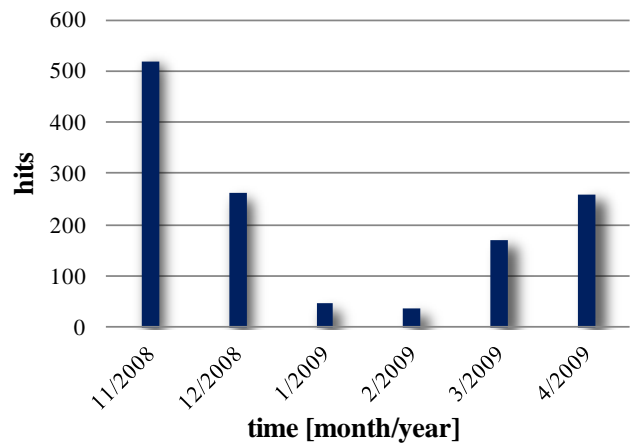


Fig. 3. The number of hits (requests), made by group of approx. 300 students since November 2008

Even though relatively high, the activity in November is subjectively assessed as significantly lower since the time right after the presentation. A noticeable decline in winter time is, further on, most probably related to exam period.

Based on the analyses of user requests and their feedback, the system would be more attractive if it included test signals that would be related to the experimental work within the scope of the syllabus. The key elements for the future development will therefore be the preparation of more targeted test signals.

The second step envisaged is to decrease the response time of the screen image refresh.

With a group of approximately 300 students, there have been no noticeable limitations in the time sharing of the instrument, since it operates 24 hours a day. The waiting queue has been rarely needed.

REFERENCES

- [1] Agilent, <http://education.tm.agilent.com>
- [2] Z. Nedic, J. Machotka, A. Nafalski, "Remote laboratories versus virtual and real laboratories", *Frontiers in Education*, pp. T3E-1-T3E-6 Vol.1, Nov. 2003
- [3] M. Joler, C.G. Christodoulou, "Virtual laboratory instruments and simulations remotely controlled via the Internet," *IEEE Antennas and Propagation Society International Symposium*, pp.388-391, 2001
- [4] I. Gustavsson, "Remote laboratory experiments in electrical engineering education," *Proceedings of the Fourth IEEE International Caracas Conference on Devices, Circuits and Systems*, pp. I025-1-I025-5, 2002