Multi-user/Multi-test-bed Remote Hardware Laboratory with Job Management System

Norihiro Fujii
Graduate School of Computer and Information Sciences, Hosei University, Japan
i02t9013@cis.k.hosei.ac.jp

Nobuhiko Koike
Faculty of Computer and Information Sciences, Hosei University, Japan
koike@hosei.ac.jp

Abstract
A new remote hardware laboratory, supporting multi-user/multi-test-bed, has been developed. Learners can perform experiments remotely utilizing actual hardware and actual measurement tools. It employs FPGAs for design implementation test-beds. It can handle many experiment sessions concurrently, by making use of actual hardware test-beds in time and space division fashions. Learners at remote sites can perform actual hardware experiments in parallel. It realized seamless remote and actual hardware laboratories. The combinatorial use of FPGA/PC connected test hardware and PC-based measurement equipment has made it possible to develop a multi-user/multi-test-bed remote hardware experiment system. In order to allocate available experimental environments for users at the same time, a job management system is employed. The prototype system of self-learning experiment environment for digital circuit designs and experiments is in use for 3rd class CS students at Hosei University.

1. Introduction
This paper describes the multi-user/multi-test-bed remote hardware experiment system for digital circuit experiments. Hardware experiment environment is usually treated as an exclusive resource, for single user usage. However, the actual test run time is rather short and most of the time is wasted leaving those precious resources idle. In order to achieve efficient usage of these environments and realize comfortable remote experiment environments, the remote laboratory system has been developed. It employs a timesharing scheme to make learners at remote sites to perform actual experiments using actual hardware equipment and tools concurrently. A job management system, named Condor[1], has been employed for the experimental environment allocations. The collaborations among the laboratory side servers, which contain PC-based testbeds/measurement equipment, and the client side PCs, which contain CAD tools, realized scalable remote laboratories organized as “Rich Client” configurations.

2. Design Motivations and Approaches
Nowadays, different kinds of distance learning systems have been developed and in use, thanks to the advancement of the Internet technology. One of the emerging areas is the natural science field, where a number of design-and-experiment runs using actual equipment are inevitable. Although, there are many existing virtual laboratory environments in the field, most such systems are employing so-called “the Client-Server architecture”. Applying simple Client-Server system for the scientific experiments has the following disadvantages: 1) to perform actual experiments concurrently, 2) to utilize actual measurement equipment remotely, 3) to analyze actual measured data, and 4) to support a number of users exceed the number of actual available resources[2].

For natural scientific experiments, especially computer design and implementation laboratory, it becomes important to make use of actual hardware, to obtain actual measured data, and to inspect actual experiment results. The existing client-server based remote laboratories cannot fulfill these requirements. In order to overcome these shortcomings, the authors have focused on the timesharing Web-based remote laboratory approach and Job management system, which realize an efficient sharing of test equipment and supports actual experiments runs/measurements concurrently.

3. System Considerations
In most existing remote laboratory systems, the experiment platform including target FPGA board, logic analyzer and pattern generator were exclusively occupied by single user for a long period of time. Once a user has started remote experiments, the measurement equipment was so expensive that it became prohibitive to prepare sufficient number of equipment. The authors observed the usage of the test benches and addressed that the actual occupation time of the test benches is rather short comparing with other tasks, such as preparation, compilation and post-analysis. So, a number of remote laboratory environments could efficiently share actual test
equipment, in a time division fashion, without introducing severe resource contention. As the modern logic analyzers and pattern generators are offering PC-based measurement environments, the control software for these PC-based equipment are also operational on the client PCs, and reproduce the same operating environments, except for the actual measurement functionalities. Learners can prepare test runs and analyze the experiment results on their own client PCs, decoupled from actual measurement equipment. The job management system is employed for the Web-based remote laboratory system, in order to perform the conflict-resolution among equipment usages, job submissions and job allocations.

The remote hardware laboratory system performs the following steps along the workflow as shown in Table 1. These steps are necessary to complete the digital hardware design cycle and to implement a circuit into FPGA. At Step 7 and Step 8, the Remote Laboratory offers the following services; circuit implementation, experimental test run, measured data acquisition to validate behaviors of circuits. From Step 1 to Step 6, and Step 9 in Table 1, learners can perform pre and post test-run tasks by making use of their own client PCs.

Table 1. Design and Implement Workflow

<table>
<thead>
<tr>
<th>Step</th>
<th>Workflow</th>
<th>Location</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>DesignEntry</td>
<td>Client</td>
</tr>
<tr>
<td>2</td>
<td>HDLC Compiler</td>
<td>Client</td>
</tr>
<tr>
<td>3</td>
<td>Logic Synthesis</td>
<td>Client</td>
</tr>
<tr>
<td>4</td>
<td>Logic Simulation</td>
<td>Client</td>
</tr>
<tr>
<td>5</td>
<td>Placement and Routing</td>
<td>Client</td>
</tr>
<tr>
<td>6</td>
<td>Measurement Conditions Setting</td>
<td>Client</td>
</tr>
<tr>
<td>7</td>
<td>Implementation of the actual target unit</td>
<td>Remote Lab.</td>
</tr>
<tr>
<td>8</td>
<td>Test and Validation</td>
<td>Remote Lab.</td>
</tr>
<tr>
<td>9</td>
<td>Analysis of obtained data (post-analysis)</td>
<td>Client</td>
</tr>
</tbody>
</table>

4. Remote Hardware Experiment Environment

To realize the remote laboratory using the Internet broadband connectivity, we should consider the following features: 1) Job Management System, 2) Service Registry, 3) Experiment Platforms, and 4) Rich Client-site programs. The overview of the proposed architecture of a multiuser remote laboratory system is shown in Figure 1.

Job Management System: It handles the multiuser remote laboratory environments and the distributed resource management. It resolves conflicts among users, when many users request the same services at the same time. Job manager “Condor” prevents contentions and arranges the service scheduling. It enables to schedule multiuser and multijob experimental environments for many learners at the same time, while eliminating the conflicts among equipment usages and avoiding the deadlocks.

Service Registry: The service registry contains specification of experiment platforms to find out suitable platforms. It consists of several related information, such as platform IDs, supported target devices, measurement equipment data and so on. It provides a mechanism for users to find out appropriate experiment platforms, necessary to perform specific experiments, dynamically.

Experiment Platform: It consists of a number of experiment platform servers. Each contains PC-based measurement equipment, such as Logic Analyzer/Pattern Generator, and a target FPGA connected PC. Their major roles are: to allocate measurement/target unit equipment resources and to control the experiment executions.

Rich Client Software: All client PCs perform the tasks from Step 1 to Step 6, and Step 9 in Table 1. As these tasks are offloaded from the server host and executed by client PCs, this rich client configuration allows the system to become scalable. The server can concentrate on other tasks, such as the job management and hardware resource managements. As the same FPGA concentrate on other tasks, such as the job management and hardware resource managements. As the same FPGA/PC CAD software is used both for the remote client-site PCs and for local PCs in the actual laboratory, the same environment can be reproduced at remote site, after getting the floating licenses form the license server through a virtual private network (VPN).

5. Conclusions

The advantages applying this remote laboratory are: 1) Location-free efficient sharing of actual experiment equipment, 2) Quick turnaround time of hardware design and experiment cycle, 3) Cost-effective way of sharing expensive measurement equipment, 4) Get the feeling of handling actual measurement equipment and hardware from the remote site, and 5) Remote acquisition of actual measured data to inspect the designed digital circuit behaviors. An effective and practical remote multi-user timesharing hardware experiment system has been realized.

References