

Designing and Implementing a Virtual 3D Microcontroller Laboratory Environment

Jeffrey Richardson, Nicoletta Adamo-Villani, Edward Carpenter, George Moore
 Purdue University, College of Technology, West Lafayette, IN 47907
 jjrichardson@purdue.edu, nadamovi@purdue.edu, ecarpent@purdue.edu, moore19@purdue.edu

Abstract - Laboratory instruction is a major component of undergraduate curriculums throughout the United States. The laboratory experiences represent fundamental instructions for all technology students. Traditional laboratory instruction is hampered by several factors including limited access to resources by students, high laboratory maintenance cost and the inability to deliver the laboratory content of a course at distance. This paper describes the development of an interactive, photorealistic 3D computer-simulated laboratory for undergraduate instruction in microcontroller technology. The virtual lab operates and produces results equal to the physical laboratory. In addition, it includes an extremely realistic representation of devices and components, thus providing the students with the mental engagement necessary to successfully complete the experiments outside the confines of a traditional laboratory. The 3D virtual lab can solve the majority of the problems associated with traditional laboratory instruction, and can provide students with the same level of understanding of the experiments as a real laboratory environment.

Index Terms – Virtual laboratory, 3D computer-simulation, Microcontroller, Distance Education.

INTRODUCTION

Laboratory instruction is a major component in engineering and engineering technology programs across the country and around the world. Purdue University is no exception. In the Electrical and Computer Engineering Technology department at Purdue University, approximately 95% of the course offerings have an accompanying laboratory. The laboratory experiences require the students to build and test prototype systems. Some of these experiments involve programming microcontrollers and connecting them to peripheral devices. In general, electrical engineering instructional laboratories follow the paradigm of students spending two or three hours a week in a lab, clustered around work benches learning to build and program circuits while using test equipment under the guidance of the faculty or teaching assistant [1].

There are several problems associated with traditional laboratory instruction. One of the major problems is limited opportunity to repeat the lab assignments, since further practice depends on availability of laboratory space and

resources. Students participating in distance education courses may face complete unavailability of laboratory facilities. For smaller schools, the cost to equip and maintain a laboratory may exceed the available funds.

The object of our research is the development of an interactive, photorealistic 3D computer-simulated laboratory for undergraduate instruction in microcontroller technology to address the shortcoming listed above. The project includes a highly realistic representation of devices and components. The virtual lab operates and produces results equivalent to the physical laboratory.

BACKGROUND

The value of simulation tools and other multi-media technologies in the learning process has been documented in numerous publications [2]. Research shows that the use of simulation tools reinforces learning and leads to performance improvements in a variety of disciplines [3]. There has been significant progress in the development of computer-based tutorial systems in many different areas over the last several years. There are several noticeable examples of engineering virtual laboratories. Del Alamo [4], a professor of electrical engineering at MIT, created a web-based microelectronics lab for his students in 1998. At Johns Hopkins University, Karweit [5] has simulated various engineering and science laboratories on the web. At the University of Illinois Urbana-Champaign (UIUC), researchers have developed a virtual laboratory for earthquake engineering [6]. Oakley, in the department of Electrical and Computer Engineering at UIUC, has developed an interactive tutorial for an introductory circuit analysis course: ‘CircuitTutor’ [7]. At Vanderbilt University Mosterman et al. [1] have created and evaluated the effectiveness of a computer-simulated lab for use in undergraduate engineering education. Electronics Workbench [8] has been producing educational based electronics software for 20 years. In particular, the MultiSIM 9 software is an electronic lab on a PC that allows for circuit simulation with schematic entry, and includes a few realistic components. At Temple University, Duarte and Butz [9] have produced the ‘Virtual Laboratory for the Disabled’ aimed at engineering students who do not have adequate mobility of hands and upper bodies to perform the physical laboratory experiments.

The problem with the majority of computer-simulated labs developed so far is that the representation of the virtual environments and their components is very schematic and

two-dimensional. In fact, most of the existing virtual labs are just software simulations. The software simulations and schematic representations of the hardware are far from the actual hardware utilized in the real laboratory and do not provide an authentic lab experience.

To address the shortcoming listed above, we have turned to a photorealistic interactive 3D computer animated solution which provides an authentic lab experience.

THE VIRTUAL LABORATORY

The virtual laboratory includes true-to-life replicas of the laboratory development boards and workstation equipment utilized in the undergraduate microcontroller curriculum at Purdue University. Figure 1 shows a screenshot of the virtual laboratory environment. The laboratory development boards shown include a microcontroller development board, a stepper motor board, and a lights & switches board. Additional development boards are also available and include an alphanumeric keypad board, a 7-segment LED display board and a liquid crystal display board. The current workstation equipment includes a variable, triple output power supply (shown in Figure 1), a digital multi-meter and a digital logic probe (not shown).

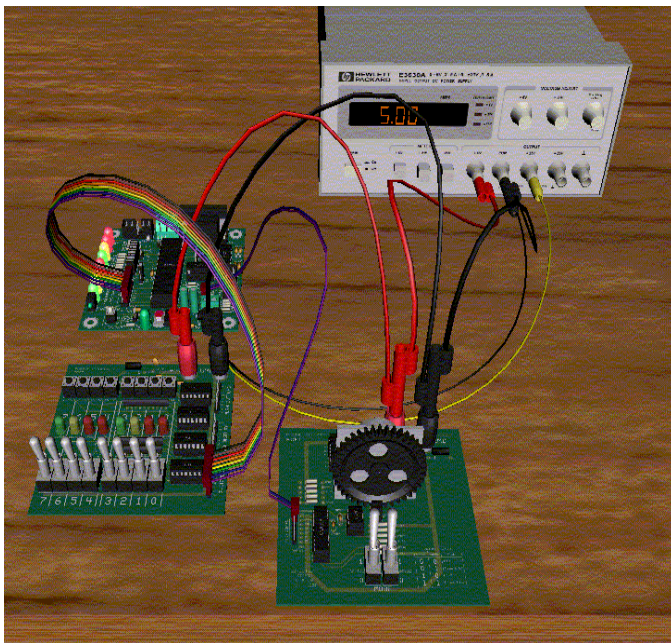


FIGURE 1 – SCREENSHOT OF THE VIRTUAL LABORATORY

All of the 3D objects were initially modeled from the actual technical drawings of the components. Once the actual sizes of the objects were determined, the real hardware and pictures of the real hardware were used to complete the models with special attention given to the details. Figure 2 shows a close up of the microcontroller development board illustrating the level of detail in the virtual models.

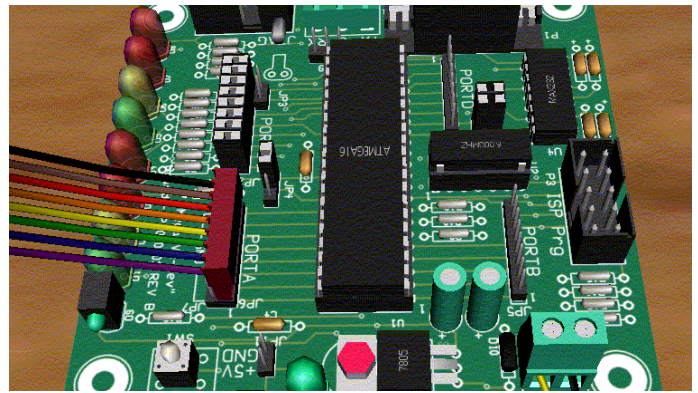


FIGURE 2 – DETAILED VIEW OF THE MICROCONTROLLER BOARD

The virtual lab was implemented using interactive 3D computer animation technology commonly found in computer video games. Maya 6.5 (Alias™) software was used to model the 3D lab components and to animate their functionality. For example, push buttons actually move up and down as they are clicked with the mouse, lights illuminate and glow when activated, toggle switches change position when actuated and knobs actually rotate. OpenGL with the C/C++ programming language was used to program the interactivity with the components and the various development boards.

VIRTUAL LABORATORY INTERFACE

The graphical interface for the virtual laboratory consists of one resizable application window and one toolbox window. The toolbox window allows the user to modify the current view by zooming in and out, moving left or right and up or down, and rotating around the X, Y and Z-axis (also known as tumbling). These controls allow the user to completely manipulate the environment to create any view necessary or desired. The toolbox also allows the user to make all of the required interconnections with the various wires and cables that are visible in Figure 1.

VIRTUAL MICROCONTROLLER DEVELOPMENT BOARD

The embedded microcontroller development board is the focal point of the virtual laboratory and is based on the ATmega16 RISC microcontroller from Atmel Corporation. The entire virtual microcontroller is organized to match the real world counterpart. The virtual microcontroller, like its real world counterpart, contains separate memory arrays to support FLASH, SRAM and EEPROM memories as well as the Register File locations and the Input/Output Register locations. In addition to these elements, the virtual microcontroller contains key single memory locations for the microcontroller's Status Register and Program Counter.

The key feature of the virtual microcontroller board lies in its ability to decode and execute the various microcontroller instructions. To accomplish this task, the microcontroller must be "loaded" with an Intel Hex file which contains the actual machine code for the microcontroller. The hex file is the industry standard format and is the same file used to

program a real microcontroller. This feature eliminates the special files (normally object files) that are commonly used by commercial simulators. The hex files can be produced by a variety of methods including any C compiler or assembler capable of generating executable code for the target microcontroller. The appropriate hex file is selected through a standard window's type, graphical interface. The interface and sequence of operation for placing the program in the microcontroller is modeled after the current preferred method used at Purdue, but it can be modified to mimic any method that may be utilized in another laboratory environment.

Once the appropriate file has been selected, the file is processed line-by-line and character-by-character to extract the information. The microcontroller instructions and supporting data (opcodes and operands) are placed into a memory array that represents the actual flash memory of the microcontroller. At this point, the data stored in the virtual microcontroller's flash memory is organized identically to the way the data would be stored in a real microcontroller programmed with the same file.

INTERCONNECTING DEVELOPMENT BOARDS

The virtual development boards and the real laboratory development boards are connected with cables that allow the pins of the microcontroller to be connected to the various input and output pins on the other peripheral boards. The virtual cables are created dynamically allowing for any desirable combination to be created. The cables are available in a 2, 4, 5, 6 and 8 pin configuration. They are placed in the virtual laboratory through the toolbox window and serve two key purposes. The first purpose is to visually mimic the real laboratory environment and the second is to *carry* the digital signals between the boards.

DEVELOPMENT BOARD INTERACTION

The header pins located on the development boards are the key interface element. The development boards were created using the object-oriented paradigm. Therefore, each of the header pins is an object with properties that include the current digital value. When the pin is an output (providing data to another device), the state of the development board will determine the appropriate digital value associated with the pin. For example, a switch on the lights & switches board is considered an output; therefore, the position of the switch determines the digital value on the corresponding pin. When the pin is an input (receives information from another device), the value on the header pin will determine the behavior of the development board. A light on the lights & switches board is considered an input; therefore, the digital value on the corresponding pin will determine whether the light is on or off.

The header pins on each of the development boards determines whether or not a cable is connected to the header. When a cable is connected, the cable serves as a mechanism to link the pins of one header to the pins of another header located on the opposite end of the cable. With a cable in place, an output signal at one end of the cable can control an

input value located on the other end. The ability of the cables to connect one header pin to the next and the ability to be dynamically placed are the most important elements in the entire virtual laboratory.

VIRTUAL CODE EXECUTION

The virtual laboratory's operations are controlled through a system clock. Each tick of the system clock instructs the virtual microcontroller to execute the next instruction provided the virtual power supply is connected properly and actually turned on. The virtual microcontroller executes an instruction by reading the machine code instruction located in the appropriate memory location. The instruction is then interpreted through an instruction decoder routine which actually performs the specified operation. As part of executing the instruction, the microcontroller's status register and program counter are updated according to the manufacturer's specifications.

MICROCONTROLLER CODE CAPABILITIES

The microcontroller instructions are broken down into five main categories: arithmetic & logic instructions, branch instructions, data transfer instructions, bit & bit-test instructions and MCU control instructions. There are four total MCU control instructions. Of these four instructions, three are not applicable to the virtual microcontroller laboratory environment and therefore have not been implemented. The remainder of the instructions from all the other categories have been successfully implemented and tested.

Most of the microcontroller instructions move data from one memory location to another. However, two of the microcontroller's instructions interact with the input/output register memory locations and are vital to the operation of the virtual laboratory. These instructions either place data into or extract data from the memory locations. The values located in the input/output register memory locations are tied directly to the header pins located on the virtual microcontroller board. When a cable is connected between the microcontroller and another development board, changing the values stored in the output register memory location will ultimately change the input value to the other development and alter or control its behavior. Likewise, a change to an output device on the development board will ultimately change the input value stored in the input register memory location allowing the microcontroller to read the current status of the output devices.

Each time the graphics are updated, the position of each output device on the virtual development boards is checked and the value of the header pin associated with the output device is updated. When the header is connected to a cable, the cable's properties are updated and new value is transferred to the header located on the other end of the cable, thus, updating the corresponding input value on the other development board. This process is repeated continuously, allowing updates to be processed almost immediately.

LABORATORY CONTENT

The virtual laboratory was designed to deliver the contents of the first five microcontroller laboratory experiments and allows the students to experimentally test a plethora of basic concepts. The virtual laboratory provides an environment where the students can learn to implement basic input and output operations and explore variable types and sizes. The laboratory also allows the students to implement mathematical equations using integer and floating point operations and compare the results. The students are able to implement decisions and apply logical operations as well as iterative operations.

The students are also able to explore the layout and operation of each of the individual laboratory peripheral boards including header & component placement and device movement such as toggle switches or push buttons. This allows the students access to development boards without the cost associated with purchasing each of the board individually.

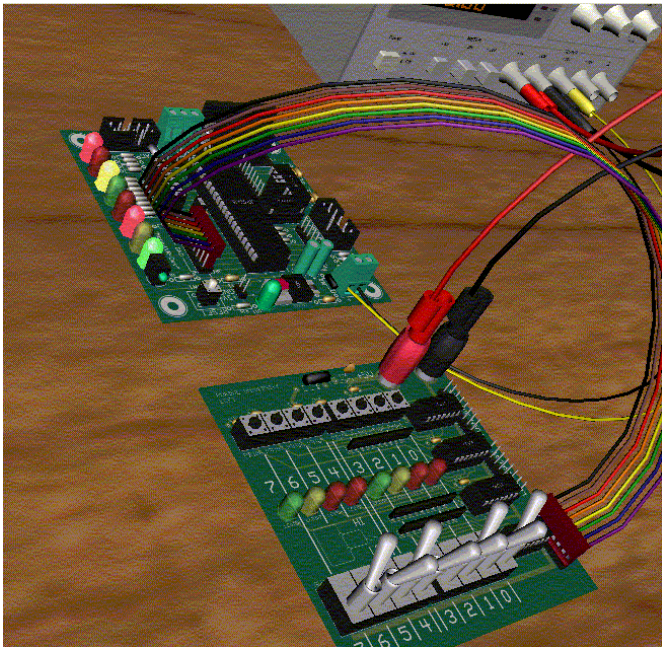


FIGURE 3 – INPUT/OUTPUT TEST PROGRAM RESULTS

LABORATORY OPERATION

In order to use the microcontroller or any of the development boards, they must be supplied with an appropriate supply voltage from the virtual power supply. The virtual laboratory has a triple output DC power supply modeled from the actual Hewlett Packard supply located in the laboratory. Properly powering the microcontroller development board involves two basic steps. The first step requires the power supply to be set to an appropriate voltage and the second step requires the supply to be connected with the power cables. Requiring the student to operate the power supply is another step closer to creating a real world laboratory experience.

LABORATORY TESTING RESULTS

The virtual laboratory environment has been evaluated and tested throughout the development process by expert users within the Electrical and Computer Engineering Technology Department at Purdue University. These experts include faculty members and graduate students. The virtual laboratory was also tested by students during the summer session of the introductory microcontroller course.

The virtual lab operates and produces results equal to the physical laboratory in several respects. The virtual laboratory operation requires the students to physically place the development board into the virtual workspace and make all the required cable connections. The virtual laboratory is programmed to respond according to the placement of the cables. Cables in the wrong place or missing cables in the virtual laboratory will produce the same results as missing or misplaced cables in the real laboratory. In addition to the cables, the power supply must be properly connected and set to the appropriate voltages. Inappropriate voltage settings or forgetting to turn on the power supply produces identical results in the virtual and real laboratory environments. The virtual microcontroller has been programmed to execute the same instructions as the real microcontroller. Each of these instructions has been tested to ensure accuracy in the results.

In addition to the functional test results above, the individual components of the laboratory were evaluated for overall appearance and operation by the experts. The summarized comments are listed below.

- **Microcontroller Board:** Overall great visual representation of the actual development board. One minor visual issue involves missing component leads on the passive elements.
- **Lights & Switches Board:** Good visual representation. Again, some passive components are missing leads (very small visual detail).
- **Stepper Motor Board:** Great overall visual representation. The stepper motor operates in full step and half step mode; however, the direct drive and wave drive modes still need implemented.
- **Power Supply:** Power supply knobs are hard to operate since they require rotation. The light gray face of the power supply is hard to read at certain angles due to lack of contrast with the other visual elements. The negative tracking operation still need implemented.
- **Cables:** The cables tend to have some sharp edges instead of being completely smooth along their lengths but overall look good.

The students found the virtual environment to be “cool” and “impressive” while producing the same results as the actual hardware. The virtual environment also provided a realistic experience by requiring the connection of all components and the power supply. The virtual laboratory operates at a slower pace than the real hardware and was the overall number one complaint of the student users. At the time of the testing, the user interface was not user-friendly and

also received poor remarks. Overall, the student feedback was very positive for the virtual environment.

VIRTUAL MICROCONTROLLER LIMITATIONS

There are two key limitations to the current virtual microcontroller laboratory. The first limitation is in the number of internal peripheral devices that are supported. The initial implementation plan for the virtual laboratory only required the internal UART to be supported in order to exchange serial data with a virtual terminal to assist students with trouble shooting their software. The internal UART has been programmed with the functionality required to exchange data with the virtual terminal window that has been incorporated within the environment. The remaining internal peripherals will be implemented in the order with which they are utilized in the course. This implementation will occur over the next several months.

The last limitation of the virtual laboratory is the speed of program execution. A program being executed in the virtual laboratory operates at a noticeably slower rate than the same program operating on the real hardware. The speed with which the virtual laboratory operates is directly related to the speed of the PC's processor and operating system. Future testing will attempt to quantify the actual performance of the virtual laboratory.

FUTURE WORK

The virtual microcontroller laboratory environment is capable of performing all the necessary functions to support the first five laboratory experiments in the introductory microcontroller course for which it was developed. Additional features will be added during the next calendar year to increase the capabilities to include the first ten laboratory experiments. The increased capability will result directly from the implementation of the internal peripheral operations of the microcontroller. Additional work will be performed to address the shortcomings identified during the expert and student evaluations.

ACKNOWLEDGMENT

This research is supported by Purdue University TLT (Teaching and Learning with Technology) grant (Proposal #00006585).

REFERENCES

- [1] MOSTERMAN, P. J., DORLANDT, M.A.M., CAMPBELL, J. O., BUROW, C., BOUW, R., BRODERSEN, A. J., BOURNE, J. R., "Laboratories: Design and Experiments", *IEEE Transactions on Education*, 39,3, 1996, pp.309-313.
- [2] CHUNG, G., K., HARMON, T. C., BAKER, E. L., "The Impact of a Simulation-Based Learning Design Project on Student Learning", *IEEE Transactions on Education*, 44, 4, 2001, pp. 390- 398.
- [3] ALLWOOD J. M., COX, B. M., LATIF, S. S., "The Structured Development of Simulation-Based Learning Tools With an Example for the Taguchi Method", *IEEE Transactions on Education*, 44, 4, 2001, pp. 347- 353.
- [4] MANNIX, M., "The virtues of virtual labs", *ASEE Prism* – Sep. 2000. Available at: <http://www.prism-magazine.org/sept00/html/toolbox.cfm> (retrieved 12/08/2005)
- [5] KARWEIT, M. A., "Virtual Engineering/Science Laboratory Course", Available at: <http://www.jhu.edu/~virtlab/virtlab.html> (retrieved 12/10/2005)
- [6] Smart Structures Technology Laboratory (SSTE) at UIUC, "Virtual Laboratory for Earthquake Engineering", Available at: <http://cee.uiuc.edu/sstl/java/> (retrieved 12/10/2005)
- [7] BURKS OAKLEY II, "Computer Aided Instruction: Implementation of Interactive Tutorials for Introductory Circuits Analysis", *Proc. of New Approaches to Undergraduate Engineering Education III*, July 1991, pp. 189-202.
- [8] Electronics Workbench. Available at: www.electronicworkbench.com (retrieved 1/15/2006)
- [9] DUARTE, M., BUTZ, B., "The Virtual Laboratory for the Disabled", *Proc. of 31st ASEE/IEEE Frontiers in Education Conference*, S1C-23, 2001.