Outreach Project Introducing Computer Engineering to High School Students

Michael Gasser, Yung-Hsiang Lu, and Cheng-Kok Koh
Purdue University, magasser@purdue.edu, yunglu@ purdue.edu, chengkok@ purdue.edu

Abstract - **This paper presents an outreach program that targets students in high school. The program is designed to be fun, hands-on, and interactive, while introducing educational building blocks that will prepare students for careers in computer engineering. Key concepts introduced to students include logic design, hardware-software interface, simple circuit design, robust design practices, and teamwork. An adaptation of the popular video game Dance Dance Revolution (DDR) is the platform on which this program is built. Teams of students construct dance pads from arts and crafts materials and aluminum foil. They also design and implement a scoring circuit based on Boolean logic, using logic gates on a solder-less breadboard. The scoring circuit is then interfaced with a preprogrammed micro controller and graphical user interface to complete the DDR design. Using DDR as a platform allows students to conquer several design challenges including building reliable dance pads and designing logic for a scoring circuit. The program was used three times during engineering summer camps hosted by the Minority Engineering Program and Women In Engineering Program at Purdue University in 2009. It received overwhelmingly positive reviews from the students that participated.**

Index Terms - Computer Engineering, Pre-College Outreach, Logic Design, Game Development.

MOTIVATION

There is a general consensus among professionals and educators in computer engineering that the future will require an increase in the number of educated workers in computer related fields. Current enrollment in education is not at a sufficient level to supply the number of professionals needed [1][2]. Along a similar vein, most computer related professions and educational environments are lopsidedly male and non-minority [3]-[5]. These problems are both large and complicated, but similar in that they both require attracting people to computer engineering. This paper presents an outreach program for high school age students designed to attract them to the field of computer engineering. This paper first describes the details of the project and then describes an implementation of the project that took place at Purdue University.

PROJECT CRITERIA

The project is designed around the following criteria. The criteria are chosen in large part based on prior educational experience, prior experimentation in outreach, and our vision of what outreach should look like.

- The project should be hands on and interactive. The students should spend most of their time *doing* the project rather than listening to lectures. Evidence shows that this is both more interesting and leads to more knowledge retention in students [6].
- The project should emphasize both design and implementation. The authors feel this is the essence of engineering. Students need to understand both why things are working and how to make them work.
- The project should involve a hardware/software interface. The project is designed to be different from commercial or typical education kits. Many of these focus singularly on either building hardware or software programming. We believe it is important to see the interaction between the two.
- The project is an engineering project and as such should involve engineering principles. The project is designed to teach the fundamentals of thinking like an engineer as well as emphasize reliability.

PROJECT DESCRIPTION

There is a large body of prior art showing that leveraging the inherent fun and interest in games can be an effective tool for educating in engineering [7]. The authors designed this project based around this idea.

An imitation of the popular video game Dance Dance Revolution (DDR) provides the vehicle for implementing the project. DDR is typically played with a television or monitor and a dance pad. The dance pad features four arrows pointed in each of the cardinal directions. On the television or monitor, arrows scroll up the screen to a point at which they become a "goal" and the player dances on the corresponding arrow in time with the scrolling graphical user interface (GUI) arrow.

This game is implemented with four distinct parts; two of which are designed and implemented by students and two of which are created by us prior to student use. Students design the physical dance pad and digital scoring logic.

Students are provided with the graphical user interface and a pre-programmed micro-controller. The students integrate these four elements into a working DDR game. This can be seen schematically in Figure 1.

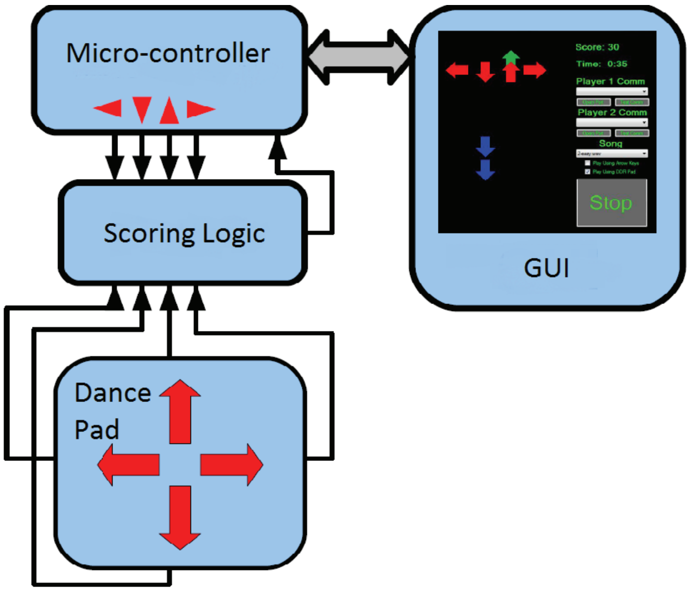

FIGURE 1
DIAGRAMATIC VIEW OF THE DDR PROJECT.

I. The Dance Pad

The dance pad is constructed from arts and crafts materials and aluminum foil. Aluminum foil allows for the creation electrical switches that are activated by pressure, ie dancing. Students design the dance pad to function like four switches (one for each arrow). The switches alter a logic value that is fed into the scoring circuit.

II. The Scoring Logic

The scoring logic is designed and implemented on a solder-less breadboard with plastic dual in-line packaged (PDIP) logic gate integrated circuits. The goal of the logic is to use the dance pad as four switches between logic high and logic low depending on whether or not the pad is being stepped on. The logic values are compared with the four output pins on the micro controller which are supplying the correct dance steps. The comparison of the dance pad to the micro controller is then fed back into the micro controller through an input pin. The students are responsible for designing the logic network for this comparison and building it on a breadboard.

Creation of the glue logic between the pad and the micro-controller requires comprehension of some Boolean algebra as well as its implementation in circuitry using PDIP logic gates. Students must understand some basic Boolean logic gates; AND, XOR (exclusive or), and NOT (inverter). This is done incrementally. First, a scoring circuit for an imaginary DDR game in which there is only one pad is designed. This requires only an XOR and a NOT gate, seen in Figure 2. Then the gate array to do a complete comparison in a full four pad DDR game is created. This can also be seen in Figure 2. The system of using logic gates as glue allows flexibility that improves the student experience. There are several valid variations and extensions of the design shown in Figure 2.

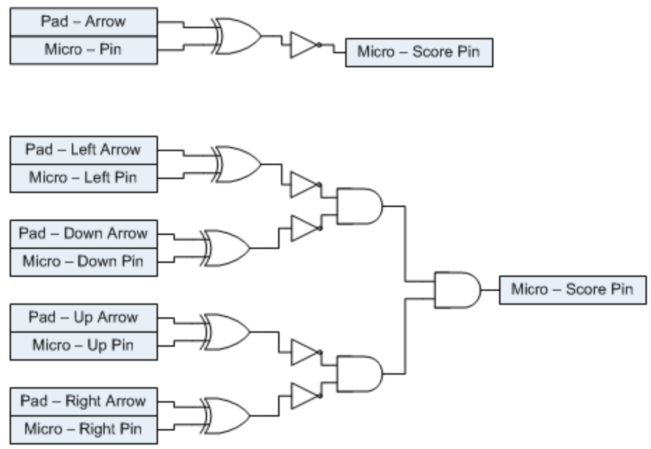

FIGURE 2
A "ONE ARROW" LOGIC GAME AND THE GENERALIZED FULL GLUE LOGIC.

III. The Graphical User Interface

The GUI provided to the students imitates the commercial DDR GUI. It provides scrolling arrows and the ability to choose between songs. It keeps track of score and the time played. A screenshot of the GUI can be seen in Figure 3.

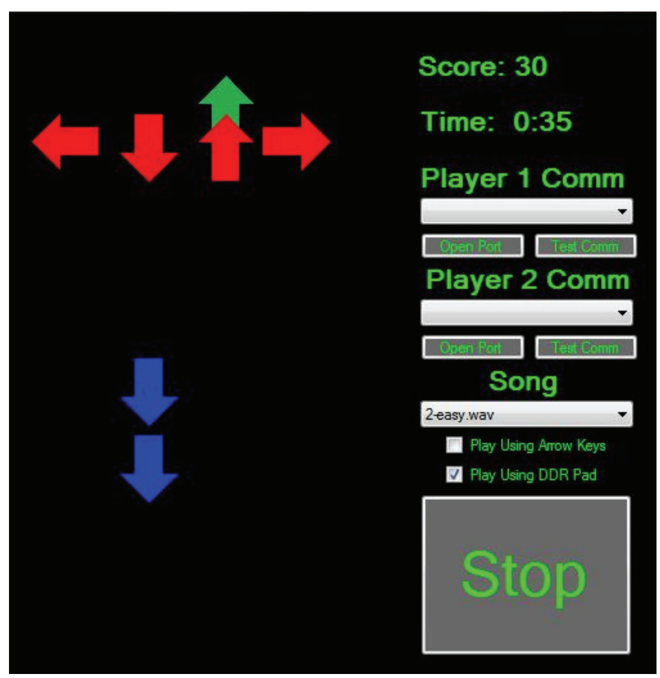

FIGURE 3
A PARTIAL SCREENSHOT OF THE DDR GUI.

IV. The Micro-controller

The micro-controller provided to students serves as an interface between the GUI and the breadboard; between student-built hardware and the provided software. It maintains four output pins, each representing the current state of a goal arrow. For example, in the screenshot in Figure 3 the current micro-controller state would have a high voltage at the pin representing the up arrow and low voltages on the other three output pins. The micro-controller also has an input pin to receive the scoring signal. It de-bounces this signal and enforces a policy of only one score per change in goal state. Additionally the micro-controller has a pin "blinking" twice every second between high and low voltage to serve as a visual indicator that the micro-controller is powered.

PROJECT IMPLEMENTATION

This project was implemented in three summer camps at Purdue University over the summer of 2009 organized by the Purdue University Minority Engineering Program (MEP) and Women In Engineering Program (WIEP). The project was used in one MEP camp and two WIEP camps. Each camp lasted a week and was composed of between 25 and 40 high school students. In each week the project was given approximately 9 hours, spread out over 3 or 4 sessions. Students completed the project in teams of three or four.

The project is divided into five phases, each with specific goals. Each phase lasts between 1.5 and 2 hours for a total of 9 hours. Each phase starts with a short lecture that explains the goals of the phase and the relevant material necessary to understand the phase. Phases also serve as stopping points to keep the students on track to complete the project.

I. Basic Circuit and Dance Pad

The primary educational goal of phase 1 is to introduce students to the basic circuit. Without dwelling on theory, students are introduced to the idea of a generic power source and high and low voltage. Then the students are taught how a switch works. With this knowledge, students are prompted to create dance pads. The pads act as switches, being an open circuit until stepped on. Students are supplied with cardboard, compressible foam, aluminum foil, wires, and arts and crafts supplies. To verify the functionality of their switches, students are provided with batteries, lights, and buzzers. Students are also encouraged to decorate their pads. A completed pad can be seen in Figure 4.

We observe that students quickly learn that creating a switch is relatively easy. Creating one that still works after being stepped on several times can be difficult. Reliability quickly becomes the primary concern of this phase. Even so, this phase requires the least technical knowledge and as such lets students become comfortable in their groups.

In practice, the implementations of the dance pads were varied and unique. Several different methods for the switching mechanism were designed by the students, including using arches or cardstock to separate layers of aluminum foil, using coiled wires as springs to make contact with a layer of aluminum foil, and using foam to separate layers of aluminum foil. These can be seen in Figure 5. The method of separating the conductive surfaces with layers of foam is the same method employed by many commercial DDR pads. Additionally, an example of the internal workings of a student created dance pad can be seen in Figure 6.

FIGURE 4
A COMPLETED DANCE PAD.

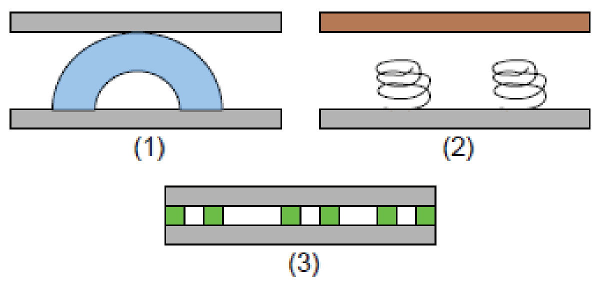

FIGURE 5
THREE EXAMPLES OF DANCE PAD SWITCHES. (1) AN ARCH MADE OF CARD STOCK OR RESILIENT PAPER HOLDING TWO LAYERS OF ALUMINUM FOIL APART. (2) WIRE COILED INTO A SPRING WITH AN ALUMINUM FOIL LAYER BELOW AND THE END OF THE WIRE STRIPPED. (3) MANY SMALL PIECES OF COMPRESSIBLE FOAM SEPARATING TWO LAYERS OF ALUMINUM FOIL.

II. Hardware-Software Interface

The second phase introduces the solder-less breadboard, LEDs, a very general understanding of what the micro-controller does, and the GUI software. Before students begin working on this phase a brief safety lecture is also given. The safety lecture explains the two potential safety hazards of the project. First, shorts are explained and students are told to unplug their breadboards if it begins behaving in a way they did not expect. Second, students are told that when IC's are powered incorrectly they can generate heat, which can sometimes be enough to burn a finger. To avoid this possibility students are required to double check every time they connected an IC. In practice neither of these issues are terribly worrisome; they are unlikely to cause a fire or injury.

After the safety briefing students begin the phase by powering the breadboard and lighting LEDs. Then the micro-controller is powered. This requires teaching students how pins on a PDIP IC are numbered and how to tell where the first pin is. Students plug in a LED to a specified micro-controller port that provides a "blinking" signal. This serves as a sanity check to ensure the micro-controller is working. Students also connect LEDs to the four output pins of the micro-controller that represent the four DDR arrows.

After the basic setup is complete students connect the micro controller to the GUI via RS232 and a level shifter. Once connected the students observe the LEDs "dancing" in accordance with the song being played and the graphics being shown. This proved to be an unexpectedly fun step. At this point most students became convinced in their ability to complete the project.

FIGURE 6
THE INSIDE OF A STUDENT IMPLEMENTED DANCE PAD. THIS PAD HAS COILED WIRES ABOVE THE LAYER OF ALUMINUM FOIL THAT MAKE CONTACT WITH THE ALUMINUM FOIL WHEN STEPPED ON FROM ABOVE.

III. Digital Logic

Phase 3 is the most technically intensive portion of the project. In this phase the concepts of digital logic are introduced. Lecture explanations of the required digital logic are accompanied with brief intermissions to verify the validity of the lecture material on the breadboard with physical logic gates. This phase begins by explaining the NOT gate, and then moving to the AND gate, and finally the XOR gate. An understanding of these gates is essential for completing the project.

Once basic explanations are complete a worksheet is provided that guides students through a variety of logic gate arrays. Students are asked to correctly identify the output of the arrays given a set of sample inputs. This listing of sample inputs is non-exhaustive as most logic gate cascades presented have approximately five inputs. Students are asked to explain in plain language what a specific gate setup is accomplishing. Students are encouraged to implement a few of the worksheet problems on their breadboard to test their results.

At the end of this phase, students are shown the diagram in Figure 1 and asked to consider what logic is necessary to provide a correct scoring signal.

IV. Implementation and Testing

This phase allows students to design and implement their digital logic. During this phase we observe the most variance between group's abilities. Some groups immediately grasp the proper design and some groups struggle or require help.

We find that the most successful way for groups to achieve an understanding of the final design is to encourage them the think about the problem modularly. The most effective method for breaking the problem down involves asking students to design logic for a fictional DDR game in which there is only a single dance pad. This involves a single XOR and a single NOT gate. Then students are then asked to generalize to a game that has four pads.

Even with the correct design drawn, implementation on a breadboard is a problem in and of itself. Students are encouraged to debug with LEDs one gate at a time. Most errors involved some combination of accidentally misplaced wires or shorts. A picture of a completed student breadboard can be seen in Figure 7.

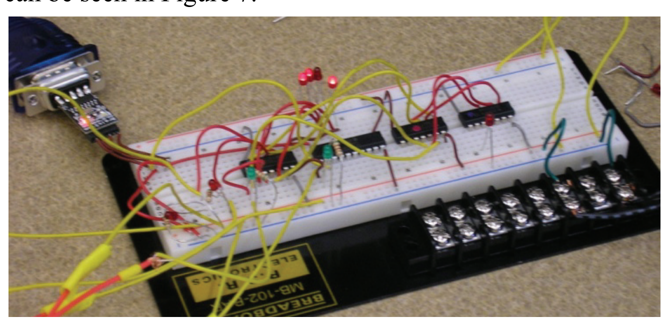

FIGURE 7
A STUDENT BREADBOARD.

This phase brought out the variety of abilities displayed by groups. Some groups finished early, decorated their pads, designed different dancing sequences, and engaged course staff in discussion about the GUI and micro-controller. Other groups needed help and could implement only the necessary functions. Current Purdue University engineering students helped struggling groups while the advanced groups were pointed to additional activities.

V. Competition

At the end of the camp a competition is held. Parents are encouraged to attend and even join in the dancing to experience what their children had created. This is a fun way for students to show off their accomplishments. While the metric for winning the competition is scoring the most points by dancing the most accurately, there is another level of engineering involved. Students that had created a very reliable pad and had taped the wires on their solder-less breadboard down had a much easier time dancing.

PROJECT RESULTS

The primary goal of this project was to introduce students to computer engineering in a fun and positive way. Students completed a survey at the beginning and ending of the camp. The end survey measured the fun of all the projects and the combination of the beginning and end surveys measured intended major and engineering major recognition before and after the camp.

The DDR project measured as the first or second most fun activity in all the camp activities, with no negative reviews. Results also indicated a 5% net shift in intended major towards computer engineering across the camps. This is particularly noteworthy because this project was held in the midst of camps that included outreach projects from many engineering disciplines.

PROJECT COST AND STAFFING

Excluding the cost of a computer for running the DDR software a complete and functional project costs less than $60. The majority of this cost comes from items that are reusable, such as the solder-less breadboard and the micro-controller. See Table I for a breakdown of this project cost.

Related to the cost is the number of staffers required to facilitate this project. We find that it is useful to have one well trained staff member for every two teams (eight students). These staff members were typically undergraduate students in electrical or computer engineering.

TABLE I
BREAKDOWN OF PROJECT COST

Item	Quantity	Price*Quantity
PDIP Logic Gates	3	$1.50
Micro-controller	1	$6.50
Misc Peripheral Hardware	1	$1.00
LEDs	10	$5.00
5V Power Supply	1	$3.00
USB Cable	1	$4.00
Jumper Wire Kit	1	$12.00
Solder-less Breadboard	1	$10.00
Wire	40 ft	$5.00
Wire Stripper	1	$3.00
Aluminum Foil, Cardboard, Crafts Supplies, etc.	N/A	$15.00
TOTAL		**$56.00**

CONCLUSION

This paper presents an outreach project for computer engineering. The project involves the use of the popular video game Dance Dance Revolution as a platform for teaching the basics of digital logic and attracting students to computer engineering. Students are introduced to basic circuit design, Boolean algebra, reliability in design practices, and the hardware-software interface. The project was implemented three times at summer camps held by Purdue University and received overwhelmingly positive reviews.

We are happy to provide interested parties with the documentation necessary to recreate this project.

ACKNOWLEDGMENT

We would like to thank the Purdue University MEP and WIEP for their support of this project as well as Hewlett Packard for donating equipment under the HP Innovations in Education grant and the National Science Foundation grant CCF-0702567 and CNS 0722212. Any opinions, findings, and conclusions are those of the authors and do not necessarily reflect the view of the sponsors.

REFERENCES

[1] Committee on the Prospering of the Global Economy of the 21'st Century An Agenda for American Science and Technology. *Rising Above the Gathering Storm Energizing and Employing America for a Brighter Economic Future*, Washington, D.C., March 2007.

[2] http://www.nsf.gov/statistics/nsf08321/.

[3] Ward, W., " Broadening participation of U.S. stem talent", *NAS Joint Meeting of Committee on Underrepresented Groups and the Expansion of the S&E Pipeline and Committee on Capitalizing the Diversity of the S&E Workforce in Industry*, June 2008.

[4] National Science Board, "Broadening Participation in Science and Engineering Research and Education", 2004, NSB-04-72.

[5] U.S. Census Bureau, "Current Population Survey, Annual Social and Economic Supplement, Racial Statistics Branch, Populaiton Division", 2004.

[6] Ibrahim, M., Al-Shara, O., "Impact of Interactive Learning on Knowledge Retention", *Springer Berlin / Heidelberg*, 2007, volume 4558.

[7] Arango, F., Aziz, E., Esche, S., Chassapis, C., "A Review of Applications of Computer Games in Education and Training", *38th ASEE/IEEE Frontiers in Education Conference*, T4A-1.