Serious Games in Computer Engineering Education

Dr. Afsaneh Minaie, Utah Valley University

Afsaneh Minaie is a Professor of Electrical and Computer Engineering at Utah Valley University. She received her B.S., M.S., and Ph.D. all in Electrical Engineering from the University of Oklahoma. Her research interests include gender issues in the academic sciences, embedded system, internet of things, wireless sensor network, and robotics.

Dr. Reza Sanati-Mehrizy, Utah Valley University

Reza Sanati-Mehrizy is a professor of Computer Science Department at Utah Valley University, Orem, Utah. He received his M.S. and Ph.D. in Computer Science from the University of Oklahoma, Norman, Oklahoma. His research focuses on diverse areas such as: D

Serious Games in Computer Engineering Education

Abstract

The purpose of a capstone design project course is to provide graduating senior students with the opportunity to demonstrate understanding of the concepts they have learned during their studies. As with many computer science and engineering programs, students of the computer engineering program at Utah Valley University (UVU) conclude their degree programs with a semester capstone design experience. The intent is for students to utilize competencies developed in the first three years of the curriculum in the solution of an embedded design problem.

Recently many of our computer engineering students have shown interest in game design and choosing to design games for their capstone project. This paper presents the details of sample game projects that the computer engineering students have done in this capstone course.

Background Information

Utah Valley University (UVU) is a comprehensive regional university with over 40,000 students charged with serving Utah County, which is the second largest county in the state. UVU has a dual mission – that of a comprehensive university offering 91bachelor's degrees and 11 master's degrees, and that of a community college offering 65 associate degrees and 44 certificates. To fill its community college mission, the institution maintains an open-enrollment policy. To facilitate academic robustness, UVU has implemented a structured enrollment policy that establishes requirements which students must meet before they can engage in all the courses of their major and provides additional access to advising and other resources. These additional preparatory course increase students' time to graduation but helps them succeed. As a large public university UVU has a very high number of low-income students – the largest proportion in the state [1]. Around 35% of students are classified as non-traditional students (age 25 or older). Nineteen percent of the students have children under the age of five [2]. UVU's students live at home or in off-campus housing, which makes it very difficult to organize activities for them. Many students do not have time to spend much time outside of class on campus, leading some to feel little connection with other students. About 80% of UVU's students will remain in their communities and pursue employment in this region [3, 4].

Engineering and Computer Science Departments

To meet one of the region's most pressing workforce needs, UVU initiated three new engineering programs in Fall 2018. The new bachelor's degree programs in Electrical Engineering, Civil Engineering, and Mechanical Engineering have joined UVU's established programs in Computer Engineering and Pre-Engineering in a new Department of Engineering. The new programs were immediately popular with students, with 300 students enrolling for Fall 2018. Currently, the new Engineering Department has more than 900 students in five programs which are housed in that department. Before forming the Engineering Department at UVU, Computer Engineering program was housed in the Computer Science department which offers a Bachelor of Science (BS) in Computer Science, Software Engineering, and Computational Data

Science. It also offers a Bachelor of Applied Science in Software Development and a Master of Computer Science. The Bachelor of Science in Computer Science program was one of the first Bachelor of Science programs implemented at UVU in 1993. The program's goal has been to provide a quality program that meets accreditation standards while providing the students with a skill set that allows them to succeed in computing careers. The Computer Science degree at UVU is accredited by Computing Accreditation Commission of the Accreditation Board for Engineering and Technology (ABET). Currently, the Computer Science Department has more than 1300 students. The Engineering programs at UVU are accredited by Engineering Accreditation Commission of the Accreditation Board for Engineering and Technology.

Computer Engineering Program's Capstone Course

Our CE capstone course is structured as a collection of independent student projects. This course is offered every semester. Usually, the students in the Computer Engineering program take this course during their last semester. Students either can come up with an embedded project independently or work on a project that is given to them by their advisors. Students write a proposal to define problems and identify solution approaches for their project and the hardware and software that is needed for their project. After several iterations, the advisor approves their project.

The faculty advisor will meet with each student individually on a weekly basis at a regularly scheduled, mutually agreeable time. At each meeting, issues associated with the project will be discussed and a written status report will be provided to the advisor. Students will keep a daily journal/work log detailing the work that was done, how much time was spent that day, and any technical details that might be needed for later reference. Faculty advisor keeps notes of each meeting as well as action items to be accomplished for the next meeting. Reviewing the log sheet from the previous meeting is a great way for the faculty to prepare for the upcoming one and provides further evidence to the student of the meeting's importance.

At the end of the semester, the students turn in a final written report and a final presentation which is evaluated by several faculty members from the department.

Introduction

To engage today's students who are always on mobile devices, new models and approaches to teaching and learning are needed from the educators. There are two types of games: fun games and serious games. One approach which has been successful in education is using serious games. Serious games are different from fun games in that they have a serious purpose that is combined with the entertainment aspect [5]. Game based learning is increasingly being used in educational settings and is widely predicted to become mainstream in the next few years [6-8]. The use of computer games for educational purposes is an important aspect that can improve the learning process. In a study conducted by Mateos, they have successfully implemented the ShopC serious game and tested it with students. The adaptation of the game followed the motivation, learning, and gaming design principles. Their evaluation validates the design of computer-based games regarding the three dimensions (motivation, learning, and gaming) and suggests a positive effect in the learning process [9]. In another practical study by Callaghan into the use of serious games for teaching, they developed the Circuit Warz game, and the

approach taken potentially offers a new engaging and highly interactive way to teach engineering related material [8]. Most students are highly accustomed to and very skillful in playing computer games. A remarkable feature of serious games is their power to motivate students to learn. Research has indicated that computer games can achieve high learning results in areas where interdisciplinary knowledge is necessary and where skills such as problem solving, critical thinking, group communication, and decision making are of high importance [10].

Games have been gaining tremendous interest as tools for teaching and learning in recent years. The merits of game-based learning (GBL) include supporting effective learning [11], enhancing higher order thinking [12], increasing problem-solving skills, and promoting engagement [13].

There are many arguments in favor of the use of games in education. Games are a very important learning tool. Playing develops capabilities and aptitude that contribute to the formation of personality. The game is a way of learning because of the increase of motivation. Different games have been demonstrated to motivate students to learn. The use of games for educational purposes is an important aspect that can improve the learning process.

Recently many of our computer engineering students have shown interest in game design and choosing to design serious games for their capstone project. The following presents the details of sample game projects that our computer engineering students have done in their capstone course.

Sample Project 1: Augmented Virtuality

Augmented virtuality (AV) is the convergence of virtual reality (VR) (where digital content overlays the physical environment) and augmented reality (AR) (where a user is immersed in a wholly digital environment) which combines the elements of both technologies to create a unique and enhanced user experience [14]. Augmented virtuality is the hybrid reality framework that incorporates real world elements into the virtual world [14]. The key characteristic of the AV is as follows:

1. **"Virtual dominance:** Augmented Virtuality takes place primarily in a virtual world, where the user interacts with computer-generated content.
2. **Real-world integration:** It integrates real-world elements, objects, or environments into the virtual environment to create a sense of presence and familiarity.
3. **Interaction and immersion:** AV offers a level of immersion and interaction that goes beyond traditional augmented reality, allowing users to manipulate and engage with the integrated real-world elements.
4. **Contextual connectivity:** It leverages sensors, cameras, and other data sources to understand the context and spatial relationship between virtual and real-world elements." [14]

Augmented virtuality has many applications such as training and simulations, gaming and entertainment, remote collaboration, architecture and design, and education [14].

In this project, a team of two computer engineering students worked together to design a game that combines both physical and digital elements to provide a positive experience for the user. This was a semester long project for these two students. This game was designed to be a racing game where the user sees the view of the camera on the top of the physical car and controls the car from a computer. The system diagram for this design is shown in Figure 1.

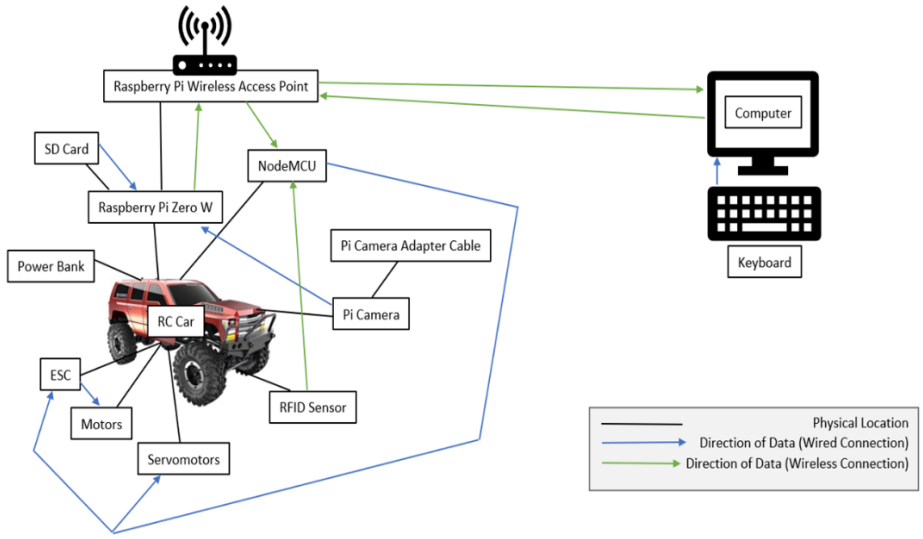

Figure 1: System Diagram [15]

The hardware components that were used in this design were Raspberry Pi Zero W, Pi Camera, RFID sensors, NodeMCU (combines a node and MCU), and an RC car. Figure 2 shows the hardware components on the remote RC car.

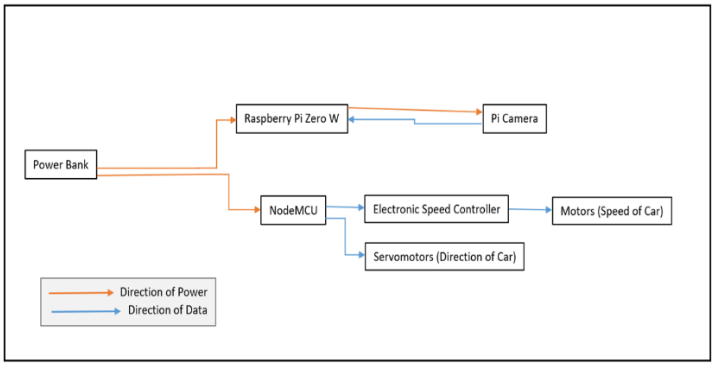

Figure 2: Components on the Remote-Controlled RC Car [15]

Figure 3 shows the remote-controlled RC car. The player of the game would be using a computer and viewing on the monitor of the computer the video stream captured by the Pi Camera which is connected on the front of the racing car. The player then can see what is in front of the car and control the car from the computer.

Figure 3: The Remote-Controlled RC Car [15]

Figure 4 shows the software hardware interconnection.

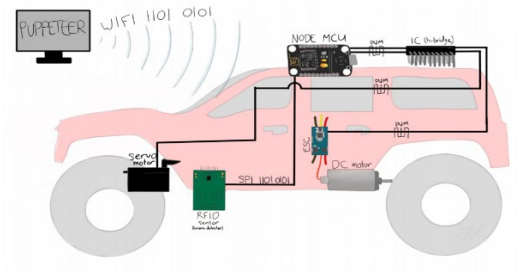

Figure 4: Code-Hardware Integration [15]

This was a successful project, and students were able to control the remote RC car over WiFi and the video stream was fast enough to control the car from video. When the RC car went over an RFID sensor/tag it disabled the controls for one second. This project was funded by NSF S-STEM Scholarship program at UVU.

Sample Project 2: Snake Game: A Verilog Implementation

In this project, a team of two computer engineering students worked together to design a Snake Game video game on a FPGA (Field Programmable Gate Array) using Verilog language. The game was uploaded to a tinyFPGA-BX board. The users controlled the game using simple push buttons that are wired to the board. A Video Graphics Array (VGA) display was used so that the game could be transported and played on modern displays.

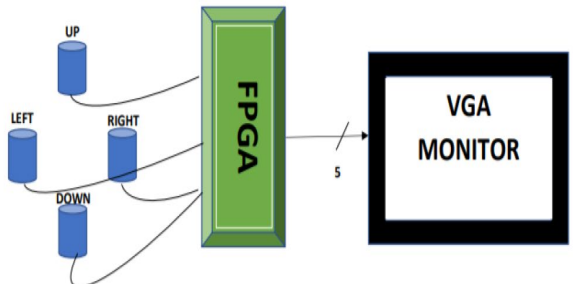

Figure 5: Top Level View [16]

Specialized hardware was built to run this game. The hardware interconnection is shown in Figure 6.

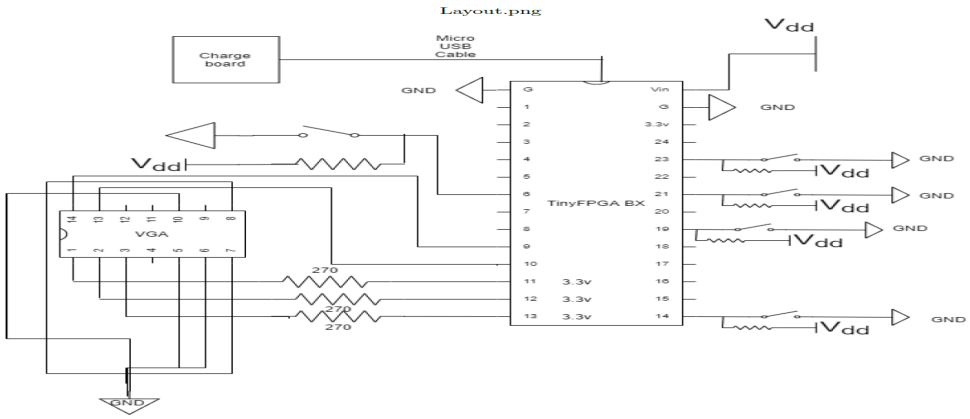

Figure 6: System Diagram for the Snake Video Game [16]

A handheld console that can be used to play the snake game. This handheld console was designed, and 3D printed to encompass the TinyFPGA BX module as well as other internal parts of the circuit.

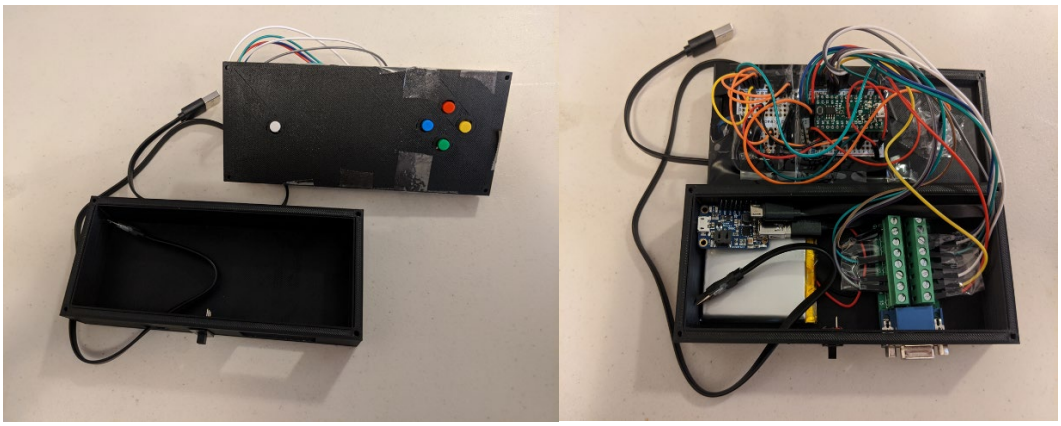

Figure 7: Handheld Console with Hardware [16]

Controls to play the game are located on the top of the console as shown in Figure 7. The console is portable and handheld. They used an open-source toolchain called APIO that did not have a built-in wave simulator viewer like other IDE's and used GTKWave to view the content of the registers. Another IDE tool that was used is called an 8BitWorkshop. This IDE is a free online tool. Overall, the project was successful, and students were able to design and prototype a handheld console for playing a snake video game which is implemented on an FPGA.

Sample Project 3 – MASS ExtINCTION – Implemented on a 16-bit FPGA Soft Processor

In this project, a team of two computer engineering students designed and implemented MASS ExtINCTION game from the ground up on a soft processor. MASS ExtINCTION is a simple 16-bit video game that was inspired by early 1980s 'retro'-style video games. The principle of MASS ExtINCTION game is simple, there is a falling square, and the user must control a character and not be crushed by the square by moving left and right. If the square hits the bottom of the screen without hitting the player, then the score is incremented. If the square hits a player, the score is reset. Additionally, every two points the speed of the falling square is increased by one to make the game progressively more difficult.

Students implemented a 16-bit RISC CPU as a soft-processor on an FPGA. They designed and built an external graphics controller using an Arduino-based microcontroller and an SPI TFT display. A program was created that takes advantage of both the processor and the graphics controller to implement a retro 16-bit style video game. The instruction set architecture used was adapted from the *16-bit RISC Processor* designed in Verilog by Van Loi Le [18]. The system diagram is shown in Figure 8. Their complete system design includes a two-button controller, soft processor, graphics controller and display.

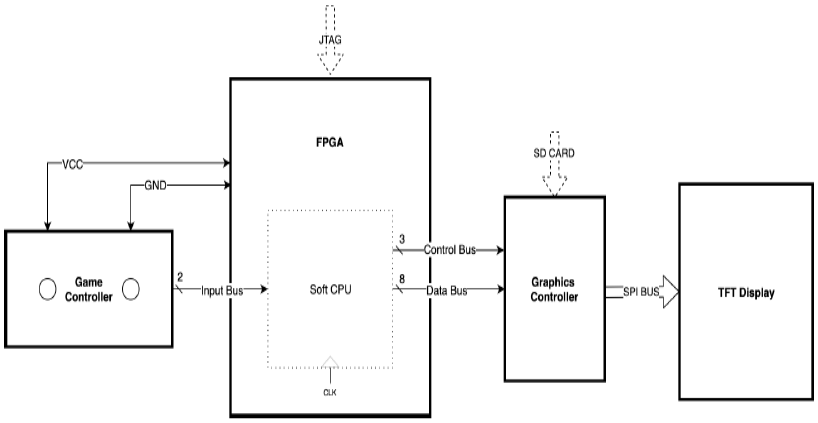

Figure 8: System Diagram [17]

A Zybo Z7-10 Development Board which is shown in Figure 9 was used for this design. The final processor was implemented on the ZYBO Z7-10 FPGA. The added I/O ports were used to output data to the graphics controller and receive user input from a custom two button controller.

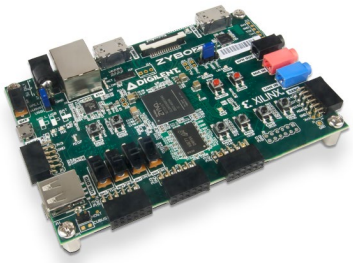

Figure 9: Zybo Z7-10 Development Board [17]

Figure 11 shows the code development process for this design. The game was prototyped in Unity and implemented in machine code in their CPU architecture. The user sends input to the CPU via a simple two button controller that sets a VDD or GND signal on one of the first two bits to the input register based on if a button is pressed, and which button is pressed. The game looks for the rising edge of the signal to trigger the player movement, which adds difficulty because the player must repeatedly press the button to get the character to move out of the way of the falling object. Their character is a dinosaur sprite (Figure 10) with a simple two state animation. The bitmaps for the two states are loaded by the graphics controller from an SD card.

Figure 10: Bitmap of the game hero “Stubby” [17]

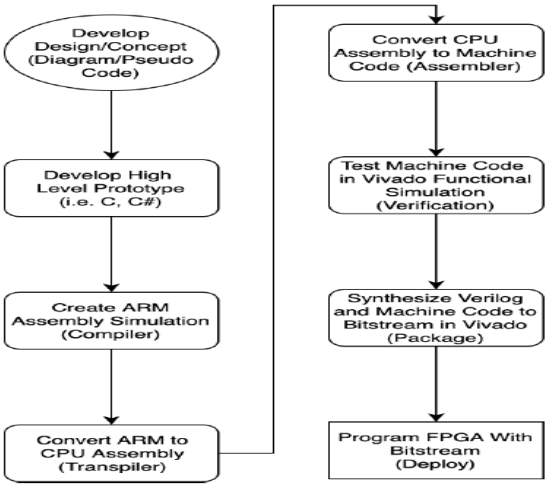

Figure 11: Code Development Process [17]

The flow chart for the game is shown in Figure 12. The game checks for a collision, then checks if the player has scored, then it checks if the user is pressing a button to move the player.

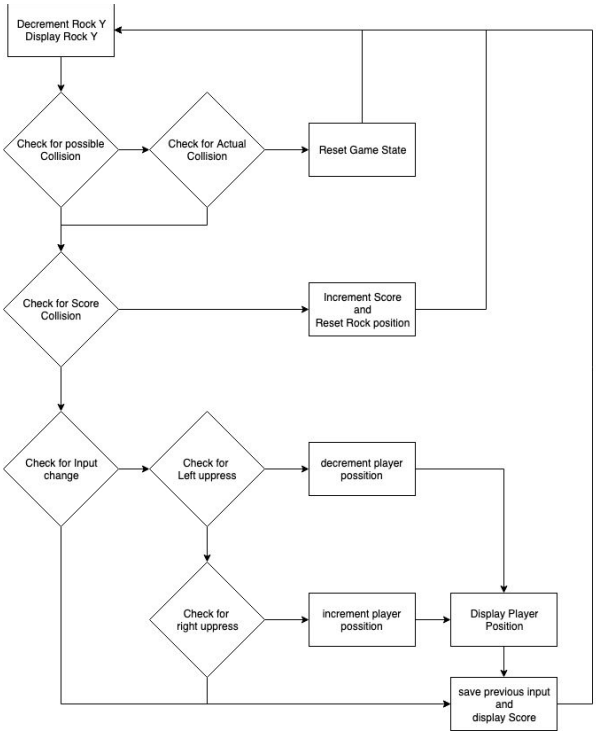

Figure 12: Game Flow Chart [17]

The graphics controller was constructed to simply read data from the input pins and then render the sprites on screen with that data. The serial communication built into the Arduino microcontroller was used to read data about the state of the graphics controller, and to send simulated inputs to the controller to generate graphical output.

This project was successful, and the student commented that "working on this project gave us a more holistic understanding of the integration of CPU architectures, hardware, and software development. We became acquainted with various I/O implementations, addressing modes, and ISAs, and communication protocols. Furthermore, performing all portions of software development by hand gave us an in-depth perspective of the operations performed by the compiler when going from a high-level programming language to assembly, to machine code."

This project was funded by LEAP NSF S-STEM Scholarship program at UVU.

Summary and Conclusion

Nowadays, most students have grown up with digital technology such as computers, the internet, video games, and mobile phones. These students are fundamentally different from previous generations in the way they learn. Currently, students prefer learning experiences that are digital, connected, experiential, immediate, and social. It appears that students prefer learning by doing rather than learning by listening and often they choose to study in groups. Most students are highly accustomed to and very skillful in playing games. A remarkable feature of games is their power to motivate. Research has indicated that computer games can achieve high learning

results in areas where interdisciplinary knowledge is necessary and where skills such as problem solving, critical thinking, group communication, and decision making are of high importance.

This paper discussed recent capstone projects in game design where three teams worked on designing various games. Our computer engineering capstone course is structured as a collection of independent student projects. Students find this course both challenging and rewarding as they are required to design, build, and troubleshoot a fully functional embedded project. These projects give the students the chance to use their technical expertise and knowledge gained during years of study. Students work very hard to have a working project by the end of the semester. These projects provide students many opportunities to engage in self-directed learning. They develop the ability to debug, seek and find information they need, and the ability to understand and reverse-engineer poorly written documentation. The students' feedback and their final project presentation indicate that they have pride in their project accomplishments and have gained confidence in their engineering abilities.

References:

1. Annual Report on the State of Poverty in Utah, 2014, Community Action Partnership of Utah, http://caputah.org/images/poverty-reports-full/2014_Annual_Report_on_Poverty_.compressed_1.pdf , access on 1-2-2020.
2. Annual Report on the State of Poverty in Utah, 2012, Community Action Partnership of Utah.
3. Information and statistics provided by the UVU Office of Institutional Research and Information – IRI.
4. NSF Proposal document, "Strengthening Outcomes for Students in Computer Science and Engineering through Leadership, Engagement, Academic Mentoring, and Preparation (LEAP)", August 2013.
5. Effelsberg, Wolfgang, et. al., "Serious Games 2014: International Workshop on Serious Games", Proceedings of the 22nd ACM international conference on Multimedia, Pages 1265-1266.
6. Kosmadoudi, Zoe, et. al., "Engineering Design using Game-Enhanced CAD: The Potential to augment the User Experience with Game Elements", Computer-Aided Design, vol.45, no.3, pp. 777-795, March 2013.
7. Aziz, E. S., et.al., "Review of the State of the Art in Virtual Learning Environments based on Multi-Player Computer Games", Computer Education Journal, vol. 20, no. 1, pp. 22 – 35, 2010.
8. Callaghan, Michael, et. al., "Mapping Learning and Game Mechanics for Serious Games Analysis in Engineering Education", IEEE Transaction on Engineering Topics in Computing, vol. 5, no. 1, March 2017.
9. Mateos, Maria, et.al., "Design and Evaluation of a Computer Based Game for Education", IEEE, 2016.
10. Pivec, M. and Kearney, P., "Games for Learning and Learning from Games", An International Journal of Computing and Informatics Journal, Vol. 31, No. 4, pp. 419-423, 2007.
11. Connolly, T.M., Boyle, E.A., MacArthur, E., Hainey, T., & Boyle, J.M., "A Systematic literature review of empirical evidence on computer games and serious games", Computers and Education, 59, 661-686, 2012.
12. Sanchez, J. and Olivares, R., "Problem solving and collaboration using mobile serious games", Computers and Education, 57, 1943-1952, 2011.
13. Akcaoglu, M., and Koehler, M.J., "Cognitive outcomes from game-design and learning (GDL) after-school program, Computers and Education, 75, 72-81, 2014.
14. "What is Augmented Virtuality?", https://www.educative.io/answers/ , accessed on 2-2-2024.
15. Brown, Brock, and Linda Siebert, "Augmented Virtuality", ECE 4800, Final Reports, Fall 2020.
16. Riding, Nathan, and Dylan Maxfield, "Snake Game: A Verilog Implementation", ECE 4800, Final Reports, Fall 2019.
17. Herrmann and Brandyn Burbank, "Mass ExtINCTION: A Video Game Implemented on a 16-bit Soft Processor", Final Reports, Fall 2019.
18. Van Loi Le, "Verilog-Code-for-16-bit-Risc-Processor", https://www.fpga4student.com/2017/04/verilog-code-for-16-bit-risc-processor.html, Accessed on 2-4-2024.