CedarLogic 2.0 Update

Colin Broberg
Cedarville University, cbroberg@cedarville.edu

Julian Pernia
Cedarville University, jpernia@cedarville.edu

Tyler Drake
Cedarville University, tylerdrake@cedarville.edu

James Von Eiff
Cedarville University, jamesvoneiff@cedarville.edu

Follow this and additional works at: http://digitalcommons.cedarville.edu/research_scholarship_symposium

Part of the Other Computer Sciences Commons, and the Software Engineering Commons
CedarLogic is the Cedarville University’s student-developed, digital logic simulator. Engineering and Computer Science students use this software for several of their classes. Our primary goal for this update is adding black boxes, buses, and cross-platform compatibility. Our additional improvements in user-friendliness and functionality will give students an improved CedarLogic experience.

**What is CedarLogic?**

Cedarville Engineering Students use CedarLogic for designing and simulating low-level computer hardware. The program has many gates and chips available for implementing all kinds of circuits and simple computers. Often the first class students will be introduced to CedarLogic in Digital Logic Design (ELE450), in which they design a digital watch, a full adder, and a 4x4 sorting stack. CpE and CS students will also design small computers called Mojo Machines in Computer Architecture.

---

**Buses**

Buses allow users to connect large buses of wires between bus-ends as a single wire, for logical and visual simplicity.

**Black Box**

Black boxes, or hierarchy, was the most time-consuming element of this project. While we managed to implement this feature to have basic functionality, we had plans to make black boxes more robust and versatile. Given more time, this is something we would love to improve.

**Buses**

Buses, while not too difficult to implement, were one of the most requested features for CedarLogic. Their implementation was well-received by students using the program. For additional functionality, we would like to add bus To/From gates to allow for further consolidation of wires.

**Stability**

Due to its nature of being a student-made program, CedarLogic has never had the stability of a commercially available product. Our first task for this project was to improve stability of the program to enhance user experience.

**Cross-platform Compatibility**

Cross-platform compatibility was one of the project requirements was to port CedarLogic to Linux to provide more compatibility and support for users. While we currently have a build working for Linux, it has some quirks we are hoping to work out.

**UI Improvements**

UI Improvements

The visual improvements to CedarLogic were not a project requirement to us, but we believe these additions dramatically improve the CedarLogic experience. Future teams working on CedarLogic could further improve it by updating to a newer version of OpenGL.

**Git**

Git – git was used as the version control system for this project.

**Visual Studio**

Visual Studio – Visual Studio is the IDE we are using to develop this project.

**WIX**

WIX – The WIX toolset is a set of tools that can be used to create an installer for the Windows installation engine. It is easily integrated into projects and Visual Studio supports it.

**OpenGL**

OpenGL – OpenGL is a 2D and 3D graphics API that is compatible with almost all platforms.