

George Engel, William Kingsley, Derek Jacobson

CSCI 460 Operating Systems

12 December 2019

Project Technical Report

Project Outline

The purpose of the project is to research and explore hardware emulation in operating systems by creating a fully functioning CHIP-8 emulator.

Deliverables Accomplished

The deliverables that we accomplished for this project are as follows:

- Basic functioning emulator
- Emulator bootloader
- Basic UI and system information displays
- Fully functioning CHIP-8 emulator

Additional deliverables added to project for presentation purposes:

- Graphic representations of Memory, input, and OpCode translation
- Ability to Halt the process the emulator is running and step through a tick at a time
- Ability to change foreground and background color of the game screen
- Speed slider, for altering delay between CPU cycles

Demonstration Accomplishments:

Our project demonstration provided a brief history of CHIP-8 technology and early computers. After which, we proceeded by providing a demonstration of the functioning CHIP-8 emulator. We discussed how the emulator converted CHIP-8 OpCodes into equivalent C# functions for use on a modern Windows operating system environment. Likewise, the key structural operating system components such as the stack, program counter, and memory layout were explained in detail using the UI as an illustration tool. Original CHIP-8 games such as Space Invaders, Pong, and Tron were also played on the emulator as part of the demonstration to showcase successful execution of CHIP-8 programs by the emulator.

Implementation Details

The project was built using C#, JSON, XML, Windows Forms, and Microsoft Visual Studio.

To start, since we knew we would be building it out as a functioning GUI application, we decided on Visual Studio's Windows Forms as a base for our project, as it provides prebuilt templates of the MVC design pattern we could build off of for our project.

So with that as our base, the 'Emulation' portion, simply needs to read in a specified chip8 file and translate it's instructions to C# functions, run the emulation cycle, and output the resulting boolean matrix to a screen element used in the GUI.

The actual OpCodes are processed into their respective C# functions by sorting through the first four bytes one at a time, and running the equivalent C# function of what the OpCode's intended function as detailed on the Chip8 Wikipedia.

However, Chip8 originally didn't run on the fast processors we now have today, and it is from an era which didn't account for drastically faster CPU speeds, and so the higher clock rate of modern CPUs would break most ROMs, making them nigh unusable. As a solution to this problem, we introduced a controlled delay into our emulation cycle, such that you could expect the game to have a certain amount of 'ticks' (OpCode processing cycles) per second. This delay allows us to experience chip8 applications on modern machines as they were in their own time. Additionally, adding a multiplier to this value associated with a slider bar in the GUI allows us to control the speed of the application, as a nice additional feature.

As for how it receives keyboard input, that is passed in from the GUI, into the Chip8 emulation portion's pressed buttons stack with a key value translated via a predetermined dictionary that determines which chip8 input keys get mapped to which keys the user pressed on their keyboard. Due to the nature of how input was taken from the GUI, it also made it convenient to display which keys were currently pressed in the GUI, so that was implemented as well.

The memory display GUI elements use an observer like software pattern, which only updates the RAM, stack, register, and counter displays when values are changed. This was necessary to prevent the application from slowing down the emulation process itself. (The game loop)

Results:

A functioning, configurable Chip8 emulator capable of running any chip8 game.

Conclusion:

Building a chip8 emulator is a fantastic programming project for anyone looking to learn more about computer architecture, or wanting an introduction to the world of emulation.

Github repository for the project: <https://github.com/Geo3ngel/CHIP-8-Emulator>