SNO

SUPER NINTENDO ONLINE

Team Members

Michael Kelly
mkelly01@my.fit.edu

Keith Johnson
kjohns07@my.fit.edu

Eric Wells
wellse@my.fit.edu

Faculty Sponsor

Dr. William H. Allen
wallen@cs.fit.edu

Project Goals

SNO is an SNES emulator written in Java as an applet to be embedded onto a website. It is a program that emulates the hardware functions of a Super Nintendo, including the CPU, graphics, sound, input, and memory. The main goals of the project are:

- Embed SNO on a web page
- Load ROM images from a user's hard drive or from the website server
- Run SNES games at a playable rate
- Accurate video and sound emulation

Motivation

SNO provides a very interesting technical challenge, as it requires both an intimate knowledge of the SNES hardware as well as considerable programming skill to implement the hardware functionality in Java. We will have to take into consideration performance issues, as well as compatibility issues; several SNES games depend on hardware "quirks" that need to be implemented.

Also, we're gamers. This is our dream project.

Milestones

Milestone 1 – September 27th

- Requirements Document
- Design Document
- Test Plan

Milestone 2 – October 25th

- Partial CPU Emulation
- Basic applet frontend
- Ability to read ROM from client
- Memory implementation

Milestone 3 – November 29th

- Improved CPU Emulation
- Basic Video Implementation
- Configuration options for GUI

Milestone 1 – Task Matrix

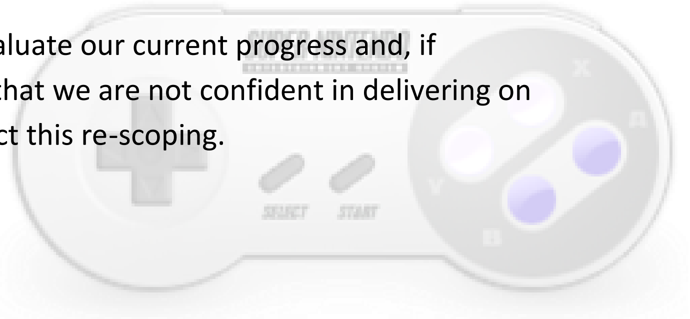
Task	Michael Kelly	Keith Johnson	Eric Wells
Requirements Document	33%	33%	33%
Design Document	33%	33%	33%
Test Plan	33%	33%	33%

Scope

Due to the difficulty of implementing a full emulator, our project's scope is currently tentative; it is difficult to accurately estimate the effort needed to complete the project without having previous experience in emulation.

Our current scope includes three major areas: CPU/memory, video, and sound. For the first semester, our goal is a nearly complete CPU and memory implementation, and a partial video implementation. For the second semester, our goal is to finish off the video and CPU implementation and to complete the sound implementation.

At the beginning of the second semester, we will evaluate our current progress and, if necessary, re-scope the project to remove features that we are not confident in delivering on time. Our second semester plan document will reflect this re-scoping.



Possible Issues

There are several major issues that have been brought up in planning that we'd like to acknowledge:

- **CPU Instruction Set:** The instruction set for the 65816 CPU (the model used in the SNES) includes 94 different operations, many of which have multiple addressing modes. Most of these are simple memory management commands, and should be relatively simple to implement and test once a base memory implementation is made.
- **Graphics Intensive:** The SNES picture processing unit (PPU) runs on the same clock as the CPU, which runs at a maximum of 3.58 MHz. We are confident that a modern computer can run Java fast enough to achieve these speeds easily; other SNES emulators have been running games at full speed for more than two decades now.
- **Java Garbage Collection:** Although we are using Java as our programming language, we will not be coding in a "Java style." Because we wish to be close to the actual hardware implementation of the SNES, we will not be creating many objects and instead will rely on relatively static data that will not be collected. This, along with the relatively low speed of the SNES processor, makes performance hits due to garbage collection a small risk.

Faculty Sponsor Approval

I have discussed with the team and approve this project plan. I will evaluate the progress and assign a grade for each of the three milestones.

Signature: ___________________________________ Date: ______________________