OHIO NORTHERN UNIVERSITY
Department of Electrical & Computer Engineering and Computer Science

NES HD

Porting the Nintendo Entertainment System to HDMI Displays

submitted by
Paul Sorensen & Samuel Roth
on
April 18, 2016

advised by
Dr. Ziad Youssfi

Acknowledgements

Northern Digital would like to extend gratitude to a number of different people, without whom our work would not be possible. First and foremost, to Ziad Youssfi, for helping define our vision and for guiding us throughout the entire academic year as our advisor. The faculty and staff of the TJ Smull College of Engineering, for providing feedback and additional insight along the way. Brad Hummel, for printing (and reprinting) our *award-winning* poster to display at the IEEE Student Activities Conference and the College of Engineering Project Showcase.

— Paul Sorensen and Samuel Roth

Table of Contents

List of Figures

List of Tables

Problem

The Nintendo Entertainment System (NES) is a 8-bit video gaming console developed by Nintendo and released internationally in the early 1980's. It captured the imagination of an entire generation by pushing the limits of what was possible with the available display technologies. While there remains a large user base for the NES, it relies on outdated RGB video technology through composite and RF connectors that are not supported by the displays currently on the market.

Figure 1. Nintendo Entertainment System as released in the United States

This presents a problem to the users of the NES, as they must retain outdated displays in order to continue playing their original, classic games. Consequently, there exists a need for a system which is compatible with more recent display standards, such as HDMI, that can play the original Nintendo game cartridges.

Objective Statement

There are a number of existing methods to play the original Nintendo games on modern display technology. One popular method is through the use of emulators - pieces of software that use the ROM (read-only memory) from NES games to recreate the system. There are multiple issues associated with this approach.

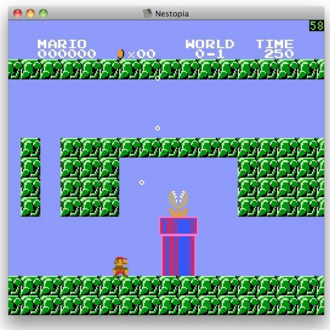

Figure 2. Nintendo Emulator Running on Mac OS

One primary concern is the legality of the system - these ROM files are easily obtainable online, so people can use them despite not having paid for them. Next, many families and enthusiasts still have the original Nintendo cartridges, and seek a solution that gets as close to the original gaming experience as possible. Another issue is the lag involved. Although it may seem small, software interrupts which handle the controller input are not as quick to respond as a pure hardware alternative. This can be frustrating for users of emulators.

In this document, we describe the process of building a recreation of the NES with a field-programmable gate array (FPGA). We sought to retain all functionality of the original NES, but also to produce an HDMI output signal that would allow the system to be used on modern displays. We also document the process of designing and printing an aesthetically pleasing enclosure. Finally, we discuss our acquired experience with large-scale system development.

Objective Tree

The project was broken down into three primary components. First and foremost, the functionality of the system was pivotal to its success. Also considered were the enhancements to input and output, as well as the system casing and documentation.

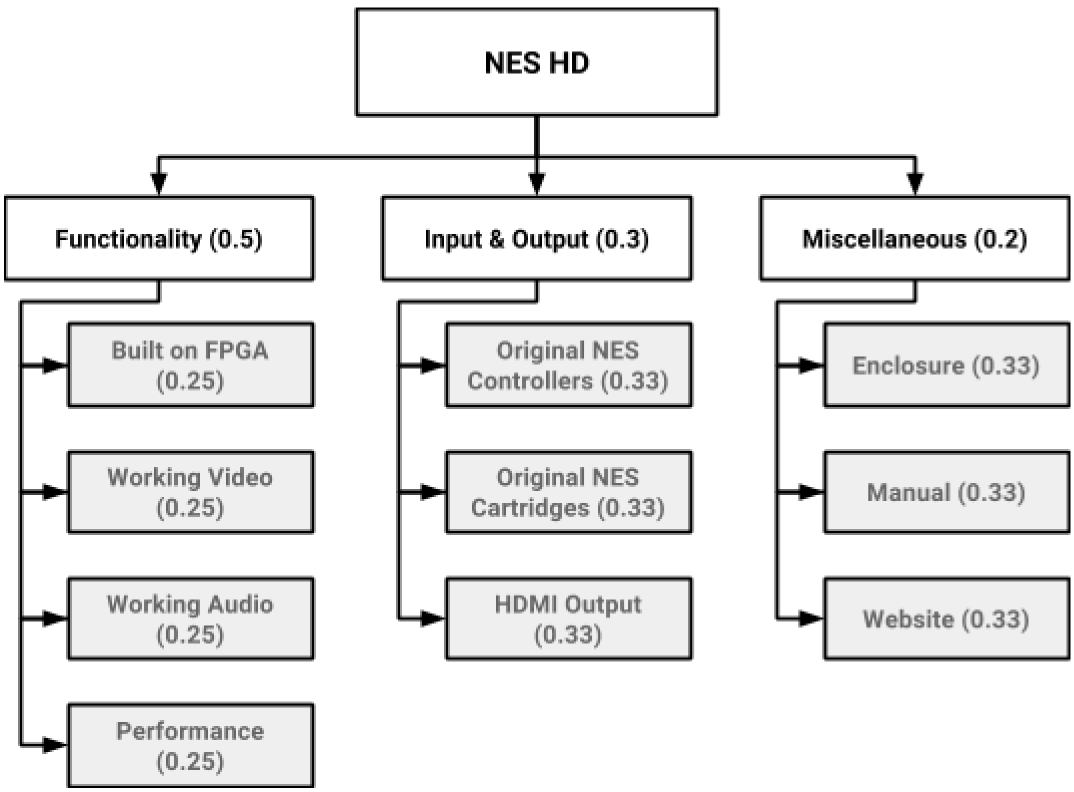

Figure 3. Objective Tree Representing the Goals of the Project and Their Weights

Constraints

Business Concerns

The team began the year by outlining a variety of different requirements that would need to be met — they are reproduced below. The satisfaction of these requirements would ensure that the final product appealed to as many potential customers as possible.

- The system should be able to use the original Nintendo cartridges.
- The system should be able to use the original Nintendo controllers.
- The system should connect to a display with an HDMI cable.
- The original game audio and video should be working without distortion.
- The system should be reasonably portable.
- The system should not require any additional hardware to work.
- The system should be enclosed in a user-friendly shell.
- The system requires user-friendly documentation, available in print or online.

Engineering Concerns

A number of constraints, across varying categories, were acknowledged before beginning the development of this system. Similar to the marketing requirements, satisfying these constraints would ensure that the resulting system is a success.

Regarding Performance

- The system should be able to perform better than similar products that rely on emulator software (e.g., less lag).
- The system should be usable on modern display technologies through the use of the HDMI interface.

Regarding Functionality

- The system should be able to read and interpret data from the original NES game cartridges.
- The system should be able to read and interpret data from the original NES game controllers.
- The output of the system — an HDMI signal — should include both the audio and video for the Nintendo game, with minimal distortion to the original data.

Regarding Economic Viability

- The system should be as affordable as the original NES.
- The system should be more affordable than other, emulator-based solutions, depending on how it performs in relation to those other solutions.

Regarding Energy Consumption

- The system should be able to operate fully when connected to home outlets, both in the U.S. and abroad.

Regarding Health & Safety

- The system should not leave the electronic components exposed, so as to alleviate the risk of electric shock.
- The system should not have any parts on its exterior that could potentially be removed and consumed by small children.
- The system should never be at risk of overheating or causing any sort of heat-related damage to its environment.

Regarding Legality

- First and foremost, the system should not violate any intellectual property holdings; particularly, those of Nintendo.
- The system should be able to be mass produced and sold within the United States, as well as abroad.
- Any parts of the project that utilize publicly available code (i.e. open source projects) should be appropriately documented.

Regarding Maintainability

- The system should be enclosed in a user-friendly case that allows for efficient transportation and storage.
- The system should be able to be reset in the event that one or more components cause(s) the system as a whole to fail.

Regarding Operation

- The system should be compatible with any hardware — game cartridges, controllers, and so on — that is compatible with the original NES.
- The system should function in a logically similar fashion to the NES.
- Ports and buttons should be properly identified.

Regarding Reliability

- The system should be able to function properly over a long period of time.
- Documentation should be made available (in print and on the web) that helps users ensure the long-term reliability of the system.
- The developers of the system should make their contact information available, in the event that the system is not behaving as expected.

Regarding Availability

- The system should be able to be mass produced and distributed within the United States as well as abroad.
- The documentation for the system and its development process should be available for those who are interested in learning more about digital electronics.
- The system should be available for users (and potential customers) to test in a hands-on fashion.

Regarding Social & Cultural Aspects

- The system should strive to recreate the original NES experience as accurately as possible.
- The system should be able to take the original experience and bring it to modern display technologies, in an effort to catch the attention of the next generation of gamers.
- Hypothetically, if made available internationally, the system should be able to provided tailored documentation and resources for different languages.

Design

Alternative Solutions

There are a number of existing methods to play original NES games on modern displays. The team believed that original NES hardware implementation on a FPGA will yield the most success. This section continues by elaborating on a few of the most popular alternatives, and why they do not stack up to the FPGA implementation.

Original NES with Video Conversion

One convenient way to use the NES on modern displays is by attaching an adapter to the output, converting the signal appropriately. While this can be a relatively cheap solution, the performance will suffer significantly. This alternative is only reasonable if a decent amount of money is put into a high-quality converter. Additionally, this method relies on the original NES hardware, which will be old and sluggish, affecting performance in its own right.

Modified NES

Another popular method involves the modification of the NES hardware in some way. As described earlier, the Analogue NT utilizes the original NES chips (obtained from previously-manufactured systems), placing them on a custom printed circuit board [1]. The absurd price aside, the Analogue NT suffers from the same issue that plagues the video-converted NES systems — it relies on old chips. Not only do these chips come burdened with age; they are also available in a harshly limited quantity. This is likely the cause of the $500+ price of this alternative, which is unacceptable, in our opinion.

Emulated NES

There are a number of different solutions that involve emulation in one way or another. The emulation can either be based in software or hardware. Software emulation involves users running a program on their PC which simulates the NES. This tends to inhibit the traditional console gaming experience, and is also legally questionable (as the ROM's used have typically been downloaded illegally from the Internet). Hardware emulation is favored because of its similarities to the traditional NES experience. At that point, the team must decide whether to use an FPGA or an ASIC. An ASIC would likely be more efficient, but would prevent the team from continually developing new features, bug fixes, and other improvements [2]. An FPGA would make the opposite trade-off [3].

Decision Matrix

One helpful way to visualize the different possible solutions is to present them using a decision matrix. Using the alternatives above, along with the objective tree described previously, the team prepared a decision matrix to determine which solution was most viable and likely to succeed. This decision matrix is reproduced below.

Table 1. Decision Matrix for NES Implementation

Criteria	Weight Factor	Modified NES HD Output		Software Emulator		Hardware Emulator FPGA		Hardware Emulator ASIC	
		Rating	Score	Rating	Score	Rating	Score	Rating	Score
Cost	25	0.2	5	1.0	25	0.7	17.5	1.0	25
Power	5	1.0	5	0.5	2.5	1.0	5	1.0	5
Performance	40	1.0	40	0.5	20	0.8	32	0.9	36
Testability	30	0.8	24	1.0	30	1.0	30	0.3	9
Total	100		74		77.5		**84.5**		75

Solution Overview

After deciding to emulate the NES hardware components on an FPGA, a lot of time went into deciding which kind of development platform to use. Due to its popularity, and the team's familiarity with it in the classroom setting, it was agreed that a Spartan 6 would be the best FPGA for this project. A number of different boards were considered, but for initial development, the team utilized the Xilinx Atlys board which was readily-available in Ohio Northern University's Digital Design Lab. A picture of the board is available in Figure 4.

This was deemed a great way to get started because the Atlys board has HDMI output functionality. However, there were two significant concerns: first, the lack of I/O pins made it inappropriate for use when it came to integrating NES controller- and cartridge-interfacing functionality. Next, the sheer size of the board, and the number of unused components, meant that it would be silly to use it as a reference when designing an enclosure.

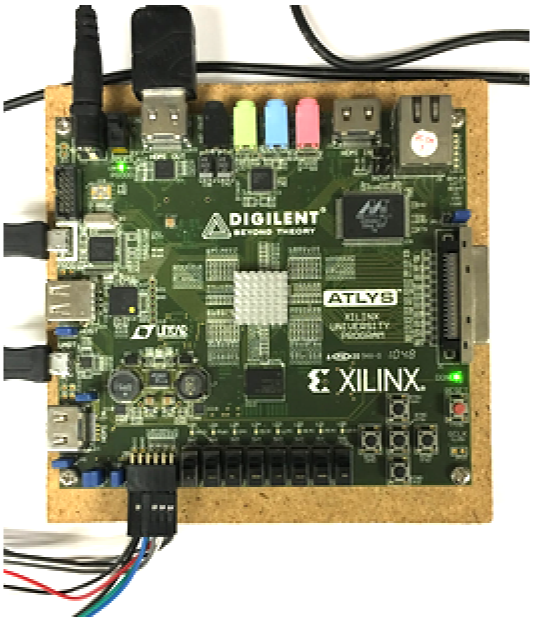

Figure 4. Xilinx Atlys Development Board

To address those concerns, we investigated other FPGA development boards. We came across the miniSpartan6+, which was crowdsourced and developed by Scarab Hardware. Not only did this board feature enough pins to handle controller and cartridge interfacing, but it also provided HDMI and SD card support. While the HDMI output was vital to the success of our project, the SD card slot would allow us to test games before the cartridge reading functionality was implemented. The miniSpartan6+ board is shown below, in Figure 5.

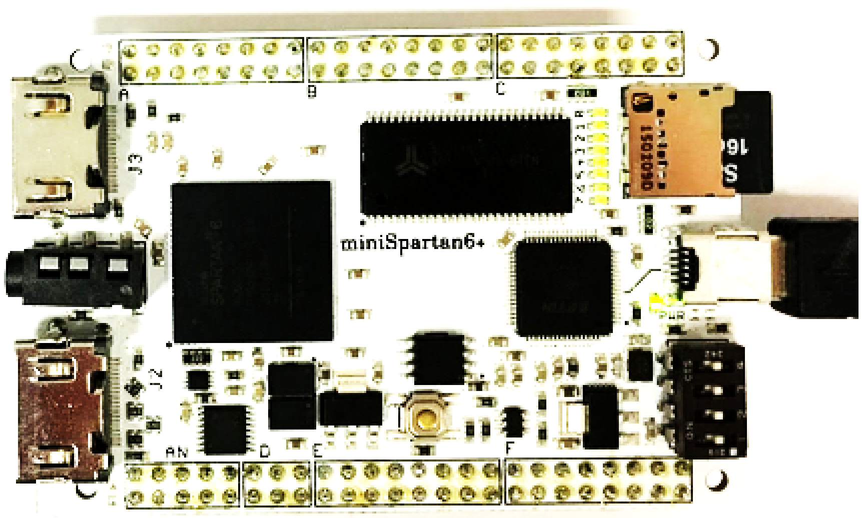

Figure 5. Scarab Hardware miniSpartan6+ Development Board

Integration

A very large portion of the Spring 2016 semester was dedicated to the process of integrating the various components of the NES HD system. While a substantial portion of the system was on the FPGA itself, there were multiple additional components that needed to communicate with that FPGA and, more specifically, the hardware programmed onto it.

NES Controllers

The controllers for the NES are obviously a fundamental component for the system, for without them, the users of our system have no way of playing the games they own. The integration of controller input ended up being pushed back due to issues with HDMI encoding, which are described below. While we are able to read data pertaining to pressed buttons, we have yet to integrate this functionality with the NES HD system. The reading of the controller data was accomplished by opening the cable which connects the controller to the system, as shown in Figure 6. Information regarding the purpose for each wire was available online.

Figure 6. Original NES Controller with Internal Wires Exposed

NES Cartridges

The team also believes the ability to read original NES cartridges is paramount to the success of our system. While we have the hardware components required to read the ROM data, we have not yet had the opportunity to implement that functionality. When testing the system's ability to actually play the NES games, we used ROM files that were transferred from our workstation to the FPGA using UART. These tests were successful for multiple games.

The games which would present an issue for our system at present are those which feature additional circuitry within the game cartridges, which gained popularity after the original release of the NES. More original (and more popular) games, such as Super Mario Bros. and Donkey Kong, did not have this issue, and were successfully loaded onto our system.

Audio and Video Output

The initial goal of the project was to be able to produce an HDMI output signal which would contain both the video and audio for the NES HD system. This output signal would be based on required information from the game cartridge ROM, as well as the controllers connected via the I/O pins.

The process of producing a converted output signal has proven to be quite challenging. The team started by producing a VGA output because we were able to find more reliable documentation on that standard. This phase of testing the video output was largely successful. One noteworthy element of the test is that we only allocated three bits for color, whereas the original NES output used 8. This meant that our output looked washed out, as shown in Figure 7.

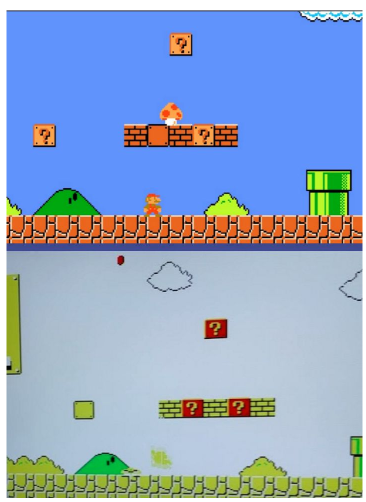

Figure 7. Colors on Original NES (top) vs. Colors on FPGA Test (bottom)

The real challenge with the output came with the integration of the HDMI standard. It turns out that HDMI has quite a complicated encoding process that included unforeseen considerations, such as the Digital Content Protection component. We have been able to produce an HDMI output with simple colored bar tests, but have not yet tested HDMI output with NES game data.

Verification

For the purpose of evaluating success at this point in time, we revisit our testing plan from our original formal proposal. For each question, we address our progress in **bold text**, if progress has been made. Problems we are currently tackling are highlighted.

Functionality

- Does the system accurately reproduce the video from the game ROM?
  - **Yes, we have done multiple tests with colored bars and other basic shapes. We have also read game ROMs through UART and have been successful in producing that video data.**
- Does the system accurately reproduce the audio from the game ROM?
  - **Yes. We were able to produce a stereo audio output, and are in the process of integrating the game audio into the HDMI output.**
- Does the system perform well, compared to other alternatives?
  - **Yes — we have not noticed any of the performance issues that are common in software-based emulator applications.**

Input & Output

- Is the system compatible with the original NES game cartridges?
  - No — at present, game data is read from our workstation using UART.
- Is the system compatible with the original NES controllers?
  - **Yes — the controller input is read and processed by the system correctly.**
- Does the system produce a proper HDMI output?
  - **Yes, an HDMI output is produced.**

Miscellaneous

- Is the system enclosure durable enough to protect its interior components?
    - **Yes — we have 3D printed a durable enclosure, although an even stronger material would likely be used in production.**
- Is the documentation (print and web) understandable and helpful?
    - **Yes!**

Budget Overview

For development purposes, the team needed a wide variety of supplies. Most of these items were already available through Ohio Northern University, so our actual budgetary requirements were not accurately reflected in our original analysis. Also note that the original analysis reflected the development of a single unit, which served as a prototype. Costs for production is still expected to be significantly smaller.

Table 2. Original NES Implementation Cost Analysis

Component	Quantity	Expected Cost (per item)	Expected Cost (all items)	Actual Cost
FPGA	1	$300	$300	$110
NES Controllers*	2	$10	$20	$0
NES Game Cartridges*	5	$5	$25	$0
Displays*	2	$100	$100	$0
Enclosure Materials	N/A (Will vary)	$100	$100	$0
Miscellaneous Components	N/A (Will vary)	$50	$50	$40
Total			**$595**	**$150**

As the above table shows, our team was able to remain significantly under our budget. Specifically, we spent $150 and have a remaining $445 dollars left of our budget as allocated by the College of Engineering. This money is unlikely to be required for the remainder of our project, largely due to the fact that we already had access to original NES controllers, game cartridges, and displays to test our system with. We were fortunate to have the material for the enclosure freely available to us as well.

Schedule

The timeframe for our project was initially laid out using Microsoft Project. The diagram indicating the flow of our work throughout the entire academic year is replicated below.

Table 3. Original Project Schedule

Task Name	Duration	Start	Finish	Predecessors	Resource Names
Problem Identification	5 days	Mon 8/24/15	Fri 8/28/15		
Research Phase	5 days	Mon 8/31/15	Fri 9/4/15	1	
Set up weekly meeting schedule	0 days	Wed 9/2/15	Wed 9/2/15		
Requirements Specification	5 days	Mon 9/7/15	Fri 9/11/15	2	
Contribution Criteria	0 days	Wed 9/9/15	Wed 9/9/15		
Concept Generation	5 days	Mon 9/14/15	Fri 9/18/15	4	
Problem Identification	0 days	Wed 9/16/15	Wed 9/16/15		
Design Phase	35 days	Mon 9/21/15	Fri 11/6/15	6	
Peer-Peer Evaluation	0 days	Wed 9/23/15	Wed 9/23/15		
Formal Written Proposal	0 days	Wed 9/30/15	Wed 9/30/15		
Proposal Oral Presentation	0 days	Wed 10/7/15	Wed 10/7/15		
Peer-Peer Evaluation	0 days	Wed 10/21/15	Wed 10/21/15		
Prototyping and Construction Phase	75 days	Mon 11/9/15	Fri 2/19/16	8	
Start Ordering Parts	0 days	Wed 11/11/15	Wed 11/11/15		
Progress Written Report	0 days	Wed 11/11/15	Wed 11/11/15		
Oral Presentation 2	0 days	Wed 11/18/15	Wed 11/18/15		
Ethics Assignment	0 days	Wed 12/2/15	Wed 12/2/15		
Prepared Abstract for conference	0 days	Wed 12/9/15	Wed 12/9/15		
Peer-Peer Evaluation	0 days	Wed 12/9/15	Wed 12/9/15		
System Integration	30 days	Mon 2/22/16	Fri 4/1/16	13	
Test Phase	20 days	Mon 4/4/16	Fri 4/29/16	20	
Delivery and Acceptance	5 days	Mon 5/2/16	Fri 5/6/16	21	

For the most part, our team was able to follow the schedule fairly accurately. While we had no issue meeting assignment deadlines, such as the peer evaluations and the oral presentations, primary scheduling issues came with establishing long-term development plans that spanned the majority of the Fall and Spring semesters. Specifically, consider the *Design*, *System Integration,* and *Test phases.*

Our team was responsible for first developing an understanding of the NES architecture, HDMI, and other platforms, and we didn't really incorporate the amount of time that required into our original schedule. As a result, the three phases mentioned previously were a largely-convoluted process involving lots of trial and error.

Conclusion

> *There are a number of different ways to play the original Nintendo games today, due to the various efforts of our generation. However, the most popular options available today fail to capture the true NES experience — sitting around a television with family and/or friends. Enjoying games together.*

Northern Digital has spent the past year working toward an *affordable* and *functional* system that brings the fun back to playing games amongst your family and friends. We have made great strides in HDMI support, as well as support for original Nintendo Entertainment System controllers and game cartridges. It is our hope, with the remaining time in this semester, that we can move even closer to our original goal, and present a working prototype before our graduation.

Again, we thank Dr. Youssfi and the faculty and staff of Ohio Northern University for supporting this learning experience.

References

1. Design. Retrieved September 30, 2015, from http://www.analogueinteractive.com/products/analogue-nt-information

2. FPGA vs. ASIC. Retrieved September 30, 2015, from http://www.xilinx.com/fpga/asic.htm

3. Ludde's FPGA NES. Retrieved September 30, 2015, from http://fpganes.blogspot.com/2013/01/luddes-fpga-nes.html

4. Nintendo Entertainment System Expired Patents Do Not Mean Expired Protection. Retrieved September 30, 2015, from http://www.gamasutra.com/view/feature/130867/nintendo_ entertainment_system__.php

5. The Psychology of Video Game Nostalgia. (2013, June). Retrieved September 30, 2015, from http://www.psychologyofgames.com/2013/11/the-psychology-of-video-game- nostalgia

6. Sieber, Jonathan. Implementing The Nintendo Entertainment System On A FPGA. 1st ed. 2013. Web. 28 Sept. 2015.