Harris, Wendy P.

NOTE:

This file is intended to collect all documentation and progress into a single file.

As project is still in development, this file is currently undergoing changes throughout the development.  
  
Once project is completed, this file will be converted to a pdf.

Enjoy 😊

EMULATION MACHINE

Documentation

Table of Contents

[Introduction 2](#_Toc180843354)

[Planning 2](#_Toc180843355)

[Purpose 2](#_Toc180843356)

[Current System Specifications 2](#_Toc180843357)

[Hardware Requirements 2](#_Toc180843358)

[Chosen Hardware 3](#_Toc180843359)

[Software Choices 3](#_Toc180843360)

[Operating System Choice 3](#_Toc180843361)

[Plans for Software: Scripts, ROM Management, and Emulators 4](#_Toc180843362)

[Summary 6](#_Toc180843363)

# Introduction

Emulation has long been a way for enthusiasts to relive the gaming experiences of the past, offering the ability to play classic games on modern hardware without the need for outdated consoles. This project aims to create a robust emulation machine capable of running games from a variety of consoles, ranging from retro systems like the Super Nintendo to more modern consoles like the Xbox 360 and PlayStation 3.

As a lifelong gamer and tech enthusiast, I’ve always been fascinated by the blend of hardware and software, and this project allows me to combine both interests. The goal of this emulation machine is to not only serve as a powerful platform for gaming but also to showcase a range of choices in hardware, software, and peripherals, ensuring the best experience for each console's emulation.

This documentation will serve as a record of my process, from planning and design through to execution, highlighting the decisions made and the challenges faced along the way.

# Planning

## Purpose

The purpose of this project is to transform my current PC into a fully functional, home theatre emulation machine: combining modern gaming with retro emulation seamlessly. Leveraging Steam’s Big Picture mode, the goal is to create a console-like experience that allows me to launch both Steam games and emulated games from a unified interface. This will not only make it easier to access and play a wide variety of games but will also allow me to fuse my passions for emulation, hardware, and software development.

I aim to create a cohesive gaming environment that feels intuitive, fluid, and powerful. Whether playing the latest PC titles or revisiting classic games from past consoles, the system will offer an all-in-one solution for gaming enthusiasts.

## Current System Specifications

* CPU: Ryzen 5 3600
* GPU: GTX 1070 Ti
* PSU: EVGA GQ 650W
* Motherboard: ASUS ROG Strix B450-F Gaming
* Case: Corsair 275R ATX
* Storage: 250GB M.2 SSD, 2TB HDD

## Hardware Requirements

* Upgrade Path: Plan to upgrade the CPU and GPU from Ryzen 5 3600 and GTX 1070 Ti to higher options.
* Performance Goal: Current setup supports all emulators and modern games at 1080p 60fps, medium to high settings. Future upgrades aim for 4K 60fps on ultra settings.
* Storage: Plan to replace current storage with a single 2TB Gen 4 NVMe SSD, potentially adding another Gen 3 NVMe SSD if more capacity is needed.
* PCI Generation: Ensure compatibility with the latest PCI generation for the GPU.
* Size Constraints: Use a micro ATX form factor to fit into the entertainment system, prioritizing aesthetics and compact design.
* Remote Power Switch: A power switch connected to a PCIe slot, controlled remotely, allowing the PC to be powered on from a distance.

### Chosen Hardware

* Motherboard: MSI B550M Mortar Micro ATX
* Case: ONZ PC Gaming Case Micro ATX Mid Tower - Dual Chamber, Compact Edition V2
* Power Switch: CERRXIAN PCIe Wireless Computer Remote Control Switch, 2.4G Wireless Remote Computer Power/Reset Switch

## Software Choices

### Operating System Choice

For this emulation machine, Windows 10 has been chosen initially, with plans to upgrade to Windows 11 following hardware upgrades. Here’s why Windows was selected over other options.

#### Original Considerations

When selecting an OS for the emulation machine, several key factors were evaluated:

* **Compatibility**: The OS needed to support a broad range of emulators and modern games, from retro consoles to more demanding systems like PlayStation 3 and of course regular steam games.
* **Performance**: Since the system will run multiple emulators and Steam Big Picture mode, stable and efficient performance is essential.
* **Customization**: The OS should allow sufficient customization to minimize the standard desktop experience and provide a console-like interface.
* **Ease of Use**: A streamlined, user-friendly interface is crucial to facilitate easy navigation between Steam games and emulated games without requiring frequent interaction with the desktop.

##### Operating Systems Considered

1. **Windows 10/11** *(Chosen OS)*:
   * **Advantages**:
     + Complete compatibility with all emulators.
     + Seamless integration with **Steam** and **Steam Big Picture mode**, allowing for a unified gaming experience.
     + Extensive customization options through registry edits, scripting, and startup settings.
   * **Drawbacks**:
     + Windows boots into windows explorer before steam big picture. This could limit the console like experience.
     + Windows updates can reset some custom settings, which requires additional scripting for automation and restoration.
2. **Linux-based Systems (ChimeraOS, HoloISO)**:
   * **Advantages**:
     + Linux-based gaming operating systems like **ChimeraOS** and **HoloISO** are built to boot directly into Steam’s interface, providing a highly console-like experience.
     + Good compatibility with many emulators (however not all) via native Linux applications and support for RetroArch.
     + Open-source flexibility for high customization.
   * **Drawbacks**:
     + Limited compatibility with some emulators, especially high-performance ones like **RPCS3** for PS3.
     + More complex setup for hardware drivers, especially for controllers and certain graphics cards, compared to Windows.
     + Fewer options for native support for non-Steam applications, like emulators, requiring workarounds to integrate emulators.
     + Some steam games may not be compatible
     + As it replicates the SteamOS interface used for the Steam Deck, there are numerous references to the Steam Deck in the UI that aren’t relevant for this machine. This could disrupt the immersive, console-like experience intended for the emulation machine, potentially reminding the user that they are navigating a PC environment rather than a dedicated gaming console.

#### Why Windows?

After considering various options, **Windows** was chosen for the following reasons:

1. **Broad Emulator Compatibility**: Windows has extensive support for a wide range of emulators, including **RPCS3** for PS3 and **Dolphin** for GameCube/Wii. Its compatibility ensures strong performance across both retro and modern systems.
2. **Ease of Use and Familiarity**: With more experience in Windows, I can start the project immediately without the steep learning curve Linux sometimes requires. Windows also tends to be less "finicky" than Linux in terms of driver and software compatibility, which means fewer unexpected issues.
3. **Existing Setup**: Since the current system is already running Windows, I can jump right into configuring emulators and software without reinstalling or configuring a new OS, saving time and simplifying the setup process.
4. **Customization and Scripting Potential**: While Windows doesn’t boot directly into Steam Big Picture mode, it offers robust scripting options through **PowerShell** and **Batch scripting**. Writing scripts to automate startup, bypass the desktop, and create a streamlined interface will not only be a fun challenge but will also help create a truly console-like experience.
5. **Future Linux Consideration**: If Windows updates repeatedly undo my custom scripts and workarounds, switching to a Linux-based system could become a viable alternative. In that case, I may create a Linux version of the emulation machine to achieve a more persistent setup without as much interference from OS updates.

### Plans for Software: Scripts, ROM Management, and Emulators

To create a seamless and console-like experience on Windows, I have several software plans I am excited to deploy: 1 **Automatic Emulator Launch with Steam ROM Manager**:

* **Purpose**: Create individual Steam shortcuts within the steam library for each emulated game, allowing them to be launched directly from Steam Big Picture mode without interacting with emulator interfaces.
* **Implementation**: **Steam ROM Manager** will be configured to generate game entries for each emulator, displaying them as native Steam games in Big Picture mode.

1. **Direct Boot into Steam Big Picture**:
   * **Purpose**: Bypass the Windows login screen and desktop environment, creating the illusion of a direct boot into Steam Big Picture mode.
   * **Implementation**: A PowerShell script will modify the Windows registry to replace Windows Explorer with Steam as the default shell, making Steam Big Picture the primary interface at startup. This setup will effectively skip the standard desktop environment, providing a console-like experience. 3 **Bloatware Removal**:
   * **Purpose**: To optimize performance and reduce unnecessary clutter, I will remove default Windows applications and services not needed for gaming, such as OneDrive and Explorer.
   * **Implementation**: A custom PowerShell script will automate the removal of these unwanted applications and services, reducing background processes and creating a cleaner, more streamlined environment.
2. **Restore Boot Configuration After Windows Updates**:
   * **Purpose**: Windows updates may reset custom configurations, undoing the direct boot setup.
   * **Implementation**: A secondary script will monitor and reapply the direct boot configuration after each Windows update. This script, likely a combination of PowerShell and a startup task, will check if Steam is still set as the default shell and reconfigure it if necessary.

#### Emulator Options

To manage emulation, I am considering two main approaches:

1. **All-in-One Solution with RetroArch**:
   * **Description**: RetroArch is a front end that consolidates multiple emulators ("cores") into one platform. It offers a convenient, centralized solution for managing emulation across various systems.
   * **Pros**: RetroArch simplifies setup and management by housing multiple emulators in one application, making it easier to switch between consoles.
   * **Potential Drawbacks**: If RetroArch doesn’t allow games to be launched directly from Steam without displaying the RetroArch interface, or if customizing each core is too limited, it may disrupt the seamless experience I aim for.
2. **Standalone Emulators for Each Console**:
   * **Description**: Using individual emulators for each console provides greater flexibility and customization, as each emulator can be fine-tuned for optimal performance.
   * **Emulators Planned**:
     + **Dolphin** for GameCube and Wii
     + **PCSX2** for PlayStation 2
     + **RPCS3** for PlayStation 3
     + **mGBA** for GameBoy Advance
     + ect ect
   * **Pros**: Standalone emulators generally offer more detailed configuration options, which may be beneficial for fine-tuning performance and visuals.
   * **Potential Drawbacks**: This setup may require additional management and scripts for seamless integration into Steam Big Picture mode.

**Final Decision:** I will start by testing RetroArch due to its convenience. If it meets the requirements of launching games directly from Steam and allows adequate customization of each core, it will be used as the primary solution. Otherwise, I will opt for individual emulators for each system to maintain flexibility and control over each console's emulation.

### Summary

Choosing Windows provides the flexibility, compatibility, and customization needed to build a powerful, all-in-one gaming system that combines the best of modern and retro gaming. By integrating PowerShell and Bash scripting, this setup will deliver a smooth, console-like experience without unnecessary interactions with the Windows desktop environment.

This setup will create a cohesive and user-friendly emulation machine that meets all the initial requirements and provides a streamlined experience for gaming enthusiasts.