

NES模擬器

專題編號:113-CSIE-S022

執行期限:112年第1學期至113年第1學期

指導教授:劉傳銘

專題參與人員:110820057 蕭嘉甫

110820060 林嘉威

一、摘要

此專題使用C++開發NES模擬器，我們將模擬並實現CPU、PPU(Picture Processing Unit)和APU(Audio Processing Unit)等元件，以及對NES遊戲的載入和執行。透過模擬器實現多款經典遊戲的模擬，並驗證功能及性能。

關鍵詞:NES模擬器、遊戲執行、硬體結構

二、緣由與目的

因為對於NES遊戲機的興趣以及對其硬體運作原理的探索。目的是用C++開發模擬器，以模擬硬體的行為並理解它們之間的相互作用。除此之外，還要能夠讓模擬器載入並執行遊戲，這也涉及到圖形渲染和音效撥放，最後還需要確保它能夠執行多款遊戲，除了能夠讓玩家在電腦上重新體驗NES遊戲的樂趣外，也學習並理解此模擬器的運作原理。

三、研究範圍

(一)CPU模擬:模擬6502處理器，這是NES的核心處理單元，需要確保指令的正確執行以及時序的精確性。

(二)PPU模擬:模擬NES的圖形處理單元，負責將遊戲畫面渲染到畫面上。

(三)APU模擬:模擬音頻處理單元，負責遊戲中的音效和音樂。這部分要求精確的音頻同步和再現經典音效。

(四)遊戲載入與執行:載入NES遊戲檔案，確保模擬器能夠正確解析並執行遊戲。

四、使用技術方法

(一)C++:高效的記憶體管理和硬體層級操控能力，非常適合模擬器開發需求。

(二)SFML(Simple and Fast Multimedia Library):跨平台的多媒體庫用來處理圖形渲染、音效撥放、以及用戶輸入等。還為模擬器提供了易用的API，能夠輕鬆的處理遊戲的多媒體需求。

(三)CMake:簡化了編譯流程，並且使得專案能夠輕鬆移植到不同平台。它幫助我們管理複雜的編譯，確保專案在多種開發環境中能夠穩定運行。

五、架構流程

(一)首先實現6502處理器的指令集模擬，並透過網路上提供的測試數據驗證CPU的正確性。

(二)在CPU模擬完成後，進入圖形處理器的模擬。PPU負責將遊戲畫面渲染到畫面上。我們逐步實現了背景的繪製，並使遊戲能夠顯示出初始畫面。

(三)模擬NES的遊戲控制器，使模擬器能夠接收並處理用戶的按鍵輸入，確保遊戲中的角色能夠正確響應玩家操作。

(四)最後，我們實現了音效處理單元的模擬，重現了遊戲中的音樂和音效，這讓遊戲體驗更加完整。

六、實驗結果

我們已經成功模擬了NES的核心硬體，並能夠運行一些經典遊戲。模擬器能夠準確的重現遊戲畫面和音效，且在性能表現上達到了預期。我們使用了一些遊戲來驗證模擬器的穩定性和流暢性，結果顯示在大部分情況下遊玩可以接近原版的

速度運行，但在某些複雜場景下，仍有少許性能下降的情況。

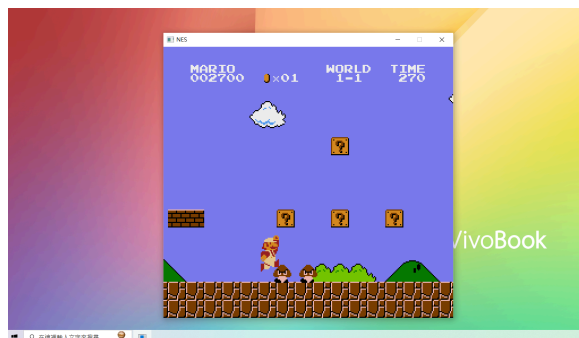


圖1. 遊玩畫面

七、結論

本專案成功實現了NES模擬器的基本功能，並支援多款遊戲的順暢運行。然而，未來我們仍然可以針對性能進行進一步優化，並擴展對更多遊戲的支援。此外，我們還可以加入更多的硬體模擬功能，進一步提升模擬器的精確性。

八、參考文獻

- [1] Nesdev Wiki
https://www.nesdev.org/wiki/Nesdev_Wiki
- [2] Blargg apu reference
https://www.nesdev.org/apu_ref.txt
- [3] An overview of nes rendering
https://austinmorlan.com/posts/nes_rendering_overview/