

PiGBoy

Game Boy Emulator

Group 15

Aleksandar Kostic

James Mitsumasu

Sean Aspinall

Sean Maas

Zhen Wang

SECTION 1. Description

The purpose of this project is to emulate the Nintendo Game Boy system on the Raspberry Pi. This will involve recreating hardware components of the console - the CPU, memory, and graphics - entirely using software. The emulator will execute original Game Boy programs (mainly video games) running on the Raspberry Pi hardware.

SECTION 2. Features

Required Features

1. Emulation: the program needs to be able to emulate the game and its operations on the Raspberry Pi; otherwise, it is not an emulator.
2. Display/Graphics: a display/graphical output is needed to see the emulated game on the Raspberry Pi; without this, it wouldn't be usable.
3. Interactivity: interactivity is needed to allow the user to communicate and use the emulator effectively; without this, the game could not be played.

Optional Features

1. Audio: audio would provide a more accurate emulation and better user experience, but it is not absolutely necessary for the emulator.
2. Save Functionality: being able to save within the emulator would allow the user to retain progress between sessions.

Wish-List Features

1. Website: hosting our emulator online would bring better portability and access to the system. Furthermore, websites are great for promotion.
2. Compatibility: better compatibility would mean more game titles can be emulated.
3. Save States: on top of the built-in save functionality, save states would allow saving and restoring from any point, which increases accessibility.
4. Online Functionality: emulating the Game Boy link cable via the Internet will allow emulation of multiplayer and data transfer between Game Boy systems. However, most Game Boy games had limited or no link cable functionality.

SECTION 3. Risks

1. Familiarity
 - a. The majority of group members have no experience working with emulators and have never worked in translating low-level assembly to high-level languages.
 - b. Strategies to deal with this include referring to online documentation on emulators and the translation of assembly to high-level programming languages. Furthermore, group members who are struggling can contact other group members who have prior experience working with emulators.

2. Complexity

- a. The nature of the project is complex: hardware must be emulated using software. Thus far in our undergraduate careers, we have not worked with hardware which is a disadvantage. Furthermore, as low-level assembly must be translated to high-level C++, there are many operation codes that must be decoded and precisely replicated by the program.
- b. Strategies to deal with this include referring to online documentation on the Game Boy hardware, its operation codes, and general system specifications. Furthermore, time management is essential and must be prioritized to implement all operation codes.

3. Compatibility

- a. There are plenty of games published on Game Boy; the emulator we create may not have all of them run well due to accuracy issues.
- b. Strategies to deal with this include conducting hardware tests so we can ensure all operation codes work, indicating compatibility with most games. Furthermore, we could focus on major titles in the case that there are issues with compatibility after conducting hardware tests.

4. Performance

- a. The emulator may be able to load the game but if the code is not properly optimized it might run slow and sluggish, making for an unpleasant experience.
- b. Strategies to deal with this include writing better code with lower time complexity, reducing the number of function calls, and reducing redundant code.

SECTION 4. Other Notes

For the UI, we will be using some sort of external library, such as GLUT or wxWidgets, to handle display drawing and button inputs.

Sean Maas has experience with emulators and will be the go-to person for questions related to documentation.

SECTION 5. References

<http://marc.rawer.de/Gameboy/Docs/GBCPUman.pdf>

<http://www.codeslinger.co.uk/pages/projects/gameboy.html>

<https://github.com/gbdev/awesome-gbdev>

<https://www.wxwidgets.org/>