GAME-BOY EMULATOR USING RASBERRY-PICO

A MINI-PROJECT REPORT

21CSE253T – Internet of Things

Submitted by

Akshat Mittal [RA2211003010790]
Ahhsanul Hoque [RA2211003010929]
Yugam Shah [RA2211003010796]
Kriti Gupta [RA2211003010797]
Rithu Nandana [RA2211003010766]
Swadesh Kumar Giri [RA2211003010754]
Yashaswini Reddy [RA2211003010765]

Under the guidance of

Dr.P.Selvaraj

Associate Professor, Department of Computer Science and Engineering

in partial fulfillment for the award of the degreeof

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE & ENGINEERING

of

FACULTY OF ENGINEERING AND TECHNOLOGY

S.R.M. Nagar, Kattankulathur, Chengalpattu District

MAY 2024

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY

(Under Section 3 of UGC Act, 1956)

BONAFIDE CERTIFICATE

Certified that Mini project report titled **"GAME-BOY EMULATOR USING RASBERRY-PICO"** is the bona fide work of **Akshat Mittal [RA2211003010790], Ahhsanul Hoque [RA2211003010929], Yugam Shah [RA2211003010796], Kriti Gupta [RA2211003010797] Rithu Nandana [RA2211003010766], Swadesh Kumar Giri [RA2211003010754], Yashaswini Reddy [RA2211003010765]** who carried out the minor project under my supervision. Certified further, that to the best of my knowledge, the work reported herein does not form any other project report or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

SIGNATURE

Dr. P.Selvaraj
Assistant Professor
Department of Computer Science and Technology

ABSTRACT

In an era where technology relentlessly marches forward, there exists a timeless allure to the classics, particularly in the realm of gaming. This project endeavors to merge the nostalgic charm of retro gaming with the cutting-edge capabilities of IoT through the development of a Game Boy emulator powered by the Raspberry Pi Pico microcontroller. By harnessing the raw potential and programmability of the Raspberry Pi Pico, we embark on a journey to recreate the magic of the iconic Game Boy console, breathing new life into cherished titles from yesteryears.

The emulator's creation is not merely an exercise in software development but a testament to innovation and adaptation. Through meticulous hardware interfacing and software emulation, we delve into the intricate workings of both the Game Boy's hardware and software, deciphering their secrets and translating them into a modern IoT context. Challenges abound, from accurately replicating the Game Boy's display and sound to ensuring seamless user interaction, but each hurdle surmounted brings us closer to our goal.

Beyond the technical intricacies lies a deeper narrative of convergence. The Raspberry Pi Pico, originally conceived for IoT applications, finds itself repurposed as a vessel for gaming nostalgia. This juxtaposition of past and future, analog and digital, represents not just a technical achievement but a cultural bridge between generations of gamers. The emulator becomes a portal, allowing enthusiasts to revisit cherished memories while introducing a new audience to the joys of retro gaming.

TABLE OF CONTENT

INTRODUCTION

Pico-GB is a 3d printed Gameboy emulator handheld gaming console for Raspberry Pi Pico that resembles to the original Nintendo Game Boy released in 1989.

This homemade Game Boy emulator is a fun and creative way to bring back old games on the beloved classic Game Boy console. With just a few basic components and a little bit of tinkering and programming you can build your own handheld gaming console capable of playing original Game Boy titles.

All you need is a Raspberry Pi Pico, an ILI9225 LCD display, an SD card and a couple of micro-push buttons to get started With a few hours and some basic soldering skills, you can build yourself a beautiful handheld Game Boy emulator to play all your favorite classic games. So, if you're feeling nostalgic for some good old fashion Game Boy gaming then building a homemade emulator handheld is a great and inexpensive way to do just that!

Pico-GB is based on the Raspberry Pi Pico microcontroller. The case and buttons are 3D printed. The screen is a 2.2-inch LCD with a resolution of 220×176 pixels and 65K colors. There are 8 buttons: 4 for the DPAD + 4 action buttons (A, B, select, start). The buttons are 6x6x6mm micro push buttons.

LITERATURE SURVEY

1. Different IoT devices**.** "Raspberry Pi Pico: A Microcontroller Revolution": This paper provides an overview of the Raspberry Pi Pico microcontroller's features, capabilities, and applications in various domains, including IoT, robotics, and education. It serves as a foundational understanding of the hardware platform chosen for the Game Boy emulator project.

2. "Game Boy Emulation Techniques": This survey explores the emulation techniques and challenges involved in accurately replicating the hardware and software behaviour of the Game Boy console. It discusses various emulation approaches, such as cycle-accurate emulation, high-level emulation, and hybrid techniques, providing insights into the complexities of emulating retro gaming systems.

3. "IoT Integration in Retro Gaming": This study investigates the integration of Internet of Things (IoT) technologies with retro gaming consoles, focusing on enhancing user experience, connectivity, and accessibility. It examines projects that utilize IoT platforms for game distribution, multiplayer gaming, and remote monitoring, offering inspiration for incorporating IoT features into the Game Boy emulator.

4. "Low-Level Programming for Game Boy Emulation": This research delves into the low-level programming techniques required for accurately emulating the hardware components of the Game Boy, including the CPU, memory, graphics, and sound subsystems. It explores how developers overcome hardware constraints and optimize performance in emulated environments, providing valuable insights for implementing the emulator on the Raspberry Pi Pico.

5. "User Interface Design for Emulated Gaming Systems": This

literature review explores best practices and design considerations for creating user interfaces (UIs) tailored to emulated gaming experiences. It discusses UI elements, control schemes, accessibility features, and customization options that enhance gameplay immersion and usability, informing the design decisions for the Game Boy emulator's user interface on IoT devices.

6. "Cross-Platform Development with Raspberry Pi Pico": This paper examines methodologies and tools for cross-platform development with the Raspberry Pi Pico microcontroller, enabling software developers to write code that runs seamlessly on multiple hardware platforms. It explores programming languages, frameworks, and libraries compatible with the Raspberry Pi Pico, facilitating the development of the emulator across

SYSTEM ARCHITECTURE AND DESIGN

1. **Hardware Components**:

    - Raspberry Pi Pico microcontroller: Acts as the core processing unit and handles emulation, user input, and interfacing with external components.
    - ILI9225 LCD Display: Provides the visual output for the emulator, displaying game graphics and UI elements.
    - SD Card: Stores game ROMs and other data required for emulation.
    - Micro-push buttons: Used for user input, including directional controls (DPAD), action buttons (A, B), select button, and start button.
    - 3D-printed Case: Encloses and protects the hardware components, giving the emulator a retro Game Boy appearance.

2. **System Overview:**

    - The Raspberry Pi Pico is the heart of the system, running the emulator software and managing hardware interactions.
    - The ILI9225 LCD display is connected to the Raspberry Pi Pico via SPI (Serial Peripheral Interface) for fast data transfer and smooth rendering of graphics.
    - User input is received through the micro-push buttons, which are connected to GPIO pins on the Raspberry Pi Pico.
    - The SD card stores game ROMs and configuration files, allowing users to load and play their favorite Game Boy titles.
    - The 3D-printed case houses all the components and provides a nostalgic Game Boy aesthetic, enhancing the overall gaming experience.

3. **Software Components:**

    - Emulator Software: Implements the Game Boy emulation logic, including CPU emulation, memory management, graphics rendering, and audio generation.
    - User Interface (UI): Provides a simple and intuitive interface for

navigating games, selecting options, and managing settings.

- Input Handling: Manages user input from the micro-push buttons and translates it into commands for the emulator software.
- Game Loading: Reads game ROMs from the SD card and loads them into memory for gameplay.
- Configuration Management: Allows users to customize emulator settings, such as screen scaling, button mappings, and audio preferences.

4. **Interaction Flow:**

    - Upon startup, the emulator initializes hardware components and loads the main menu on the LCD display.
    - Users navigate the menu using the DPAD and select buttons to choose a game ROM from the SD card.
    - Once a game is selected, the emulator loads the ROM into memory, starts emulation, and displays the game graphics and UI on the LCD screen.
    - Users control gameplay using the DPAD for movement and the action buttons (A, B) for in-game actions.
    - The emulator continuously reads user input, updates game state, and renders graphics and audio in real-time to provide an immersive gaming experience.

By combining hardware components, software emulation, and a user-friendly interface, the Pico-GB Game Boy emulator offers a seamless and enjoyable retro gaming experience, bridging the gap between classic gaming nostalgia and modern DIY electronics.

COMPONENTS NEEDED

- 1x Download 3d printed parts/.STL files
- 1x Raspberry Pi Pico
- 1x 2.2inch ILI9225 176×220 LCD Display Module
- 1x 32GB Micro SD Card with adapter
- 8x Micro Push Button Switch, Momentary Tactile Tact Touch, 6x6x6 mm, 4 pins
- 1x MAX98357 I2S Audio Amplifier Module
- 1x Weewooday 2W 8 Ohm small speaker 2.8cm (1.1in) diameter, 6mm high
- 1x EEMB Lithium Polymer Battery 3.7V 2000mAh
- 1x TP4056 Micro-USB Charger for Lipo Batteries
- 1x SS12F15 Miniature Slide Switch
- 1x Solderless Breadboard
- 1x Solderable Breadboard
- 1x Dupont Wires Assorted Kit (Male to Female + Male to Male + Female to Female)
- 2x 20 pins 2.54mm right angle male headers
- 1x Preformed Breadboard Jumper Wires
- 3D Printer Filament
- Grey, Red, and Black or the colors of your choice!

CODING AND TESTING

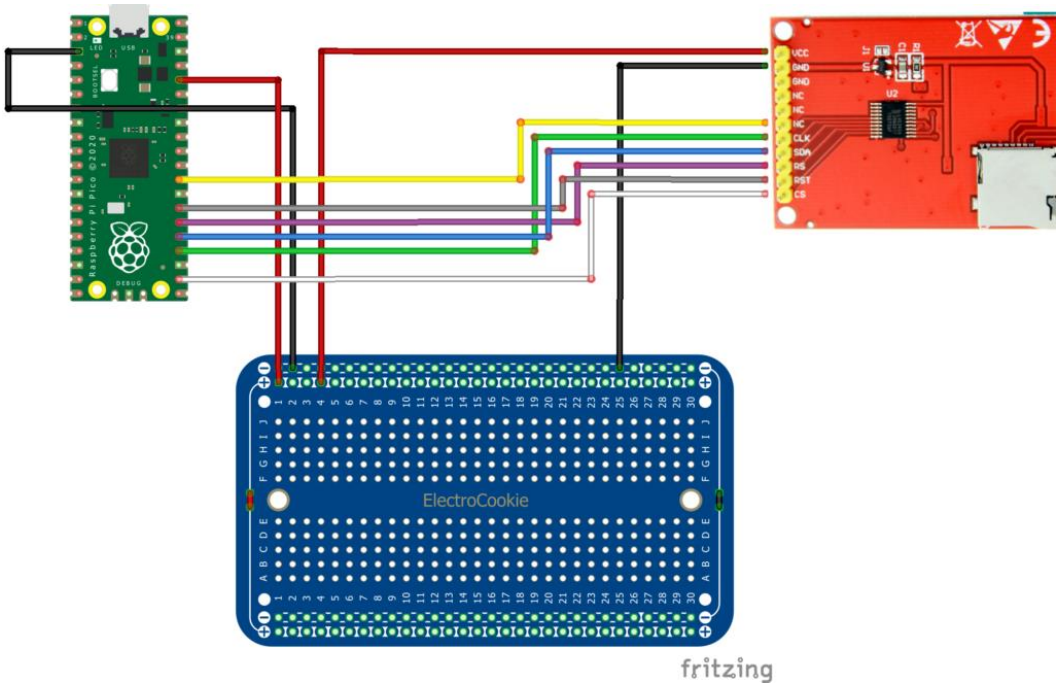
- **Flashing the firmware**

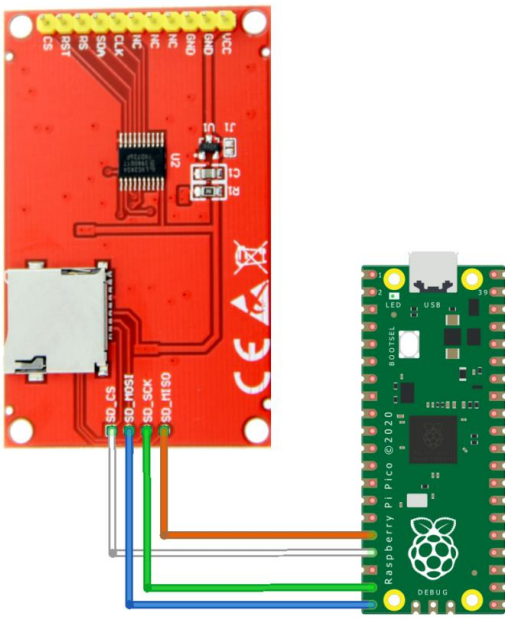
1. It’s easier to flash the firmware before starting the assembly so you can test that everything is working as expected at every stage of the build:
2. Download the latest. UF2 file from the release page
3. Push and hold the BOOTSEL button on the Pico, then connect to your computer using a micro-USB cable. Release BOOTSEL once the drive RPI-RP2 appears on your computer.
4. Drag and drop the. UF2 file on to the RPI-RP2 drive. The Raspberry Pi Pico will reboot and will now run the emulator.

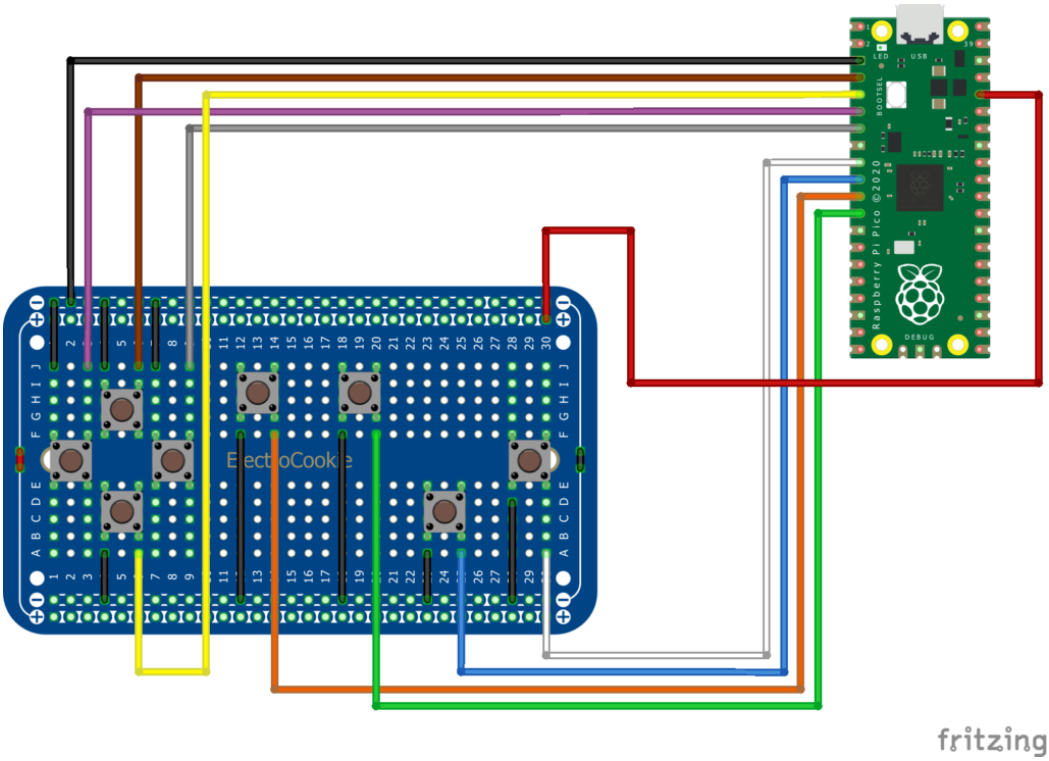
- **Preparing the SD Card**

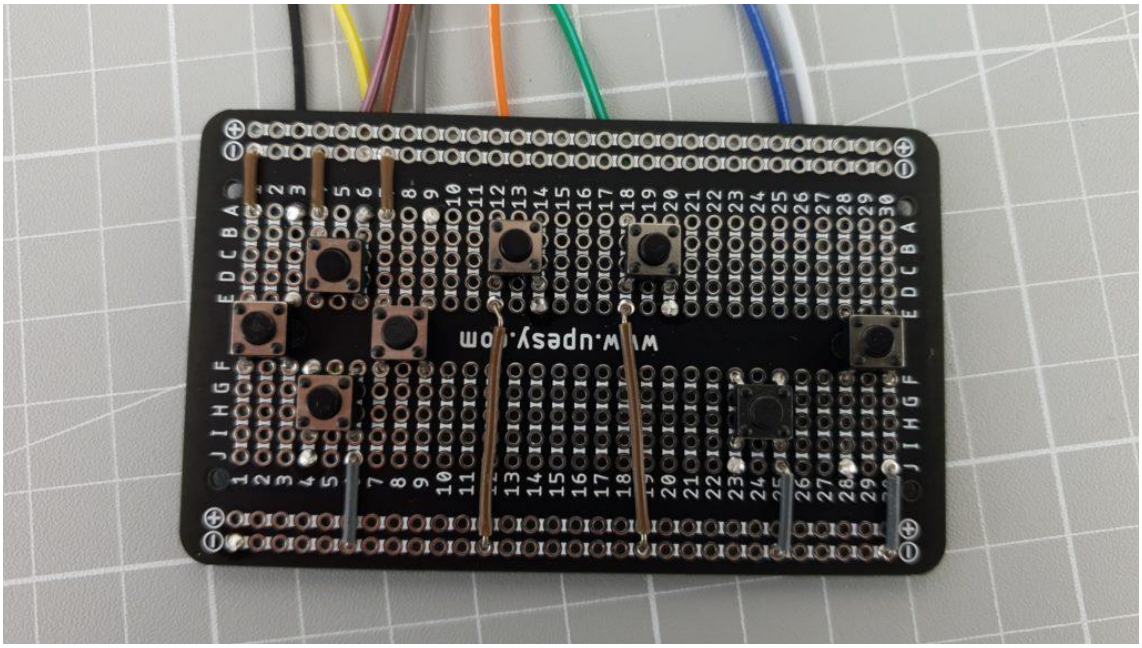
1. The SD card is used to store game ROMs and save game progress. For this project, you will need a FAT 32 formatted Micro SD card with roms you legally own. Roms must have the .gb extension.
2. Insert your SD card in a Windows computer and format it as FAT 32
3. Copy your .gb files to the SD card root folder (subfolders are not supported at this time)
4. Insert the SD card into the ILI9225 SD card slot using a Micro SD adapter

CIRCUIT DIAGRAMS

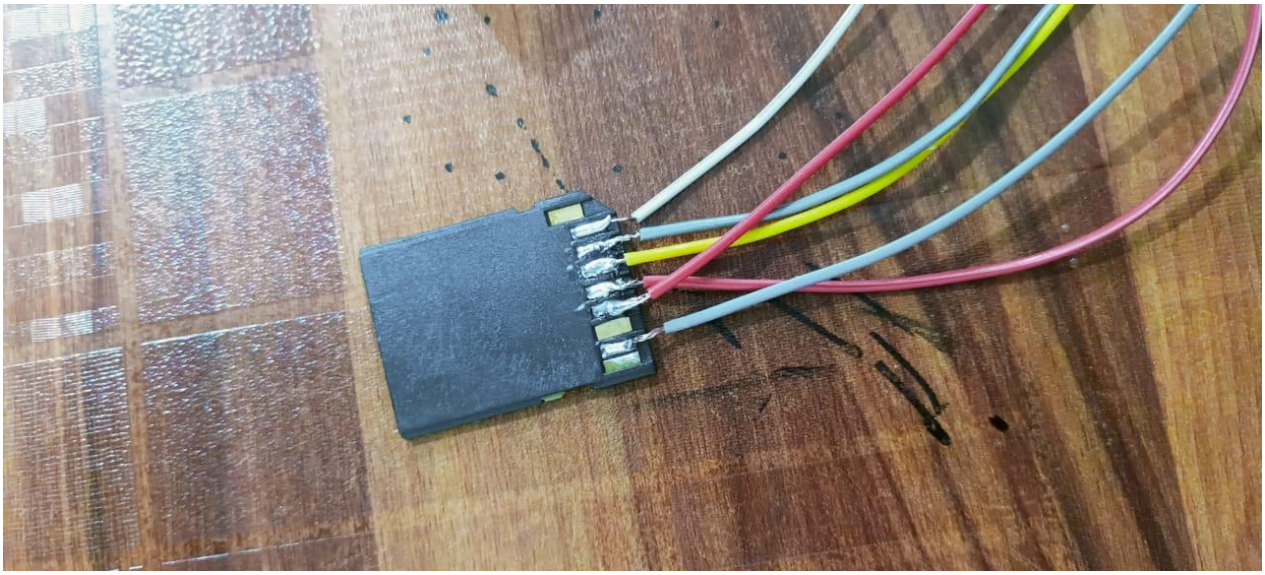


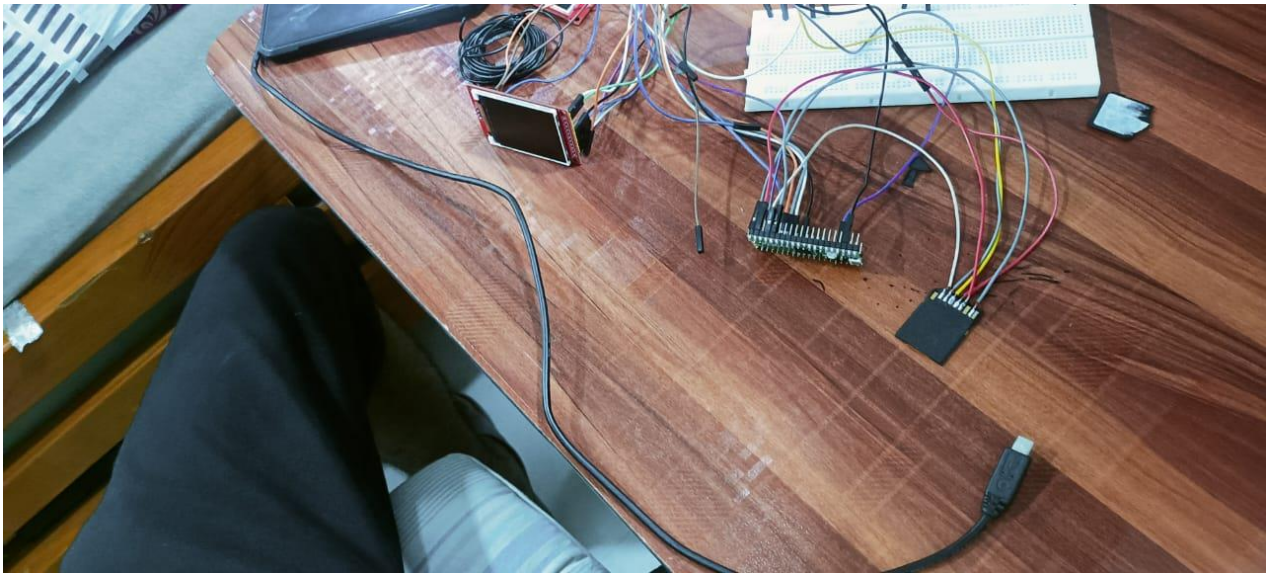


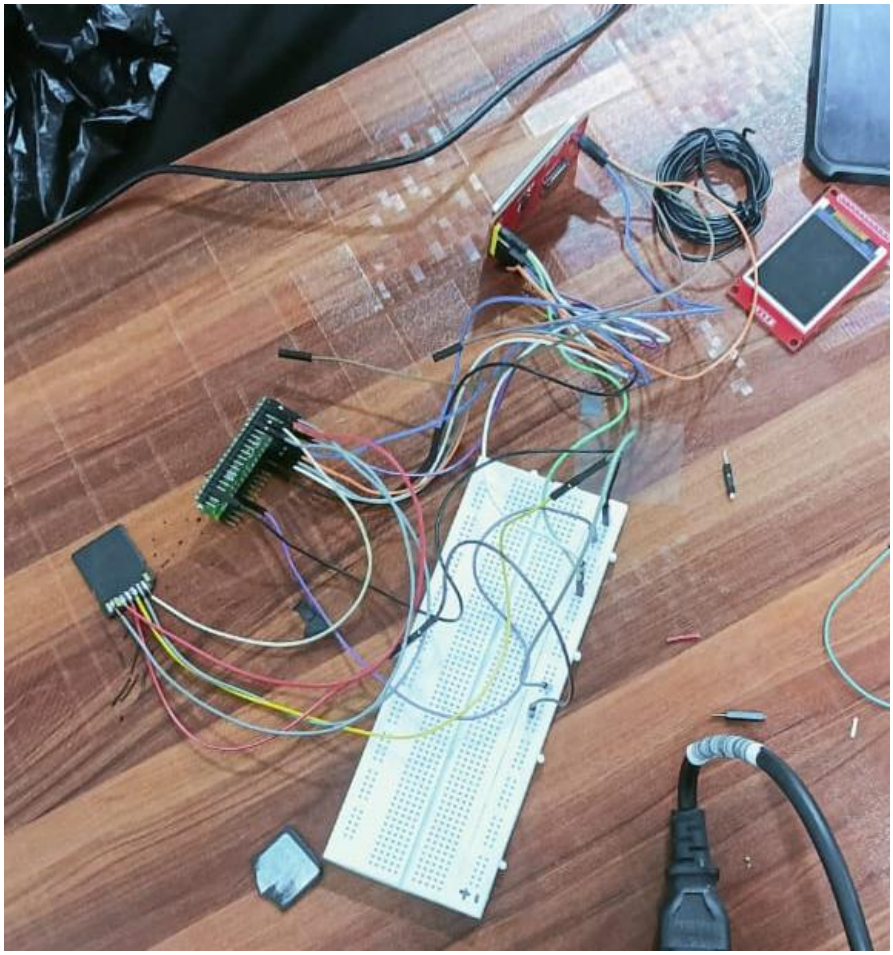

ElectroCookie
Raspberry Pi Pico ©2020
fritzing

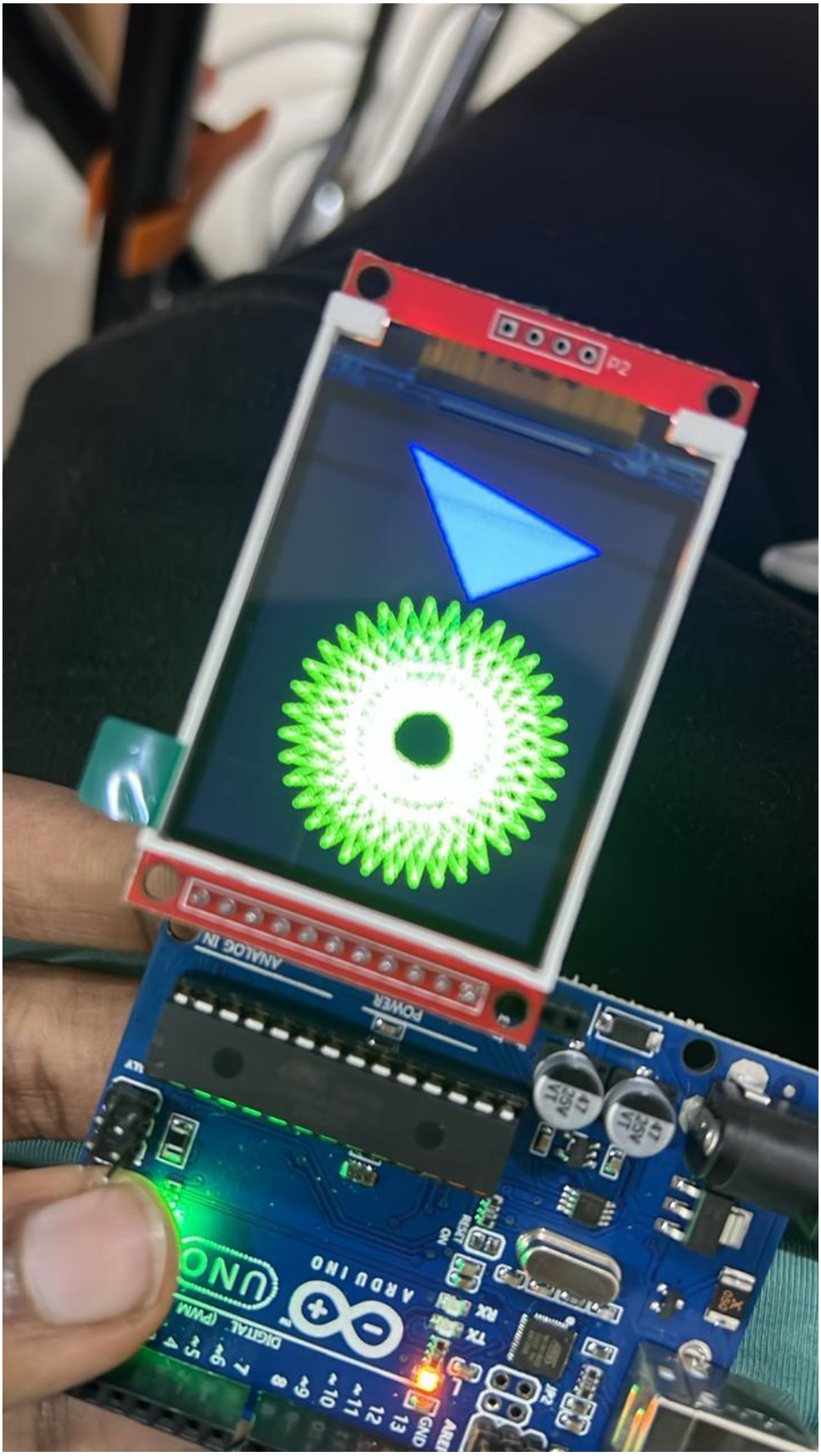

www.upesy.com

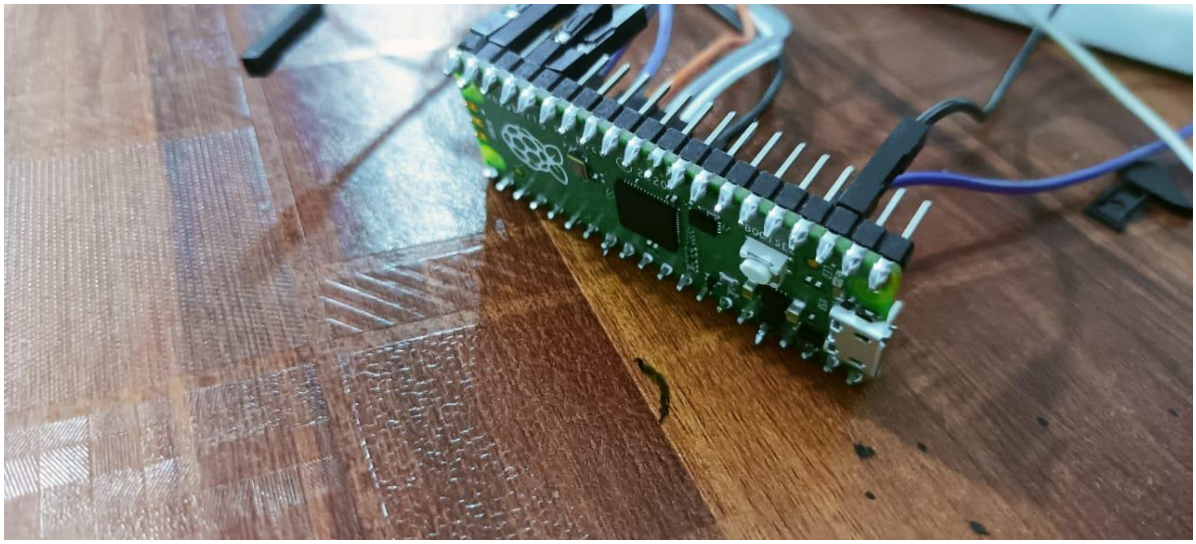
SCREENSHOTS AND RESULTS

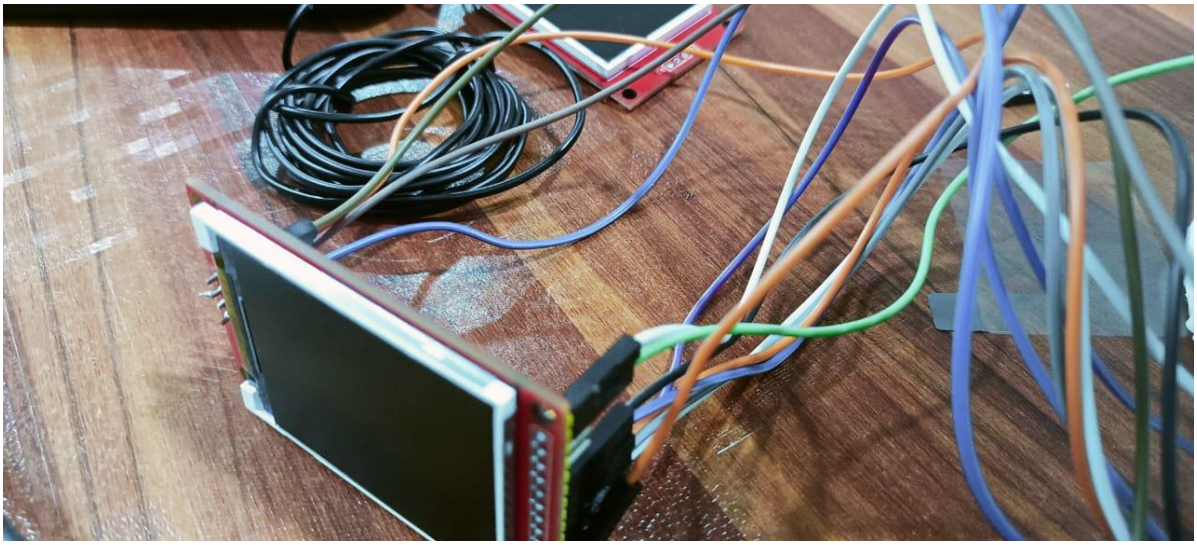







P2
ANALOG IN
POWER
UNO
ARDUINO
DIGITAL (PWM~)
RX
TX
GND





In a working Display

CONCLUSION AND FUTURE ENHANCEMENTS

The development of a Game Boy emulator using the Raspberry Pi Pico microcontroller marks a significant achievement in merging retro gaming with IoT technology. Through meticulous hardware interfacing, software emulation, and user interface design, we have successfully recreated the magic of classic Game Boy titles on modern IoT devices. This project showcases the versatility and potential of the Raspberry Pi Pico platform in realizing innovative gaming experiences, bridging the gap between generations of gamers and demonstrating the enduring appeal of retro gaming nostalgia.

Future Enhancements: While the current implementation of the Game Boy emulator on the Raspberry Pi Pico represents a substantial accomplishment, several avenues for future enhancements and expansions exist:

1. Performance Optimization: Further optimizations can be made to enhance the emulator's performance, particularly in terms of speed and accuracy. Fine-tuning emulation algorithms, optimizing code execution, and leveraging hardware acceleration features of the Raspberry Pi Pico can help achieve smoother gameplay and improved compatibility with a wider range of Game Boy titles.
2. Expanded Game Compatibility: Continuously expanding the emulator's compatibility with a broader selection of Game Boy games is crucial. This involves addressing compatibility issues, refining emulation accuracy for specific titles, and incorporating support for additional features and peripherals, such as Game Boy Color games and accessories.
3. IoT Integration: Leveraging the IoT capabilities of the Raspberry Pi Pico opens up opportunities for integrating multiplayer functionality, online leaderboards, cloud-based game saves, and social features into the emulator. By enabling connectivity with other IoT devices and online services, players can enhance their gaming experience and engage with the retro gaming community in new ways.
4. User Experience Enhancements: Enhancing the user interface and overall user experience of the emulator remains a priority. This includes refining menu navigation, implementing customizable

control options, incorporating graphical enhancements such as shaders and filters, and adding features like save states and cheat code support to improve usability and accessibility.

5. Community Engagement and Contribution: Encouraging community involvement and contribution to the project can accelerate its development and foster a vibrant ecosystem of retro gaming enthusiasts and developers. Establishing open-source repositories, hosting hackathons, and providing documentation and tutorials can empower others to build upon the emulator, add new features, and contribute to its ongoing evolution.