



Raspberry Pi-Powered Arcade Machine with RetroPie

Cagri Goksu Ustundag

IT Project Volunteer

Digital Inclusion A.S.B.L.

1, Dernier Sol, Bonnevoie, L-2543

September 2025, Luxembourg

Table of Contents

LIST OF FIGURES	II
1 Hardware Components	1
1.1 General Hardware Components	1
2 SD Card Preparation	2
2.1 Writing Images to SD Card	2
2.1.1 Windows	2
2.1.2 Linux	3
2.2 Extracting Images from SD Card	3
2.2.1 Windows	3
2.2.2 Linux	4
3 Arcade Machine Setup	5
3.1 Hardware Assembly	5
3.1.1 Installing Joysticks and Buttons	5
3.1.2 Connecting USB Encoder	6
3.1.3 Connecting Hardware to Raspberry Pi	6
3.2 Running the System	7
4 Configurations	8
4.1 Emulator Input Mapping	8
4.1.1 Keyboard	8
4.1.2 Joystick and Buttons	9
4.2 Emulator UI Settings	10
4.2.1 UI Mode Settings	10
4.2.2 Game Settings	10

List of Figures

3.1	Joystick and button layout.	5
3.2	USB-encoder wiring schema.	6
3.3	USB blocks on Raspberry Pi.	7
3.4	Fully-connected system schema.	7

1 Hardware Components

This chapter outlines the essential hardware components needed to build a fully functional arcade machine using a Raspberry Pi, RetroPie, input controls and other units.

1.1 General Hardware Components

In this project, the core of the arcade machine on Raspberry Pi relies on a few key hardware elements that bring the system to life.

1. Raspberry Pi 3B+
2. Screen (Raspberry Pi 3B+ supports full-size HDMI)
3. 5V-3A micro-USB power supply (to power Raspberry Pi)
4. SD card (min. 32GB to store RetroPie and game ROMs)
5. Input devices (keyboard, joystick, and button)
 - Keyboard (to configure the system)
 - Joystick
 - 6 arcade buttons (Start, Select, A, B, X, Y)
 - USB-Encoder
6. Speaker (3.5mm audio jack, preferably USB-powered, optional)

The input controls give the ability to configure the system and play the games. The joysticks enable directional input to navigate menus and gameplay. The arcade buttons are arranged to handle game actions. A USB encoder connects joystick and buttons to the Raspberry Pi, translating physical inputs into digital signals that RetroPie can interpret. The keyboard input is used solely to configure the machine.

2 SD Card Preparation

Before the arcade machine can run, the Raspberry Pi needs a bootable SD card with the RetroPie image. This chapter explains how to write a system image to an SD card and how to extract an image from an existing SD card for backup. Instructions are provided for both Windows and Linux-based systems (e.g., Ubuntu, Debian, ...). The image of this project requires **minimum 32GB SD card**.

2.1 Writing Images to SD Card

Writing an image to the SD enables Raspberry Pi to boot up and run the RetroPie system. The image writing process erases all data on the SD card, so ensure that any important files are backed up before proceeding.

2.1.1 Windows

On Windows, tools like **balenaEtcher** or **Win32 Disk Imager** can be used.

1. Download and install balenaEtcher (recommended for simplicity; <https://etcher.balena.io/>).
2. Insert the SD card into your computer.
3. Open balenaEtcher and select the RetroPie image file (`.img` or `.zip`).
4. Select the target SD card.
5. Click **Flash** and wait for the process to finish.

Once the process completes, safely eject the SD card. This SD card is ready to be used to boot up the Raspberry Pi with RetroPie.

2.1.2 Linux

On Linux-based systems, an image can be written to an SD card with **balenaEtcher** or simply **dd** command without installing anything. To process with balenaEtcher, refer to Section 2.1.1.

To process using **dd** command;

1. Insert the SD card into your computer.
2. Identify the device name using **lsblk** (e.g., **/dev/sda**).

Note: Device name can be different than **/dev/sdX** depending on your operating system. Something like **/dev/sdX1** points a partition in the device. Do not include numbers in device name.

3. Use the **dd** command:

```
sudo dd if=retropie.img of=/dev/sdX bs=4M status=progress conv=fsync
```

Replace **retropie.img** with the actual image filename and **/dev/sdX** with your SD card device. Once the process completes, safely eject the SD card. This SD card is ready to be used to boot up the Raspberry Pi with RetroPie.

2.2 Extracting Images from SD Card

Extracting an image from an SD card creates a complete backup of the system. The extracted image can be written to an SD card in order to run the games on multiple arcade machines. This method is also useful to keep a copy in case of corruption. Note that extracting image will have the size of full capacity of your SD card. For instance, an image of an 32GB SD card will have 32GB size.

2.2.1 Windows

On Windows, the tool for extracting SD card images is **Win32 Disk Imager**.

1. Download and install Win32 Disk Imager.
2. Insert the SD card into your computer.

3. Open the program and select the drive letter that corresponds to your SD card.
4. Choose a location and filename for the image file (`.img`).
5. Click **Read** to copy the SD card contents into the image file.

2.2.2 Linux

On Linux-based systems, SD card images can be extracted using the `dd` command.

1. Insert the SD card into your computer.
2. Identify the device name using `lsblk` (e.g., `/dev/sda`).
Note: Device name can be different than `/dev/sdX` depending on your operating system. Something like `/dev/sdX1` points a partition in the device. Do not include numbers in device name.
3. Run the following command in a terminal:

```
sudo dd if=/dev/sdX of=~/backup.img bs=4M status=progress
```

Replace `sdX` in `/dev/sdX` with the correct device name, and then wait for the process to complete. In case of an interruption during the process, retry step 3.

3 Arcade Machine Setup

This chapter explains how to assemble the arcade machine hardware and run the system on the Raspberry Pi.

3.1 Hardware Assembly

The hardware assembly stage involves connecting all physical components of the arcade machine. This includes mounting the joysticks and buttons, wiring them to the USB encoder, and linking the encoder and other peripherals to the Raspberry Pi.

3.1.1 Installing Joysticks and Buttons

Ensure that the buttons are arranged in a comfortable layout for gameplay. While the layout can be adjusted to personal preference, this project follows the configuration shown below.

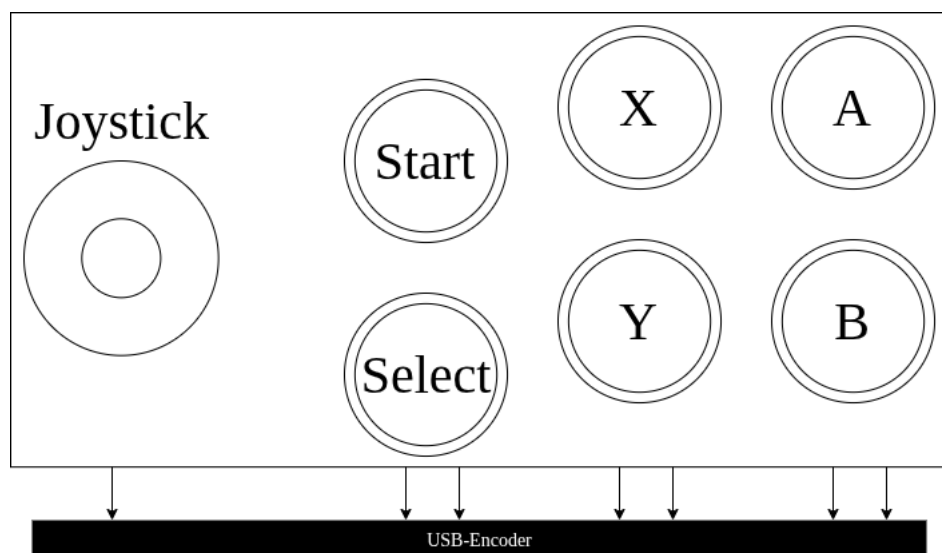


Figure 3.1: Joystick and button layout.

3.1.2 Connecting USB Encoder

Attach the arcade buttons and joystick wires to the USB encoder in accordance with the Figure 3.2. The encoder converts physical inputs into signals recognized by the Raspberry Pi over USB. Note that the joystick uses 5-pin, arcade buttons use 3-pin cable for connection.

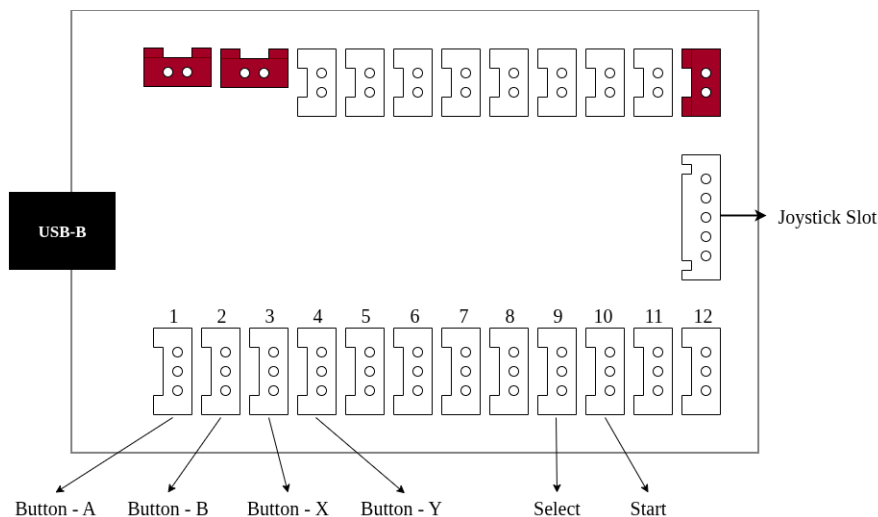


Figure 3.2: USB-encoder wiring schema.

3.1.3 Connecting Hardware to Raspberry Pi

In order to complete the setup, all connections must be completed before running the system. The following components should be connected to the Raspberry Pi:

- A display to full-size HDMI port.
- A prepared SD card to SD card slot.
- A keyboard to a USB port, if configuration necessary.
- A speaker to 3.5mm jack(optional, preferably USB-powered).
- One or multiple USB-Encoders to USB ports.

Important!: The Raspberry Pi 3B+ includes two USB blocks, each with two ports (see Figure 3.3). Port priority is determined first by block, with Block-A taking precedence over Block-B, and then by position, with the top port prioritized over the bottom. The resulting order is: A-top, A-bottom, B-top, B-bottom.

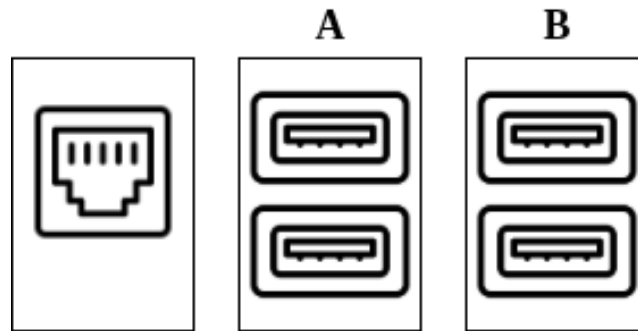


Figure 3.3: USB blocks on Raspberry Pi.

These steps provides all the necessary connections to complete the hardware assembly, except power supply. Once the power is connected over micro-USB to the Raspberry, the system will start up. Figure 3.4 shows the fully-connected system schema.

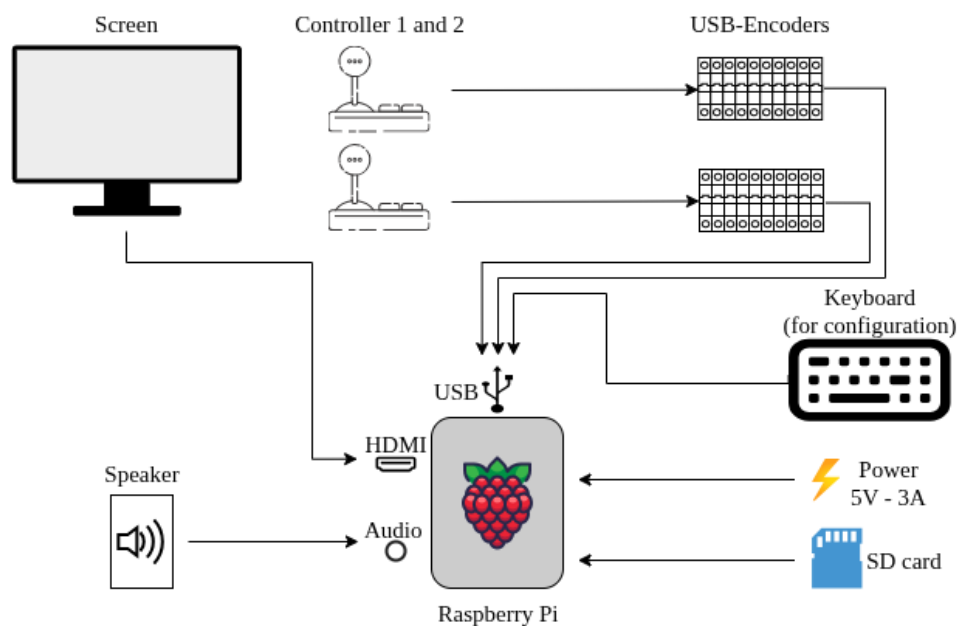


Figure 3.4: Fully-connected system schema.

3.2 Running the System

Once the prepared SD card inserted into the Raspberry Pi, power the device on by plugging power cable. It is important to inserting SD card before powering on the device. After booting, the system will run on the RetroPie.

4 Configurations

It is necessary to configure inputs and interface settings to ensure proper operation. This chapter explains how to map keyboard and joystick inputs to default setup, adjust UI modes, and manage game visibility.

4.1 Emulator Input Mapping

Input mapping allows the emulator to recognize user actions correctly, whether from a keyboard or physical arcade controls. The menu to configure buttons is accessible in EmulationStation screen, by pressing the assigned **Start** button to enter the main menu and selecting **Configure Input**. In order to keep the consistency between multiple arcade machines, please use the same button mapping as indicated below.

4.1.1 Keyboard

Keyboard inputs are mainly used for configuration and system navigation. To set up keyboard controls:

1. From the RetroPie main menu (Enter button as default), navigate to **Configure Input**.
2. Follow the on-screen prompts to assign keys for system functions, hotkeys, and navigation, as follows.
 - D-Pad Up, Down, Left, Right → Up, Down, Left, Right
 - Start → Enter
 - Select → Right Shift
 - Button-A → X
 - Button-B → Z

- Button-X → S
- Button-Y → A
- Left Shoulder → Q
- Right Shoulder → W
- Hotkey → Left Ctrl

These are some useful key or combinations:

- Hotkey + Enter: Exit to EmulationStation
- F4: Enter terminal when on EmulationStation

4.1.2 Joystick and Buttons

Joystick and arcade button mapping is used exclusively for gameplay. RetroPie automatically detects the controllers on startup. To configure:

1. From the RetroPie main menu (Enter button as default), navigate to **Configure Input**.
2. Follow the on-screen prompts to assign keys for system functions, hotkeys, and navigation, **as follows**.
 - D-Pad Up, Down, Left, Right → Up, Down, Left, Right on joystick
 - Start → Start
 - Select → Select
 - Button-A → A
 - Button-B → B
 - Button-X → X
 - Button-Y → Y
 - Hotkey → **Important:** Do not enter any button for hotkey by skipping it. When you exit the configuration, a hotkey button will be asked to use Select button as default. Choose **Yes** if you have multiple games, **No** if you have single game.

4.2 Emulator UI Settings

These settings help secure configurations and make the arcade machine user-friendly. Using the settings below, the configuration menus will be only accessible by keyboard. The gamers are prevented from entering these menus.

4.2.1 UI Mode Settings

This setting manages access to the menus by the user. To configure:

1. From the main menu (Enter button), go to **UI Settings**. Under UI Mode, there are three different modes:
 - **Full Mode** – Complete access to all menus and settings.
 - **Kiosk Mode** – Simplified interface hiding advanced menus.
 - **Kid Mode** – Restricts access to configuration screens to prevent accidental changes by gamers.
2. Choose the mode: Full mode to configure system, Kid mode before end-user access to machine.
3. To return to Full mode while in Kid mode, press Up, Up, Down, Down, Left, Right, Left, Right, B, A buttons in order.

4.2.2 Game Settings

Game metadata determines which games are visible in each UI mode:

1. Navigate to game in game list and open **Options** by pressing Select button.
2. Edit game metadata and make **KIDGAME** on.
 - Only games tagged appropriately for Kid mode will appear when the arcade is running in that restricted mode, ensuring children access approved games.
 - Full mode will display the complete game library.