



GRS Viper SBC User Manual

1. Table of Content

1. Table of Content	2
2. Introduction	3
3. Hardware Explained	4
4. 40PIN GPIO mapping	5
5. Batocera Mapping with GPIO.....	6
6. Build-A-Cade Compatibility	7
A. DSI and HDMI output handling	7
B. Build-A-Cade Mapping with GPIO using Batocera System	8
7. Modification via USB direct connection to a PC.....	11
A. Enable read and write for Batocera system files	11
B. Accessing to Batocera system files and adding ROMs via direct PC connection	11
8. Backing up Batocera system files into an SD card/USB stick/via USB.....	15
A. MTP Storage Files Backup.....	15
B. Backup Image to Your Desired Storage Device via 'dd' in Linux.....	15
9. Adding ROMs and Play games via SD Card	17
10. Adding ROMs and play games via USB Drive	18
11. Adding ROMs to Viper SBC via SSH	19
12. Firmware upgrade/Burning image to eMMC	20
13. Viper SBC GPU Overclock.....	22
14. Boot from SD card	22
15. Boot from USB stick	23
16. Boot from eMMC	23
17. Additional Information.....	23

2. Introduction

Introducing the GRS Viper Single Board Computer (SBC), a product designed with one primary goal: to deliver an unparalleled DIY retro gaming experience to a broad community of enthusiasts. Our industry veteran, Glen from Glen's Retro Show, has dedicated his life to DIY arcade machine construction and continues to introduce impressive products that rekindle the nostalgia of our childhood gaming center experiences.

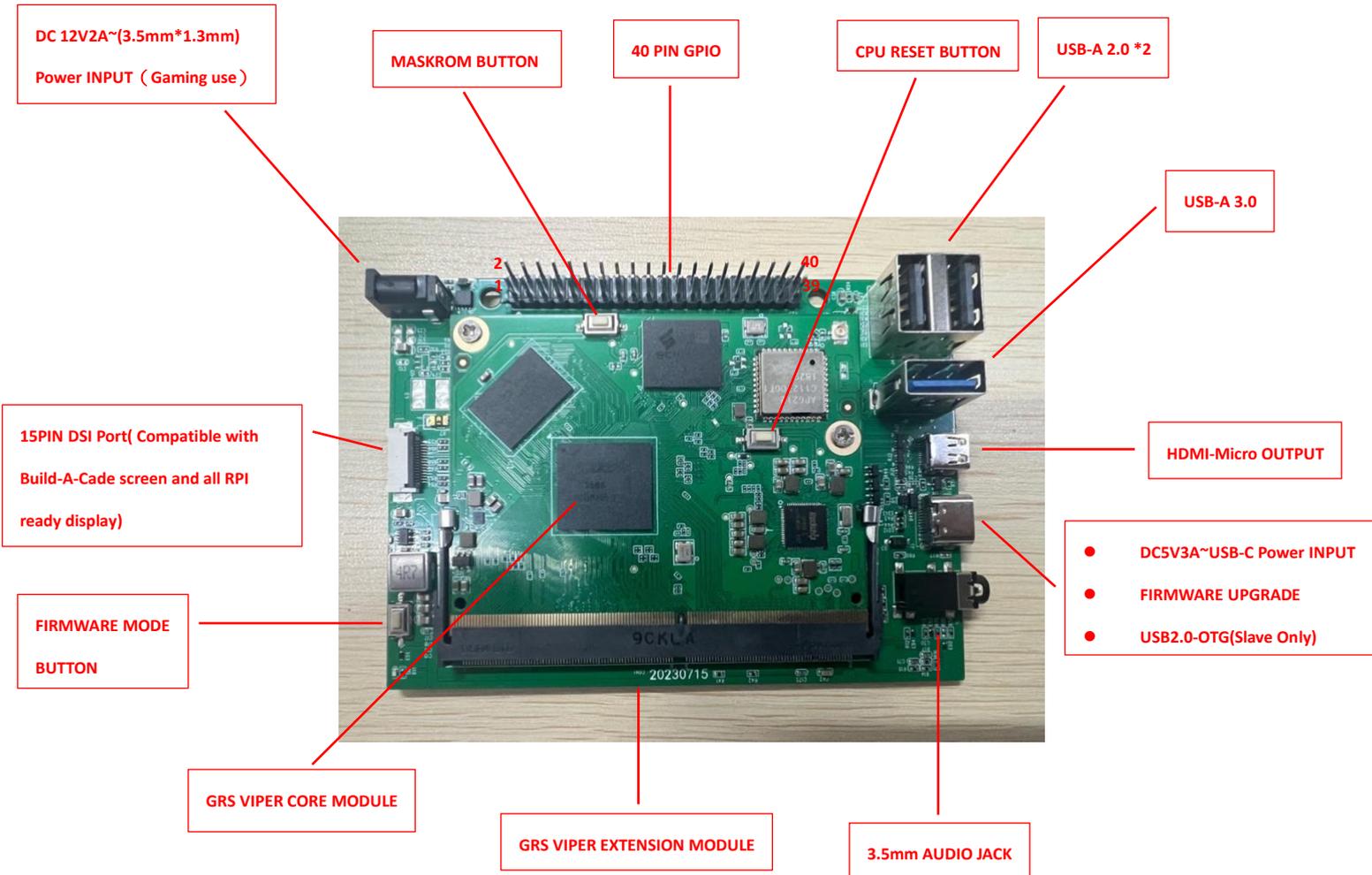
Our range features several unrivaled products, such as the GRS Spinner, GRS Flight Yoke, and the GRS Super Joystick. Each of them employs innovative technology to recreate retro gaming, thus elevating the gaming experience to a new level. Our ambition extends beyond retro gaming; we strive to offer products of "Industrial Arcade quality without Arcade price."

Today, Thunderstick Studio and Glen unite to bring even more to the table. We take great pride in presenting our GRS Viper SBC, a single board computer designed and manufactured explicitly for the DIY Arcade community. Here's a glimpse of the features it offers:

- Seamless compatibility with the GRS iiRcade Ultimate system switcher, allowing iiRcade owners using the GRS iiRcade Ultimate Control Panel to rapidly switch between the iiRcade system and the Viper SBC.
- Plug-and-Play capability for the Sinden Lightgun and the GRS Sinden Ultimate Shotgun.
- Compatibility with the GRS Build-A-Cade (note: users will need to purchase an additional wiring harness kit for this feature from [Link:xxx](#)).
- The unique feature of MTP Storage Device for swift access to Batocera files by simply connecting the Viper SBC to a PC through a single USB cable, rendering an experience similar to accessing photos on an Android device.
- A DSI port, A HDMI port, two USB 2.0 type A ports, one USB 3.0 Type A port.
- DC inputs of 12V through a DC round head port and 5V through a USB Type C port.
- Firmware upgradability, 40-pin GPIO, and a modularized structure: Core PCB and Extension Module, ensuring future hardware upgradeability.

3. Hardware Explained

Below is a brief hardware overview for the GRS Viper SBC:



Additional Information:

- **DC12V2A Power Input:** This port serves as either the primary or secondary power source for the Viper SBC. The default setup of 'Build-A-Cade' relies on the DC5V USB-C port for its power supply.
- **Firmware Mode Button:** This button is intended for use solely during the firmware upgrade process.
- **CPU Reset Button:** Press this button to perform a hardware reboot.
- **USB-C Port:** This port hosts multiple functions: it can act as the power source for the Viper SBC, serve as a connection to a PC for firmware upgrades, or link to a PC as a Media Transfer Protocol(MTP) USB device for file transfer. Detailed instructions on these functionalities will be covered in the remaining sections of this document.

4. 40PIN GPIO mapping

This section outlines how the 40-pin connector on the Viper SBC maps to the functions of the RK3566 CPU. This information is valuable for users looking to leverage the diverse functionalities offered by the Viper SBC.

GPIO Functions	PIN		GPIO Functions
VCC-3V3 SYS	1	2	ADP 5V
I2C3 SDA M0 CON1	3	4	ADP 5V
I2C3 SCL M0 CON1	5	6	GND
AUDIO CTRL GPIO3 C4	7	8	UART2 TX M0 DEBUG
GND	9	10	UART2 TX M0 DEBUG
HP DET L GPIO3 A1	11	12	LCD0 RST L GPIO3 a3
GSSENSOR INT L GPIO3 A2	13	14	GND
SARADC VIN1 EVB HW ID	15	16	UART4 RX M1
VCC-3V3 SYS	17	18	UART4 TX M1
I2S3 SCLK M1 CON1	19	20	GND
I2S3 SDO M1 CON1	21	22	AUDIO CTRL GPIO3 B5
I2S3 MCLK M1 CON1	23	24	I2S3 SDI M1 CON1
GND	25	26	SARADC VIN2 LCD ID
I2C5 SDA M0	27	28	I2C SCL M0
SPDIF TX M1	29	30	GND
PWM11 M0 FSYNC IN GPIO3 B6	31	32	PCIE20 PRSNT L GPIO3 B7
IR TX GPIO3 C0	33	34	GND
LCD RST L GPIO3 A4	35	36	INT GPIO3 A7 CON1
CAMERA1 PDN L GPIO3 B0	37	38	MIPICAM0 RST L GPIO3 A6
GND	39	40	MIPICAM1 RST L GPIO3 A5

5. Batocera Mapping with GPIO

In its default configuration with the dedicated Batocera System OS, the Viper SBC automatically maps Control Port 1 to specific GPIO pins on the board. This mapping serves as the basis for the Batocera Key Map, which details the actual functions within the Batocera system. By default, the Viper SBC acts as a controller known as 'GPIO' within the Batocera environment. Each button on this 'GPIO' controller is represented by a numerical value displayed in Batocera's Controller Mapping Menu. This setup allows for easy identification and customization. The table that follows provides a comprehensive guide, showing the correlation between each physical GPIO pin on the Viper SBC and its corresponding key function in Batocera, as well as how these mappings are presented in the Controller Mapping Menu.

GPIO controller Value	Gamepad Controller	PIN		Gamepad Controller	GPIO Controller value
		1	2		
		3	4		
		5	6		
13	Up	7	8		
		9	10		
14	Down	11	12	A(SOUTH)	4
15	Left	13	14		
		15	16		
		17	18		
16	Right	19	20		
10	Start	21	22	L1	2
9	Select/Hotkey	23	24		
		25	26		
5	L2	27	28		
7	R2	29	30		
6	L3	31	32	Y(NORTH)	11
8	N/A	33	34		
3	X(WEST)	35	36	N/A	N/A
0	B(EAST)	37	38	R1	1
		39	40	R3	12

6. Build-A-Cade Compatibility

A. DSI and HDMI output handling

The Viper SBC is expertly crafted for immediate compatibility with the Build-A-Cade ecosystem. It boasts a DSI port, pre-set to a 90-degree orientation, to seamlessly fit all standard Build-A-Cade kits. This strategic design presents a compelling alternative to the conventional Raspberry Pi SBCs, offering ease of integration right out of the box.

Engineered to exclusively support the Batocera System, the Viper SBC introduces a distinctive approach to toggling between the Build-A-Cade's DSI display and an external HDMI monitor. It incorporates an advanced hardware HDMI detection feature, streamlining the display selection process. Upon startup, the system evaluates the HDMI connection status in the background. If no HDMI monitor is detected, it defaults to the DSI display, automatically adjusting the orientation for optimal Build-A-Cade compatibility. Conversely, connecting an HDMI display triggers an automatic switch to HDMI output, leveraging Batocera's dynamic display resolution adjustment. This ensures users achieve the best possible display quality with high compatibility, enhancing the overall gaming experience.

B. Build-A-Cade Mapping with Viper's GPIO in Batocera System

This table illustrates the physical mapping of GPIO pins to the GRS Build-A-Cade console when it is equipped with the Viper SBC and running the specialized Batocera System designed for the Viper SBC.

Cable color	Build-A-Cade	PIN		Build-A-Cade	Cable Color
Red/White		1	2		Red
		3	4	Mainboard VIN	Red
		5	6		Black
Red/Black	P1 Joystick Up	7	8		Green/White
Black		9	10		Blue/White
Green/Black	P1 Joystick Down	11	12	Button A	Yellow
Blue/Black	P1 Joystick Left	13	14		Black
		15	16		
Red/White		17	18		
Yellow/Black	P1 Joystick Right	19	20		Black
Dark green	P1 Start	21	22	L1	Green
Pink	Left Coin Door	23	24		
Black		25	26		
Purple/Black	P2 Joystick Up	27	28		
Orange/Black	P2 Joystick Down	29	30		Black
Brown/Black	P2 Joystick Left	31	32	Right Coin Door	Orange
Grey/Black	P2 Joystick Right	33	34		Black
Grey	Rear Button 2	35	36	Rear Power Button	
Blue	P2 Start	37	38	R1	Brown
Black	Control Panel GND	39	40	Rear Button 1	Purple

C. Build-A-Cade Control Mapping Diagrams



Front



Back

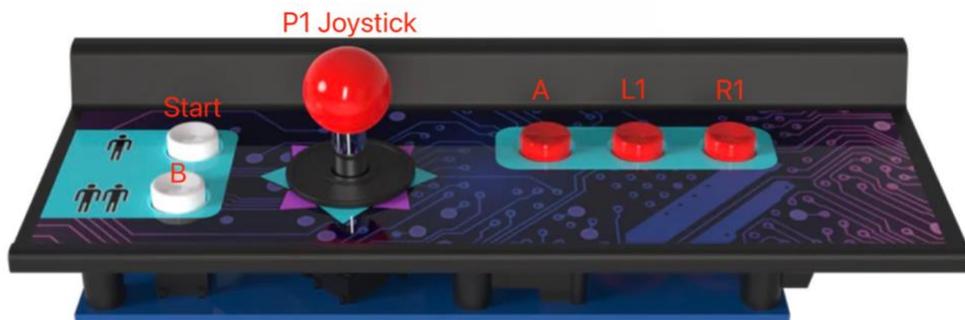
Control Panels



A-Joystick with 3 buttons



B-Dual joystick with 3 buttons



C-Joystick with 5 buttons



D-Spinner with 5 buttons and GRS mini steering wheel



E-Trackball with 5 buttons

7. Modification via USB direct connection to a PC

A. Enable read and write for Batocera system files

The Batocera system is by default read only, so in here we introduce a method to enable read/write permission for the Batocera system. Here are the steps:

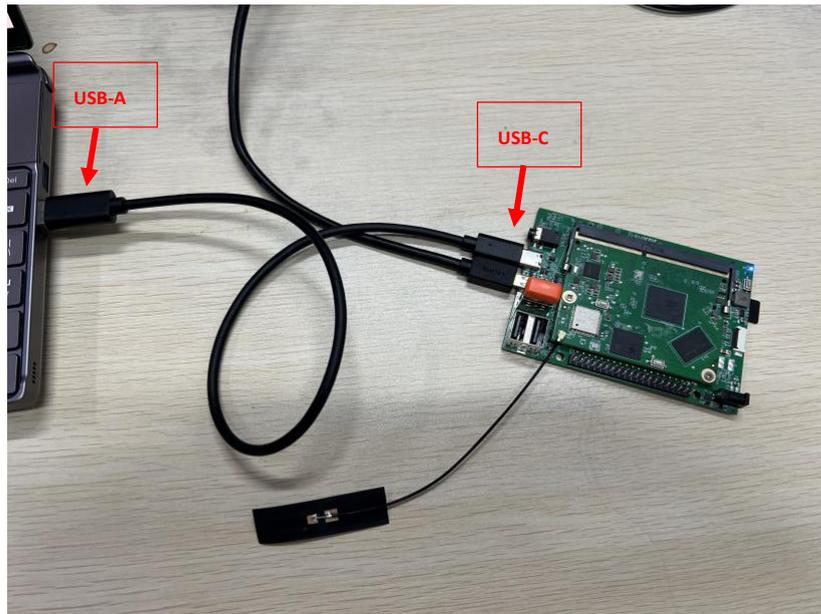
1. SSH to to your Batocera system in the Viper SBC.
2. Type in the terminal, type in command `‘/bin/mount -o remount,rw /dev/mmcbk0p4 /boot’`, then hit enter.
3. Then type in `‘mkdir -p /boot/userdata/system/lower /boot/userdata/system/upper /boot/userdata/system/work’`, then hit enter.
4. Now we download the file “ulnitrd” from link and copy this file to the boot/boot directory.
“ulnitrd” download link:
<https://www.dropbox.com/scl/fo/k77yu2kyzvy3k0rxcykld/h?rlkey=00k3e6olty8vuyzl72xsg6dtw&dl=0>
5. Then type in command `‘reboot’`, hit enter. After the Viper SBC is rebooted, all the system files are read and writable. This means that you can any modification to the Batocera system as you like. And all the changes would be saved in this folder `/boot/userdata/system/upper`

B. Accessing to Batocera system files and adding ROMs via direct PC connection

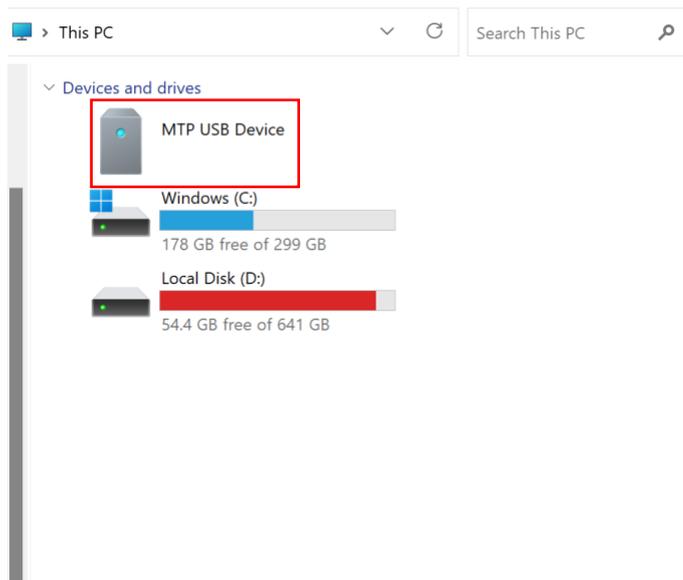
The purpose of this section is to assist users in connecting their Viper SBC to a PC directly through the USB cable provided. This enables direct access to the Batocera system folders for modifying files and the addition of ROMs, thereby enhancing customization and usage efficiency. For modification of Batocera’s system files, make sure to perform steps in [7. A Enable read and write for Batocera system files](#) beforehand.

Steps:

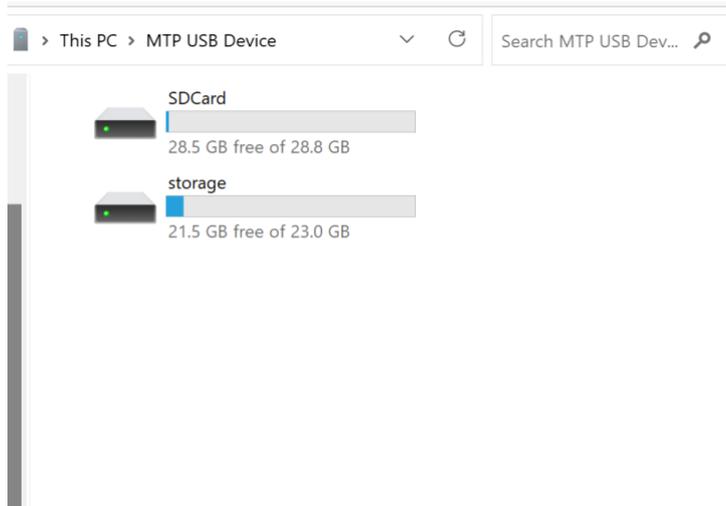
1. Insert the USB-C end of the USB-A to USB-C cable into the corresponding USB-C port on your Viper SBC. Affix the other end of this cable into the USB port of your PC, Linux or Mac system. **Note:** MacOS users must first download and install the MTP USB device driver to enable successful connection.



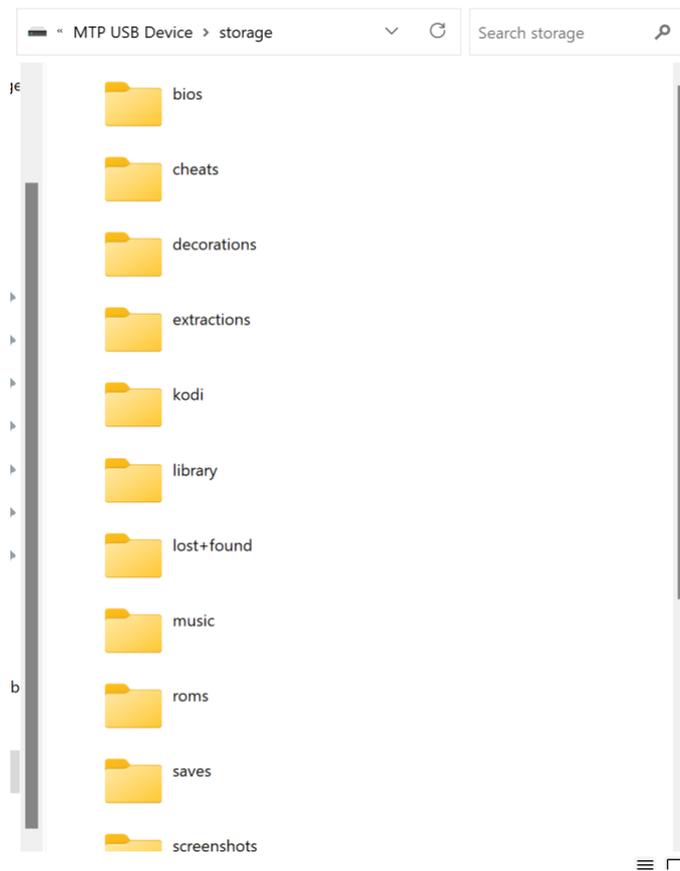
2. Wait for a period of approximately 5-10 seconds. Your computer will then detect and display the MTP USB device as a new storage device.



3. For access, double-click on the displayed device. Inside, you will see a storage device labeled "storage". This is the system folder of the Batocera operating system, stored in the EMMC drive of the Viper SBC. An additional storage device represents the other connected storage devices. As an example, there could be a storage device named "SDCard". This is the platform where you will have the ability to access both storage spaces and manage file transfers between various storage device.



4. To visualize the complete, customizable folder structure of the Batocera system, double-click on the "storage" entry. This will open a view where you can freely paste ROMs or customized elements into any desired location.



Important Note

You need to make sure you configure your storage device in Batocera system to your desired storage device as Batocera only reads ROMs from a single storage device at a time. This means that if you select Internal you will only be able to see the games

stored in you EMMC drive. If you are hooking up a blank disk or SD card please follow the guide in "*Add ROMs to SD card and play ROMs on SD card but run the Batocera system on EMMC*" section.

8. Backing up Batocera system files into an SD card/USB stick/via USB

A. MTP Storage Files Backup

Please note that the GRS SBC does not support the built-in Batocera image backup function, as the RK3566 chip lacks compatibility with the Batocera system. The cost associated with developing this feature at a low level for the SBC is also a critical factor. To address this, we've leveraged our expertise to devise an alternative backup method that offers improved functionality and ease of use.

Through the enabled MTP USB device feature, users can now access all user data in Batocera, including customized content and ROMs. The backup process is straightforward - simply copy files from the MTP USB Device onto your PC or any desired storage device. Then, zip this data and share it with friends or the community; transferring it to another SBC is as easy as pasting it in the same MTP USB Device pathway and rebooting the system.

B. Backup Image to Your Desired Storage Device via 'dd' in Linux

Creating a disk image backup with Linux's 'dd' command can be a daunting task as it requires substantial technical expertise. Users with limited familiarity with Linux backup and firmware packaging techniques are advised against this approach due to the potential risk of system damage or irreversible issues.

However, the following sections provide a simplified step-by-step guide to backing up your system image to an SD card:

Steps

1. **Set Up the Environment:** First, insert the SD card and create a new directory, as in `mkdir /dev/tmp`. Then, mount the SD card by typing `mount /dev/mmcblk1p1 /dev/tmp/`. Navigate to the SD directory using the command `cd /dev/tmp/`.
2. **Copy the Partition:** Identify the appropriate partition to copy. The current partitions are `/dev/mmcblk0p4(BATOCERA)` and `/dev/mmcblk0p5(SHARE)`. If you

wish to export the BATOCERA partition, your command would look like this: `dd if=/dev/mmcblk0p4 of=./backup.img bs=1M`.

3. **Sync and Unmount:** Once the command execution completes, run `sync` to ensure all the data from the memory has been fully written to the card. Unmount the SD card using: `cd /; umount /dev/tmp/`.

4. **Export and Package:** Finally, eject the SD card and insert it into your PC. Copy the file from the SD card and repackage it as firmware.

To dig deeper into firmware packaging, refer to the document `Rockchip_Developer_Guide_Linux_Software_EN.pdf`, specifically the "3.2 Development Burning Tool" and "3.3 Packaging Tool" sections. For more data on partitioning, consult `Rockchip_Introduction_Partition_EN.pdf`.

Documentation download link:

<https://www.dropbox.com/scl/fo/hpekmlxt1mtkalfe0r3zc/h?rlkey=85x0shltyy3va87zd01zwh1xe&dl=0>

9. Adding ROMs and Play games via SD Card

Enable Batocera to utilize the folder structure on an SD card, allowing the system to pull settings from the SD card instead of the Viper SBC's EMMC. The Viper SBC will still run the system locally.

Steps

1. **Format SD Card:** Format the SD card to FAT32 using either Pi Imager or BalenaEtcher software.
2. **Insert SD Card:** Insert the formatted SD card into the Viper SBC before powering it on.
3. **Access Batocera Settings:** Turn on the Viper SBC. Navigate to Batocera's MAIN MENU > SYSTEM SETTINGS. Select STORAGE DEVICE and choose the inserted SD card.
4. **Restart System:** Return to Batocera's MAIN MENU, click QUIT, then select RESTART SYSTEM.
5. **Folder Creation:** Allow 5-10 minutes for Batocera to create the folder structure on the SD card. The system will reboot twice during this process.
6. **Add ROMs:** Once Batocera is up, turn off the Viper SBC and safely remove the SD card to load ROMs onto it.
7. **Final Steps:** Reinsert the SD card into the Viper SBC then boot up. Your games will now appear in Batocera's user interface.

Important Note

When using this setup, the Batocera system configuration is stored on the SD card. If you switch the storage device back to "Internal," you only access to both the games and the system settings stored in the eMMC drive which is isolated from the SD card data.

10. Adding ROMs and play games via USB Drive

Enable Batocera to utilize the folder structure on an USB Stick, allowing the system to pull settings from the USB Stick instead of the Viper SBC's EMMC. The Viper SBC will still run the system locally.

Steps

1. **Format USB Stick:** Format the USB stick to FAT32 using either Pi Imager or BalenaEtcher software.
2. **Insert USB Stick:** Insert the formatted USB stick into the Viper SBC before powering it on.
3. **Access Batocera Settings:** Turn on the Viper SBC. Navigate to Batocera's MAIN MENU > SYSTEM SETTINGS. Select STORAGE DEVICE and choose the inserted USB Stick.
4. **Restart System:** Return to Batocera's MAIN MENU, click QUIT, then select RESTART SYSTEM.
5. **Folder Creation:** Allow 5-10 minutes for Batocera to create the folder structure on the USB stick. The system will reboot twice during this process.
6. **Add ROMs:** Once Batocera is up, turn off the Viper SBC and safely remove the USB stick to load ROMs onto it.
7. **Final Steps:** Reinsert the USB stick into the Viper SBC then boot up. Your games will now appear in Batocera's user interface.

Important Note

When using this setup, the Batocera system configuration is stored on the USB stick. If you switch the storage device to "Internal," you will lose access to both the games and the system settings stored on the USB stick.

11. Adding ROMs to Viper SBC via SSH

Connect to the Viper SBC via WiFi to access its EMMC drive and upload ROMs, which will then be accessible through the Batocera user interface.

Steps

1. **WiFi Connection:** Ensure the Viper SBC is connected to WiFi. For optimal connectivity, the use of an external WiFi antenna is recommended.
2. **Check IP Address:** Once connected, locate the IP address displayed in the Batocera system.
3. **Access EMMC Drive:** From your computer, use either Final Shell or SecureFX to connect to the Viper SBC's EMMC drive. Use the IP address displayed in Batocera. The username is 'root' and the password is 'linux'.
4. **Transfer ROMs:** After connecting, you'll have access to Batocera's file structure. Use this access to copy ROMs into the system.
5. **Reboot for Game Access:** After transferring the ROMs, reboot the Viper SBC. The games will then be available on the Batocera user interface.

Note

Make sure to reboot the system for the changes to take effect.

12. Firmware upgrade/Burning image to eMMC

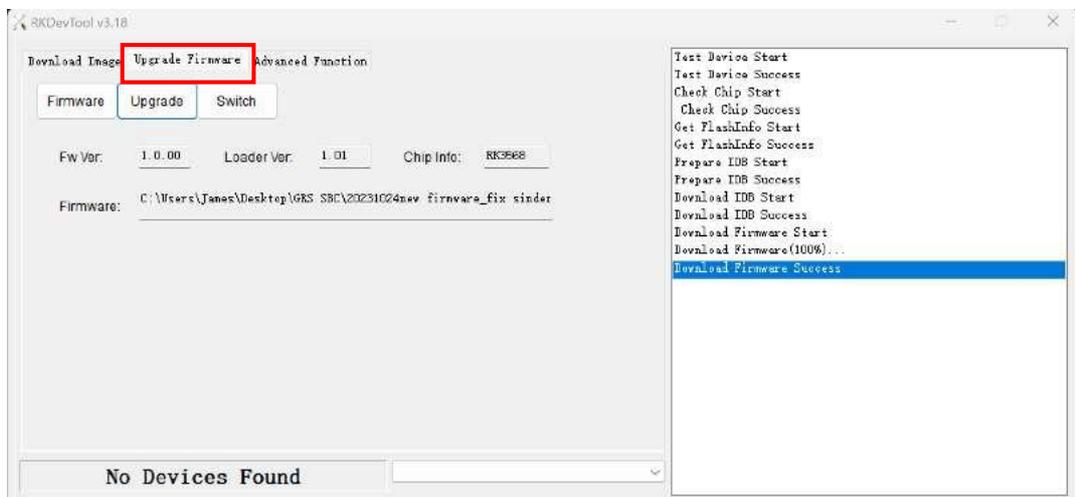
To successfully upgrade the firmware on the Viper SBC's EMMC drive. Note that this process will erase all current data, so a backup is essential.

Pre-Upgrade Steps

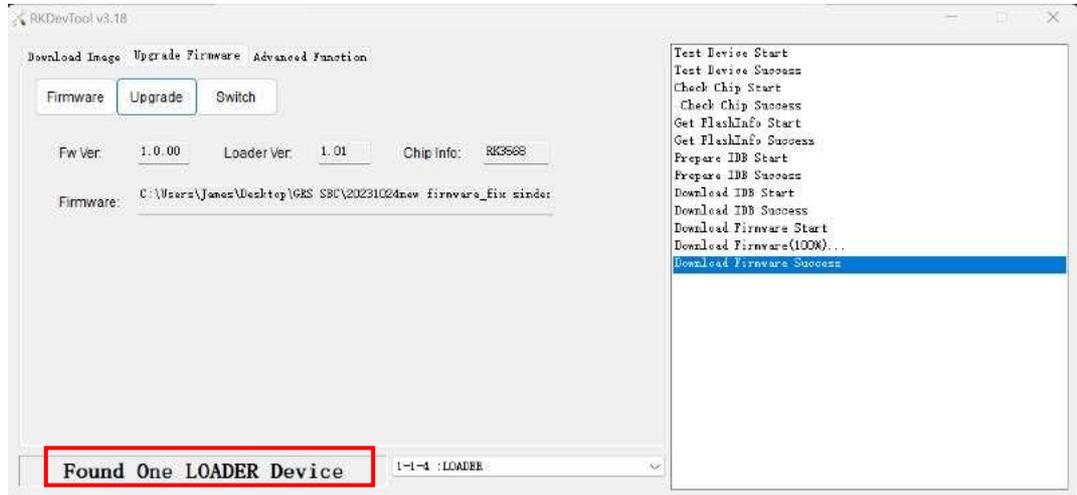
- **Backup Data:** Ensure you've backed up your image to preserve your existing data stored in the Viper SBC's EMMC.

Steps

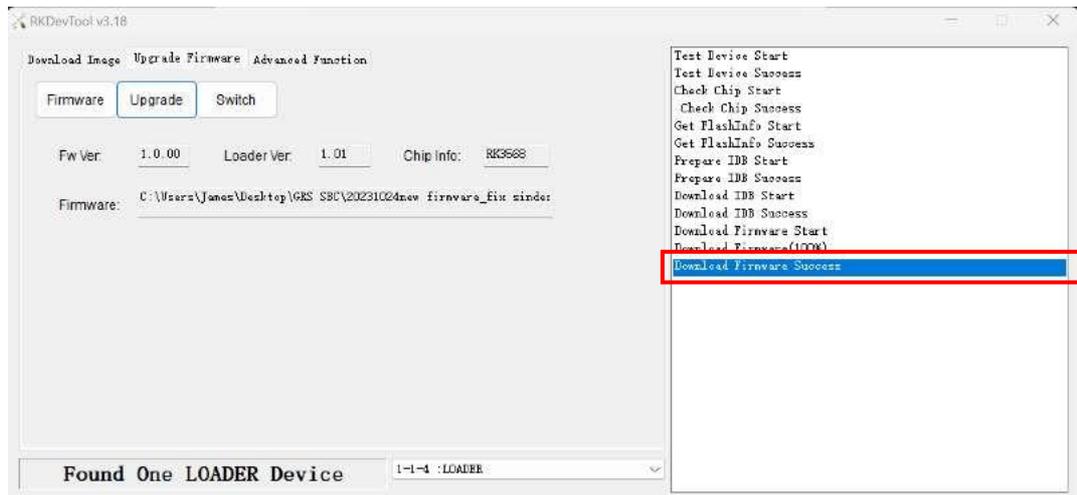
1. **Install Tools and Image:** Download the RKDevTool and RockChip's DriverAssistant software and have your image file (*.img) ready for use. Install the DriverAssistant software. The RKDevTool is executable so no need to install.
DriverAssistant download link:
<https://www.dropbox.com/scl/fo/7mllkizy7i0mkxk8sm0za/h?rlkey=us6vm8e33wz5ogvosiba6bhsz&dl=0>
RKDevTool download link:
<https://www.dropbox.com/scl/fo/sdirr1y6vr0pb6esnrla/h?rlkey=ri93k3d0pmwnmg068h43se4nx&dl=0>
2. **Prepare Cable:** Make sure you have a proper USB-C to USB-A cable on hand. Charging cables will not work for this process.
3. **Connect USB-C:** Plug the USB-C end into the Viper SBC. Note that there should be only one USB-C port on the board. Do not connect the other end to your PC yet.
4. **Open RKDevTool:** Launch RKDevTool and navigate to the 'Upgrade Firmware' tab.



- Firmware Mode:** Press and hold the firmware mode button on the Viper SBC's extension PCB. While holding, plug the USB-A end into your PC. A message "Found One Loader Device" should appear in RKDevTool, indicating the Viper SBC is ready for the firmware upgrade. Release the firmware mode button.



- Select Firmware:** In RKDevTool, click the 'Firmware' button. Select the *.img file you wish to use for the upgrade, and then click 'OK'.
- Start Upgrade:** Click the 'Upgrade' button within the 'Upgrade Firmware' tab to initiate the firmware update process.
- Completion:** Wait for the log to display "Download firmware Success." The Viper SBC will reboot twice to complete the firmware upgrade.



13. Viper SBC GPU Overclock

It's crucial to caution users that overclocking the GPU comes with potential risks. Overclocking can put excessive stress on the chip, leading to potential failure or even burnout. This not only harms the device's longevity but, in extreme cases, poses possible safety hazards. Hence, we strongly advise against overclocking the GPU beyond its standard operational limits.

Steps

1. Set Performance Mode

Use the command `echo performance | tee /sys/bus/cpu/devices/cpu[046]/cpufreq/scaling_governor /sys/class/devfreq/dmc/governor`` followed by `echo performance > /sys/devices/system/cpu/cpufreq/policy0/scaling_governor`` to set the system to performance mode.

2. Specify CPU Frequency (1296MHz/1514MHz)

Set a specific frequency for the CPU with `echo userspace > /sys/devices/system/cpu/cpufreq/policy0/scaling_governor`` and `echo 1514000 > /sys/devices/system/cpu/cpufreq/policy0/scaling_setspeed``.

3. Set GPU to Performance Mode

Apply `echo performance > /sys/devices/platform/fde60000.gpu/devfreq/fde60000.gpu/governor`` and `echo performance > /sys/class/devfreq/fde60000.gpu/governor`` to set the GPU to performance mode.

For more advanced CPU/GPU settings, such as adjusting their voltage, modify the ``rk3568.dtsi``. Proceed cautiously, as this method could potentially damage the chip and void any warranty.

14. Boot from SD card

We have not yet tested with the method in the RockChip website:

[http://opensource.rock-chips.com/wiki/Boot_option#Boot from SD.2FTF Card](http://opensource.rock-chips.com/wiki/Boot_option#Boot_from_SD.2FTF_Card)

For a thorough guide, please refer to the "Boot from SD/TF Card" section on this webpage. Be advised that implementing these steps necessitates a strong understanding of Linux and backend development. We strongly suggest backing up

all valuable files prior to undertaking these procedures.

15. Boot from USB stick

We have not yet tested with the method in the RockChip website:

http://opensource.rock-chips.com/wiki_Boot_option#Boot_from_SD.2FTF_Card

For detailed instructions, please refer to the "Boot from U-Disk" section on this webpage. Keep in mind that carrying out these steps requires substantial background knowledge in Linux and backend development. We highly recommend backing up all crucial files prior to attempting these steps.

16. Boot from eMMC

By default, the Viper SBC is configured to boot from the eMMC. Users and developers interested in understanding this further should refer to the "Boot from eMMC" section in the provided link below:

http://opensource.rock-chips.com/wiki_Boot_option#Boot_from_SD.2FTF_Card

17. Additional Information

This is the end of the document for the GRS Viper SBC, a device that lets you play games and customize your gaming system. We are Thunderstick studio and Glen's Retro Show, a group of enthusiasts who made the Viper SBC to bring joy to the retro gaming community. We welcome any feedback and collaboration from developers who want to help us make the Viper SBC better. If you are interested in learning more about the Viper SBC and its features, you can visit the RockChip's wiki (http://opensource.rock-chips.com/wiki_Main_Page) for more instructions on how to use the RK3566 chip in our build. Thank you for using the Viper SBC and happy gaming!