

Sony PlayStation Portable



The PlayStation Portable (PSP) is a handheld console launched by Sony first in Japan in December 2004, and in the rest of the world in 2005. It's been fairly successful in Japan, but struggled against the Nintendo DS in America and Europe.

The PSP game library is expansive and high quality, made at the peak of quality studios making games for it without having to dedicate as much a budget to their home console equivalents. Most PSP games handle upscaling and HD textures quite nicely.

This system scrapes metadata for the "psp" group and loads the psp set from the currently selected theme, if available.

Quick reference

- **Accepted ROM formats:** .iso, .cso, .pbp
- **Folder:** /userdata/roms/psp

Emulators
PPSSPP
libretro: PPSSPP

BIOS

No PlayStation Portable emulator in Batocera needs a BIOS file to run.

ROMs

Place your PlayStation Portable ROMs in /userdata/roms/psp.

Saves

In Batocera **v33**, PSP game saves are stored in /userdata/system/configs/ppsspp/PSP/SAVEDATA/.

Don't ask why. This will probably change in the future.

Emulators

PPSSPP

PPSSPP is an incredibly performant piece of software. A lot of PSP games run at or near full speed on a Raspberry Pi4 or an Odroid Go Advance/Super.

PPSSPP configuration

Standardized features available to all cores of this emulator: `psp.videomode`, `psp.rewind`, `psp.ratio`


ES setting name batocera.conf_key	Description ⇒ ES option key_value
GRAPHICS BACKEND <code>psp.gfxbackend</code>	Choose your graphics rendering ⇒ OpenGL 0 (OPENGL), Vulkan 3 (VULKAN).
VIDEO RESOLUTION <code>psp.internal_resolution</code>	Improve the fidelity of 3D models (does not affect 2D sprites) ⇒ 1x 480×272 1, 2x 960×544 2, 3x 720p 3, 4x 1080p 4, 5x 2400×1360 5, 6x 1620p 3K 6, 7x 3360×1904 7, 8x 2160p 4K 8, 9x 4320×2448 9, 10x 4800×2720 10.
FRAME SKIP <code>psp.frameskip</code>	Skip frames to improve performance (smoothness) ⇒ Off 0, Autodetect automatic, 1 1, 2 2, 3 3, 4 4, 5 5.
VSYNC INTERVAL <code>psp.vsyncinterval</code>	Enable for more fluidity, disable if screen tearing ⇒ Enable 1, Disable 0.
TEXTURE UPSCALING LEVEL <code>psp.texture_scaling_level</code>	Choose multiple of original texture resolution ⇒ Off 1, Automatic 0, 2x 2, 3x 3, 4x 4, 5x 5.
TEXTURE UPSCALING TYPE <code>psp.texture_scaling_type</code>	Choose upscaling method when upscaling is turned on ⇒ xBRZ 0, Hybride 1, Bicubique 2, Hybride + Bicubique 3.
TEXTURE DEPOSTERIZE <code>psp.texture_deposterize</code>	Fix texture banding when upscaling ⇒ Off False, On True.
ANISOTROPIC FILTERING <code>psp.anisotropic_filtering</code>	Enhance the quality of distant perspective textures ⇒ Off 0, 2x 1, 4x 2, 8x 3, 16x 4.
GAMES CHEATS <code>psp.enable_cheats</code>	For cheating in games as with Action Replay ⇒ Off False, On True.

RetroArch

RetroArch (formerly SSNES), is a ubiquitous frontend that can run multiple “cores”, which are

essentially the emulators themselves. The most common cores use the [libretro](#) API, so that's why cores run in RetroArch in Batocera are referred to as "libretro: (core name)". RetroArch aims to unify the feature set of all libretro cores and offer a universal, familiar interface independent of platform.

RetroArch configuration

RetroArch offers a **Quick Menu** accessed by pressing [HOTKEY] +  which can be used to alter various things like [RetroArch and core options](#), and [controller mapping](#). Most RetroArch related settings can be altered from Batocera's EmulationStation.

Standardized features available to all libretro cores: `psp.videomode`, `psp.ratio`, `psp.smooth`, `psp.shaders`, `psp.pixel_perfect`, `psp.decoration`, `psp.game_translation`

ES setting name batocera.conf_key	Description ⇒ ES option key_value
Settings that apply to all cores of this emulator	
GRAPHICS BACKEND <code>psp.gfxbackend</code>	Choose your graphics rendering ⇒ OpenGL <code>opengl</code> , Vulkan <code>vulkan</code> .
AUDIO LATENCY <code>psp.audio_latency</code>	Audio latency in milliseconds, turn it up if you hear crackles ⇒ 256 256, 192 192, 128 128, 64 64, 32 32, 16 16, 8 8.
THREADED VIDEO <code>psp.video_threaded</code>	Improves performance at the cost of latency and more video stuttering. Use only if full speed cannot be obtained otherwise. ⇒ On <code>true</code> , Off <code>false</code> .

libretro: PPSSPP

libretro: PPSSPP configuration

All configuration must be done within RetroArch's **Quick Menu** ([HOTKEY] + ).

PSP upscaling and enhancements

One of the best features of PPSSPP is its ability to enhance the visual quality of PSP games. It's particularly interesting for 3D models, but even 2D fonts and textures can be enhanced through HD textures replacement (see the chapter below).

From the Batocera EmulationStation menu, you can enable several enhancement by entering the **ADVANCED GAME OPTIONS** menu:

- Video Resolution: to improve 3D modeling
- Frame Skip: skip frames to improve performance (but a less smooth experience)
- Textures Upscaling: automatically upscale textures for 3D models
- Texture Enhancement: scales up a nearest-neighbor version of a texture
- Texture Deposterize: fix an artifact causing bands to appear on the textures
- Anisotropic Filtering: enhance the quality with perspective on texture




PSP textures packs

You can replace the original textures from a PSP game with High Definition versions, in order to make it look better, especially when playing on a large screen with Batocera.

PSP texture packs can be easily found online, and a good starting point could be [this thread on PPSSPP forums](#).

Replacing textures

First we need to dump the game's textures:

1. Create the saves directory for the game you want to replace the textures from. The easiest way is to launch the PSP game from EmulationStation as usual and create a savestate ([HOTKEY] + ). It will create a saves directory path in /userdata/saves/psp/PSP/. Create a sub-folder TEXTURES as /userdata/saves/psp/PSP/TEXTURES/.
2. Once you are in the game, press [HOTKEY] +  to open the main PPSSPP emulator menu.
 - Go to **Settings** → **Tools** → **Developer Tools**.
 - Then, in the Developer menu, go to the **Texture Replacement** section at the end of the menu, make sure that **Replace textures** is ticked, and tick also **Save new textures**.
 - Get back to the game, and play a little bit to let the emulator dump the textures on the SD card/hard drive. The gameplay might lag a bit because of the texture dump.
 - Once a few texture files have been dumped, return to the PPSSPP Emulator menu with [HOTKEY] +  and unset the **Save new textures** option that was set earlier.



Be sure to keep **Replace textures** on!

Now we just need to replace those textures:

1. In /userdata/saves/psp/PSP/TEXTURES/<game ID>/ with the ID for the game you played, inside this directory you'll find a sub-directory called new with a bunch of .png files, corresponding to the first textures that have been dumped. We won't use them, we will replace them with the enhanced texture pack that you downloaded. For example, for Wipeout Pure, the directory for the replacement textures is /userdata/saves/psp/PSP/TEXTURES/UCES00001/.
2. Unzip the texture pack you downloaded for this game. A texture pack comes with a textures.ini file that describes all the textures that will be replaced in the game. You can check that it corresponds to the game you have, usually the codename of the game is referenced in the textures.ini file.



Specificity would be nice.

3. Once you have unzipped everything you can remove the new directory that you dumped

previously, it won't be used any more. In my example, the resulting files are:

UCES00001				
Name	Date Modified	Size	Kind	
> font	3:48 PM	--	Folder	
> in_game	3:48 PM	--	Folder	
> intro	3:48 PM	--	Folder	
> main_menu	3:48 PM	--	Folder	
README.md	3/18/21	584 bytes	Document	
textures.ini	3/18/21	363 KB	Document	

Now, it's time to launch your game through PPSSPP again... and enjoy beautiful textures for a much more modern look!

Notes about texture packs

Texture packs are pretty heavy! That actually is the largest component in modern video games, and that explains why games went from being a few kilobytes to several gigabytes by today's standards. That said, some PSP games, when upscaled to HD or 4K with the right texture packs look absolutely **gorgeous**. It's a pleasure to rediscover some of your favorite games with a higher standard.

See below the difference it can make, **click** on the thumbnail below to actually see the details in the new texture pack.



Original PSP fonts

PPSSPP uses different fonts than the original PSP fonts from Sony. Some games rely on these original system fonts, not on fonts embedded in the game, and it can provide visual artifacts when an unexpected font is used, or even some words completely missing. For example:



If you have access to Sony's fonts (from your PSP firmware or through a set of files), you need to:

1. Replace all the files in `/usr/share/ppsspp/PPSSPP/flash0/font` with Sony's files. Overwrite all the `ltn*.pgf`, `jpn0.pgf` and `krn0.pgf` existing files, and put the additional ones in this folder.
2. Do a `batocera-save-overlay` from the console or SSH to keep these files saved upon reboot (they are in the system folder, not in `/userdata/`).

Once this is done, you should have the new original PSP fonts on screen, like on the right picture above.

No system font when using libretro-ppsspp core?

The system fonts, that you can see when saving/loading data are working with PPSSPP standalone, but not the libretro-ppsspp core? That's a known issue and there is a workaround for it:

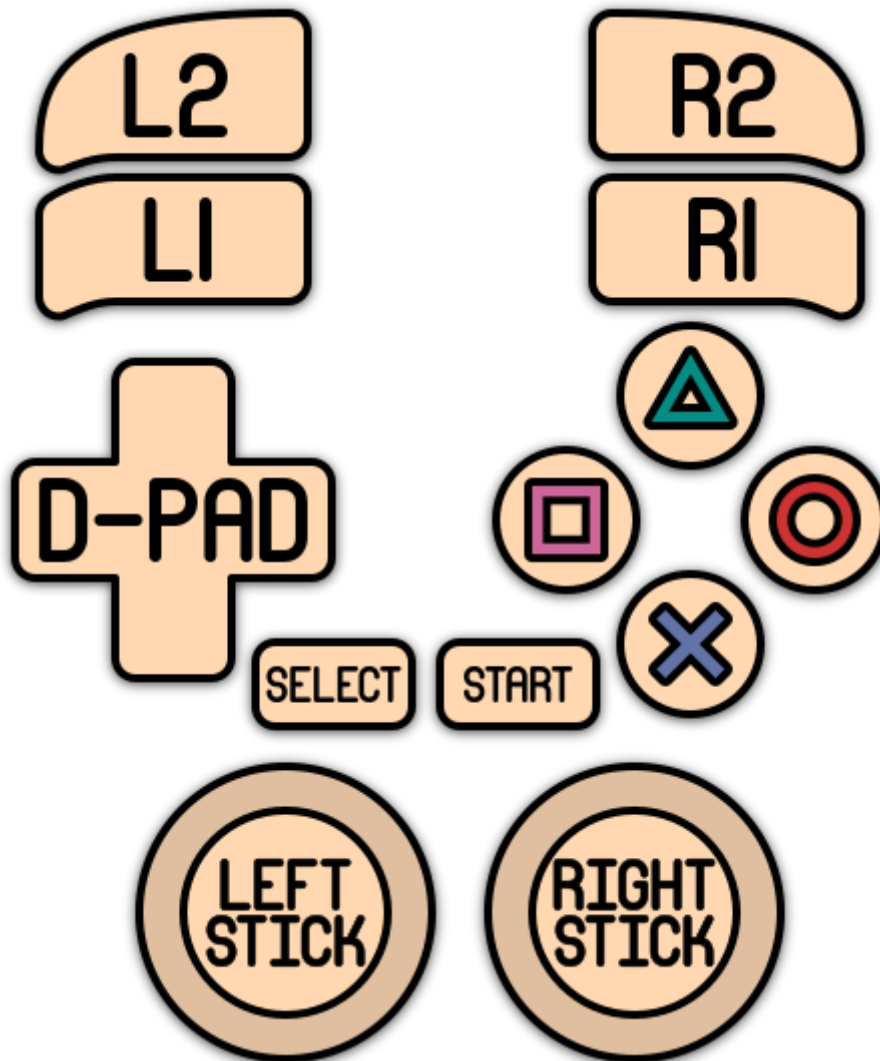
- If your `/userdata` file system supports symbolic links (ext4 the default filesystem or btrfs), you can [log into Batocera through SSH](#) and type `ln -sf /usr/share/ppsspp/PPSSPP /userdata/bios/`
- If you don't know if your `/userdata` file system supports symbolic links or not, you can just make a copy of the files once you are [logged with SSH](#) with `cp -af /usr/share/ppsspp/PPSSPP /userdata/bios/` This copy method works in all cases.

RetroAchievements

[RetroAchievements](#) are supported from Batocera **v33** and higher. To enable them make sure to choose the libretro-ppsspp core and that the image format is `*.iso`. Compressed `*.cso` images are not supported by libretro at the moment.

Controls

Here are the default Sony Playstation Portable's controls shown on a [Batocera Retropad](#):



You might be thinking: "Right stick? Huh? The PSP didn't have a right stick!" and you would be right, the PSP has no physical right stick.

But it has a virtual one.

Maybe it was coded like this for a potential future accessory that never got developed, but technically all games could support input via an additional analog stick. Fortunately, there are some community game patches available that can retroactively add right stick support to particular games, and there are some homebrew that natively take advantage of it. Batocera simply assigns this to your modern controller's right stick in the rare case that you use this functionality, if you don't then it's completely safe to ignore it forever.



Make a version of the overlay that hides the right stick by default for those who want to say give the system to people and not to give them false hopes.

Troubleshooting

Further troubleshooting

For further troubleshooting, refer to the [generic support pages](#).

From:

<https://www.wiki.batocera.org/> - **Batocera.linux - Wiki**

Permanent link:

<https://www.wiki.batocera.org/systems:psp?rev=1639548152>

Last update: **2021/12/15 06:02**

