Advanced RetroArch Settings

You might want to check if the option you want to change has already been implemented into EmulationStation! Most game-related settings are in GAME SETTINGS, but some RetroArch specific options may be in the DEVELOPER menu.

With ordinary distributions (such as installing RetroArch on Windows), RetroArch is configured by adding/editing lines in retroarch.cfg. Batocera generates and replaces the lines in this file with every emulator launch, so the better option is to let Batocera know which RetroArch options you would like to use. Most of these settings can instead be changed within EmulationStation. Exceptions will be noted.

For options not already implemented into ES, you can change the file /userdata/system/batocera.conf and add the following keys if needed. Most RetroArch settings not mentioned in the default batocera.conf can be edited and saved from within RetroArch itself (accessed by pressing [HOTKEY] while in-game in a Libretro core), these will be saved to /userdata/configs/retroarch/retroarchcustom.cfg (with core options being saved to retroarch/cores/retroarch-core-options.cfg instead), however the vast majority of RetroArch settings will be overridden by Batocera at emulator launch.

In order to use the configuration generator to create a RetroArch setting (letting Batocera know which value is being used, avoiding it being overwritten), follow the global.retroarch.retroarch_setting=value syntax. Batocera supports the massive majority of RetroArch's settings, but for the few that it doesn't (such as bleeding edge new features or obscure/forgotten ones), those settings can still be adjusted by editing them directly into /userdata/configs/retroarch/retroarchcustom.cfg as they will not be overwritten (at least, until Batocera supports them as well).

RetroArchievements

RetroArchievements is a website to log your achievements on several console games. It's built into RetroArch and is only supported by a selection of cores. Available from GAMES SETTINGS → RETROACHIEVEMENTS SETTINGS.

You first need a valid account there and then add your credentials inside the configuration file. Get more detailed information from the RetroArchievements Settings page.

```json
## Enable retroachievements (0,1)
## Set your www.retroachievements.org username/password
## Escape your special chars (# ; $) with a backslash : $ => \\

global.retroachievements=0
global.retroachievements.hardcore=0
global.retroachievements.leaderboards=0
global.retroachievements.verbose=0
global.retroachievements.screenshot=0
```
A.I. Translation

The following section enables an automatic translation of in-game dialogs in your language. So it's possible to play Japanese game titles without hassle. Available from **GAMES SETTINGS → AI GAME TRANSLATION**

```yaml
## Enable RetroArch AI game translation service
global.ai_service_enabled=0
global.ai_service_url=http://ztranslate.net/service?api_key=BATOCERA
global.ai_target_lang=
```

Cheats

Auto apply cheats selected by the user.

```yaml
## Auto apply cheats selected by the user
global.retroarch.apply_cheats_after_load=true
global.retroarch.apply_cheats_after_toggle=true
```

Savestate Screenshots

Take a screenshot of the savestate. For when you have many savestates and don't remember which is which... now in RetroArch its thumbnail will be a screenshot of the moment you saved! Not available as an ES option.

```yaml
## Take a screenshot of the savestate
global.retroarch.savestate_thumbnail_enable=true
```

Start Screen Message

Hide the welcome message in Retroarch. Having to wait for 3 secs before the message goes away is really boring. This has become the default setting in Batocera. Not available as an ES option.

```yaml
## Hide the welcome message in Retroarch
global.retroarch.rgui_show_start_screen=false
```
**All Users Control Menu**

Allow any RetroPad to control the menu. This changes nothing in 1 player mode, but it's a dream to disable when you have friends who don't know the emulators or young children... Only Player 1 (YOU) can use RetroArch options like savestate, change slot, rewind, etc. A much better gaming experience. Not available as an ES option.

```bash
## Allow any RetroPad to control the menu (Only the player 1)
global.retroarch.all_users_control_menu=false
```

**Swap OK and Cancel buttons**

Swap the OK and CANCEL buttons on Retroarch menu. Many users have SNES gamepads and in EmulationStation, B is OK and A is NO. And in RetroArch, buttons are inverted, and not really convenient. Not available as a menu option.

```bash
## Swap the Ok and the CANCEL button on Retroarch menu
global.retroarch.menu_swap_ok_cancel_buttons=true
```

**OSD Badges**

Enable OSD (on-screen display) messages. Retroarch shows users some important messages, like Savestate, Screenshot, RetroAchievements, etc. They look beautiful on badges. But sometimes, it shows other less important messages, like your name when you connect to RetroAchievements. Those messages look like crappy yellow text lines, not like the badges. This option hides these messages but not the badges. This has become the default option in Batocera. Not available as an ES option.

```bash
## Enable usage of OSD messages (Text messages not in badge)
global.retroarch.video_font_enable=false
```

**Sync to Exact Content Framerate (G-Sync, FreeSync)**

This will enable variable refresh rate for compatible displays. This sync your display's refresh rate to the emulated console's display rate (unlike traditional v-sync that will simply delay the frames until the display is ready). Only compatible on displays that explicitly have support for “variable refresh rate” (most of them don't). Not available as an ES option.

```bash
## Enables syncing to exact content framerate (variable refresh rate, G-sync, Freesync, etc.)
global.retroarch.vrr_runloop_enable=true
```
Core options

It might be worth checking if the option you want to set is already available in the ADVANCED SYSTEM OPTIONS first! You can quickly access this for your system by pressing [SELECT] in the system's game list.

In order to set specific core options (ones that are saved to userdata/system/configs/retroarch/cores/retroarch-core-options.cfg (may cause conflicting issues, advanced users only) use the global.retroarchcore.<core name>_<core option>=value syntax (takes priority over other settings, even settings that are normally required to be adjusted to run games, be warned). For example: global.retroarchcore.genesis_plus_gx_region_detect=pal would set the region to PAL for games run with GenesisPlusGX.

Although you can use global. to apply the core option to all systems that core supports, it might be wise to specify the system you want to apply it to (considering that most cores only support a single system anyway).

Each core option is (usually) unique to the core, you can see what terminology it uses with the following:

1. Launch a game and go into the RetroArch Quick Menu and navigate to the core's options ([HOTKEY]+→ Options).
2. Edit your options if you want to, and then go to Manage Core Options → Save Content Directory Options.
3. Check the newly created file at system/.config/retroarch/config/<core name>/<system>.opt

   You may need to turn on “Show hidden folders” in your file manager to see this folder.

4. Transplant those options to batocera.conf by appending them with global.retroarchcore. first. It's okay to enclose the values in double-quotes like RetroArch does (eg. = "value") or not include any double-quotes like Batocera does (eg. = value).
5. Delete the system/.config/retroarch/config/<core name>/<system>.opt file (or just Delete Content Directory Options (which has replaced Save Content Directory Options from earlier)).

To run with our previous example, I would:

1. Launch “Shaq Fu” for the Sega Megadrive, mash [HOTKEY]+ and then go into the Options menu.
2. Change the **System region** option from “auto” to “pal”.
3. Go into **Manage Core Options** and then **Save Content Directory Options**.
4. Check the option file that was created at system/.config/retroarch/config/Genesis Plus GX/megadrive.opt

5. Take a gander at the option I set, and note that it is genesis_plus_gx_region_detect = "pal".

6. Open up batocera.conf and add
   
   global.retroarchcore.genesis_plus_gx_region_detect = "pal"

7. Delete the megadrive.opt file.

All my Mega Drive games now run at PAL speeds, as they should be. 😊
Reconfiguring the Keyboard for RetroArch

This is for if you'd like to reconfigure the default keyboard bindings for all systems emulated by a libretro core (such as if you don't have a controller to use or you're in an all-in-one device like a netbook). Setting this will prevent you from using a controller configured from ES in libretro cores until you undo this. Not available as an ES option.

Keep in mind that this is just translating your keyboard to the RetroPad. The RetroPad is independent of the system and will be assigned to the appropriate system controls when that emulator is booted up. You can read more about that and its philosophy here.

Open your /system/batocera.conf file and add the following lines as appropriate to it. Here is an example of the default bindings:

```plaintext
# Face buttons
global.retroarch.input_player1_a=x
global.retroarch.input_player1_b=z
global.retroarch.input_player1_x=s
global.retroarch.input_player1_y=a

# D-pad
global.retroarch.input_player1_down=down
global.retroarch.input_player1_left=left
global.retroarch.input_player1_right=right
global.retroarch.input_player1_up=up

# Shoulder buttons
global.retroarch.input_player1_l=q
global.retroarch.input_player1_r=w
```
global.retroarch.input_player1_l2=nul
global.retroarch.input_player1_r2=nul

# Start and Select
global.retroarch.input_player1_select = rshift
global.retroarch.input_player1_start = enter

# Left analog
global.retroarch.input_player1_l_x_minus=nul
global.retroarch.input_player1_l_x_plus=nul
global.retroarch.input_player1_l_y_minus=nul
global.retroarch.input_player1_l_y_plus=nul

# Right analog
global.retroarch.input_player1_r_x_minus=nul
global.retroarch.input_player1_r_x_plus=nul
global.retroarch.input_player1_r_y_minus=nul
global.retroarch.input_player1_r_y_plus=nul

# Analog buttons
global.retroarch.input_player1_l3=nul
global.retroarch.input_player1_r3=nul

# The Hotkey
global.retroarch.input_enable_hotkey=nul

If you'd like your keyboard bindings to only apply to a particular system, replace the global part with the shortname for your system (eg. psx for the Sony PlayStation).

It must be stressed that this is only for libretro cores; this does nothing for standalone emulators! Refer to this page for more info.

Rebinding RetroArch's hotkeys

If you were looking for how to change the functions of your handheld's function keys, refer to this page instead.

The hotkey combos that Batocera offers are nice, but what if you wanted to bind a hotkey action directly to an unused button (such as putting fast-forward on the unused [R2] button for GBA games)? This is for that. Once again, keep in mind this will only affect libretro cores, not standalone emulators.

Open your /system/batocera.conf file and add the following lines:

```plaintext
## The hotkey enable button. Equivalent to Batocera's [HOTKEY]. Making this "nul" will disable the requirement for holding the hotkey first to activate
```

commands with button presses.

```
global.retroarch.input_enable_hotkey_btn=nul
```

Keep in mind that you really should specify the system you want to make the edits for by replacing `global.` with `<system shortname>`, applicable to that system (eg. `psx.retroarch.<setting>` for the Sony PlayStation). Otherwise, more complicated systems that utilize buttons which you assign to hotkey functions here will activate their button commands instantly without first pressing the hotkey, interrupting gameplay.

You'll have to figure out what your internal button names are using `evtest` from the terminal/`ssh` session. Here is a list of the common Batocera names (using an Xbox 360 controller):

---

<table>
<thead>
<tr>
<th>Event code</th>
<th>RetroArch binding</th>
<th>Type</th>
<th>Label</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>304</td>
<td>0</td>
<td>BTN</td>
<td>SOUTH</td>
<td></td>
</tr>
<tr>
<td>305</td>
<td>1</td>
<td>BTN</td>
<td>EAST</td>
<td></td>
</tr>
<tr>
<td>307</td>
<td>3</td>
<td>BTN</td>
<td>NORTH</td>
<td></td>
</tr>
<tr>
<td>308</td>
<td>2</td>
<td>BTN</td>
<td>WEST</td>
<td></td>
</tr>
<tr>
<td>310</td>
<td>9</td>
<td>BTN</td>
<td>TL</td>
<td>[L1], can also appear as “pageup”</td>
</tr>
<tr>
<td>311</td>
<td>5</td>
<td>BTN</td>
<td>TR</td>
<td>[R1], can also appear as “pagedown”</td>
</tr>
<tr>
<td>314</td>
<td>8</td>
<td>BTN</td>
<td>SELECT</td>
<td>[SELECT]</td>
</tr>
<tr>
<td>315</td>
<td>7</td>
<td>BTN</td>
<td>START</td>
<td>[START]</td>
</tr>
<tr>
<td>316</td>
<td>?</td>
<td>BTN</td>
<td>MODE</td>
<td>[HOTKEY]</td>
</tr>
<tr>
<td>317</td>
<td>?</td>
<td>BTN</td>
<td>THUMBL</td>
<td>[L3], pushing the left stick in</td>
</tr>
<tr>
<td>318</td>
<td>?</td>
<td>BTN</td>
<td>THUMBR</td>
<td>[R3], pushing the right stick in</td>
</tr>
<tr>
<td>0</td>
<td>?</td>
<td>ABS</td>
<td>X</td>
<td>Horizontal axis of the left stick</td>
</tr>
<tr>
<td>1</td>
<td>?</td>
<td>ABS</td>
<td>Y</td>
<td>Vertical axis of the left stick</td>
</tr>
<tr>
<td>2</td>
<td>?</td>
<td>ABS</td>
<td>Z</td>
<td>[L2], Left analog trigger</td>
</tr>
<tr>
<td>3</td>
<td>?</td>
<td>ABS</td>
<td>RX</td>
<td>Horizontal axis of the right stick</td>
</tr>
<tr>
<td>4</td>
<td>?</td>
<td>ABS</td>
<td>RY</td>
<td>Vertical axis of the right stick</td>
</tr>
<tr>
<td>5</td>
<td>?</td>
<td>ABS</td>
<td>RZ</td>
<td>[R2], Right analog trigger</td>
</tr>
<tr>
<td>17</td>
<td>h0up/h0down</td>
<td>ABS</td>
<td>HAT0Y</td>
<td>D-pad up/down, value -1 for up, 0 for neutral, 1 for down</td>
</tr>
<tr>
<td>16</td>
<td>h0left/h0right</td>
<td>ABS</td>
<td>HAT0X</td>
<td>D-pad left/right, value -1 for left, 0 for neutral, 1 for right</td>
</tr>
</tbody>
</table>

BTN = Button, ABS = Absolute position (commonly analog sticks, but also includes analog triggers and the D-pad (yep, the D-pad is just a virtual analog stick)).

Then add/edit the following lines as appropriate to `batocera.conf`. Here is a non-exhaustive list of
RetroArch input commands (with the default buttons for an Xbox 360 controller):

```plaintext
## Exit the current emulator. Equivalent to [HOTKEY]+[START].
global.retroarch.input_exit_emulator_btn=7

## Open RetroArch's Quick Menu. Equivalent to [HOTKEY]+[SOUTH].
global.retroarch.input_menu_toggle_btn=0
## Save current state to the selected slot. Equivalent to [HOTKEY]+[WEST].
global.retroarch.input_save_state_btn=2
## Load the state from the selected slot. Equivalent to [HOTKEY]+[NORTH].
global.retroarch.input_load_state_btn=3
## Reset the emulated system (can cause bugs with certain settings/games).
Equivalent to [HOTKEY]+[EAST].
global.retroarch.input_reset_btn=1
## Increase the currently selected save state slot. Equivalent to [HOTKEY]+[D-PAD UP].
global.retroarch.input_state_slot_increase_btn=h0up
## Decrease the currently selected save state slot. Equivalent to [HOTKEY]+[D-PAD DOWN].
global.retroarch.input_state_slot_decrease_btn=h0down

## Toggle the AI translation service. Equivalent to [HOTKEY]+[R1]
global.retroarch.input_ai_service_btn=5
## Fast-forward (hold). Equivalent to [HOTKEY]+[D-PAD RIGHT]
global.retroarch.input_hold_fast_forward_btn=h0right
## Rewind. Equivalent to [HOTKEY]+[D-PAD LEFT]
global.retroarch.input_rewind_btn=h0left
## Take a screenshot. Equivalent to [HOTKEY]+[L1].
global.retroarch.input_screenshot_btn=9
## Next shader. Equivalent to [HOTKEY]+[R2].
global.retroarch.input_shader_next_btn=nul
## Previous shader. Equivalent to [HOTKEY]+[L2].
global.retroarch.input_shader_prev_btn=nul

## Hotkey (for holding). Setting this to anything but nul will re-enable the combo requirement.
global.retroarch.input_enable_hotkey_btn=8
## Eject the virtual disc.
global.retroarch.input_disk_eject_toggle_btn=4
## Select the next disc.
global.retroarch.input_disk_next_btn=nul
## Select the previous disc.
global.retroarch.input_disk_prev_btn=nul
## Toggle the FPS display.
global.retroarch.input_fps_toggle_btn=nul
## Advance one frame.
global.retroarch.input_frame_advance_btn=nul
## Slow-motion (hold).
global.retroarch.input_hold_slowmotion_btn=nul
## Record movie toggle.
global.retroarch.input_movie_record_toggle_btn=nul
```
If you'd like to be able to open the **Quick Menu** with an alternative button combo, you can utilize any of RetroArch's built-in combos:

## Only open the RetroArch Quick Menu with the hotkey command.
```
global.retroarch.input_menu_toggle_gamepad_combo=0
```

## Open the RetroArch Quick Menu with Down + Y + L1 + R1.
```
global.retroarch.input_menu_toggle_gamepad_combo=1
```

## Open the RetroArch Quick Menu with L3 + R3.
```
global.retroarch.input_menu_toggle_gamepad_combo=2
```

## Open the RetroArch Quick Menu with L1 + R1 + Start + Select.
```
global.retroarch.input_menu_toggle_gamepad_combo=3
```

## Open the RetroArch Quick Menu with Start + Select.
```
global.retroarch.input_menu_toggle_gamepad_combo=4
```

## Open the RetroArch Quick Menu with L3 + R1.
```
global.retroarch.input_menu_toggle_gamepad_combo=5
```

## Open the RetroArch Quick Menu with L1 + R1.
```
global.retroarch.input_menu_toggle_gamepad_combo=6
```

## Open the RetroArch Quick Menu by holding Start for 2 seconds.
```
global.retroarch.input_menu_toggle_gamepad_combo=7
```

## Open the RetroArch Quick Menu by holding Select for 2 seconds.
```
global.retroarch.input_menu_toggle_gamepad_combo=8
```

## Open the RetroArch Quick Menu with Down + Select.
```
global.retroarch.input_menu_toggle_gamepad_combo=9
```

## Open the RetroArch Quick Menu with L2 + R2.
```
global.retroarch.input_menu_toggle_gamepad_combo=10
```
From: https://wiki.batocera.org/ - Batocera.linux - Wiki

Permanent link: https://wiki.batocera.org/advanced_retroarch_settings

Last update: 2022/02/09 06:18