

# Borne d'arcade BioHackLab

(page créée le 7 février 2025)

Une borne pour jouer aux jeux réalisés pendant les ateliers / game jam

## Matériel

Écran Gateway 19" TFT LCD, connexion par VGA avec adaptateur. Résolution 1280 x 1024, 59.907Hz

## Préparation Raspberry Pi

Quoi de neuf sur Debian 12 Bookworm pour Raspberry ?

<https://www.framboise314.fr/bookworm-la-nouvelle-version-de-raspberry-pi-os/>

- Adresse MAC Wifi : d8:3a:dd:ee:23:f3
- Adresse MAC ethernet : d8:3a:dd:ee:23:f2

### Préparer la carte micro-SD avec l'OS

C'est très simple en utilisant Raspberry Pi Imager qui s'occupe de préparer la carte micro-SD avec un OS choisi dans une liste assez garnie.

Installation de Raspberry Pi Imager à télécharger ici : <https://www.raspberrypi.com/software/> On peut le démarrer avec

```
rpi-imager &
```

Ce logiciel permet de choisir l'OS à installer (ici : Raspberry Pi OS 64-bit = Debian Bookworm) et de définir quelques réglages :

- nom de l'hôte : biohacklab.local
- réseau wifi : FabLab\_24
- utilisateur : xor / lpl
- SSH activé (si jamais on oublie de l'activer il est toujours possible d'[activer SSH sur la carte microSD](#))

### Afficher l'adresse IP du rpi

```
hostname -I ou ip addr
```

### Se connecter au rpi en SSH

```
ssh xor@192.168.1.35
```

### Copier un fichier vers le rpi

```
scp ./fondecran_01.png xor@192.168.1.35:/home/xor/borne_biohacklab_ui_v5
```

### Premier démarrage

- pas d'image sur le moniteur (branché avec un adaptateur micro-HDMI / VGA)
- biohacklab.local ne répond pas
- `sudo nmap -sP 192.168.1.*` : le rpi n'apparaît pas

Passage obligatoire sur <https://www.raspberrypi.com/documentation/computers/getting-started.html>

On peut aussi trouver l'adresse IP à partir de l'adresse MAC

```
sudo nmap -sn 192.168.1.1/24|awk '/^Nmap/{ip=$NF}/D8:3A:DD/{print ip}'
```

On peut trouver l'adresse MAC du Pi en ligne de commande avec `ifconfig`

Même si `nmap` ne «voit» pas le rpi, on peut s'y connecter en SSH!

```
ssh xor@biohacklab.local
ssh xor@192.168.1.30
```

## Ressources générales sur le raspberry pi

- cheat sheet : <https://gist.github.com/hofmannsven/9132419>
- keyboard shortcuts : <https://g7rdx.co.uk/linux/raspberry-keyboard-shortcuts/>

## Questions d'écran

Essai pour configurer le moniteur directement dans le fichier `/boot/firmware/config.txt` > sans succès  
Boot avec HDMI branché sans adaptateur : ok

### Faire fonctionner l'adaptateur micro-HDMI - VGA

Il s'agit de ce modèle acheté chez Smallcab : [adaptateur micro HDMI / VGA](#)

Après un premier essai non concluant sur un moniteur, le second testé a fonctionné!

### Problème de moniteur

L'affichage "saute", puis le message "video mode not supported" s'affiche. On modifiant le fichier `/boot/firmware/cmdline.txt` et en ajoutant au début :

```
video=HDMI-A-2:720x576M@50
```

ça fonctionne de nouveau ...

mais ce n'est pas du 720x576, il doit y avoir un autre composant qui agit après le démarrage

doc : <https://www.raspberrypi.com/documentation/computers/configuration.html#command-line-options>

voir aussi : <https://pimylifeup.com/raspberry-pi-screen-resolution/>

le rpi est-il en wayland ou en x ? pour l'adaptateur surtout pas le D! (dans `video=HDMI-A-2:720x576M@50D`) D pour digital, A pour analog

Réponse : en wayland ( <https://www.framboise314.fr/bookworm-la-nouvelle-version-de-raspberry-pi-os/> )

## Chromium en mode kiosque

On trouve pas mal de choses sur le sujet mais en grande partie obsolète car le système du raspberry pi change au fil du temps...

La meilleure référence semble être : <https://forums.raspberrypi.com/viewtopic.php?t=378883>

Avant tout : **on sort du «kiosk mode» avec `Alt` + `F4`, ou `Ctrl` + `↑ Shift` + `Backspace`**

### Refs plus ou moins obsolètes :

- <https://www.raspberrypi.com/tutorials/how-to-use-a-raspberry-pi-in-kiosk-mode/>
- <https://www.scalzotto.nl/posts/raspberry-pi-kiosk/>
- <https://www.airdroid.com/mdm/chromium-kiosk-mode/>
- [https://wiki.houptlab.org/index.php?title=Raspberry\\_Pi\\_Kiosk&mobileaction=toggle\\_view\\_desktop](https://wiki.houptlab.org/index.php?title=Raspberry_Pi_Kiosk&mobileaction=toggle_view_desktop)
- <https://reelyactive.github.io/diy/pi-kiosk/>

## Configurer le démarrage automatique

### Choix d'un serveur

Plusieurs possibilités exposées ici : <https://github.com/processing/p5.js/wiki/Local-server>

Essai avec SimpleHTTPServer en python qui est déjà installé (

<https://github.com/processing/p5.js/wiki/Local-server#python-simplehttpserver> )

On le lance avec `python -m http.server` depuis un répertoire qui devient la racine du serveur.

### Masquer la souris

On peut installer le paquet `unclutter` pour masquer le pointeur de souris quand il n'y a pas d'activité

```
sudo apt install unclutter
```

**NON**, ça ne fonctionne plus sur Debian 12 Bookworm avec Wayland :/

Autre solution à tester ici : <https://forums.raspberrypi.com/viewtopic.php?t=358285#p2175588>

### Masquer automatiquement la barre des tâches

Ajouter les lignes suivantes dans le fichier `.config/wf-panel-pi.ini`

```
autohide=true
autohide_duration=500
heightwhenhidden=0 # test sinon la barre est à moitié visible ...
```

Problème maintenant, la barre ne réapparaît plus au survol de la souris! Après un redémarrage, elle réapparaît à moitié et reste fixe! TODO : améliorer

Exemple très complet du fichier `wf-panel-pi.ini` : <https://github.com/WayfireWM/wf-shell/blob/master/wf-shell.ini.example>

## Étape 1 : démarrer sans logging manuel

```
sudo raspi-config
```

Sélectionner : System Options > Boot / Auto Login > Desktop Autologin: Desktop GUI, automatically logged in as 'xor' user.

## Étape 2 : préparer le fichier autostart

Créer le fichier

```
cd /home/xor/.config/labwc
touch autostart
nano autostart
```

Pour le modifier par la suite

```
nano /home/xor/.config/labwc/autostart
```

Le fichier autostart contient

```
cd /home/xor/borne_biohacklab_ui_v5/
/usr/bin/python -m http.server & # <- & : important de lancer le processus en arrière-plan, sinon la suite du script ne se réalise pas
# unclutter -idle 1 & # masquer le pointeur après 1 seconde d'inactivité, ne fonctionne pas pas sur Debian 12 avec Wayland
sleep 8
chromium --start-fullscreen --start-maximized --noerrdialogs --disable-infobars --kiosk http://localhost:8000
```

## Étape 3 : quand tout fonctionne, passer en lecture seule

```
sudo raspi-config
```

Dans les menus :

- sélectionner Performance options
- sélectionner Overlay File System
- confirmer l'activation de «Overlay File System»
- confirmer la protection en écriture de la partition de boot
- confirmer le reboot

Voir :

<https://pip.raspberrypi.com/categories/685-whitepapers-app-notes/documents/RP-003610-WP/Making-a-more-resilient-file-system.pdf>

### Rétablir la partition en lecture / écriture

Temporairement (jusqu'au reboot) : `sudo mount -o remount,rw /boot` → ça ne fonctionne pas :/

Pour rétablir la partition en lecture, il faut 2 reboot : dans `raspi-config`, désactiver l'Overlay File System, puis reboot dans `raspi-config`, passer la partition en écriture (overlay file system), puis reboot

(Voir ici pour le détail de la procédure complète : <https://core-electronics.com.au/guides/read-only-raspberry-pi/>)

## Interface hardware

Premier schéma







Sur X11 et Wayland : <https://nullrequest.com/posts/whywaylandisthefuture>

On peut connaître le système d'affichage utilisé avec : `echo $XDG_SESSION_TYPE`

Essai avec le serveur de debugging : sans succès / TODO ajouter code autostart + script python (cf. rep /home/xor/.config/labwc)

Essai avec firefox en mode kiosque : sans succès :/ (à creuser... cf. <https://wiki.mozilla.org/Firefox/CommandLineOptions> )

Finalement comme solution temporaire, on ajoute un mini clavier USB dans la borne, sur lequel il faut appuyer sur F5 quand ça mouline sur un premier onglet blanc : ça fonctionne mais c'est moche

## mars 2026

Passage en mode écriture

On peut choisir entre X et Wayland (via raspi-config / advanced options / Wayland) : je choisis openbox + Xorg

Infos sur X.org : <https://doc.ubuntu-fr.org/xorg>

Ce qui rend possible l'utilisation de xdotoool, nouveau test réalisé après modification du fichier autostart

On peut recharger la page avec

```
xdotoool search --onlyvisible --class chromium windowactivate --sync %1 key F5 windowactivate $(xdotoool getactivewindow)
```

Ça fonctionne **si** la page a déjà été correctement chargée mais comme ce n'est pas le cas, ce n'est toujours pas la solution ...

utilisation de xdotoool : <https://unix.stackexchange.com/a/87839> Pour la suite

```
#!/bin/bash

# X plutôt que Wayland doit être activé (via raspi-config)
cd /home/xor/borne_biohacklab_ui_v5/
/usr/bin/python -m http.server &
unclutter -idle 1 & # ne fonctionne qu'avec X pas wayland
sleep 8
#chromium --start-fullscreen --start-maximized --noerrdialogs --disable-infobars --kiosk http://localhost:8000
chromium --kiosk --noerrdialogs --disable-session-crashed-bubble --disable-infobars http://localhost:8000/index.html &
#chromium-browser --kiosk "http://localhost:8000/index.html" --noerrdialogs --disable-session-crashed-bubble --disable-infobars --remote-
debugging-port=9222 --remote-allow-origins=http://localhost:9222 &
#chromium-browser --kiosk --noerrdialogs --disable-session-crashed-bubble --disable-infobars &
#sleep 15
#/home/xor/.config/labwc/chromium_reload_first_tab.py
sleep 15
xdotoool search --onlyvisible --class chromium key F5 # ne fonctionne pas avec wayland!
#wtype -M ctrl r -m ctrl
#firefox --kiosk "http://localhost:8000/index.html"

xdotoool search --onlyvisible --class chromium windowactivate --sync %1 key F5 windowactivate $(xdotoool getactivewindow)
```

Petite fonction pour vérifier que la page recharge bien

```
function changeBackground(color) {
    document.body.style.background = '#' + (Math.random()*0xFFFFFFFF<<0).toString(16);
}
window.addEventListener("load",function() { changeBackground('red') });
```

## mai 2026

Le Pi est en mode Xorg + openbox réglé via raspi-config

### test 1 : maj chromium

Mettre à jour chromium pour voir si la nouvelle version corrige le bug... chromium-common/oldstable  
1:148.0.7778.167-1~deb12u1+rpt1 arm64 [upgradable from: 1:130.0.6723.116-1~deb12u1+rpt1]

Rpi connecté au réseau wifi local

```
sudo nmap -sn 192.168.1.1/24|awk '/^Nmap/{ip=$NF}/D8:3A:DD/{print ip}'
```

Renvoie l'adresse IP : 192.168.1.30

```
ssh xor@192.168.1.30
sudo apt update
sudo apt list --upgradable|grep chrom
chromium-common/oldstable 1:148.0.7778.167-1~deb12u1+rpt1 arm64 [upgradable from: 1:130.0.6723.116-1~deb12u1+rpt1]
chromium-sandbox/oldstable 1:148.0.7778.167-1~deb12u1+rpt1 arm64 [upgradable from: 1:130.0.6723.116-1~deb12u1+rpt1]
chromium/oldstable 1:148.0.7778.167-1~deb12u1+rpt1 arm64 [upgradable from: 1:130.0.6723.116-1~deb12u1+rpt1]
rpi-chromium-mods/oldstable 20250506 arm64 [upgradable from: 20240904]
sudo apt install chromium
```

Au passage, j'installe micro pour avoir un éditeur de texte pratique en ssh

```
sudo apt install micro
```

Puis edit du script de démarrage

```
micro /home/xor/.config/labwc/autostart
```

Pour tester le script via SSH

```
kill python; kill chromium
export DISPLAY=:0
export XAUTHORITY=/home/pi/.Xauthority
bash -x /home/xor/.config/labwc/autostart
```

Mais ça ne marche toujours pas... et l'autostart non plus :/

## test 2 : bidouille

ÇA FONCTIONNE!

Éditer le fichier /etc/xdg/lxsession/LXDE-pi/autostart (c'est ce fichier qui s'active au démarrage de session)

Bien vérifier que WebGL est activé dans chrome chrome:gpu'' ===== Utile ===== ALT+F4 ferme chrome lorsqu'il est en mode kiosque

Article extrait de : <http://www.lesporteslogiques.net/wiki/> - **WIKI Les Portes Logiques**  
Adresse : [http://www.lesporteslogiques.net/wiki/projets/biohacklab/borne\\_arcade?rev=1780503489](http://www.lesporteslogiques.net/wiki/projets/biohacklab/borne_arcade?rev=1780503489)  
Article mis à jour: **2026/06/03 18:18**