

Hyperspin Partie 4

Hyperlaunch et les modules AHK

Extrait du tuto De « [Heavyarms](#) » administrateur sur HFSPLAY (lien original : [ICI](#))

Table des matières

I.	Qu'est-ce qu'Hyperlaunch ?	2
II.	Qu'est-ce qu'un fichier AHK et un module ?	4
III.	Installation et liaison d'HyperLaunch avec Hyperspin	5

4ème et monstrueuse partie de ce dossier qui va traiter du point le plus obscure d'Hyperspin, le détail qui génère le plus d'erreurs, je veux bien sur parler d'Hyperlaunch et de ses fichiers AHK que l'on connait sous le nom de modules.

Le tuto va s'orienter en plusieurs parties, à savoir la présentation d'Hyperlaunch, des fichiers AHK (modules), l'installation d'Hyperlaunch et la configuration de toutes ses options annexes. Je terminerai par quelques petits conseils.

Pour l'élaboration de ce tuto, je me suis appuyé sur mes connaissances et en grande partie sur le topic officiel de Bad Boy Bill (que je traduis en grande partie) que je remercie énormément pour son travail.

I. Qu'est-ce qu'Hyperlaunch ?



Hyperlaunch (nommé aussi HL) est un lanceur de script AutoHotKey (traduit AHK), ce qui signifie pour nous autres utilisateurs d'Hyperspin qu'il est un lanceur séparé d'émulateurs, de jeux ou de programmes. Hyperspin peut lancer des jeux par lui-même, mais le résultat est assez moche et instable alors que la mise en place d'Hyperlaunch en complément ouvre d'énormes possibilités grâce aux scripts (traduit modules) configurables.

Le lancement de programmes (émulateurs ou autres) par l'utilisation de scripts est bien meilleur car ces derniers contrôlent exactement ce qui se passe lors du lancement ou de l'extinction d'un émulateur ou d'un jeu. Ce point est particulièrement important pour une utilisation dans une borne d'arcade car les joueurs recherchent toujours un résultat le plus propre possible.

Vous n'êtes pas forcément obligé de savoir comment HL fonctionne pour l'utiliser, vous devez simplement l'activer dans HyperHQ, avoir le bon module et configurer 2 ou 3 petites choses pour que tout fonctionne correctement mais le but de ce tuto est d'éclaircir l'ensemble du bazar.

Pour résumé, Hyperlaunch à uniquement pour but, via Hyperspin, de lancer, de faire tourner les jeux et d'en sortir le plus proprement possible.

Il y a eu 2 versions d'HyperLaunch (petite description de Soft-bug du forum HyperFreeSpin) :

HyperLaunch 1.x :

Il y avait un seul fichier AHK qui était le lanceur de tous les émulateurs que vous utilisiez. Dans cet AHK, vous deviez mettre (il y avait des éléments par défaut pour les consoles comme la SNES ou la Megadrive ...) le lancement de l'émulateur en relation avec votre installation d'HS et de votre config.

Mais pour cela, vous deviez compiler le fichier Hyperlaunch.ahk (via le compilateur AutohotKey) pour chaque nouvel émulateur.

HyperSpin 2.x :

Actuellement dans une version transitoire 2.25c, il y a un fichier AHK par émulateur. Bref, le fichier de la v1.x peut maintenant être découpé en rondelles. Sauf que le nouveau set ne demande plus la compilation des modules, vous pouvez soit les trouver sur le site officiel (à cet adresse, il vous suffit juste de trouver votre système et de cliquer sur l'icône verte « **AHK** » dans la colonne de droite : <http://hyperlist.hyperspin-fe.com/>), soit les créer vous-même ou modifier ceux déjà existants.

L'on parle généralement de "module" quand on parle des éléments à mettre dans un AHK pour lancer un émulateur.

Le but de cette dernière version d'Hyperlaunch est :

1/ De rendre son utilisation plus simple pour les nouveaux utilisateurs. L'ancienne version nécessitait de savoir comment mettre à jour et de recompiler un fichier AHK pour le modifier. Cette dernière version élimine ce processus, vous n'avez plus à compiler HL ou ses modules.

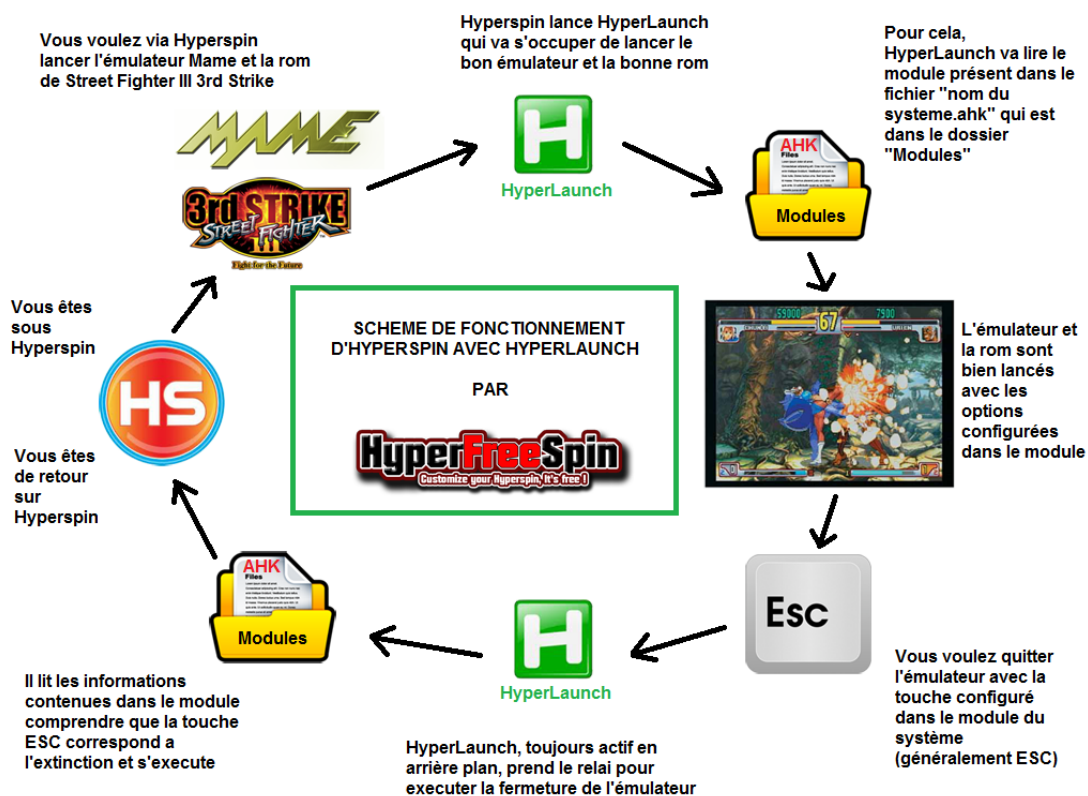
2/ Pour rendre HL modulaire. Cela signifie que chaque émulateur a un script différent que vous pouvez télécharger. Vous avez donc juste à trouver le module correspondant à la version de votre émulateur et basta. Cela enlève donc le fait de devoir ajouter les émulateurs dans HL via scripting.

3/ De configurer une touche dans le script pour quitter automatiquement l'émulateur (normalement sans modification du module par l'utilisateur, mais c'est loin d'être le cas), cela dans le but d'une installation propre.

4/ HL EST un programme automatique dont Hyperspin se sert pour lancer et quitter un émulateur proprement et dans cette version, il dispose d'une petite interface, si vous le lancer seul, qui vous permet de tester votre configuration module + émulateur sans devoir lancer Hyperspin, donc gain de temps.

Hyperlaunch est donc le programme qui permet de dire à Hyperspin de lancer tel ou tel émulateur avec tel rom, de faire tourner le jeu en premier plan et ensuite de quitter l'émulateur pour revenir sur le frontend directement. Pour schématiser, HyperSpin serait le chef d'orchestre et Hyperlaunch serait les musiciens. Et pour que les musiciens puissent jouer, ils leur faut bien sûr des instruments et dans ce cas, ces instruments seront les modules qui sont les données des fichiers AHK que vous mettrez dans le dossier « Hyperspin\Modules ».

Voici un petit schéma de conclusion pour vous expliquer le fonctionnement de l'ensemble :



II. Qu'est-ce qu'un fichier AHK et un module ?

Nous parlons souvent de fichiers AHK ou de modules. Ces termes désignent en fait une seule et même chose mais dans les faits, ces 2 choses sont un peu différentes.

Un fichier AHK est un script AutoHotKey qui contient des informations comme « démarrer cet émulateur avec ce jeu » ou encore « quitter l'émulateur et revenir sur Hyperspin par la touche ESC » par exemple, des données lisibles par HyperLaunch pour que l'émulateur fonctionne correctement avec Hyperspin. C'est l'ensemble de ces données qui constituent le module à proprement parlé.

Ces fichiers AHK seront à placer dans le dossier « Hyperspin\Modules » et devront être nommés de la même manière que le système avec lequel il est lié (par exemple, si votre système se nomme « MAME » dans Hyperspin, le AHK devra se nommer « MAME.ahk »).

Il faut savoir qu'un module n'est pas propre à un système (Megadrive par exemple) mais à un émulateur (Kega Fusion, Mess, Haze MD...), comprenez par là qu'en fonction de l'émulateur que vous choisirez, vous aurez besoin d'un module différent. L'explication vient du fait que tous les émulateurs ne sont pas fait de la même façon et que le fichier AHK ne contiendra pas le même module à donner à Hyperlaunch, logique.

Veillez aussi à vérifier la version de votre émulateur et la version pour lequel le module est écrit. Dans certains cas, cela entrainera des problèmes de compatibilité entre votre émulateur et Hyperlaunch, donc des erreurs avec Hyperspin par la suite.

Voici à quoi ressemble un module vu de l'intérieur (éditable avec le note pad) :

```
1 ;-----
2 ; Super Nintendo Entertainment System
3 ; zsnsw v1.51
4 ; by djvj
5 ; 1.1
6 ;
7 ; Notes:
8 ; Be sure to set the settings you want below
9 ;-----
10 Fullscreen = true
11 Stretch = false
12 resX = 1024
13 resY = 768
14 DisplayRomInfo = false ; Display rom info on load along bottom of screen
15 ;-----
16
17 SetKeyDelay, 50
18 zsnswFile := CheckFile(emuPath . "zsnsw.cfg")
19 FileRead, zsnswCfg, %zsnswFile%
20
21 GUI_ID := FadeInStart()
22
23 xLine := TF_Find(zsnswCfg,"","","CustomResX=") ; find location in zsnsw cfg where it stores its custom X res
24 yLine := TF_Find(zsnswCfg,"","","CustomResY=") ; find location in zsnsw cfg where it stores its custom Y res
25 modeLine := TF_Find(zsnswCfg,"","","cvidmode=") ; find location in zsnsw cfg where it stores its custom vid m
26 zsnswCfg := TF_ReplaceLine(zsnswCfg,xLine,xLine,"CustomResX=" . resX) ; update custom X res in zsnsw cfg file
27 zsnswCfg := TF_ReplaceLine(zsnswCfg,yLine,yLine,"CustomResY=" . resY) ; update custom Yres in zsnsw cfg file
28
29 If ( Fullscreen = "true" && Stretch = "true" ) ; sets fullscreen, stretch, and filter support
30     vidMode = 39
31 Else If ( Fullscreen = "true" && Stretch != "true" ) ; sets fullscreen, correct aspect ratio, and filter supp
```

Tout ce qui est placé entre ces 2 premières rangées de tirets sont des infos laissées par le créateur du module, A LIRE IMPERATIVEMENT !

Tout ce qui se trouve derrière un point virgule à la fin d'une valeur, c'est la description du réglage sur lequel influe cette valeur

MODULE
↓ ↓ ↓ ↓

III. Installation et liaison d'HyperLaunch avec Hyperspin

Avant d'attaquer à proprement parler l'installation d'HyperLaunch et la configuration des menus, attardons nous sur les fichiers de configuration .INI qui doivent se trouver dans le dossier « **Hyperspin\Settings** ». Vous devriez trouver ces 3 fichiers :

Settings.ini : Ce fichier contient les options spécifiques de vos systèmes. Il devra contenir les différentes clés, chemins d'accès et paramètres de chacun des systèmes.

Global Settings.ini : Ce fichier contient les mêmes clés que les fichiers « nom du système ».ini, mais celui-ci aura tous les paramètres par défaut qui affecteront l'ensemble des systèmes. Si vous voulez par exemple configurer toutes les wheels d'un seul coup sans avoir à modifier un par un tous les « nomdusysteme.ini », c'est ici que ça se passe.

« **Nom du système** ».ini : Ce fichier contient les mêmes clés que le fichier Global Settings.ini mais est spécifique pour le système concerné. Si vous changez une valeur dans ce fichier, cela n'affectera que ce système. Les valeurs par défaut pour toutes ces clés sont dans la partie « **use_global** ».

Quel que soit la clé ou la valeur que vous modifierez, veillez à être dans le bon fichier INI sinon rien ne sera reconnu par Hyperspin. Vous pouvez essayer de modifier ces valeurs pour trouver de nouveaux réglages pour résoudre vos problèmes.

Si je vous parle de ces fichiers INI, c'est qu'au premier lancement d'HyperLaunch, il copiera ses données à partir de vos fichiers « **Settings.ini** » et « **Global Settings.ini** » existants. Cela vaudra aussi pour la prochaine version 2.0 de Hyperspin.

Etape 1 :

Au premier lancement, HyperLaunch créera beaucoup de nouvelles entrées dans « Settings.ini » et « Global Settings.ini ». La première fois que vous lancerez un jeu, il créera aussi d'autres clés dans ces INI.

Donc quand vous créez un nouveau système dans Hyperspin, il est vivement conseillé de juste lancer un jeu, même si il ne fonctionne pas, simplement pour que ces clés soient générées pour vous dans les INI. Une fois créées, vous pourrez modifier ces valeurs si vous en avez besoin.

Exemple des clés générées dans Settings.ini :

[HyperLaunch]

```
Exit_Script_Key=~q ~s
Exit_Emulator_Key=~Esc
Toggle_Cursor_Key=~t
Path_To_Daemontools=
Path_To_CPWizard=
Last_System=MAME
Last_Rom=1942
```

Etape 2:

Créez un nouveau dossier appelé “Modules” dans votre Hyperspin (si vous avez suivi tous mes tutos et avez récupéré mon archive d’Hyperspin déjà mis à jour, le dossier est déjà créé). Je vous conseille d’ailleurs de mettre votre Hyperspin à la racine de votre disque dur (par exemple C:\Hyperspin) pour minimiser les erreurs de chemins d’accès car Hyperspin est sensible aux longs nommages (par ailleurs, Windows a une limite de 255 caractères pour les chemins d’accès).

Téléchargez votre module sur HyperList ou le cloud du forum. Chaque module devra être nommé de la même manière que le système et placé dans un sous dossier lui aussi du même nom que le système.

Exemple pour le module du système « Sega Megadrive » :

D:\Hyperspin\Modules\Sega Megadrive\Sega Megadrive.ahk

Etape 3 (facultative si vous avez récupéré mon archive d’HS déjà mis à jour, j’ai déjà tout fait) :

Téléchargez le nouveau HyperLaunch sur le topic officiel ([http://www.hyperspin-fe.com/forum/showt ...x-Released](http://www.hyperspin-fe.com/forum/showt...x-Released) toutes les mises à jour d’HL seront postées la bas) et extraire l’archive à la racine de votre Hyperspin.

L’archive contient :

HyperLaunch.exe (le lanceur compilé, le code source peut être trouvé sur HyperList)

autohotkey.dll (c’est grâce à lui que nous n’avons plus besoin de compiler les modules)

7z.dll (Requis pour les fonctions 7z)

7z.exe (Requis pour les fonctions 7z)

Etape 4 :

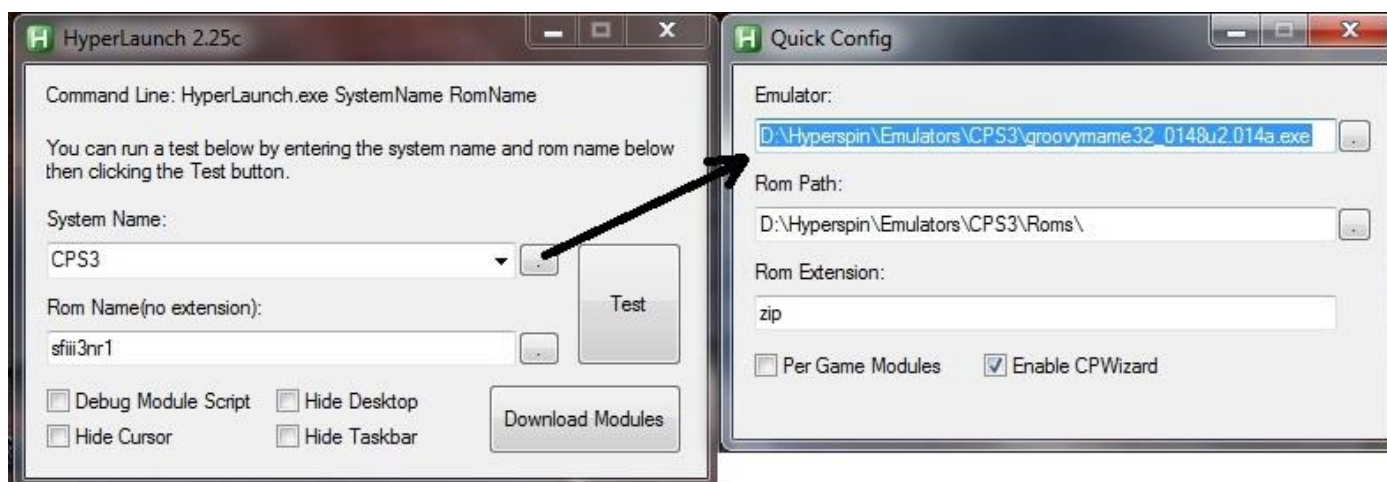
Comme montré précédemment, **chaque module contient des notes dans le haut de la page que vous devez impérativement lire et suivre pour éviter des erreurs**. De nombreux modules ont des valeurs que vous pourrez éditer beaucoup plus facilement en suivant ces informations.

La plupart des modules sont disponibles sur la page d’Hyperlist mais d’autres ne sont disponibles que sur le FTP payant du site officiel (payer 40€ pour avoir accès à 3 fichiers AHK que l’on peut arriver à créer... merci à nos amis américains pour leur esprit de partage !). N’hésitez pas à demander de l’aide dans la section appropriée du forum si l’un des AHK vous manque ou si vous voulez des conseils lors de l’édition de vos modules. Mais avant toute question, je vous encourage à chercher un minimum et à essayer différentes possibilités d’édition par vous-même.

Si vous avez respecté toutes ces étapes, HyperLaunch devrait être correctement lié à Hyperspin et minimiser le risque d’erreurs. Voyons maintenant comment utiliser HyperLaunch seul.

Utilisation d'HyperLaunch hors d'Hyperspin

Ouvrez HyperLaunch en double cliquant sur « hyperlaunch.exe », vous devriez voir simplement la fenêtre de gauche (celle de droite est accessible en cliquant à droite de la ligne System Name) :



Dans cet exemple, le système nommé « CPS3 » est sélectionné et un menu déroulant est présent pour choisir un autre système déjà installé dans Hyperspin. Si vous voulez utiliser un émulateur non installé dans HS **ET** que vous possédez le module correspondant, vous pouvez cliquer sur le point à droite qui vous affichera la fenêtre de droite qui vous demande de joindre votre émulateur, votre dossier contenant les roms et d'inscrire l'extension des roms. Les cases « Per Games Modules » et « Enable CPWizard » permettent pour la première, de lancer un module différent en fonction du jeu utilisé (pour utilisation avec des jeux PC par exemple, je détaille cette partie un peu plus bas) et pour la seconde, de lier votre système avec CPWizard si vous l'utilisez.

Cliquez ensuite sur la case à droite de la ligne « Rom Name » pour arriver directement sur le dossier contenant les roms du système (que vous aurez lié dans Hyperspin ou dans HL) et choisir une des roms pour la tester (dans cet exemple Street Fighter III 3rd Strike No CD JAP Rev.A).

Il ne vous reste plus qu'à appuyer sur le bouton « **Test** » pour vérifier si le lancement, le fonctionnement et l'extinction de l'émulateur s'effectuent correctement, ce qui signifiera que votre système est bien lié à Hyperspin et que le module est bon.

Utilisation de Daemon Tools Lite avec HyperLaunch

Daemon Tools Lite est requis pour l'utilisation de nombreux émulateurs utilisant des jeux au format CD. Il a l'avantage de pouvoir monter et démonter proprement les CD/DVD et surtout de prendre en compte les pistes audio originales des jeux.

Quelques émulateurs sont programmés avec la faculté de monter un Iso, mais cela n'est pas forcément un avantage dans une utilisation avec Hyperspin car souvent, ces émulateurs perdent des options de configurations quand l'iso est monté en interne, ce qui n'arrive pas avec l'utilisation de Daemon Tools.

Etape 1

Téléchargez Daemon Tools Lite sur le site officiel : <http://www.daemon-tools.cc/eng/products/dtLite>

Installez le où vous voulez, idéalement à l'emplacement par défaut ou sur le même disque dur qu'Hyperspin, mais ça n'a pas vraiment d'importance. Lors de l'installation, veillez à décocher l'installation des programmes tiers qui ne serviront à rien (barre d'outils, barre widget, etc). Vous pouvez aussi choisir de ne pas installer SPDT, nécessitant un redémarrage du PC pour son installation), qui procure quelques avantages d'émulation et de lecture CD mais que nous n'avons pas vraiment besoin dans notre cas.

Etape 2 :

Configurez maintenant le chemin d'accès de Daemon Tools dans votre fichier « Hyperspin\Settings\Settings.ini ». Ouvrez le avec le note pad et dans la section [HyperLaunch], éditez votre valeur. Par exemple pour une installation classique, voilà ce que ça donne :

Path_To_Daemontools=C:\Program Files (x86)\DAEMON Tools Lite\DTLite.exeAssurez-vous que les 2 chemins correspondent. Si la section HyperLaunch n'existe pas dans votre « Settings.ini », lancer HyperLaunch une fois pour qu'il les ajoute automatiquement (voir installation d'HyperLaunch plus haut).

Etape 3 :

Si Daemon Tools vous demande de redémarrez, faites le directement.

Vous devriez normalement avoir une petite icône DT dans votre barre de tâches actives (si elle n'y est pas, lancez DT, allez dans le menu de configuration, onglet général et cochez les cases « Utilisez l'Agent DT » et « Auto Démarrage ») et un nouveau lecteur virtuel installé et assigné à une lettre que Daemon Tools a installé pour vous. Si cela n'est pas le cas ou si vous voulez plusieurs lecteurs, faites un clic droit sur l'icône DT et sélectionnez « Périphériques Virtuels ◇ Ajouter un lecteur virtuel SCSI. Il en créera un nouveau avec la prochaine lettre disponible.

La configuration principale de DT est maintenant terminée. Vous devrez trouver des instructions additionnels dans chaque module qui utiliseront Daemon Tools pour vous permettre de le configurer avec l'ensemble module/émulateur.

Utilisation de la fonction « Use Per Game Module »

Cette fonction vous permet de créer un module par jeu, requis par exemple pour certains jeux PC si vous avez besoin de plus de fonctions que PCLauncher peut vous proposer.

HyperLaunch ajoutera par défaut cette valeur dans le fichier « nomdusysteme.ini » en le lançant (à changer par true si vous voulez l'utiliser):

[exe info]

Per_Game_Modules=false

Utilisation du support de CPWizard

CPWizard peut être utilisé en liaison avec Hyperspin via HyperLaunch. Pour ceux qui ne savent pas ce qu'est, je les invite à se rendre à cette adresse pour découvrir de formidable software, l'essayer c'est l'adopter : <http://headsoft.com.au/index.php?catego ... page=about>

HyperLaunch ajoutera ces valeurs par défaut automatiquement :

Dans "Settings.ini" :

[HyperLaunch]

Path_To_CPWizard=false Dans "Global Settings.ini" :

[HyperLaunch]

CPWizard_Enabled=false

CPWizard_Delay=8000

CPWizard_Params=-minimized -timeout 9000 Dans "nomdusysteme.ini" :

[exe info]

CPWizard_Enabled=use_global

CPWizard_Delay=use_global

CPWizard_Params=use_global

Path_To_CPWizard – Remplacez "false" par l'emplacement de l'exé de CPWizard

CPWizard_Enabled – Active ou désactive le support de CPWizard pour tous les systèmes ou un seul selon dans quel INI où vous l'éditez.

CPWizard_Delay – Le temps que prendra HyperLaunch, après que CPWizard soit lancé, avant de continuer à lancer le module, en millisecondes.

CPWizard_Params – Aucune info là dessus, même sur le site officiel.

Utilisation du support de 7-Zip

Si vous avez compressé toutes vos roms dans une seule et même archive 7z, 7-Zip est nécessaire pour pouvoir lancer les roms à partir de cette grosse archive.

Avec lui, vous pouvez maintenant utiliser n'importe quelle rom compressé avec son émulateur. Il requiert juste 2 lignes de commande dans chaque module (la plupart des modules ont été mis à jour pour le supporter). HyperLaunch va d'abord contrôler si la rom existe dans votre grosse archive avant de tenter l'extraire pour y jouer. Ensuite il l'extraira dans le dossier de la clé « 7z_extract_dir » et enverra le bon emplacement et la bonne extension de l'archive à l'émulateur qui le lancera alors.

Seule modification à faire, vous devrez rajouter l'extension 7z dans le fichier « nomdusysteme.ini ». Par exemple, si vous avez compressé toutes vos roms Super Nintendo dans une seule archive 7z, les extensions de rom devraient être smc,7z.

Par contre toutes vos roms présentes dans la grosse archive devront correspondre au nom de l'archive de la rom, qui elle-même devra correspondre au nom du jeu dans votre XML. Par exemple, si vous avez une rom nommée « Mario (USA).rom », son archive devra se nommer « Mario (USA).7z » dans la grosse archive.

La rom que vous voulez lancer (la seule envoyée à l'émulateur), ne doit pas être dans un dossier dans cette archive. Le système de dossier marche, comme par exemple avec les jeux POP Cap, mais pour les simples roms ou isos, n'utilisez pas de dossiers.

Seule la rom qu'HyperLaunch cherche devra porter le même nom. C'est par exemple le cas avec les CD/DVD avec plusieurs fichiers de pistes sonores. Vous devez, par exemple avoir une archive nommé "Wipeout (USA).7z" et dedans avoir "Wipeout (USA).cue, Track1.bin, Track2.bin, Track3.bin, etc". Le tout marchera à partir du moment où le fichier cue contiendra les noms correct des fichiers de pistes sonores et que vous avez spécifié « cue » dans vos extensions de roms.

Vous avez la possibilité, pour chaque système, de supprimer ou de garder les roms décompressées temporairement pour y jouer. Le fait de les garder garantit un lancement plus rapide et la tenue d'un historique de lancement, ce qui peut être très appréciable si vous avez stocké vos roms sur un serveur ou un NAS.

« Global Settings.ini » devra contenir ces valeurs par défaut :

[HyperLaunch]

7z_Enabled=false

7z_Extract_Dir=%A_Temp%\

7z_Delete_Temp=true Pour chaque « nomduystème.ini », vous pouvez définir un dossier d'extraction de roms séparés des autres, et garder les roms temporaires ou non par rapport à ce que vous avez rentré dans « Global Settings.ini » :

[exe info]

7z_Enabled=use_global

7z_Extract_Dir=%use_global

7z_Delete_Temp=true **7z_Enable** – Voulez-vous laisser HyperLaunch décompresser les roms pour vous ? Mettez « true » comme valeurs si vos émulateurs ne supportent pas le format de compression 7z.

7z_Extract_Dir – Rentrer le chemin d'accès de votre dossier qui contiendra vos roms une fois décompressées.

7z_Delete_Temp – Si édité sur « true », cela supprimera le dossier des roms temporaires une fois que quitterez l'émulateur. Mettre « false » pour les conserver.

Utilisation de bezel ou d'images lors des chargements

Créez un dossier nommé « GameStart » dans « Hyperspin\Media\NomDuSysteme\Images ».

HyperLaunch cherchera et chargera ces images si il trouve : top.png (bannière le long du haut de l'écran lors du chargement), middle.png (bannière en milieu d'écran), bottom.png (bannière le long du bas de l'écran).

Vous pouvez aussi placer une image pour chaque jeu (nommé de la même façon que la rom) dans ce dossier et HyperLaunch remplacera la bannière middle.png avec celle-ci, ce qui permet une personnalisation accru de son interface pendant le chargement des jeux.

HL va générer automatiquement ces valeurs par défaut :

Dans “Settings.ini” :

```
[HyperLaunch]
Fade_In_Interrupt_Keys=anykey
Dans « Global Settings.ini » :
[HyperLaunch]
Fade_In=false
Fade_Out=false
Fade_Color=0x000000
Fade_In_Duration=500
Fade_Out_Duration=500
Fade_In_Delay=0
Dans chaque « nomdusysteme.ini »
```

[exe info]

Fade_In=use_global

Fade_Out=use_global

Fade_Color=use_global

Fade_In_Duration=use_global

Fade_Out_Duration=use_global

Fade_In_Delay=use_global

Quoi qu’il en soit, aussi longtemps que vous garderez la valeur “use_global” dans votre “NomDuSystème.ini”, il utilisera les valeurs présentes dans « Settings.ini ». Si vous voulez que votre système ait ses propres réglages, changez juste « use_global » par le même type de valeur que celle présente dans « Settings.ini ».

Fade_In – Si “true”, Hyperspin fera un effet fondu vers la couleur mise en “Fade_Color”. Réglez sur « false » pour résoudre des problèmes de lancement de jeux.

Fade_Out – Si « true », l’émulateur fera un effet fondu pour sortie de la couleur mise en « Fade Color ». Réglez sur « False » si vous lancer les jeux en plein écran, si vous voyez des flashes ou des artefacts apparaître sur votre écran ou pour résoudre des problèmes de lancement de jeux. **Ces options marchent uniquement avec des jeux ayant des marges en plein écran, donc incompatible avec la majorité des jeux « réellement plein écran »**

Fade_Color – Rentrez la couleur que vous souhaitez avoir en fondu lors des chargements de lancement ou d’extinction (composés de 6 chiffres correspondant a la valeur d’une couleur RGB)

Fade_In_Duration – Réglez ici la vitesse d’exécution du fondu Fade_In, en millisecondes.

Fade_Out_Duration – Réglez ici la vitesse d’exécution du fondu Fade_Out, en millisecondes.

Fade_In_Delay – Valeur réglant la durée d’activation de votre image à l’écran avant le chargement réel de l’émulateur. Elle pourra être interrompue en pressant le bouton réglé dans Fade_In_Interrupt_Keys.

Fade_In_Interrupt_Keys – Si vous rentrez la valeur « anykey » (par défaut), toutes les touches du clavier interrompent l’écran de fondu de démarrage. Si vous voulez paramétrer quelle touche le fera, changez celle valeur par la liste des touches que vous désirez. Par exemple, si vous voulez utilisez F1, F5, MAJ Gauche et CTRL Droit, ces variables devront être inscrites comme suit : {F1}{F5}{LShift}{RCtrl}. Les joysticks ne

sont pas supportés, sauf si vous les remappez en touches de clavier via un logiciel de KyRemapping. La liste de toutes les touches définissables se trouvent ici : <http://www.autohotkey.com/docs/KeyList.htm>

Note : Les lettres et numéros n'ont pas besoin d'être définis, ils sont automatiquement activés.

Utilisation des logiciels de KeyMapping avec HyperLaunch

Seuls Xpadder (versions 2012), JoyToKey (v5+) et des Scripts AutoHotKey sont supportés par la fonction de Key Mapping d'HyperLaunch. Ne fonctionne pas avec JoyToKey v3 ou inférieur.

Vous pouvez utiliser celui que vous souhaitez (je détaille les 3) mais sachez qu'Xpadder est le meilleur au niveau du fonctionnement avec HS et HL.

Créez un dossier dans Hyperspin nommé « Keymapper Profiles » (il peut être modifié par la suite dans Settings.ini) dans lequel vous placerez vos profils de Keymap. Ce dossier sera structuré de la manière suivante (je vais prendre MAME comme système d'exemple) :

Hyperspin\Keymapper Profiles\ - Placez ici votre profil pour Hyperspin (ex : Hyperspin.xpadderprofile) – Ce profil est seulement requis si vous avez modifié vos contrôles pour la navigation dans Hyperspin.

Hyperspin\Keymapper Profiles\ - Placez ici votre profil par défaut (ex : default.xpadderprofile) - Il sera votre profil principal et utilisé pour tous vos systèmes n'ayant pas de profil spécifique.

Hyperspin\Keymapper Profiles\MAME – Placez ici votre profil pour le système MAME (ex : MAME.xpadderprofile) et chaque profil spécifique à une rom MAME en particulier qui doit être nommé avec le nom de la rom (ex : altbeast.xpadderprofile pour Altered Beast).

Quand vous lancerez un jeu, HyperLaunch regardera s'il existe un profil pour la rom. Dans le cas contraire, il cherchera un thème spécifique au système, puis un profil par défaut. Enfin, si il n'y a aucun profil par défaut d'existant, il lancera le profil d'Hyperspin. Et s'il n'existe aucun profil, il y aura une erreur et le jeu ne se lancera pas.

Pour récapituler, l'ordre des profils chargés sont :

1. Profil spécifique à une rom
2. Profil spécifique à un système
3. Profil par défaut
4. Profil Hyperspin

Lorsque vous quitterez l'émulateur, HyperLaunch fermera et déchargera le logiciel de keymapping. Il remettra ensuite en place le profil Hyperspin si vous en avez un.

Configuration d'Xpadder avec HyperLaunch

Il vous suffit tout simplement de le lancer et de suivre l'assistant de configuration initial.

Support du Multi-Joueur Xpadder : Si vous souhaitez lancer plusieurs profils, un par joueur par exemple (jusqu'à 4), vous devez simplement ajouter un nouveau profil dans le même dossier que la rom, avec le même nom et d'ajouter [p#] à la fin du nom. Par exemple si vous lancer Mame.xpadderprofile pour le joueur 1, il vous faudra Mame [p2].xpadderprofile pour le joueur 2. **N'oubliez pas l'espace !** Pour 4 joueurs, cela nous donnera donc "Mame" "Mame [p2]" "Mame [p3]" "Mame [p4]".

Configuration de JoyToKey avec HyperLaunch

Je vous invite à lire ce tuto du forum fait par Irving22Pac qui détaille très bien la configuration de Joy2Key : [viewtopic.php?f=28&t=402](http://www.hyperlaunch.com/viewtopic.php?f=28&t=402)

Il suffit ensuite de le lancer en mode minimisé et de cacher l'icône de la barre de tâche quand il est minimisé.

Configuration d'un script AutoHotKey de remapping

1. Créez un nouveau fichier AHK et placez les données de remap que vous voulez utiliser dans le code AHK. Vous pouvez trouver beaucoup d'informations sur comment remap via un fichier AHK ici (une liste de clés AHK valables peut y être trouvée) : <http://www.autohotkey.com/docs/misc/Remap.htm>
2. Enregistrez le fichier comme profil de jeu, de système, d'Hyperspin ou par défaut. HyperLaunch le chargera comme tel.
3. Pour exemple, vous voulez remap le A en B et le C en D en utilisant un AHK pour la rom « Mario ». Vous devrez donc créer «Mario.ahk » dans « Keymapper Profiles\Mame\ » et dans cet ahk, vous devrez trouver ce genre de valeurs de remap :
a::b
c::d
Vous devrez aussi avoir modifié la valeur « keymapper=ahk » dans vos fichiers « Settings.ini », « Global Settings.ini » ou dans le « MAME.ini » correspondant au système MAME. Nous voyons ça de suite.

Configuration des INI

Pour que le support du KeyRemapping soit activé, vous devrez modifier quelques valeurs dans différents fichiers ini comme précédemment.

HyperLaunch aura généré ces paramètres par défaut :

Dans « Settings.ini » :

```
[HyperLaunch]
Keymapper_Path=
```

Keymapper_Profiles_Path=%A_ScriptDir%\Keymapper Profiles\Dans « Global Settings.ini » :

[HyperLaunch]

Keymapper_Enabled=false

Keymapper=

Dans « NomDuSysteme.ini » :

[exe info]

Keymapper_Enabled=use_global

Keymapper=use_globalQuoi qu'il en soit, aussi longtemps que vous garderez la valeur "use_global" dans votre "NomDuSystème.ini", il utilisera les valeurs présentes dans « Settings.ini ». Si vous voulez que votre système ait ses propres réglages, changez juste « use_global » par le même type de valeur que celle présente dans « Settings.ini ».

Keymapper_Enabled – Mettez cette valeur sur « true » pour activer le support du KeyMapping pour tous les systèmes qui utiliseront la valeur « use_global » dans leur ini respectif.

Keymapper – Par défaut vierge, cette valeur est en fait le chemin vers votre logiciel de KeyMapping, par exemple C:\Hyperspin\Utilities\xpadder\xpadder.exe

Keymapper_Profiles_Path – Rentrez ici le chemin du dossier qui contiendra tous vos profils de keymapper. Par défaut, « Hyperspin\Keymapper Profiles\ » mais peut être modifié avec le nom de dossier que vous souhaitez

Note : Si vous avez le Key Mapping activé sur un système, vous devez vous assurer d'avoir au moins un exemplaire des 4 profils nécessaires créés dans le dossier devant les accueillir. Par exemple, si vous avez XPadder d'activé pour Mame et n'avez édité qu'un seul profil juste pour Mario donc pas de profil par défaut, pour Hyperspin ou pour Mame et que vous décidez de jouer à Tekken 3, vous aurez une erreur puisque vous n'avez pas de profil trouvable par HyperLaunch lors de sa recherche (voir l'ordre de recherche d'Hyperlaunch). Donc si vous ne voulez pas remapper à chaque fois pour tous les jeux Mame, vous avez juste à créer un profil appelé Mame.xpadderprofile pour résoudre votre problème.

Quelques conseils pour minimiser les erreurs avec le support du Keymapping :

- Supprimez tous les profils que vous auriez pu sauver dans le dossier de base d'XPadder (celui ou xpadder.exe va chercher normalement).
- Assurez-vous qu'XPadder est bien configuré pour lancer le dernier profil utilisé (qui sera forcément celui d'Hyperspin vu que vous quittez avec) au lancement de Windows (indispensable dans une utilisation à l'intérieur d'une borne).
- Préparez à l'avance 2 profils (un pour HS et un par défaut) pour ne pas avoir d'erreur de lancement.

Configuration du support des jeux Multi-Disques

Cette option assez attendu, et que BadBoyBill a mis pas mal de temps à mettre au point, va vous permettre de changer de disque/disquette/cassette de jeu en cours de jeu, en utilisant une belle interface entièrement paramétrable.

Si vous avez déjà joué à un jeu vous demandant d'insérer le Disque 2 en pleine partie sur un émulateur, vous vous êtes sûrement dis que c'était vraiment chiant de démonter son disque du lecteur virtuel, de mettre

le nouveau et de relancer l'émulateur pour continuer la partie. Sans compter sur certains systèmes utilisant d'anciennes cartouches ou cassettes qui ne permettent pas de reprendre au même point du jeu et nécessitant donc des disques ou scripts adaptés.

Cette nouvelle fonction dans Hyperlaunch 2.25c va tout faire pour vous en lançant un petit script qui va automatiquement trouver le bon cd et le lancer.

Fonctions :

- Customizable Image Artwork (Artwork personnalisable) – Vous pouvez utiliser l'Artwork par défaut ou le remplacer par celui de votre création, il s'affichera lors du chargement du nouveau disque.
- Vous pouvez utiliser vos Artworks d'Hyperspin (des boîtes de jeu, cartouches, etc). Comme l'option précédente, vous avez juste à changer le chemin du dossier de base par celui contenant vos Artworks.

Utilisation du SkipChecks

Cette option, prévu pour Hyperspin 2.0, permettra de sauter la plupart des checks de démarrage de certains émulateurs comme MUGEN, OpenBor ou PopCap. Hyperspin assumera le fait que vous avez tout configuré correctement et lancera juste ce que vous lui direz. Utilisant MUGEN pour l'exemple, HS lancera juste ce que MUGEN requiert pour démarrer, sans avoir besoin de créer un fichier txt vide ou batch/lnk pour passer le check de rom initial.

HyperLaunch générera automatiquement ces valeurs dans « NomDuSysteme.ini » :

[exe info]

Skipchecks=false**Skipchecks** – Si placé sur « true », HL, combiné à Hyperspin 2 seulement, sautera tous les checks roms/cchemins d'accès. Avec HS 1.3, il n'en saute que quelques uns.

Comment cacher son curseur

Cette option vous permet de cacher le curseur de souris, soit pour tous les systèmes, soit juste pour l'un d'entre eux. Rentrez « true » pour l'activer.

HyperLaunch générera automatiquement ces valeurs dans « Global Settings.ini » :

[HyperLaunch]

Hide_Cursor=falseHyperLaunch génère automatiquement ces valeurs dans « NomDuSysteme.ini » :

[exe info]

Hide_Cursor=use_globalQuoi qu'il en soit, aussi longtemps que vous garderez la valeur "use_global" dans votre "NomDuSystème.ini", il utilisera les valeurs présentes dans « Settings.ini ». Si vous voulez que votre système ait ses propres réglages, changez juste « use_global » par le même type de valeur que celle présente dans « Settings.ini ».

Comment cacher sa barre de taches

Cette option vous permet de cacher votre barre de taches Windows, soit pour tous les systèmes, soit juste pour l'un d'entre eux. Rentrez « true » pour l'activer.

HyperLaunch générera automatiquement ces valeurs dans « Global Settings.ini » :

[HyperLaunch]

Hide_Taskbar=falseHyperLaunch génère automatiquement ces valeurs dans « NomDuSysteme.ini » :

[exe info]

Hide_Taskbar=use_globalIl existe une alternative pour cacher cette barre, assurez vous juste de désactiver cette option dans les fichiers ini avant d'utiliser cette méthode (précisé dans l'étape 4).

1. Téléchargez l'archive HideTaskbar.zip à cet adresse :
https://mega.co.nz/#!1MYAQJbR!CjyZWipn3...ltb7i_kAlw
2. L'extraire où vous voulez et ouvrez HyperHQ.
3. Allez dans Main Settings-Startup/Exit et liez HideTaskbar.exe dans Startup Program.
4. Assurez-vous que la valeur Hide_Taskbar est réglé sur « false » dans votre fichier « Setting.ini » pour éviter tout conflit entre les 2 applications.

Comment cacher son bureau

Cette option vous permet de cacher votre bureau, soit pour tous les systèmes, soit juste pour l'un d'entre eux. Elle rendra le background noir, mettez « true » comme valeur. Cette option ne cache pas votre barre de taches, appliquez la partie juste au dessus pour cela. Cette option peut rentrer en conflit avec l'utilisation de fondu, donc elle est faite pour être utilisé dans une configuration sans fondu.

HyperLaunch générera automatiquement ces valeurs dans « Global Settings.ini » :

[HyperLaunch]

Hide_Desktop=falseHyperLaunch génère automatiquement ces valeurs dans « NomDuSysteme.ini » :

[exe info]

Hide_Desktop=use_globalQuoi qu'il en soit, aussi longtemps que vous garderez la valeur "use_global" dans votre "NomDuSystème.ini", il utilisera les valeurs présentes dans « Settings.ini ». Si vous voulez que votre système ait ses propres réglages, changez juste « use_global » par le même type de valeur que celle présente dans « Settings.ini ».