Guide de Programmation de la ArcadePiHat pour Raspberry Pi

# Introduction

Ce guide explique comment configurer et programmer une carte ArcadePiHat pour connecter des joysticks et boutons rétro à un Raspberry Pi. Le processus inclut la préparation du système, la programmation de l'EEPROM, et la vérification du bon fonctionnement.

# 1. Préparation pour la programmation de l'EEPROM

1.1 Activer le bus I2C :  
  
Assurez-vous que le jumper WP (Write Protect) sur la carte ArcadePiHat n'est pas ponté. Ensuite, exécutez les commandes suivantes pour activer le bus I2C pour la programmation de l'EEPROM :

sudo bash -c 'echo "dtparam=i2c\_vc=on" >> /boot/config.txt'  
sudo reboot  
  
La première commande ajoute le paramètre `dtparam=i2c\_vc=on` au fichier de configuration `/boot/config.txt`, permettant ainsi l'accès I2C.  
La seconde commande redémarre le Raspberry Pi pour appliquer les modifications.

# 2. Installation et compilation des outils

2.1 Cloner le dépôt GitHub et compiler les outils nécessaires :  
  
**git clone** [**https://github.com/Modding-Gaming/Raspberrypi/hats.git**](https://github.com/Modding-Gaming/Raspberrypi/hats.git)

**cd hats/eepromutils/  
make clean ; make**  
  
- `**git clone**` télécharge le dépôt contenant les outils nécessaires pour la gestion de l'EEPROM.  
- `**cd hats/eepromutils/**` change le répertoire de travail pour celui des utilitaires d'EEPROM.  
- `**make clean ; make**` nettoie les fichiers de compilation précédents (le cas échéant) et compile les outils.

# 3. Effacement de l'EEPROM

3.1 Nettoyer l'EEPROM :  
  
**dd if=/dev/zero ibs=1k count=4 of=blank.eep  
sudo ./eepflash.sh -w -f=blank.eep -t=24c32**  
  
- `**dd if=/dev/zero ibs=1k count=4 of=blank.eep**` crée un fichier vierge de 4 Ko rempli de zéros.  
- `**sudo ./eepflash.sh -w -f=blank.eep -t=24c32**` écrit ce fichier vierge dans l'EEPROM, effectuant ainsi un effacement.

# 4. Création du fichier EEPROM

4.1 Créer le fichier DTBO (Device Tree Blob) :  
  
**dtc -@ -b 0 -I dts -O dtb -o retrohat.dtbo retrohat.dts**  
  
- `**dtc**` compile le fichier Device Tree Source (DTS) `retrohat.dts` en un fichier Device Tree Blob (DTBO) `retrohat.dtbo`.

4.2 Créer le fichier EEPROM :  
  
**./eepmake retrohat.txt retrohat.eep retrohat.dtbo**  
- `**eepmake**` utilise le fichier de configuration `retrohat.txt` et le fichier DTBO `retrohat.dtbo` pour créer le fichier EEPROM `retrohat.eep`.

# 5. Programmation de l'EEPROM

5.1 Écrire le fichier EEPROM sur l'EEPROM :  
  
**sudo ./eepflash.sh -w -f=retrohat.eep -t=24c32**  
**sudo reboot**  
  
- `**sudo ./eepflash.sh -w -f=retrohat.eep -t=24c32**` écrit le contenu du fichier `retrohat.eep` dans l'EEPROM.  
- `**sudo reboot**` redémarre le Raspberry Pi pour appliquer les modifications.

# 6. Désactiver l'I2C et protéger l'EEPROM

6.1 Désactiver le bus I2C et activer la protection en écriture :  
  
**sudo bash -c 'sed -i "/^dtparam=i2c\_vc=on$/d" /boot/config.txt'  
sudo reboot**  
  
- `**sudo bash -c 'sed -i "/^dtparam=i2c\_vc=on$/d" /boot/config.txt**'` retire la ligne `**dtparam=i2c\_vc=on**` du fichier de configuration, désactivant ainsi l'accès I2C.  
- `**sudo reboot**` redémarre le Raspberry Pi pour appliquer les modifications.

# 7. Vérification du bon fonctionnement

7.1 Vérifier que les broches GPIO sont configurées comme prévu :  
  
**sudo raspi-gpio get  
sudo cat /sys/kernel/debug/gpio**  
  
- `**sudo raspi-gpio get**` affiche l'état actuel des broches GPIO du Raspberry Pi.  
- `**sudo cat /sys/kernel/debug/gpio**` fournit des informations de débogage sur les broches GPIO.

7.2 Utiliser `evtest` pour vérifier le fonctionnement des touches :

evtest  
  
- `**evtest**` est un utilitaire qui permet de tester et de vérifier les événements des dispositifs d'entrée comme les joysticks et les boutons.