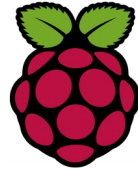


## Raspberry Pi Adafruit 2.2" PiTFT HAT - 320x240

### Display



#### 1 Présentation

La carte PiTFT est un écran couleur 320\*240px qui peut être connecté directement sur le raspberry. Cette carte utilise le bus SPI broches (SCK, MOSI, MISO, CE0, CE1) ainsi que le GPIO # 25. Tous les autres broches du GPIO sont inutilisés. Il y a 4 boutons poussoirs connecter au GPIO, que vous pouvez utiliser pour créer une interface utilisateur de base. Par exemple, vous pouvez en utiliser un comme bouton on / off.

#### 2 Installation

Le plus simple est de télécharger l'image de raspbian modifiée sur le site d'Adafruit et de la copier sur une carte SD.

#### 3 Modification de l'affichage console

La taille des caractères sur l'écran PiTFT est vraiment petite, il est nécessaire de reconfigurer l'affichage.

Pour changer la font exécuter

```
sudo dpkg-reconfigure console-setup
```

puis choisir :

- UTF-8
- Choix automatique du jeu de caractères adapté
- Terminus
- 10x20 (uniquement tampon vidéo (« framebuffer »))

#### 4 Afficher une image avec fbi

```
pi@raspberrypi ~ $ sudo fbi -T 2 -d /dev/fb1 -noverbose -a sarthe_running.jpg
```

fbi - framebuffer imageviewer est un visualiseur d'images en console (nécessite le frame buffer). Le **framebuffer** est une couche d'abstraction graphique indépendante du matériel pour afficher des éléments graphiques dans une console ici nous utilisons **fb1** pour l'écran PiTFT.

Options :

```
-T start on virtual console <arg>
-d device indique le framebuffer à utiliser
-noverbose disable la ligne de status en bas de
l'écran.
-a -autozoom automatically pick useful zoom factor
```

#### 5 Afficher une image avec python & pygame

```
#!/usr/bin/python
import os
import pygame
from pygame.locals import *

os.environ["SDL_FBDEV"] = "/dev/fb1"
CIEL = 0, 200, 255
NOIR = 0, 0, 0
pygame.init()
fenetre = pygame.display.set_mode((320, 240))
fond = pygame.image.load("background.jpg")
fenetre.blit(fond, (0, 0))
pygame.display.flip()

continuer = 1 #Boucle infinie
while continuer:
    continuer
```