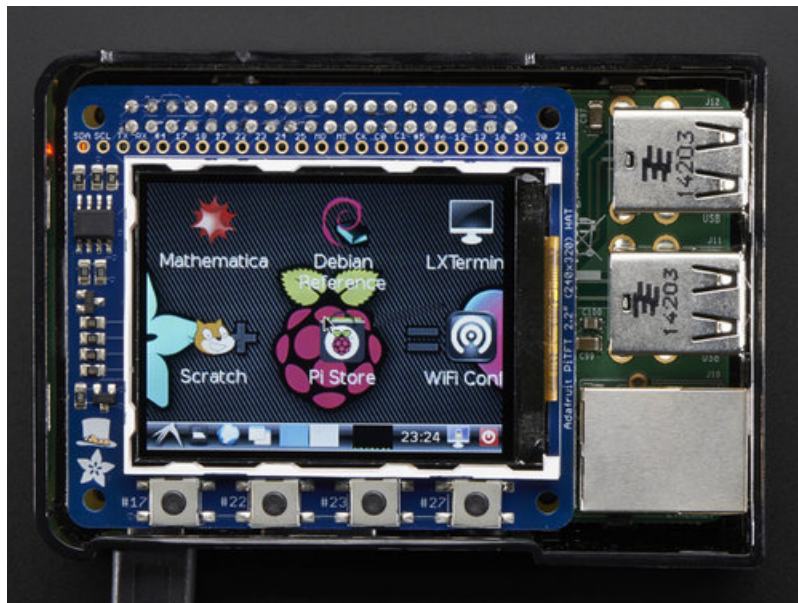


## Adafruit 2.2" PiTFT HAT - 320x240 Display

Created by lady\_ada



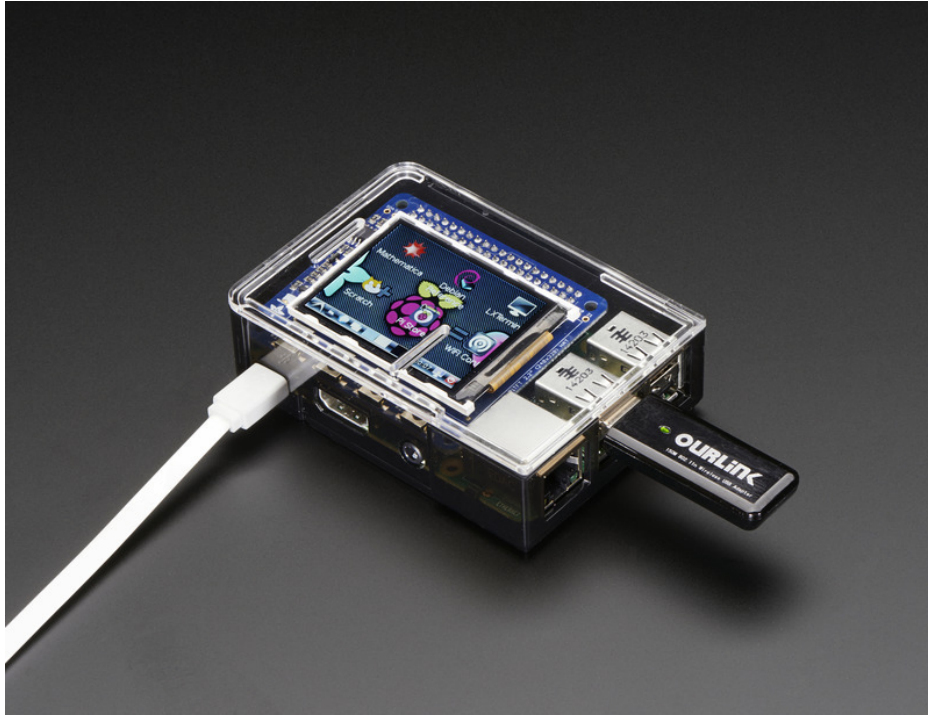
Last updated on 2019-06-14 08:25:18 PM UTC

## Overview



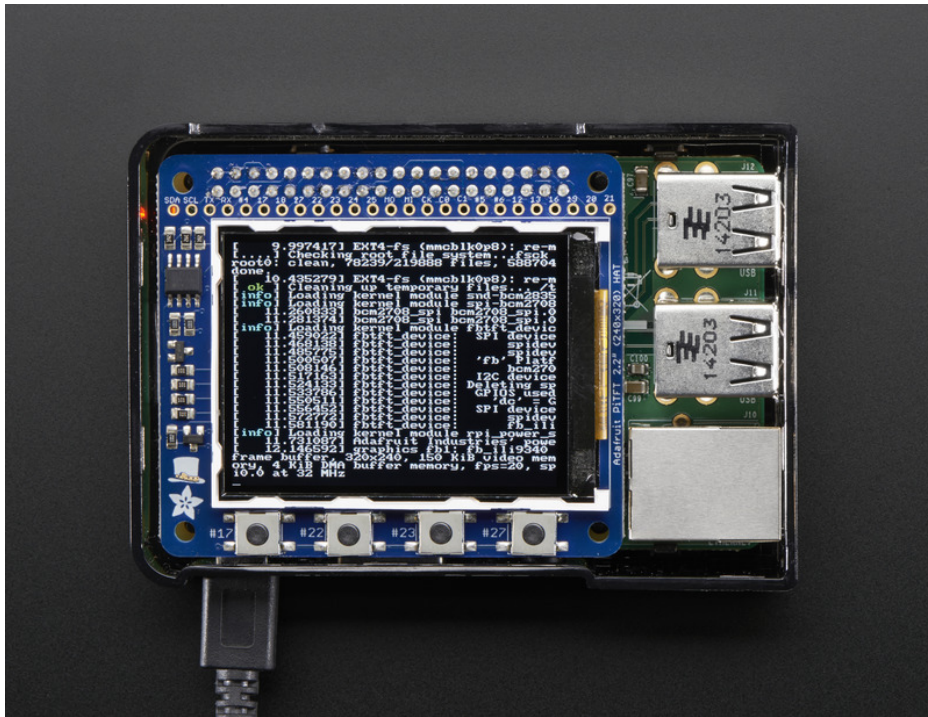
The cute PiTFT got even more adorable with this little primary display for Raspberry Pi in HAT form! It features a 2.2" display with 320x240 16-bit color pixels. The HAT uses the high speed SPI interface on the Pi and can use the mini display as a console, X window port, displaying images or video etc. Best of all it plugs right in on top of your Model A+ or B+ and fits into our case quite nicely.

It's designed to plug directly onto the Pi Model A+ or B+. While not specifically designed for Pi Model A or B, [you can use it with A/B if you solder in an extra-tall 2x13 header \(not included\) \(https://adafru.it/eib\)](https://adafru.it/eib) instead of the included 2x20 header

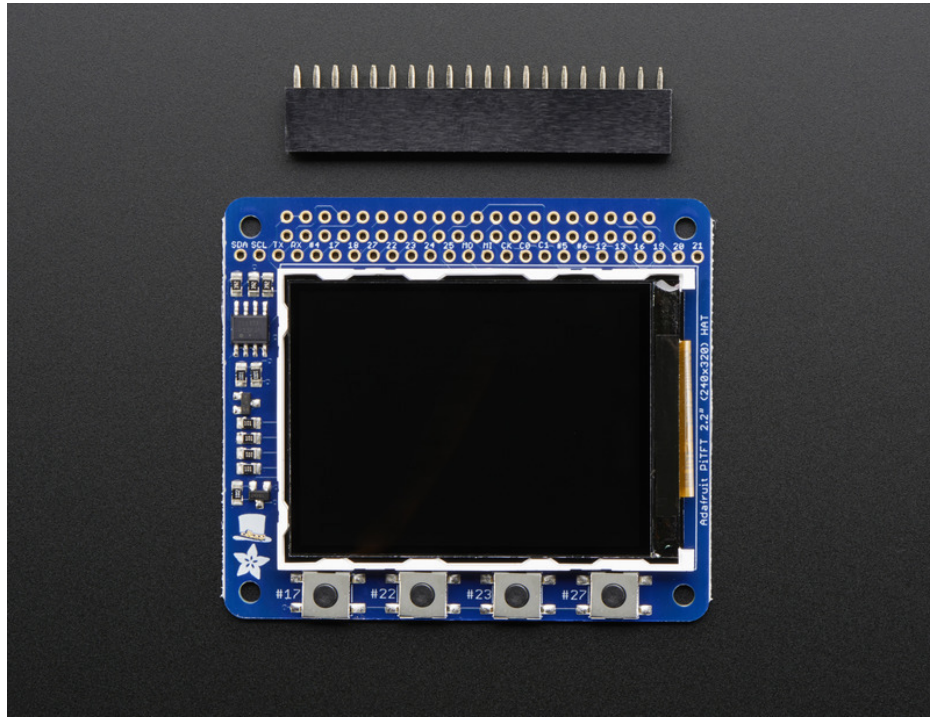


This design uses the hardware SPI pins (SCK, MOSI, MISO, CE0, CE1) as well as GPIO #25. All other GPIO are unused. Since we had a tiny bit of space, there's 4 flat tactile switches wired to four GPIOs, that you can use if you want to make a basic user interface. For example, you can use one as a power on/off button.

All the other pins are available on a 25-pin long breakout line.



To make it super easy for use: we've created a custom kernel package based off Notro's awesome framebuffer work, so you can install it over your existing Raspbian (or derivative) images in just a few commands.



Comes as a fully assembled display PCB and an additional 2x20 GPIO header. Some light soldering is required to attach the 2x20 GPIO header to the HAT but it's fast and easy for anyone with a soldering iron and solder

This tutorial series shows you how to install the software, play small videos, or display images such as from your PiCam and more!

## Easy Install



The PiTFT requires some device tree support and a couple other things to make it a nice stand-alone display. If you just want to get going, check out the following for easy-install instructions!



The same installer is used for all PiTFTs, you will pick and configure the setup during installation!

## Install Raspbian on an SD Card

You'll need to start with Raspbian or Raspbian Lite.

The last known for-sure tested-and-working version is March 13, 2018  
(<https://downloads.raspberrypi.org/raspbian/images/raspbian-2018-03-14/> (<https://adafru.it/F2N>)) from  
<https://downloads.raspberrypi.org/raspbian/images/> (<https://adafru.it/BFU>)

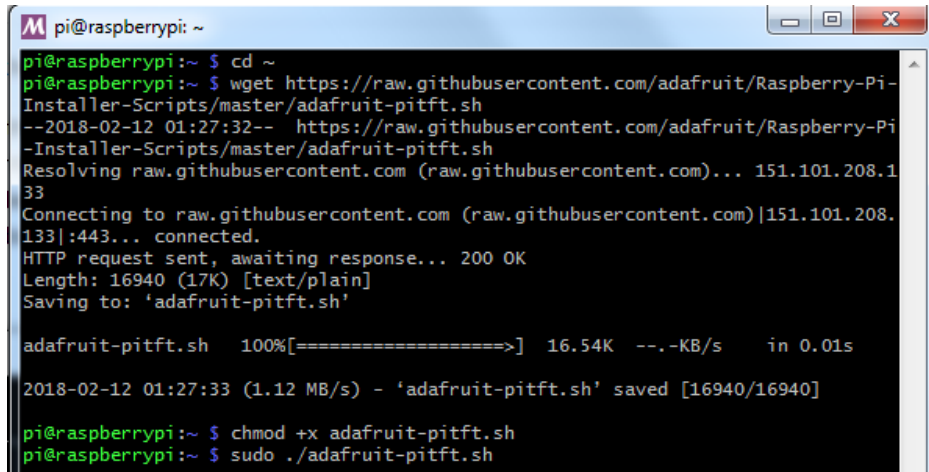
Raspbian does often 'break' stuff when new versions come out so to be safe, if you are having problems try this version!

## Installer script

This script will do all the work for you, and install both device tree overlay support as well as configure rotation and any HDMI mirroring. PiTFT no longer needs any custom kernels or modules, so you can continue to update/upgrade your Pi and it will work with the most recent releases.

Here's the commands to run. Make sure your Pi has network access, it needs to download the software!

```
cd ~
wget https://raw.githubusercontent.com/adafruit/Raspberry-Pi-Installer-Scripts/master/adafruit-pitft.sh
chmod +x adafruit-pitft.sh
sudo ./adafruit-pitft.sh
```



```
pi@raspberrypi: ~
pi@raspberrypi:~ $ cd ~
pi@raspberrypi:~ $ wget https://raw.githubusercontent.com/adafruit/Raspberry-Pi-
Installer-Scripts/master/adafruit-pitft.sh
--2018-02-12 01:27:32-- https://raw.githubusercontent.com/adafruit/Raspberry-Pi
-Installer-Scripts/master/adafruit-pitft.sh
Resolving raw.githubusercontent.com (raw.githubusercontent.com)... 151.101.208.1
33
Connecting to raw.githubusercontent.com (raw.githubusercontent.com)|151.101.208.
133|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 16940 (17K) [text/plain]
Saving to: 'adafruit-pitft.sh'

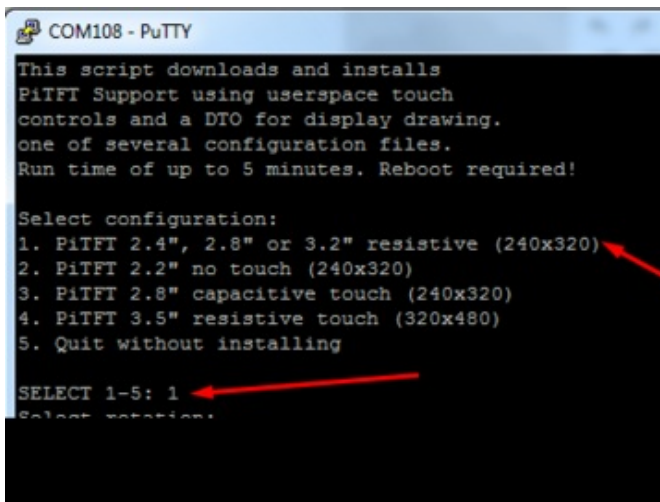
adafruit-pitft.sh  100%[=====>]  16.54K  --.-KB/s   in 0.01s

2018-02-12 01:27:33 (1.12 MB/s) - 'adafruit-pitft.sh' saved [16940/16940]

pi@raspberrypi:~ $ chmod +x adafruit-pitft.sh
pi@raspberrypi:~ $ sudo ./adafruit-pitft.sh
```

## PiTFT Selection

Once you run it you will be presented with menus for configuration.



```
COM108 - PuTTY
This script downloads and installs
PiTFT Support using userspace touch
controls and a DTO for display drawing.
one of several configuration files.
Run time of up to 5 minutes. Reboot required!

Select configuration:
1. PiTFT 2.4\", 2.8\" or 3.2\" resistive (240x320)
2. PiTFT 2.2\" no touch (240x320)
3. PiTFT 2.8\" capacitive touch (240x320)
4. PiTFT 3.5\" resistive touch (320x480)
5. Quit without installing

SELECT 1-5: 1
Select rotation:
```

For the 2.4\", 2.8\" and 3.2\" PiTFT with resistive touchscreen overlay select #1

```
This script downloads and installs
PiTFT Support using userspace touch
controls and a DTO for display drawing.
one of several configuration files.
Run time of up to 5 minutes. Reboot required!

Select configuration:
1. PiTFT 2.4", 2.8" or 3.2" resistive (240x320)
2. PiTFT 2.2" no touch (240x320)
3. PiTFT 2.8" capacitive touch (240x320)
4. PiTFT 3.5" resistive touch (320x480)
5. Quit without installing

SELECT 1-5: 2
```

For the 2.2" PiTFT select #2

```
pi@raspberrypi: ~
This script downloads and installs
PiTFT Support using userspace touch
controls and a DTO for display drawing.
one of several configuration files.
Run time of up to 5 minutes. Reboot required!

Select configuration:
1. PiTFT 2.4", 2.8" or 3.2" resistive (240x320)
2. PiTFT 2.2" no touch (240x320)
3. PiTFT 2.8" capacitive touch (240x320)
4. PiTFT 3.5" resistive touch (320x480)
5. Quit without installing

SELECT 1-5: 3
```

For the 2.8" Capacitive PiTFT select #3

```
COM108 - PuTTY
This script downloads and installs
PiTFT Support using userspace touch
controls and a DTO for display drawing.
one of several configuration files.
Run time of up to 5 minutes. Reboot required!

Select configuration:
1. PiTFT 2.4", 2.8" or 3.2" resistive (240x320)
2. PiTFT 2.2" no touch (240x320)
3. PiTFT 2.8" capacitive touch (240x320)
4. PiTFT 3.5" resistive touch (320x480)
5. Quit without installing

SELECT 1-5: 4
Select rotation:
```

For the 3.5" PiTFT select #4

## Rotation

Next you will be asked for the rotation you want, don't worry if you're not 100% sure which you want, you can always change this later by re-running the script

```
SELECT 1-5: 2
Select rotation:
1. 90 degrees (landscape)
2. 180 degrees (portait)
3. 270 degrees (landscape)
4. 0 degrees (portait)

SELECT 1-4: 1
```

It will take a few minutes to install the software and download all the things...



```
pi@raspberrypi: ~
1. PiTFT 2.4", 2.8" or 3.2" resistive (240x320)
2. PiTFT 2.2" no touch (240x320)
3. PiTFT 2.8" capacitive touch (240x320)
4. PiTFT 3.5" resistive touch (320x480)
5. Quit without installing

SELECT 1-5: 2
Select rotation:
1. 90 degrees (landscape)
2. 180 degrees (portait)
3. 270 degrees (landscape)
4. 0 degrees (portait)

SELECT 1-4: 1
[PITFT] Checking init system...
Found systemd
/boot is mounted
[PITFT] System update
Updating apt indexes...
.....
Reading package lists...
.....
[PITFT] Installing Python libraries & Software...
Installing Pre-requisite Software...This may take a few minutes!
```

## Configuring what shows where

You have a few different ways to set up the PiTFT, we ask 2 questions to figure out what you want

### PiTFT as Text Console (best for Raspbian 'Lite')

This is the simplest to set-up type of use. Its great if you have a simple text based or pygame/SDL based interface. If you want the PiTFT to act as a text console you can expect:

- HDMI will be 'deactivated' - nothing appears on the HDMI output but a black screen
- The login prompt appears on the Pi
- The Pi is all text, not a GUI (no PIXEL desktop)
- Keyboard and mouse are used only by the PiTFT interface
- Framebuffer-capable software (such as **fbi** for displaying images, **mplayer** for videos, or pygame software, etc) appear on the PiTFT
- OpenGL accelerated software *will not appear on the PiTFT* (it is unaccelerated framebuffer only)
- But, non-OpenGL-accelerated graphics software is a bit faster than using HDMI mirroring (not tons faster but you're not running **fbcp** which will always make it faster)

If you want that say **Yes** to the question **Would you like the console to appear on the PiTFT display**

```
Would you like the console to appear on the PiTFT display? [y/n] y
[PITFT] Updating console to PiTFT...
Remove fbcp from /etc/rc.local...
Configuring boot/config.txt for default HDMI
Set up main console turn on
Updating /boot/cmdline.txt
Turning off console blanking
Setting raspi-config to boot to console w/o login...
Created symlink /etc/systemd/system/default.target -> /lib/systemd/system/multi-user.target.
[PITFT] Success!

Settings take effect on next boot.

REBOOT NOW? [y/N]
```

Then simply reboot. Once rebooted you will not see anything on HDMI, but the console will appear on the PiTFT. That's it!

## PiTFT as HDMI Mirror (Best for Raspbian Full/PIXEL)

This option is the easiest to understand: whatever appears on the HDMI display will be 'mirrored' to the PiTFT. Note that HDMI is much higher resolution so it's not like it turns the PiTFT into a 1080p display. This is great for when you want to run OpenGL-optimized software, PIXEL desktop software, or really anything. The down-side is its a little slower than drawing directly to the framebuffer. You may not notice it but it's worth us mentioning!

- HDMI will be 'activated' but at a lower resolution - you can change this later but it looks best at 320x240 (PiTFT 2.2", 2.4", 2.8" and 3.2") or 480x320 (PiTFT 3.5")
- The login prompt or GUI appears on both HDMI and PiTFT at the same time
- Keyboard and mouse are shared, since the display is mirrored
- All graphics appear on both HDMI and PiTFT, thanks to **fbcp**

If you want that say **Yes** to the question **Would you like the HDMI display to mirror to the PiTFT display?**

## PiTFT as Raw Framebuffer Device

For advanced users who are comfortable using framebuffer devices, it is possible to have the PiTFT and HDMI graphics be *both* active and display different data.

- HDMI will be active and act like a normal Pi
- The login prompt or GUI (PIXEL) appears on the HDMI
- PiTFT appears black, nothing appears on it
- Keyboard and mouse are used by the HDMI interface but can, in theory, be captured and used to change graphics on PiTFT through programming
- Framebuffer-capable software (such as **fbi** for displaying images, **mplayer** for videos, or pygame software, etc) *can* appear on the PiTFT if you set it up to display to `/dev/fb1`
- OpenGL accelerated software *will never appear on the PiTFT* (it is unaccelerated framebuffer only)

If you want that, say **No** to both of the configuration questions!



You can always change your mind after setting up one of the configurations, depending on your needs! Just re-run the script

## Unsupported Full Images

Historically, we provided full 'images' of Raspbian. This worked OK until Raspbian started doing releases every few months. These are no longer supported, and won't even boot on Pi 3B+, so we recommend the script above.

There's the larger 'classic Jessie' image that will boot into X by default, and requires a 8G image, it has a lot more software installed. There's also the smaller 'Jessie Lite' that will boot into the command line, and can be burned onto a 2G card! Click below to download and install into a new SD card. [Unzip and follow the classic SD card burning tutorials \(https://adafru.it/aMW\)](https://adafru.it/aMW)

## PiTFT 2.2" Images

- [Raspbian Jessie 2016/10/23-based image \(https://adafru.it/sbg\)](https://adafru.it/sbg)
- [Raspbian Jessie Lite 2016/10/23-based image \(https://adafru.it/sbh\)](https://adafru.it/sbh)
- [Raspbian Jessie 2016/03/25-based image \(https://adafru.it/mAe\)](https://adafru.it/mAe)
- [Raspbian Jessie Lite 2016/03/25-based image \(https://adafru.it/mAf\)](https://adafru.it/mAf)
- [Raspbian Jessie 2015/09/24-based image \(https://adafru.it/iDC\)](https://adafru.it/iDC)

- [Raspbian Wheezy 2015/09/09-based image \(https://adafru.it/idt\)](https://adafru.it/idt)

## PiTFT 2.4"/2.8"/3.2" Resistive Images

- [Raspbian Jessie 2016/9/23-based image \(https://adafru.it/s7f\)](https://adafru.it/s7f)
- [Raspbian Jessie Lite 2016/9/23-based image \(https://adafru.it/s7A\)](https://adafru.it/s7A)
- [Raspbian Jessie 2016/03/25-based image \(https://adafru.it/mA9\)](https://adafru.it/mA9)
- [Raspbian Jessie Lite 2016/03/25-based image \(https://adafru.it/mAa\)](https://adafru.it/mAa)
- [Raspbian Jessie 2015/09/24-based image \(https://adafru.it/iDA\)](https://adafru.it/iDA)
- [Raspbian Wheezy 2015/09/09-based image \(https://adafru.it/idJ\)](https://adafru.it/idJ)
- [Raspbian 2014/06/20-based image \(https://adafru.it/dSM\)](https://adafru.it/dSM)
- [Raspbian 2014/09/09-based image \(https://adafru.it/e12\)](https://adafru.it/e12)

## PiTFT 2.8" Capacitive

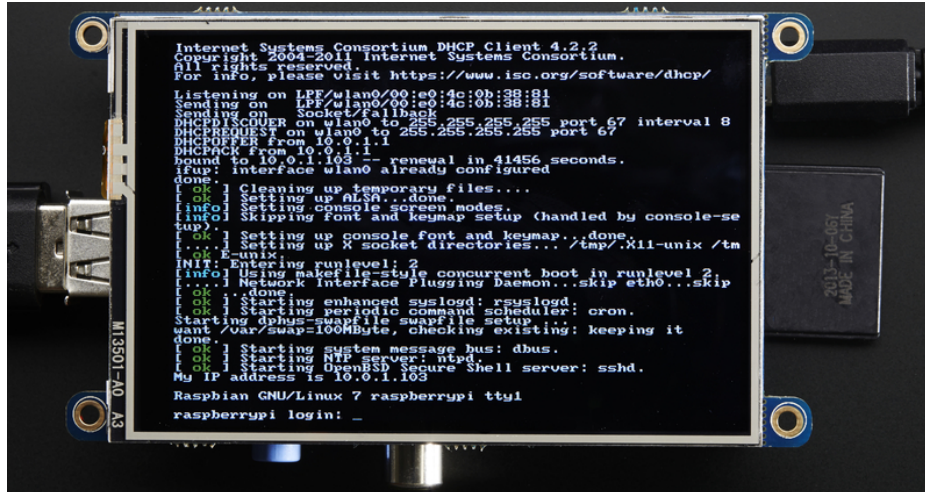
- [Raspbian Jessie 2016-09-23-based image \(https://adafru.it/saM\)](https://adafru.it/saM)
- [Raspbian Jessie Lite 2016-09-23-based image \(https://adafru.it/saN\)](https://adafru.it/saN)
- [Raspbian Jessie 2016-03-25-based image \(https://adafru.it/mAc\)](https://adafru.it/mAc)
- [Raspbian Jessie Lite 2016-03-25-based image \(https://adafru.it/mAd\)](https://adafru.it/mAd)
- [Raspbian Jessie 2015/09/24-based image \(https://adafru.it/iDy\)](https://adafru.it/iDy)
- [Raspbian Wheezy 2015/09/24-based image \(https://adafru.it/idz\)](https://adafru.it/idz)
- [Raspbian 2014/09/18-based image \(https://adafru.it/e11\)](https://adafru.it/e11)
- [Raspbian 2014/06/20-based image \(https://adafru.it/dSO\)](https://adafru.it/dSO)
- [Raspbian image from 2015/03/03 \(https://adafru.it/eUI\)](https://adafru.it/eUI)

## PiTFT 3.5" Images

- [Raspbian Jessie 2016/9/23-based image \(https://adafru.it/siF\)](https://adafru.it/siF)
- [Raspbian Jessie Lite 2016/9/23-based image \(https://adafru.it/sja\)](https://adafru.it/sja)
- [Raspbian Jessie 2016/03/25-based image \(https://adafru.it/mAb\)](https://adafru.it/mAb)
- [Raspbian Jessie 2016/03/25-based image \(https://adafru.it/mAG\)](https://adafru.it/mAG)
- [Raspbian Jessie 2015/09/24-based image \(https://adafru.it/iDD\)](https://adafru.it/iDD)
- [Raspbian Wheezy 2015/09/24-based image \(https://adafru.it/idy\)](https://adafru.it/idy)
- [Raspbian 2014/09/09-based image \(https://adafru.it/e10\)](https://adafru.it/e10)
- [Raspbian 2015/03/12 image \(https://adafru.it/eUE\)](https://adafru.it/eUE)

## Console Configuration

If you've used our installer script, this step is not required, it's already done! This is just for advanced users who are curious on how to configure and customize the console



One fun thing you can do with the display is have it as your main console instead of the HDMI/TV output. Even though it is small, with a good font you can get 20 x 40 of text. For more details, check out <https://github.com/notro/fbtf/wiki/Boot-console> (<https://adafru.it/cXQ>)

First up, we'll update the boot configuration file to use the TFT framebuffer `/dev/fb1` instead of the HDMI/TV framebuffer `/dev/fb0`

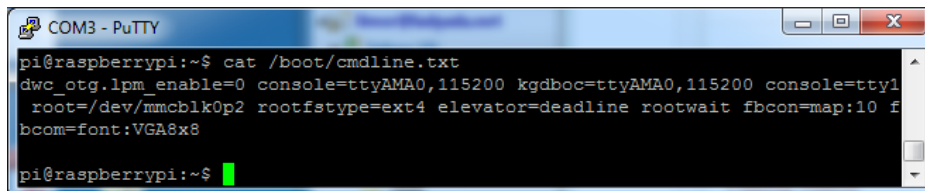
```
sudo nano /boot/cmdline.txt
```

you can also edit it by putting the SD card into a computer and opening the same file.

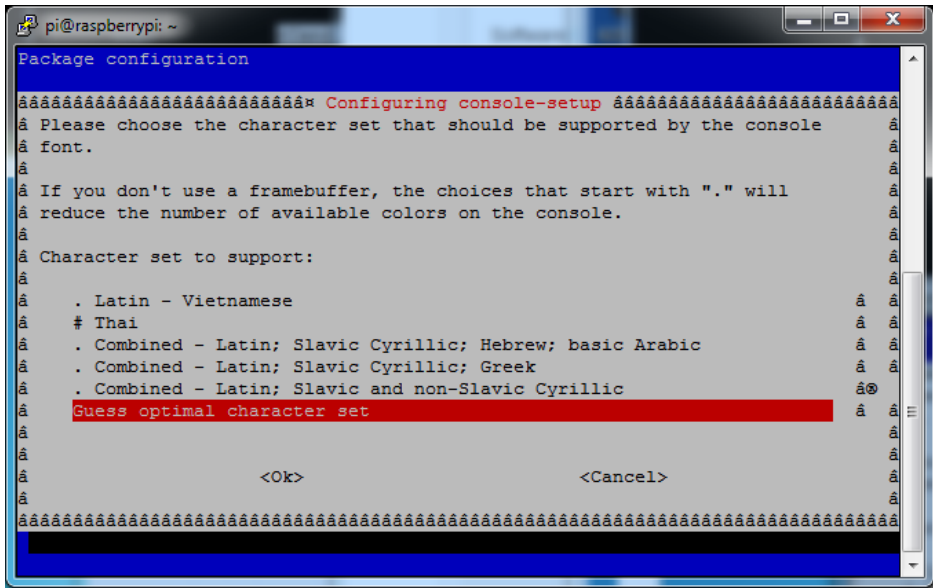
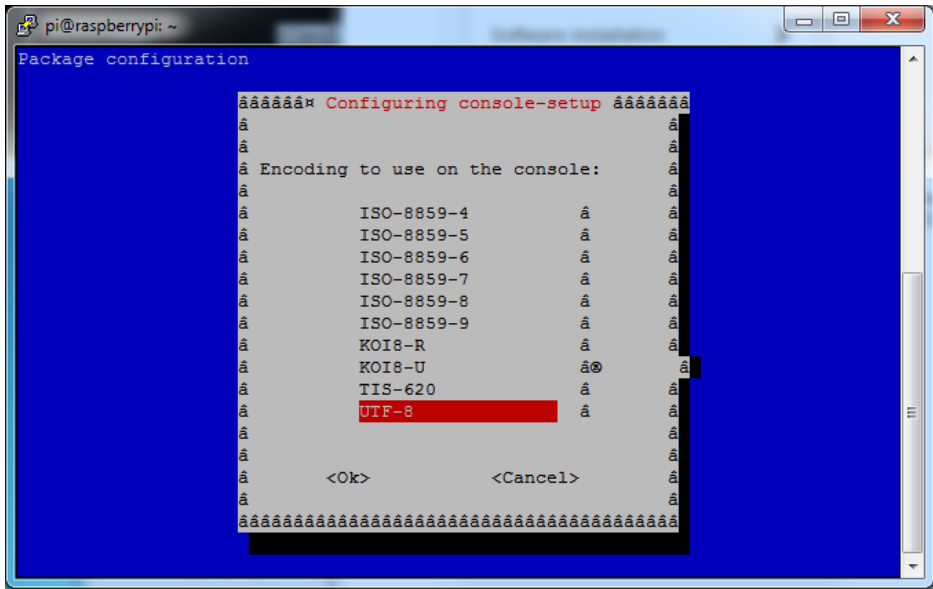
At the end of the line, find the text that says `rootwait` and right after that, enter in: `fbcon=map:10 fbcon=font:VGA8x8` then save the file.

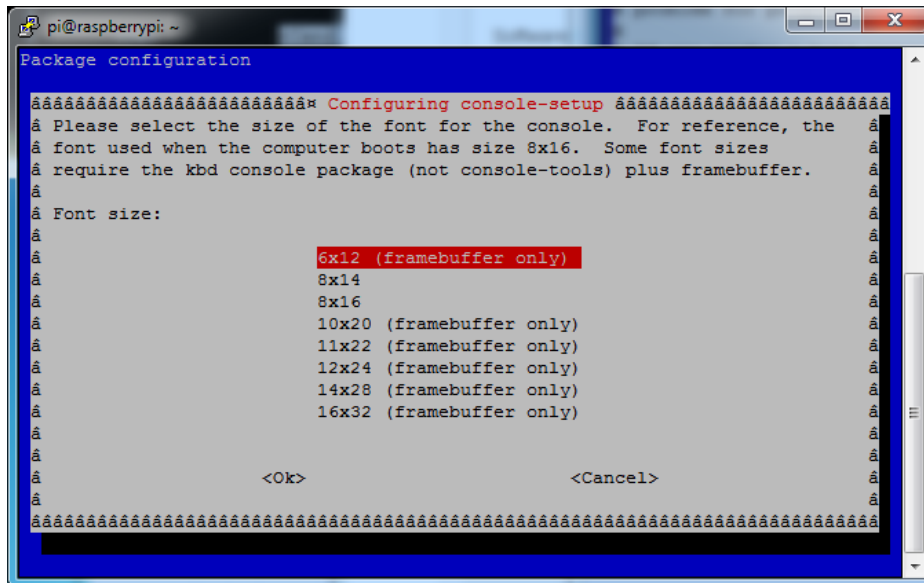
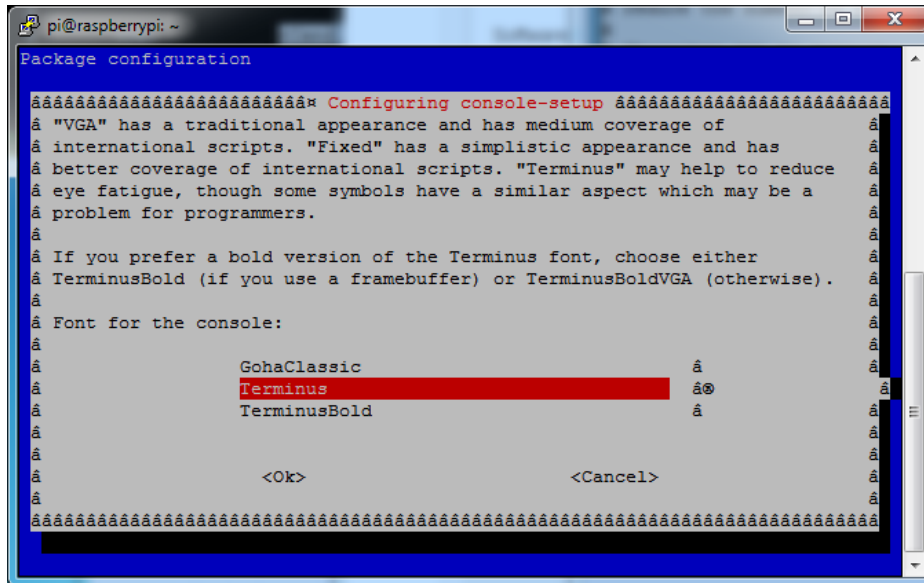
On the next boot, it will bring up the console.

Note that the kernel has to load up the display driver module before it can display anything on it so you won't get the rainbow screen, a NooBs prompt, or a big chunk of the kernel details since the module is loaded fairly late in the boot process.



I think the VGA8x8 font is a bit chunky, you probably want 12x6 which is what is shown in the photo above. To change the font, run `sudo dpkg-reconfigure console-setup` and go thru to select Terminus 6x12





## Turn off Console Blanking

You may notice the console goes black after 30 minutes, this is a sort of 'power saving' or 'screensaver' feature.

### Raspbian Jessie

Add the following line to `/etc/rc.local`

```
sudo sh -c "TERM=linux setterm -blank 0 >/dev/tty0"
```

on the line before the final `exit 0`

### Raspbian Wheezy

You can disable this by editing `/etc/kbd/config` and looking for

```
BLANK_TIME=30
```

and setting the blank time to 0 (which turns it off)

```
BLANK_TIME=0
```



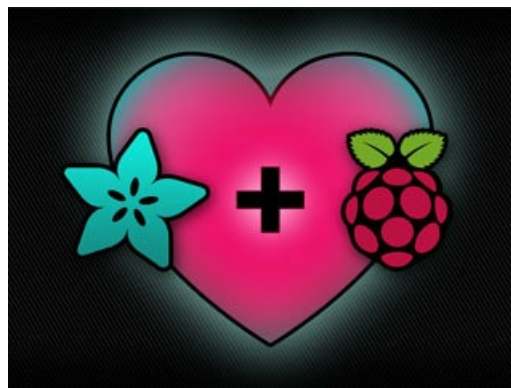
## Displaying Images

You can display every day images such as GIFs, JPGs, BMPs, etc on the screen. To do this we'll install **fbi** which is the **frame buffer image** viewer (not to be confused with the FBI agency!)

`sudo apt-get install fbi` will install it

```

COM3 - PuTTY
pi@raspberrypi:~$ sudo apt-get install fbi
Reading package lists... Done
Building dependency tree
Reading state information... Done
Suggested packages:
  imagemagick
The following NEW packages will be installed:
  fbi
0 upgraded, 1 newly installed, 0 to remove and 52 not upgraded.
Need to get 59.7 kB of archives.
After this operation, 157 kB of additional disk space will be used.
Get:1 http://mirrordirector.raspbian.org/raspbian/ wheezy/main fbi armhf 2.07-10 [59.7 kB]
Fetched 59.7 kB in 1s (40.0 kB/s)
Selecting previously unselected package fbi.
(Reading database ... 64758 files and directories currently installed.)
Unpacking fbi (from ../archives/fbi_2.07-10_armhf.deb) ...
Processing triggers for mime-support ...
Processing triggers for man-db ...
Setting up fbi (2.07-10) ...
pi@raspberrypi:~$
  
```



Grab our lovely wallpapers with

```

wget http://adafruit-download.s3.amazonaws.com/adapiluv320x240.jpg
wget http://adafruit-download.s3.amazonaws.com/adapiluv480x320.png (https://adafru.it/cXU)
  
```

For 320x240 PiTFTs (2.2", 2.4", 2.8" or 3.2") view it with

```

sudo fbi -T 2 -d /dev/fb1 -noverbose -a adapiluv320x240.jpg
  
```

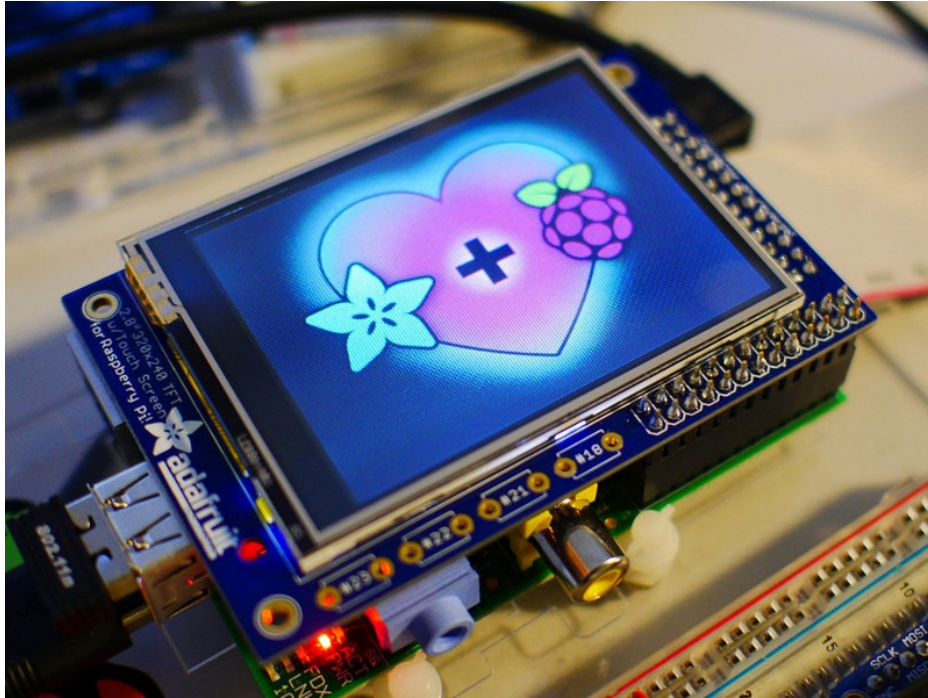
or for 3.5" PiTFTs:

```

sudo fbi -T 2 -d /dev/fb1 -noverbose -a adapiluv480x320 (https://adafru.it/cXU).png
  
```



That's it!



## Playing Videos



### How To Play Videos

You can play many types of videos on the screen, using mplayer you don't even need to run X and you can script the movies to play using Python. We'll show you how to just play one video for now.

To demo, we'll use an mp4 of Big Buck Bunny for 320 pixel wide screens. Below we show you how to create/resize videos, but to make it easy, just download our version with:

```
wget http://adafruit-download.s3.amazonaws.com/bigbuckbunny320p.mp4 (https://adafru.it/cXR)
```



The video is 30MB which is a lot if you haven't expanded your SD card yet. Before you do this, run `sudo raspi-config` to expand the SD card so you don't run out of space!

If you don't have `mplayer` yet, run

```
sudo apt-get update
```

```
sudo apt-get install mplayer
```

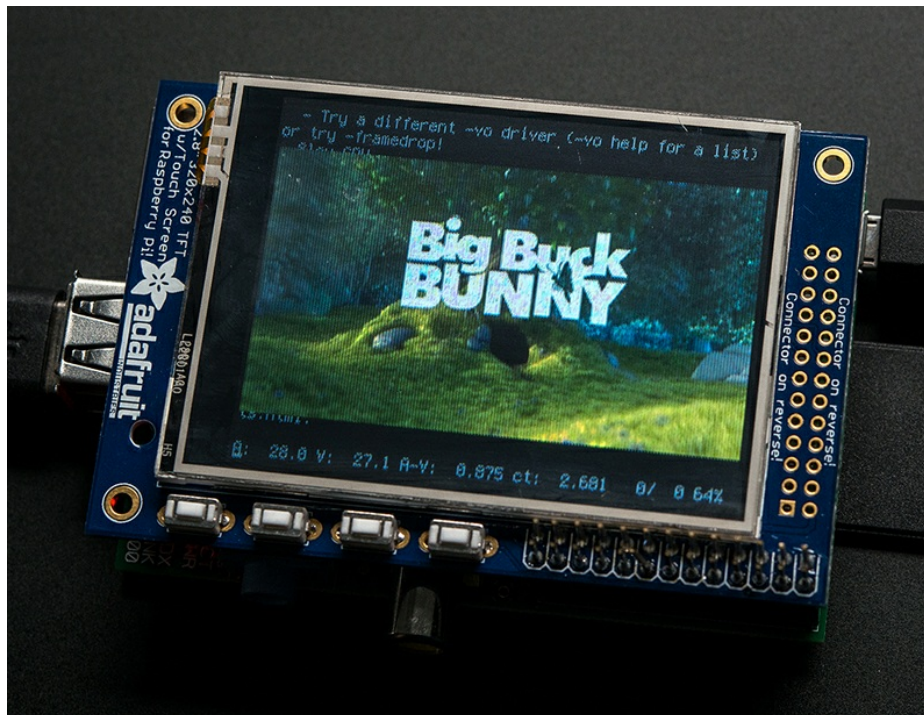
to install it. It may take a few minutes to complete

```
pi@raspberrypi: ~
pi@raspberrypi ~ $ sudo apt-get install mplayer
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following extra packages will be installed:
 esound-common libaa1 libaudiofile1 libavcodec53 libavformat53 libavutil151
 libcdparanoia0 libdca0 libdirac-encoder0 libdvnav4 libdvread4 libenca0
 libesd0 libfaad2 libfribidi0 libgpm2 libgsm1 libjack-jackd2-0 liblircclient0
 liblzo2-2 libmp3lame0 libmpeg2-4 libopenal-data libopenal1 libpostproc52
 libschroedinger-1.0-0 libspeex1 libswscale2 libtheora0 libva1 libvpx1
 libx264-123 libxvidcore4 libxvnc1
Suggested packages:
 libdvcss2 pulseaudio-esound-compat gpm jackd2 lirc libportaudio2
 libroar-compat2 speex mplayer-doc netselect fping
The following NEW packages will be installed:
 esound-common libaa1 libaudiofile1 libavcodec53 libavformat53 libavutil151
 libcdparanoia0 libdca0 libdirac-encoder0 libdvnav4 libdvread4 libenca0
 libesd0 libfaad2 libfribidi0 libgpm2 libgsm1 libjack-jackd2-0 liblircclient0
 liblzo2-2 libmp3lame0 libmpeg2-4 libopenal-data libopenal1 libpostproc52
 libschroedinger-1.0-0 libspeex1 libswscale2 libtheora0 libva1 libvpx1
 libx264-123 libxvidcore4 libxvnc1 mplayer
0 upgraded, 35 newly installed, 0 to remove and 52 not upgraded.
Need to get 9,296 kB of archives.
After this operation, 20.6 MB of additional disk space will be used.
Do you want to continue [Y/n]?
```

OK now you just have to run:

```
sudo SDL_VIDEODRIVER=fbcon SDL_FBDEV=/dev/fb1 mplayer -vo sdl -framedrop bigbuckbunny320p.mp4
```

If your video is not sized for 320 wide, you may need to add a `-zoom` after `-framedrop` so that it will resize - note that this is quite taxing for the Pi, so it may result in a choppy or mis-synced video!

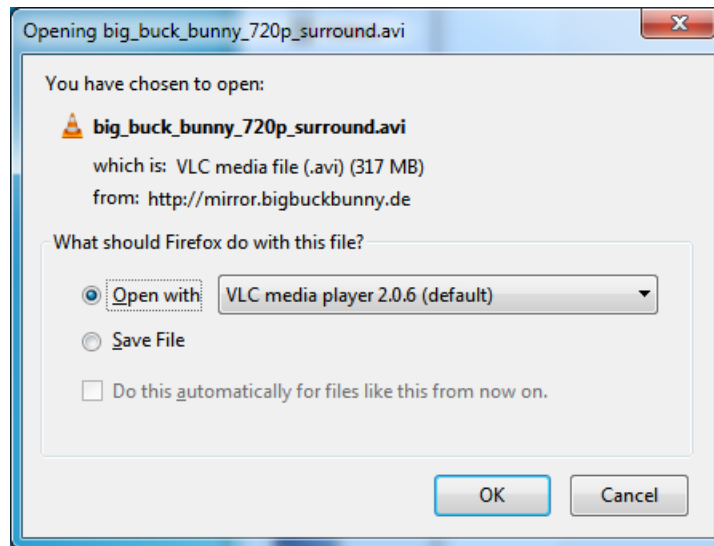


## Converting/Resizing Videos

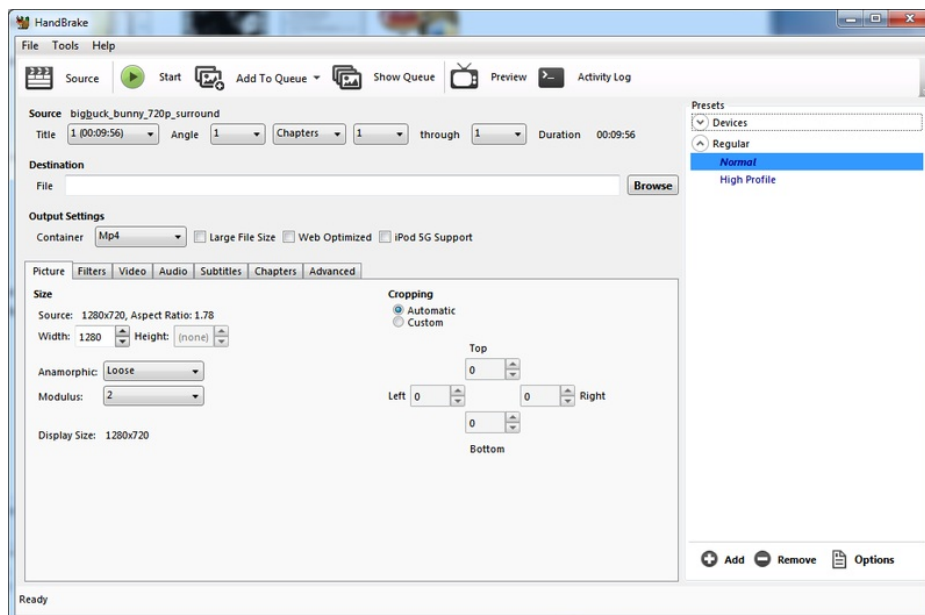
It's possible to play full length videos on the TFT plate, but since the screen is small and the Pi cant use hardware acceleration to play the videos its best to scale them down to 320x240 pixels. This will be easier for the Pi to play and also save you tons of storage space. For this demo, we'll be using the famous [Big Buck Bunny \(https://adafru.it/cXS\)](https://adafru.it/cXS)

video, which is creative commons and also very funny!

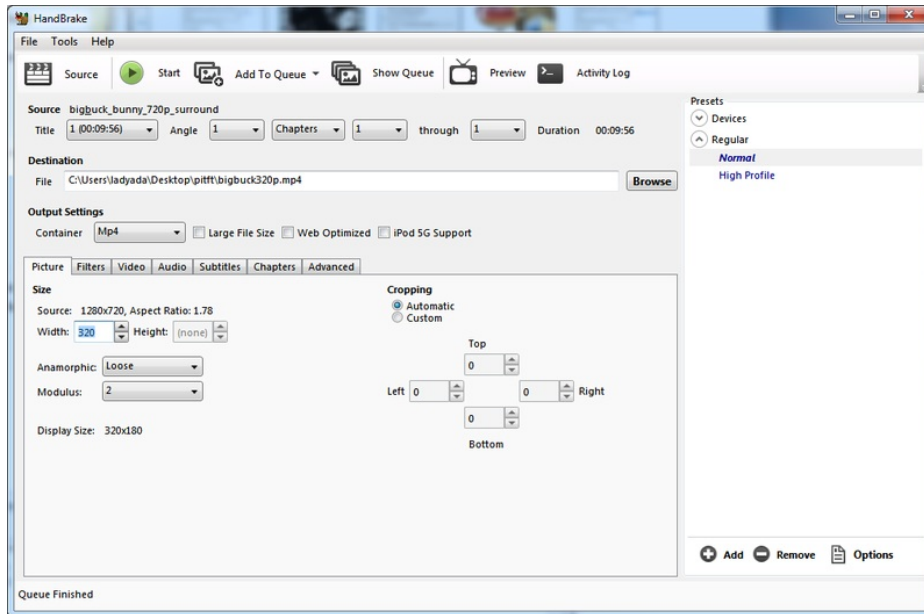
You can download it from the link above, we'll be using the 720p AVI version.



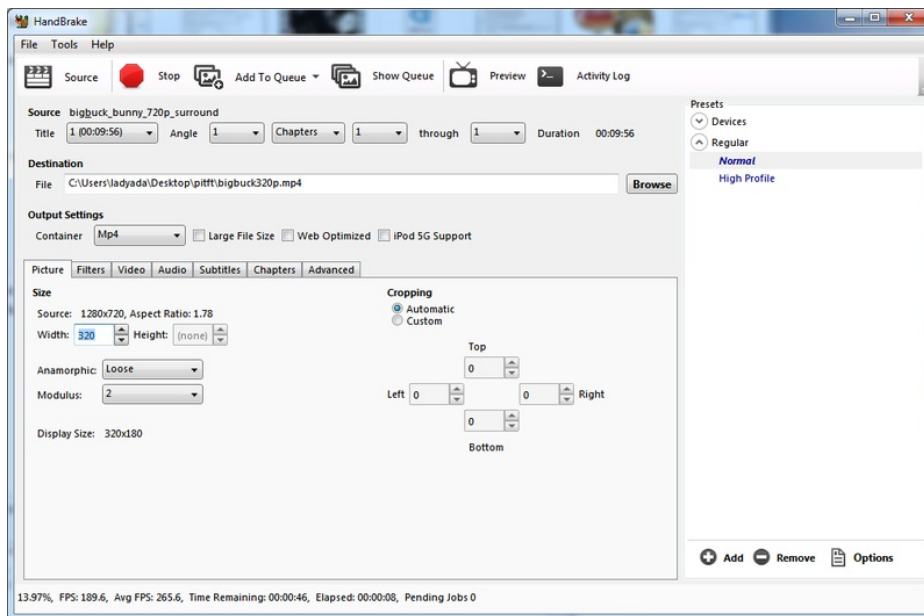
To do the conversion itself, we suggest [HandBrake](https://adafruit.it/cXT) (<https://adafruit.it/cXT>) which works great and is open source so it runs on all operating systems! Download and install from the link. Then run the installed application and open up the AVI file from before. The app will pre-fill a bunch of information about it.



Under **Destination** click **Browse...** to select a new MP4 file to save. Then under **Picture** change the **Width** to 320 (the height will be auto-calculated)



Click **START** to begin the conversion, it will take a minute or two.



That's it! You now have a smaller file. Don't forget to play it on your computer to make sure it plays right before copying it to your Pi

## Using FBCP



**The Ideal:** Adafruit's PiTFT displays are razor sharp. Whereas small composite screens on the Raspberry Pi usually require some video scaling (resulting in blurriness), PiTFT uses the GPIO header, digitally controlled pixel-by-pixel for a rock steady image. Though not a *lot* of pixels, it works great for retro gaming (and the display neatly stacks above the board, no side protuberances for video cables).

**The Downside:** this GPIO link entirely bypasses the Pi's video hardware, including the graphics accelerator. Many games and emulators *depend* on the GPU for performance gains. So the PiTFT has traditionally been limited to just a subset of specially-compiled emulators that can work and run well enough without the GPU.

**The Solution:** our latest PiTFT drivers, along with a tool called *fbc* (framebuffer copy), careful system configuration, and (optionally) the more potent Raspberry Pi 2 board open the doors to many more gaming options. Existing emulator packages (such as RetroPie, with *dozens* of high-performance emulators and ports) — previously off-limits to the PiTFT — can run quite effectively now!

<https://adafru.it/fbe>

<https://adafru.it/fbe>

## Backlight Control

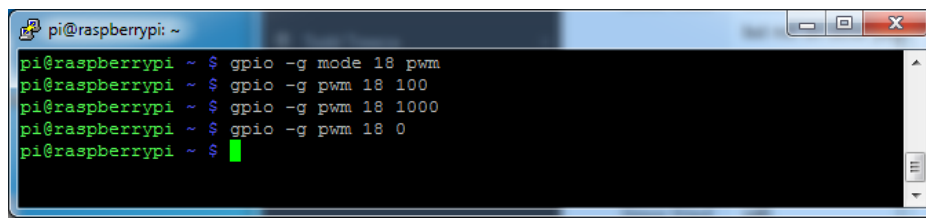
Unlike the resistive 2.8" PiTFT, this little PiTFT does not have a resistive touch controller chip that we can take advantage of as an extra backlight control pin. Instead, you can set use GPIO #18 as an on/off or PWM control.

Note that if you are playing audio out the headphone jack, you can't use the PWM capabilities of GPIO #18 at the same time, the PWM function is reassigned to do audio. However, you can use it as a simple on/off pin

There's python code available for controlling the PWM on #18 but you can also just use the kernel module and shell commands.

With these basic shell commands, you can set the GPIO #18 pin to PWM mode, set the output to 100 (out of 1023, so dim!), set the output to 1000 (out of 1023, nearly all the way on) and 0 (off)

```
sudo apt-get install wiringpi
gpio -g mode 18 pwm
gpio -g pwm 18 100
gpio -g pwm 18 1000
gpio -g pwm 18 0
```

A screenshot of a terminal window on a Raspberry Pi. The window title is "pi@raspberrypi: ~". The terminal shows the following commands and their outputs:

```
pi@raspberrypi ~ $ gpio -g mode 18 pwm
pi@raspberrypi ~ $ gpio -g pwm 18 100
pi@raspberrypi ~ $ gpio -g pwm 18 1000
pi@raspberrypi ~ $ gpio -g pwm 18 0
pi@raspberrypi ~ $
```

The terminal prompt is green, and the output of the last command is a green cursor.

Try other numbers, from 0 (off) to 1023 (all the way on)!

## Extras!

### Boot to X Windows on PiTFT

---

Use the Pi installer script, select **N** on having the console appear on the Pi, and say **Y** to mirroring HDMI. Then enable PIXEL using `sudo raspi-config` (boot to desktop)



## HELP! (FAQ)

---

### ☐ My PiTFT used to work, now it doesn't!

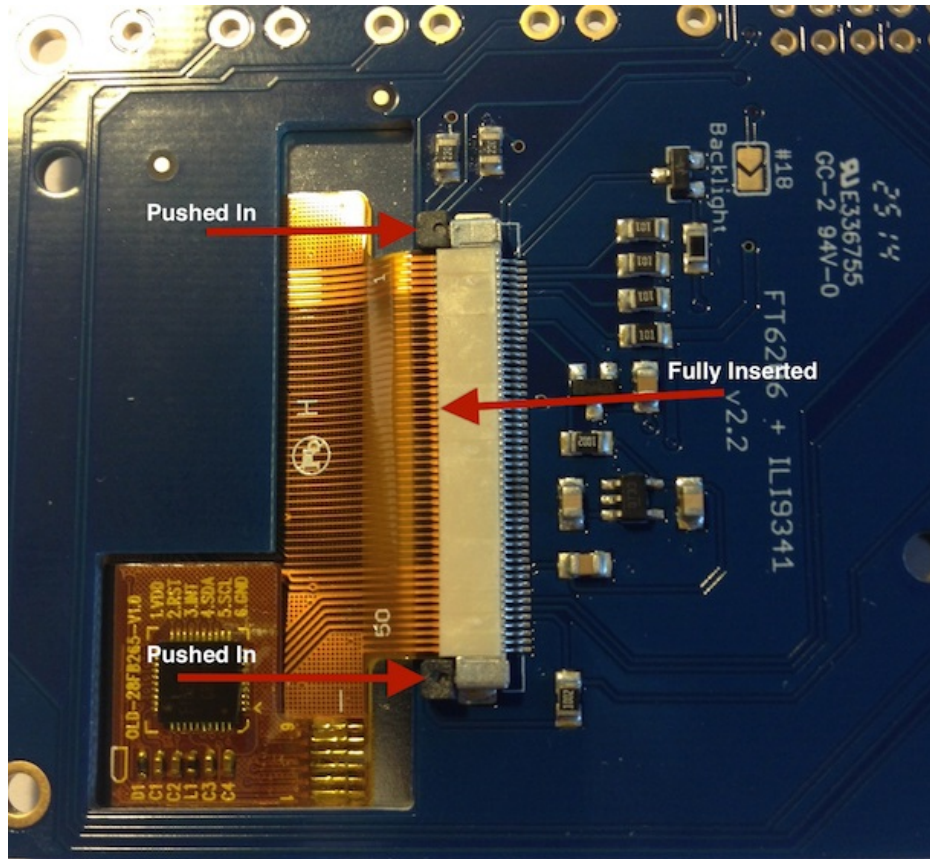
If you messed with `/boot/config.txt` or `/etc/rc.local` you may have removed or disabled some of the elements required for the PiTFT to work. Try re-running the Easy Installer script!

---

□ I'm booting my Pi with the PiTFT and the HDMI output 'locks up' during boot!

**It looks like the Pi is 'halting' or 'locking' up during boot** but what is really happening is the console is switching from the HDMI output to the PiTFT console output.

Check your PiTFT connections, particularly make sure you seated the PiTFT on the Pi properly, nothing is in the way, and the TFT flex connector is seated properly.



□ My PiTFT works for a bit and then I get a black screen with a short line of white pixels in one corner

Sounds like you tried to configure your Pi to 'boot straight to X', that is, start up the graphics interface on boot. This doesn't work by default because the Pi operating system is not expecting a PiTFT so it boots to the HDMI output. See below for how to set up your Pi to boot to X on the PiTFT

To 'fix' this, you can either connect an HDMI monitor, then in a terminal window run `sudo raspi-config` and configure the Pi to boot to the command line not X! If you do not have an HDMI monitor, you can also try a console cable

---

□ I'm trying to run startx and I get FATAL: Module g2d\_23 not found.

don't forget you have to remove the turbo file!

```
sudo mv /usr/share/X11/xorg.conf.d/99-fbturbo.conf ~
```

---

□ How come OMX-Player and Minecraft and other programs don't appear on the PiTFT display?

Some programs are graphics-optimized, particularly the video playback tools and some other programs like Minecraft. They write 'directly' to the HDMI output, and cannot write to the PiTFT so there is no way to directly make them work. However, you *can* have the output go to HDMI and then mirror the HDMI onto the PiTFT with **fbcp**. Using the Easy Installer, select **Mirror HDMI**

---

□ Why doesn't the tactile button on GPIO #21 work?

On some older PiTFTs we had one of the buttons labeled #21 - that's the original RasPi name for that pin. If you're using a V2 (chance is, you are!) that is now called #27.

All the PiTFT's we ship now have the button labeled #21 and #27

---

□ I want better performance and faster updates!

You can change the SPI frequency (overclock the display) by editing `/boot/config.txt` and changing the `dtoverlay` options line to:

```
dtoverlay=pitft28r,rotate=90,speed=62000000,fps=25
```

Or whatever you like for speed, rotation, and frames-per-second. BUT, here's the thing, the Pi only supports a *fixed number* of SPI frequencies. So tweaking the number a little won't do anything. The kernel will round the number to

the closest value. You will always get frequencies that are 250MHz divided by an even number. Here's the only SPI frequencies this kernel supports

- 15,625,000 (a.k.a 16000000 = 16 MHz)
- 17,857,142 (a.k.a. 18000000 = 18 MHz)
- 20,833,333 (a.k.a 21000000 = 21 MHz)
- 25,000,000 (= 25 MHz)
- 31,250,000 (a.k.a 32000000 = 32MHz)
- 41,666,666 (a.k.a 42000000 = 42MHz)
- 62,500,000 (a.k.a 62000000 = 62MHz)

So if you put in 48000000 for the speed, you won't actually get 48MHz, you'll actually only get about 42MHz because it gets rounded down. We tested this display nicely with 32MHz and we suggest that. But you can put in 42MHz or even try 62MHz and it will update faster

You can tweak fps (frames per second) from 20 to 60 and frequency up to 62MHz for tradeoffs in performance and speed. Reboot after each edit to make sure the settings are loaded properly. There's a trade off that if you ask for higher FPS you're going to load the kernel more because it's trying to keep the display updated.

---

## □ How can I take screenshots of the little screen?

We took the screenshots for this tutorial with **fbgrab**

```
wget http://fbgrab.monells.se/fbgrab-1.2.tar.gz
tar -zxvf fbgrab*.gz
cd fbgrab/
make
```



.fbgrab screenshot.png

```
COM3 - PuTTY
pi@raspberrypi:~$ wget http://fbgrab.monells.se/fbgrab-1.2.tar.gz
--2014-04-21 19:26:22-- http://fbgrab.monells.se/fbgrab-1.2.tar.gz
Resolving fbgrab.monells.se (fbgrab.monells.se)... 66.33.214.148
Connecting to fbgrab.monells.se (fbgrab.monells.se)|66.33.214.148|:80... connect
ed.
HTTP request sent, awaiting response... 200 OK
Length: 12836 (13K) [application/x-tar]
Saving to: `fbgrab-1.2.tar.gz'

100%[=====>] 12,836      --.-K/s   in 0.03s

2014-04-21 19:26:22 (497 KB/s) - `fbgrab-1.2.tar.gz' saved [12836/12836]

pi@raspberrypi:~$ tar -zxvf fbgrab-1.2.tar.gz
fbgrab/
fbgrab/fbgrab.c
fbgrab/INSTALL
fbgrab/fbgrab.1.man
fbgrab/COPYING
fbgrab/Makefile
pi@raspberrypi:~$ cd fbgrab/
pi@raspberrypi:~/fbgrab$ make
cc -g -Wall  fbgrab.c -lpng -lz -o fbgrab
gzip --best --to-stdout fbgrab.1.man > fbgrab.1.gz
pi@raspberrypi:~/fbgrab$ ./fbgrab
Usage:  ./fbgrab      [-hi] [-(C|c) vt] [-d dev] [-s n] [-z n]
          [-f fromfile -w n -h n -b n] filename.png
pi@raspberrypi:~/fbgrab$ ./fbgrab filemanager.png
Resolution: 320x240 depth 16
Converting image from 16
Now writing PNG file (compression -1)
```

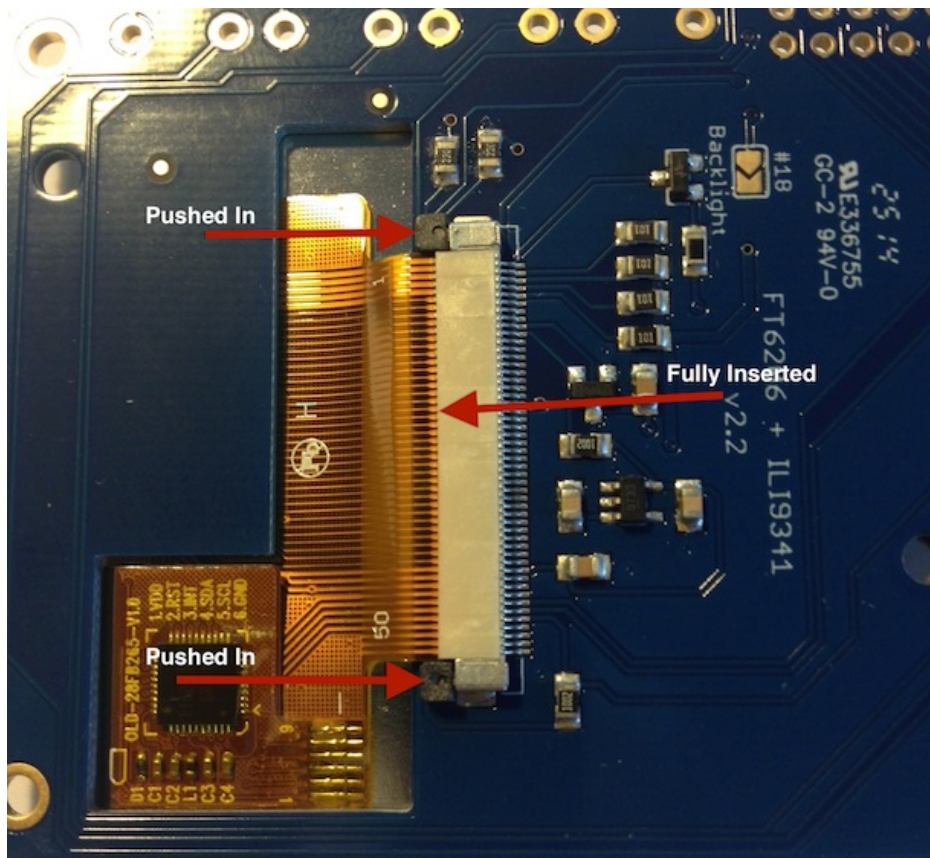
□ How do I automatically boot to X windows on the PiTFT?

Make sure your Pi boots to the graphical PIXEL desktop on the HDMI output monitor, then using the Easy Installer, select **Mirror HDMI**

---

□ My screen isn't working/works erratically/looks funny

Check to make syre that the flat flex cable is fully seated in the connetor and the 'ears' are pushed in to secure it. See the picture for what it should look like:



---

□ On my first run of startx I get a window saying "GDBus Error.org.Freedesktop Policy Kit1 Error: Failed Cannot determine user of subject"

This happens on the Raspberry Pi the first time you run startx, no matter what display. You can just re-start X and it wont appear again.

---

□ Can I get a right-click from the touch-screen?

Yes! Please see this post:

<https://forums.adafruit.com/viewtopic.php?f=47&t=77528&p=393280#p393322>

---

📄 I'm having difficulties with the STMPE resistive touch screen controller

[Here's a hack for the device tree overlay that can force different SPI modes, sometimes that helps!](#)

---

## □ My PiTFT's rotation/calibration isn't working in X11

X11 (the graphical system) has changed how it gets touchscreen input, so if you rotate the display and the calibration isn't being picked up:

Check `/usr/share/X11/xorg.conf.d` for a file called `10-evdev.conf`

If you don't see that file

1. You need to `sudo apt-get install xserver-xorg-input-evdev`, and then...
2. If you do have a `40-libinput.conf` in that same directory, you must remove it even if/once `evdev` is installed, since it will override the `10-evdev.conf` otherwise.

[Thanks to cerebrate in the forums for the hint!](#)



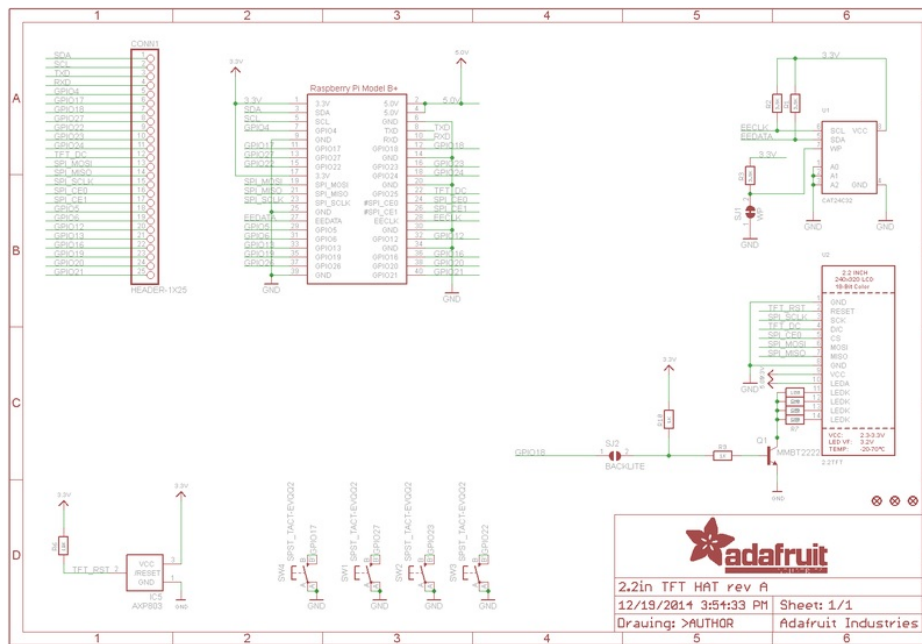


## Downloads

- The latest kernel fork that adds all the TFT, touchscreen, and other addons is here on github (<https://adafru.it/aPa>)
- ILI9340 (datasheet) (<https://adafru.it/ehr>) controller with built in pixel-addressable video RAM buffer
- Display datasheet (<https://adafru.it/ehs>)
- EagleCAD PCB files on GitHub (<https://adafru.it/rFP>)
- Fritzing object in the Adafruit Fritzing Library (<https://adafru.it/aP3>)

## Schematic

click to enlarge



## Fabrication Print

