

3,5" TFT LCD Shield pro Raspberry 320 x 480 SPI RGB Dotykový

1. POPIS

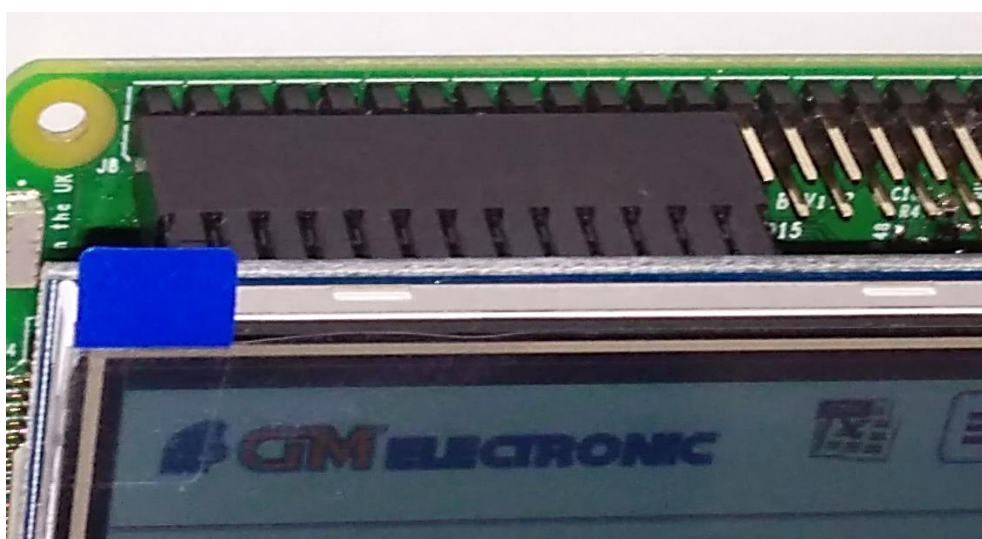
Display Shield navržený pro Raspberry PI model b / b+ / 2 je ideální alternativou pro HDMI monitor. Velice slušné rozlišení 320 x 480 px, barevný s dotykovou odporovou vrstvou. Instalace na Raspbian během několika minut.

2. SPECIFIKACE

Napájení	3,3 VDC	Displej	3,5" TFT
Komunikační sběrnice	SPI	Rozlišení	320 x 480
Kompatibilita SW	Raspbian a podobné	Rozměry (cm)	8,5 x 5,5 x 1,7
Kompatibilita HW	PI model b / b+ / 2	Hmotnost	51,1 g

3. Zapojení

Tento shield nevyžaduje žádné externí zapojení, pouze vsuňte modul do vývojového kitu Raspberry.





4. Jednoduchý návod na instalaci

PiScreen Driver Install Instructions

Bellow you will instructions for installing the appropriate drivers for PiScreen on a Raspberry Pi running Raspbian. These have been tested on all models of the Raspberry Pi. (including the Pi 2)

These instructions are based on a vanilla install of 2015-05-05-raspbian-wheezy.img. They should work for almost all versions of Raspbian. If installing drivers onto an existing Raspbian image, **Please Backup All Data** or make a copy of your SD Card.

1. Initial Config of a New Raspberry Pi Install

After booting your Raspberry Pi for the first time on 2015-02-16-raspbian-wheezy.img, we will need to perform the normal tasks of setting up our Raspberry Pi. E.g expand filesystem, enable SSH, overclocking, etc...

```
pi@raspberrypi ~ $ sudo raspi-config
pi@raspberrypi ~ $ sudo rpi-update
pi@raspberrypi ~ $ sudo reboot
```

2. Update

We now want to update our software and OS;

```
pi@raspberrypi ~ $ sudo apt-get update
pi@raspberrypi ~ $ sudo apt-get upgrade
pi@raspberrypi ~ $ sudo reboot
```

3. Enable PiScreen Drivers

Open /boot/config.txt

```
pi@raspberrypi ~ $ sudo nano /boot/config.txt
```

For PiScreen 1;

Add this line to the bottom

```
dtoverlay=piscreen,speed=16000000,rotate=90
```

(You can try and go at a higher speed on the SPI, which will increase FPS. look for the value of 16000000 above and change it to 24000000 or even 32000000. If you are noticing funny colors after the change, then decrease the speed)

For PiScreen 2;

Add this line to the bottom

```
dtoverlay=piscreen2r
```

Now reboot

```
pi@raspberrypi ~ $ sudo reboot
```

Once your Raspberry Pi comes back up, PiScreen should change from white to black.

4. Perform a quick test

We will use fbi to display an image.

```
pi@raspberrypi ~ $ sudo apt-get install fbi
pi@raspberrypi ~ $ wget http://ozzmaker.com/piscreen/image-test.gif
pi@raspberrypi ~ $ sudo fbi -noverbose -T 1 -a -d /dev/fb1 image-test.gif
```

If the image doesn't look right, try changing the speed in step 5 from speed=16000000 to speed=8000000.

5. Extra step for Jessie

By default, Jessie will automatically load X out the HDMI port which will then blank out PiScreen which will make it look like it isn't working.

If you want X to automatically load onto PiScreen. Open up the framebuffer conf file;

```
pi@raspberrypi ~ $ sudo nano /usr/share/X11/xorg.conf.d/99-fbturbo.conf
```

Change the below line;

```
Option "fbdev" "/dev/fb0"
to
Option "fbdev" "/dev/fb1"
```

Or, if you would like to disable X from starting automatically, just run `sudo raspi-config` and look for the option to disable it.

Enable Console on PiScreen

To move the console from the HDMI/RCA to PiScreen you have to updated /boot/cmdline.txt

1. Backup cmdline.txt

```
pi@raspberrypi ~ $ cp /boot/cmdline.txt ~/cmdline.txt
```

2. Update cmdline.txt

```
pi@raspberrypi ~ $ sudo nano /boot/cmdline.txt
```

Add the text below to the end of the first line. *Not on a new line as this will not work.*

Rotate will rotate the console on PiScreen. Rotate can be 0, 1, 2 or 3.

```
fbcon=map:10 fbcon=rotate:2 fbcon=font:ProFont6x11
```

3. Reboot

Reboot for the changes to take effect.

```
pi@raspberrypi ~ $ sudo reboot
```

Calibrating Console

First, we need to find out what file handle is used for the touchscreen on the PiScreen.

```
pi@raspberrypi ~ $ ls -l /dev/input/event*
```

A file handle is created for each input device currently connected and can change depending on what is connected and how many input devices are connected.

If you have a keyboard and mouse connected, there should be 3 files. One for the keyboard, one for the mouse and the last one for the touchscreen. If you have just a keyboard connected, there would be two. One for the keyboard and one for the touchscreen.

To see what file handle is used for the touchscreen, view the file below.

```
pi@raspberrypi ~ $ cat /proc/bus/input/devices
```

Look for **ADS7846 Touchscreen** and within the same section you should see **H: Handlers=** followed by an *eventnumber*. This will tell you what file under /dev/input/ is used for the touchscreen.

Going forward, use the input event number associated to your touchscreen in the commands below.

ts_lib is used to calibrate the touche screen for console use. ts_lib also includes some other tools to test your touchscreen.

```
pi@raspberrypi ~ $ sudo apt-get install libts-bin
```

Now run the `ts_calibrate` utility, while setting the framebuffer device and touchscreen device variables;

```
pi@raspberrypi ~ $ sudo TSLIB_FBDEVICE=/dev/fb1 TSLIB_TSDEVICE=/dev/input/event0
ts_calibrate
```

`ts_lib` also includes a test utility to test your touchscreen;

```
pi@raspberrypi ~ $ sudo TSLIB_FBDEVICE=/dev/fb1 TSLIB_TSDEVICE=/dev/input/event0
ts_test
```

Running `ts_print` will output the coordinates of the touchscreen when pressed;

```
pi@raspberrypi ~ $ sudo TSLIB_FBDEVICE=/dev/fb1 TSLIB_TSDEVICE=/dev/input/event0
ts_print
```

To view raw events of the input system, E.g. touchscreen input, we can use `evtest`.

```
pi@raspberrypi ~ $ sudo apt-get install evtest
```

Run `evtest` with;

```
pi@raspberrypi ~ $ evtest /dev/input/event0
```

Console Blanking

By default, the console will blank out after 30 mins.
This can be changed by editing the value

```
BLANK_TIME=30
```

in `/etc/kbd/config`.

Setting this value to '0' will disabling blanking.

Once you have enabled the console on PiScreen, you should head over to [here](#) to learn how to enable X Windows on your PiScreen.

Enable X windows on PiScreen

There are two things that need to be done before we can use X on PiScreen.

1. Remove the hard coding to `/dev/fb0` in one of the framebuffer driver config file
2. Calibrate the touchscreen for X.

We will install a `xinput_calibrator` and a script to load the calibration data each time X starts.

The first time you run X, you will be presented with a calibration screen. This will only run once, and you will be asked to touch the touchscreen a few times.

The calibration program will create a file which stores the calibration data(
`/etc/pointercal.xinput.`)

To perform calibration again, just delete `/etc/pointercal.xinput` and restart X. You will be presented again with the calibration program once X starts.

1. Disable framebuffer driver option for `/dev/fb0`

Open up the framebuffer conf file;

```
pi@raspberrypi ~ $ sudo nano /usr/share/X11/xorg.conf.d/99-fbturbo.conf
```

Comment out the line of text below;

```
Option "fbdev" "/dev/fb0"
```

This can be done by adding a hash (#) at the beginning of line.

```
#Option "fbdev" "/dev/fb0"
```

If the above file is empty, copy in the text below;

```
Section "Device"

    Identifier      "PiScreen"

    Driver          "fbturbo"

    Option          "fbdev" "/dev/fb1"

EndSection
```

2. Install all the prerequisites required for calibration

```
pi@raspberrypi ~ $ sudo apt-get install libtool libx11-dev xinput autoconf libx11-dev
libxi-dev x11proto-input-dev -y
```

3. Download and install xinput_calibrator

```
pi@raspberrypi ~ $ git clone https://github.com/tias/xinput_calibrator
pi@raspberrypi ~ $ cd xinput_calibrator/
pi@raspberrypi ~ $ ./autogen.sh
pi@raspberrypi ~ $ make
pi@raspberrypi ~ $ sudo make install
```

4. Download and setup the calibration script

```
pi@raspberrypi ~ $ cd ~
pi@raspberrypi ~ $ wget http://ozzmaker.com/piscreen/xinput_calibrator_pintercal.sh
pi@raspberrypi ~ $ sudo cp ~/xinput_calibrator_pintercal.sh
/etc/X11/Xsession.d/xinput_calibrator_pintercal.sh
```

Open X autostart;

4.1 On older images

```
pi@raspberrypi ~ $ sudo nano /etc/xdg/lxsession/LXDE/autostart
```

4.2 On Newer images, which includes Jessie

```
pi@raspberrypi ~ $ nano /home/pi/.config/lxsession/LXDE-pi/autostart
```

Add the text below to the bottom of the file;

```
sudo /bin/sh /etc/X11/Xsession.d/xinput_calibrator_pintercal.sh
```

5. Start X Windows

```
pi@raspberrypi ~ $ FRAMEBUFFER=/dev/fb1 startx
```

Force X windows to Load to PiScreen Automatically on boot

To enable automatic login to X after boot, you will need to make a modification to two files.

File one;

```
pi@raspberrypi ~ $ sudo nano /etc/inittab
```

Scroll down to one of these lines;

Older Raspbian images;

```
1:2345:respawn:/sbin/getty 115200 tty1
Newer Raspbian images
1:2345:respawn:/sbin/getty -noclear 38400 tty1
Add a hash in front to comment it out;
#1:2345:respawn:/sbin/getty 115200 tty1
and add this line right below it;
1:2345:respawn:/bin/login -f pi tty1 /dev/tty1 2>&1
File two;
```

```
pi@raspberrypi ~ $ sudo nano /etc/rc.local
```

And add the line below to the bottom of the file, just before the exit statement;

```
su -l pi -c "env FRAMEBUFFER=/dev/fb1 startx &"
```

Disable Screen Blanking

By default, X will blank out the screen after 10 minutes of no activity. Once you start using it again, the display will come back on.

If you want to disable this or increase the timeout, you will need to create a small script to disable power saving and disable screensaver which then gets loaded when X starts.

Disable X blanking

1. Install X server utilities, they will most likely already be installed.

```
pi@raspberrypi ~ $ sudo apt-get install x11-xserver-utils
```

2. Create a small script which is used to change the screen saver settings.

```
pi@raspberrypi ~ $ sudo nano /etc/X11/Xsession.d/disableblank.sh
```

3. Add these three lines.

```
xset s off          # don't activate screensaver
xset -dpms         # disable DPMS (Energy Star) features.
xset s noblank     # don't blank the video device
```

4. Change the permissions on the file.

```
pi@raspberrypi ~ $ sudo chmod +x /etc/X11/Xsession.d/disableblank.sh
```


5. Force the file to run when X starts

```
pi@raspberrypi ~ $sudo nano /etc/xdg/lxsession/LXDE-pi/autostart
```

Or /etc/xdg/lxsession/LXDE/autostart on older version of Raspbian

And add this line to the bottom;

```
/etc/X11/Xsession.d/disableblank.sh
```

Zdroje:

<http://ozzmaker.com/piscreen-driver-install-instructions-2/>

<http://ozzmaker.com/enable-console-on-piscreen/>

<http://ozzmaker.com/enable-x-windows-on-piscreen/>