

Making a Pocket Pi using a Pi Zero board.

This project combines a Raspberry Pi Zero board, a 4 port USB hub and a phone recharger into a package that measures just 92x42x25mm.

The parts needed can be bought here:

USB 4 port hub £2.38 from ebay.co.uk
5600mAh phone recharger £3.79
on / off switch £0.99 for 5 of

<http://www.ebay.co.uk/itm/390567004512>

<http://www.ebay.co.uk/itm/361403301976>

<http://www.ebay.co.uk/itm/201499604819>

Above parts are available from many different eBay vendors for around the same price.

Pi Zero board £4.20+ P&P

Pimoroni.com or thepihut.com

Print your own 3D printed PocketPi case

<http://www.securipi.co.uk/pocketpi.stl>

follow @securipi on Twitter for links to updates and to purchase case via eBay £9.99 soon.
(case design is a Creative Commons Attribution, Non-commercial, ShareAlike 3.0 license)



You'll also need:

Soldering iron & basic soldering skills.

Hot-glue gun – used to hold parts in place

Wire cutters/strippers

Black PVC electrical tape – used to insulate battery and contacts

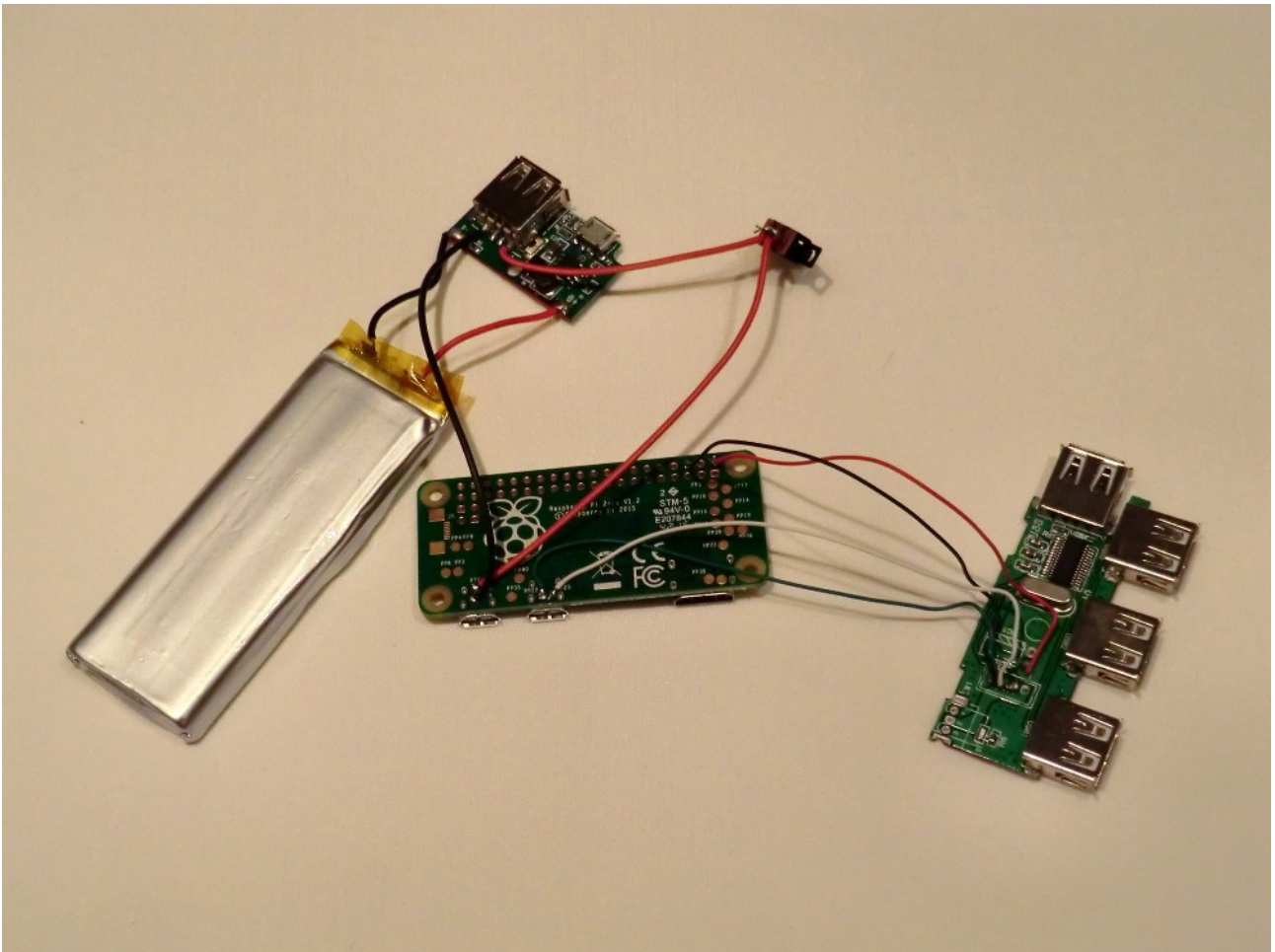
Things to do before you begin.

Before you start pulling the battery & USB hub apart, you need to setup the Raspberry Pi Zero with an SD card containing the latest Raspbian build and a USB WiFi adapter, which needs configuring to use your home router, or your Android phone in WiFi Hotspot mode.

You can learn how to configure the Pi's USB WiFi adapter here:

<https://www.raspberrypi.org/documentation/configuration/wireless/wireless-cli.md>

This is how the parts fit together:



There's a higher resolution photo of this image at <http://www.tpr2.co.uk/pocketpi3.jpg>

Before we glue the parts inside the 3D printed case we need to setup the WiFi dongle, and configure it so the Pi sends an email to the phone telling it what IP address it has been allocated by the Android phone Hotspot. At this point I'm assuming your Pi now has a working WiFi connection.

We can then use ConnectBot SSH software on the Android phone (free in Play store) to log into the Pi.

With the Pi still loose on your bench, attach a USB keyboard and HDMI monitor while we make some changes. Hopefully you've already run `raspi-config` and enabled SSH mode and expanded the Micro SD card to use all the available space.

Setting up a spare Gmail account, just for your Pi to use.

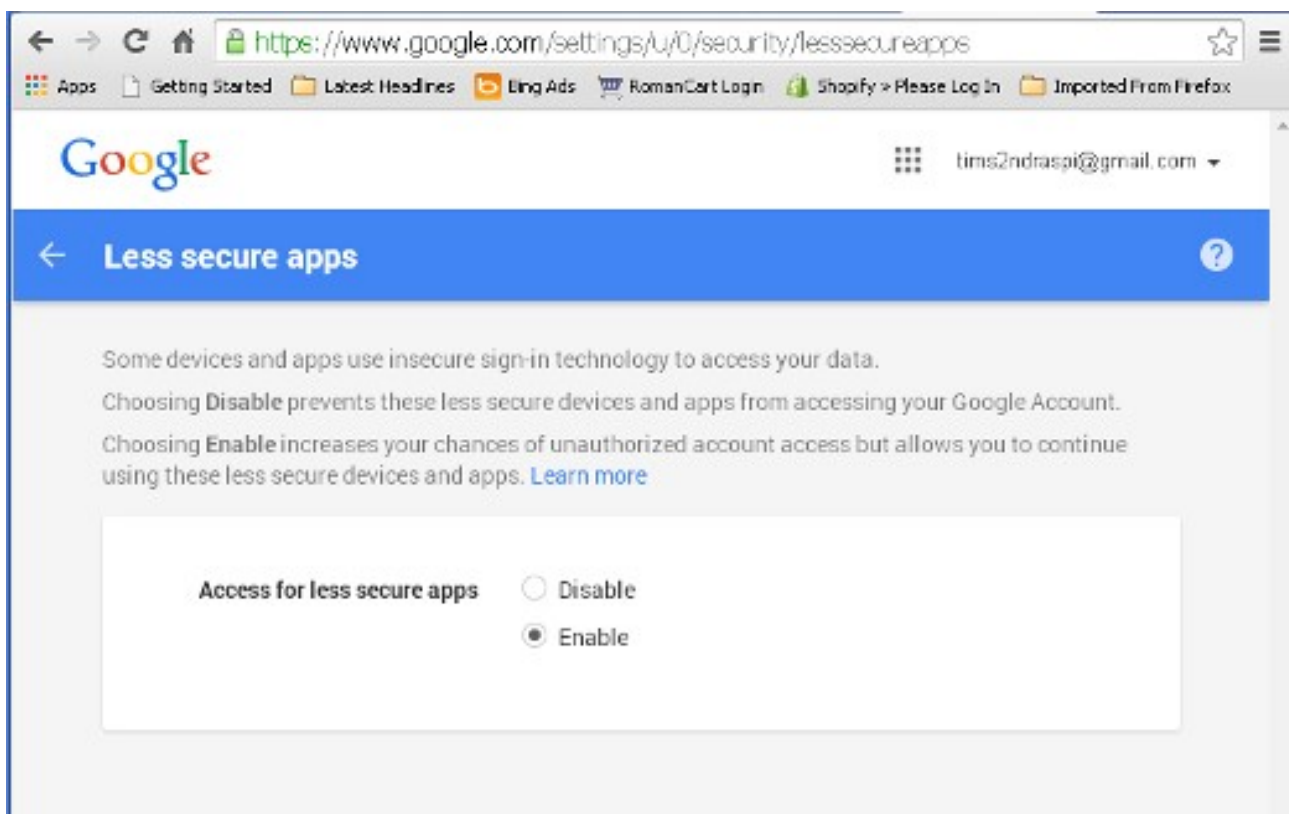
You need to create the new Gmail account in the web browser on your PC or Mac @

<https://accounts.google.com/SignUp?service=mail>

and note down the login & password for later. Don't use the # symbol in your password as it causes the Pi problems. Lower & upper-case letters & number combinations are always fine though.

Next, you need to set the new Gmail account to Enable “less secure apps”. While logged in go to:

<https://www.google.com/settings/u/0/security/lesssecureapps>



Now we have a working email account, just for the Pi to use when sending emails. Emails from the Pi can now be sent to any other existing email address on your phone or PC.

Configure the Pi to use the new Gmail account

This is how we setup the email account on the Pi.

```
sudo apt-get install ssmtp mailutils mpack
```

Setup default settings for SSMTP.

```
sudo nano /etc/ssmtp/ssmtp.conf
```

```
AuthUser=your-pi-gmail-account@gmail.com
AuthPass=your-user-password
FromLineOverride=YES
mailhub=smtp.gmail.com:587
UseSTARTTLS=YES
```

save file & exit.

Send a test email with this command, substituting the email address below with your own.

```
echo "testing 1 2 3" | mail -s "Subject" you@yourdomain.co.uk
```

Assuming that worked okay, and you received the email on your phone.

Script to email the IP address to our phone each time we turn on the Pi.

```
nano ipemail.sh
```

enter the following, changing the email address to your phones:

```
#!/bin/sh
sleep 10
ip=`ifconfig | grep 192.168 -B 1`
echo $ip | mail -s "PocketPi IP Address" your-name@your-phone.com
```

save and exit.

make it executable with

```
chmod a+x ipemail.sh
```

test it with

```
./ipemail.sh
```

hopefully you got an email that looks like this:

eth0 Link encap:Ethernet HWaddr 00:00:00:00:00:4e inet addr:192.168.1.33 Bcast:192.168.1.255 Mask:255.255.255.0
and 192.168.1.33 is the ip address of my Pi

Make the script run each time the Pi powers up.

We now just need to make the ipemail.sh script run each time the Pi powers up, (and this is why the script has a 10 second delay in it).

```
sudo nano /etc/rc.local
```

scroll down to just above the last line which reads exit 0 and type in

```
sudo /home/pi/ipemail.sh &
```

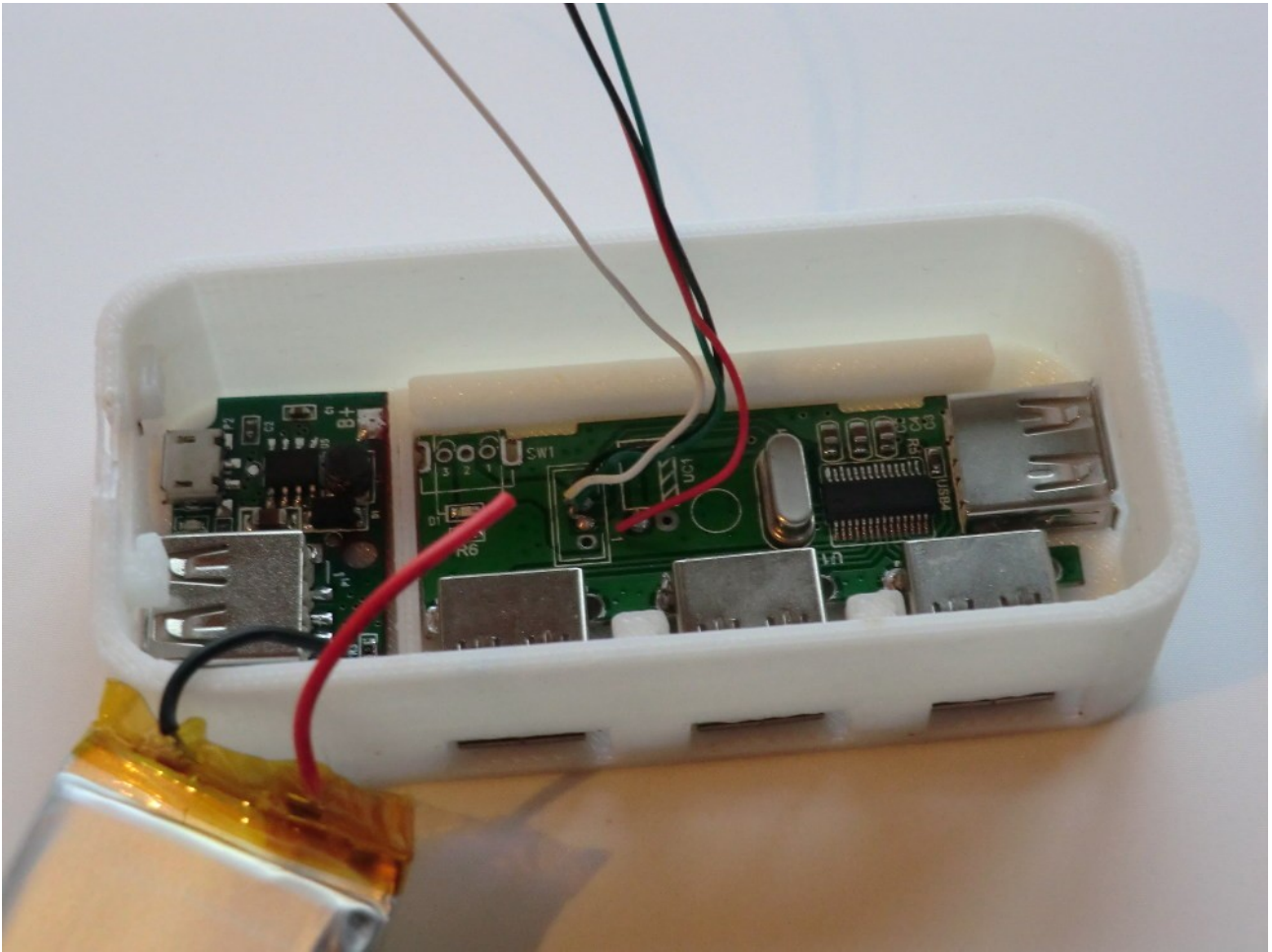
save and exit with Ctrl-O and Ctrl-X

now reboot your Pi with

```
sudo reboot
```

and you should get an email on your phone with the IP address. This should work whether you're using a WiFi dongle or a wired USB-ethernet adapter with your Pi Zero.

Fitting the boards into the 3D printed case



As you can see in the photo, there's just enough room for the USB hub and power supply board in the bottom of the case – we held the boards in place with four blobs of hot-glue on each. (a good alignment tip is to attach a micro usb cable and usb cable while applying hot-glue)

We then lined the top & bottom of the battery with black electrical insulation tape, and taped around the black and red terminals on the battery to prevent the possibility of an electrical short.

Then we fitted the power switch into the 3D printed slots and held it in place with another small blob of hot-glue.

Finally, we sat the Raspberry Pi Zero board on top & clipped the lid on. There is just enough room inside for all the parts.

The unit will act as a normal charger for your phone, when the Pi is turned off, if you attach a regular USB lead to the power board. You can charge the battery when the Pi is turned off by attaching a micro USB lead to the power board. The Pi should only be powered by the battery.

Be careful not to confuse the USB ports on the hub with the USB port on the power board.

If you want to alter the design of the PocketPi case slightly – maybe you want to add a HDMI port or a hole for a GPIO header - you can do that here:

<https://tinkercad.com/things/eZgIrStOwwx>

Connecting to the Pi using ConnectBot on an Android phone.

So you've now successfully completed the build of the PocketPi, have an Android phone running in Hotspot mode, and have configured the Pi to connect to it when it boots up. So what next?

Download ConnectBot from the Google Android Play store for free, and use it to SSH into the Raspberry Pi command line.

Start the ConnectBot app and you'll be asked for [username@hostname:port](#)

So lets say the email sent from your Pi to your phone said:

```
wlan0 Link encap:Ethernet HWaddr 00:00:00:00:00:4e inet addr:192.168.1.33 Bcast:192.168.1.255
Mask:255.255.255.0
```

Then you would enter [pi@192.168.1.33](#)

and then you should be asked for your password, which by default would be raspberry.

If you logged in correctly, you can type

```
ls
```

to see a list of all the files in your Home directory on the Pi.



If you need a bit more screen space, then you could also log into the Pi from an Android tablet (Connectbot again on Android) or laptop (using Putty), still using the mobile phone as the Hotspot.

Use VNC to remotely access the Pi desktop environment.

If you normally interact with your Pi using the desktop, then you might prefer to run a VNC server on the Pi, and access the Pi's desktop by running a VNC client on your tablet or PC. There's a full guide to this at the Raspberry Pi website <https://www.raspberrypi.org/documentation/remote-access/vnc/>

Install VNC server on your Pi with

```
sudo apt-get install tightvncserver
```

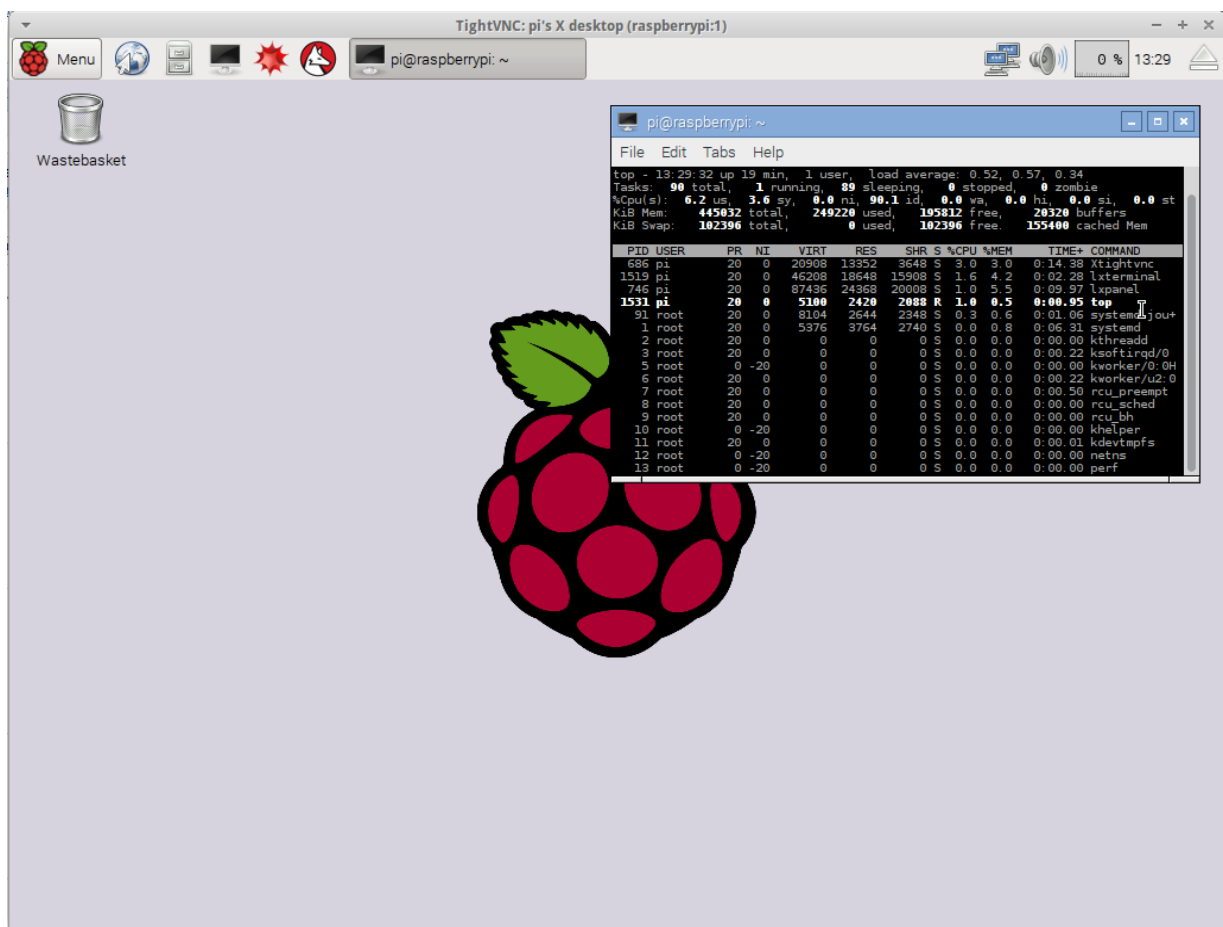
Then each time you want to launch VNC Server on your Pi

```
tightvncserver  
vncserver :1 -geometry 1920x1080 -depth 24
```

If you want to kill an instance of VNC Server on your Pi, do this:

```
tightvncserver -kill :1
```

Then on your PC or tablet, launch a VNC client (<http://www.tightvnc.com/download.php>) and give it the IP address of the Pi and screen number, like 192.168.1.33:1 and the password you setup when you launched tightvncserver on the Pi.



VNC Client on Windows

Here's the TightVNC client running on a £129 Toshiba Click Mini Windows 8 PC with 1920x1080 display.

http://www.ebay.co.uk/itm/351577788623?_trksid=p2057872.m2749.l2649&ssPageName=STRK%3AMEBIDX%3AIT



Other things you can do with the PocketPi.

If you have two USB WiFi adapters and one is capable of running in monitor mode, you can run Aircrack from your pocket using your phone to control the Pi. We successfully installed & tested Aircrack on Jessie Raspbian using this script:

```
#!/bin/bash
wget http://download.aircrack-ng.org/aircrack-ng-1.2-rc1.tar.gz
tar -zxvf aircrack-ng-1.2-rc1.tar.gz
cd aircrack-ng-1.2-rc1
sudo apt-get -y install libnl1
sudo apt-get -y install libnl-dev
sudo apt-get -y install libnl-3-dev
sudo apt-get -y install libnl-genl-3-dev
sudo apt-get -y install libssl-dev
sudo make
sudo make install
```

You could also run a pocket server; make the Pi act as an WiFi access point; make a portable TOR gateway; run a portable dropbox device...