ZERO360: PART 2 CONNECTING, VIA OTG, A CLUSTER OF RASPBERRY PI ZEROS TO A PI 3

Posted In: Blogroll

WELCOME TO PART 2 OF THIS LITTLE ZERO 360 ADVENTURE!

In this part I will only really be looking at what I did to setup the cluster of Pi Zeros to work with the Pi 3. I won't be looking at anything like load balance or anything like that. This is purely the hardware and networking required for the next step in building the Zero360.

If you're looking into a similar cluster setup (4 Pi Zeros connected to a Pi3) these notes should be for you.

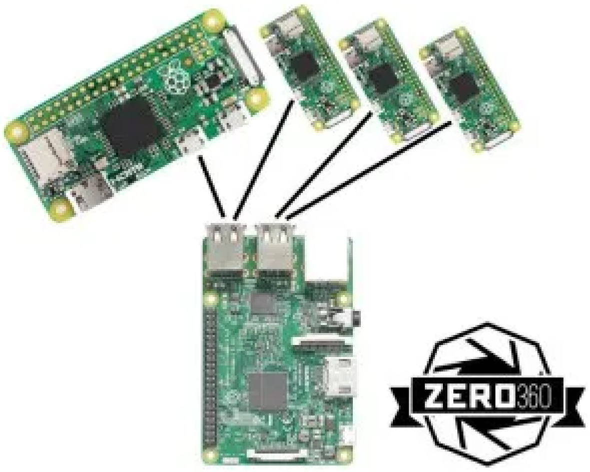
I am no expert on networking or linux systems. It might be possible there are easier ways to do what I have done, so if you know please leave a comment and we can keep notes and improve the setup.

Finally, this is a big part of the setup so this post will get rather large. I have listed the basic steps and then linked them to sections further down the post. That way, if you are using this for reference, you don't need to read everything again!

Let's Start!

WHAT IS GOING TO BE ACHIEVED

I want to set up four Pi Zeros to connect via OTG to one Pi 3. That Pi 3 should power all the Zeros using its own power supply (which could either be battery or mains). The Pi 3 should be able to access all the Pi Zeros via SSH to allow the transfer of data or commands. It should look like this.



CONSIDERING THE CABLES

I can't stress the importance, if you are considering cloning this setup, of getting the

NEXT RASPBERRY JAM

NEW CODING EVENING

BLOG

right cables and also of getting a 2.5amp power supply (the official one is best). If anything is substandard, this setup won't work.

I have also had mixed results using Powered USB hubs so I feel it's simply better to stick to the Pi 3 as a host.

PIZERO SETUP

To start, I flashed the latest Jessie Lite image to the SD card. I would avoid using the NOOBS image as you need a monitor, keyboard and mouse for the initial setup. (If you don't know how to setup an SD card, you can find out how to here: Installing Operating System Images)

Each step is listed here:

- OTG configuration with 'g_ether.host_addr' and 'g_ether.dev_addr' with unique MAC addresses
- Change the Hostname to a unique name
- Define a unique static IP

That's if for the Pi Zero side. The fun starts on the Pi3 side!

PI3 HOST SETUP

Again, I used the Jessie Lite image.

- Change the Hostname (Just like the Pi Zero but I used the name 'Batman')

- Setup UDEV rules to change the network interface names matching to the Pi Zero’s MAC addresses
- Configure the network interfaces with static IP to match the Pi Zero’s subnets

At this point I could configure some bridge connections to allow internet access but I found the connection times to be rather slow. For the Zero360 all I need is direct connection to each Pi Zero from the host Pi 3.

TA-DA CLUSTER IS BORN!

If you follow those steps and take into account my notes, I hope you would have a working cluster of Pi Zeros connected to a Pi 3. As for the Zero360, since I have 8 Pi Zeros, I just repeated the same process on the other Pi Zeros and another Pi 3 host (which I called ‘robin’… yes my 2 host Pi 3’s are called Batman and Robin!

Special thanks to Mike for proof reading my post

Note:Below this line are the notes made from the steps above. The steps themselves are links to these sections.

PI ZERO: OTG CONFIGURATION

OTG means ‘On The Go’ connection for devices, normally portable devices with micro USB. This connection should be standard on all portable devices similar to standard USB. (The exception is some phones which do not have drivers for some devices. For example: My Samsung S4 wont register wifi or Ethernet devices due to lack of drivers.)

In the case of the Pi Zero, OTG means you can connect your Zero via USB>Micro USB cable to a computer and the computer would register the Zero as a USB device like a USB Stick or an Ethernet adapter. It is even possible to setup the Zero as a Web camera (assuming the camera works with this OTG setting).

Andrew Mulholland has written a great tutorial on OTG and the different modes you can set up. He has also covered how to quickly setup OTG without a keyboard and mouse. Definitely worth checking out : Click Here

After the image is written, I reinsert my SD card in to my Mac/computer and look for the normal boot partition.

Here I want to edit two files: cmdline.txt and config.txt to setup OTG.

> config.txt should have the following appended to the bottom of the file in a new line: ‘dtoverlay=dwc2’
>
> cmdline.txt should have ‘modules-load=dwc2,g_ether g_ether.host_addr=00:22:82:ff:ff:01

g_ether.dev_addr=00:22:82:ff:ff:11' after the 'rootwait'. Do not put these on a new line! Everything has to be on the same line.

If you followed Andrew's guide already, you might have already noticed I have put in the two parameters 'g_ether.host_addr' and 'g_ether.dev_addr'. These are important for our setup and each SD card for each Pi Zero should have their own set of addresses. Make sure you change the last number of both parameters… :02 and :12 – :03 and :13 etc…

Every time you boot an OTG Ethernet Pi Zero, it is creating two new network interfaces. One on the host it's connecting to (in this case it would the Pi 3) and one on the Pi Zero itself. Every network interface needs a unique MAC address so that the device is identifiable. When connecting to a router, for example, it will check your MAC address and then assign you an IP. If you disconnect and then reconnect the router remembers you and gives you the same IP. But every time the Pi Zero boots these MAC addresses are randomly generated.

'g_ether.dev_addr' will assign the MAC address to a network interface on the Pi Zero and 'g_ether.host_addr' lets me assign the MAC address for the network interface created on the host Pi 3.

These MAC addresses will be needed later in the setup.

NEXT RASPBERRY JAM

NEW CODING EVENING

BLOG

PI ZERO: CHANGE THE HOSTNAME

If I connected to my Pi Zero (with OTG configured) using 'ssh pi@raspberrypi.local' on my Mac, there is a discovery service called Bonjour that will find the connected device using its hostname and direct my request to connect to that Pi Zero. In this case, the '.local' part just indicates that the Pi Zero should be directly connected and since I don't have a DNS server to direct the hostnames to IP addresses the '.local' is really what makes this work.

Debian also has a similar service called AVAHI. You can find out more about it here: wikipedia.

To change the hostname I only need to edit 2 files '/etc/hosts' and '/etc/hostnames' and swap out 'raspberrypi' for (in my case) 'pan01' Reboot and the new hostname should be in place.

Each Pi Zero needs its own unique hostname. In my case, the other Pi Zeros would be called 'pan02', 'pan03' etc... This is to prevent any confusion when identifying each Pi Zero

PI ZERO: DEFINE A UNIQUE STATIC IP

Edit the '/etc/dhcpcd.conf' file and add the following settings to the bottom of the file:

```
interface usb0
static ip_address=10.0.11.2
```

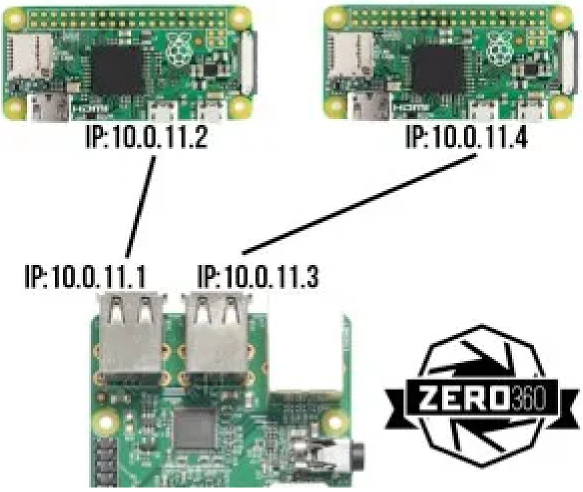
Each Pi Zero should have a unique IP . In this case, 10.0.11.2 is the IP for pan01 and pan02 should have 10.0.12.2, pan03 should have 10.0.13.2 etc…

Why am I only changing the third number group in the IP?

When all the OTG configured Pi Zeros are connected to the host Pi 3, the IP addresses are broken down to a subnet, for example 10.0.11.0 (which is the subnet for pan01), and are added to a route table of IP addresses matching the network interface that it is connected to. It means when I ping 10.0.11.1 or .2 the host Pi will check the route table and say that this ping should go to the network interface that belongs to the subnet 10.0.11.0

The issue comes into play if you have all the Pi Zeros in the same subnet (meaning similar range of IP addresses).

Let’s say we have the following setup:



The Pi Zeros have similar IP’s so they are apart of the same subnet. If I ping pan01 I will get a response. But if I ping pan02 I will not. Whats happening is the host has assigned the subnet 10.0.11.0 to both interfaces so when I ping the IP for pan02 it will check the first interface as it has that subnet and is first in the route table. But since pan02 is not connected via that interface the ping will not get a response and it will stop there. It won't ping through the other interfaces.

Therefore, each Pi Zero connecting to the host Pi 3 needs its own unique subnet.

PI 3 HOST: SETUP UDEV RULES

Each time a Pi Zero with OTG is connected to the Pi 3 host, the host creates a network interface called USB0. If you have more than one Pi Zero connected, network interfaces USB1, USB2 etc.. are also created.

Unfortunately, these interface names are reused when a device is disconnected and another similar device is connected.

For example, you could assume I would assign a static IP for each interface. If I only plugged in the Pi Zero called pan01 it would have the network interface USB0 on the host Pi 3, I would make USB0 have the IP:10.0.11.1. If I ping pan01 I would get a response! If I unpluged pan01 and then put pan02 in… USB0 interface is then re-used and that's where the setup falls over.

Because pan02's IP does not match the same IP range I assigned USB0 you won't be able to ping pan02.

Unless you plug in each Pi Zero in order one by one, you would never know which Pi Zero would be assigned which network interface name or IP.

To fix this, I am using UDEV rules (see wikipedia) to change the network interface names for the connecting Pi Zeros, based on the MAC address I assigned during the OTG configuration.

Remember, I added 'g_ether.host_addr=00:22:82:ff:ff:01' to the cmdline.txt in the OTG configuration.

I created a new file '/etc/udev/rules.d/90-panocluster.rules' and inserted the following

```
SUBSYSTEM=="net", ATTR{address}=="00:22:
SUBSYSTEM=="net", ATTR{address}=="00:22:
SUBSYSTEM=="net", ATTR{address}=="00:22:
SUBSYSTEM=="net", ATTR{address}=="00:22:
```

When I connect pan01 to the host Pi 3, it will tell the host that pan01 is an OTG ethernet device and that the host Pi should create a new interface with the MAC address '00:22:82:ff:ff:01'. The host Pi checks the UDEV rules and see's that the MAC address matches one of its rules and it should create the interface but name it 'ethpi1' instead of 'USB0'

NEXT RASPBERRY JAM

NEW CODING EVENING

BLOG

PI 3 HOST: CONFIGURE THE NETWORK INTERFACES WITH STATIC IP

Now that the UDEV rules are assigning a network interface name based on the Pi Zero connected, I can set up static IPs for each network interface to match.

Edit the '/etc/dhcpcd.conf' and add the follow settings to the bottom of the file.

```
interface ethpi1
static ip_address=10.0.11.1/24

interface ethpi2
static ip_address=10.0.12.1/24

interface ethpi3
static ip_address=10.0.13.1/24

interface ethpi4
static ip_address=10.0.14.1/24
```

At the end of each IP I added the '/24' to set the subnet for each network interface. The subnet for ethpi1 would then be 10.0.11.0. This means each Pi Zero should be reachable.

SHARE THIS:

RELATED

Introducing Zero360
28th June 2016
In "Blogroll"

Zero360: Part 1 Building the Zero360
30th June 2016
In "Blogroll"

Buyers Guide: Raspberry Pi Zero In Germany
25th February 2016
In "Blogroll"

Posted on July 6th, 2016 by

James Mitchell

NEXT RASPBERRY JAM

NEW CODING EVENING

BLOG

<< PREVIOUS

NEXT >>

23 COMMENTS

aBUGSworstnightmare
July 7, 2016 at 11:32AM

Hey there,

thanks for the info on how to configure the USB ethernet gadget. Missed the unique MAC address config so far (not mentioned in Andrew’s blog); will make use of the gadget quite simple now (that it has unique MAC address).

Thanks!

Will keep watching this nice project…

REPLY

Taki Uddin
September 9, 2016 at 1:22PM

so hyped for part 3..:D

REPLY

Graham Toal
September 25, 2016 at 4:45AM