Parallella E16G301: a starter kit under Linux

***

1°) Introduction

Parallella is a fascinating coding platform.
Its potential is huge, some of its proterties are really unique.
And you can use it in so many ways it may be difficult to know where to begin with.

My purpose is simple ; as a hobbyist and loving any minimalist though efficient way of coding, this tutorial is intended as a gift for beginners that want to start coding with minimal time, money, hardware, pain - but with average Linux experience at least.

... We need to boot the beast first ;)
Here is the **text-based** tutorial for an **SSH access**, requiring the **minimal hardware** stuff :)
It's updated from my previous ESDK 2015.1 tutorial,
this one is tested and intended for the ESDK 2016.3.1 Zynq 7020 image (May 2016).
It's even simpler, the image has improved.

2°) In practice

<!> We need a computer of course ; with an Ethernet plug ; running under any Linux flavour.
I'll talk about what I know: my Fedora Core 23 ; it should be understandable for any Ubuntu user for instance. I'm afraid you need at least some basic experience of Linux to go on.
There are of course many Windows tutorials on the forum or some popular video platform... that will do as well. But my goal is the Parallella programming under Linux...

a) Hardware

So. What to buy:
- 100 € ; a 16-core Parallella, of course.
- 15 € ; a fan
- 15 € ; a 5V power
- 9 € ; a micro SD card reader for your computer
- 6 € ; a 8 GB micro SD card
- 5 € ; a standard Ethernet cable

=> 150 € for French people like me, approximately 150 $ for American people I guess.
Tax create some weird situations :/ anyway pay your tax.

About the fan

The early Kickstarter models absolutely need it, or you'll burn your Parallella.
Avoid my naive mistake: cheap USB fans may be deadly noisy.

Picture 1 under
https://www.parallella.org/quick-start/

«Before you begin the tutorial, please follow these instructions to properly install the heatsink for the board. Do NOT attempt to use the board without the heatsink!»

Credits: Suzanne MATTHEWS. Her tutorial on the forum helped me a lot.
Basically if you have an early Kickstarter kit you have to stick the heatsink to the central black IC ZYNQ. It takes 30 s, however do NOT put your fingers anywhere on the board, remember that the static electricity may damage or kill some IC (Integrated Circuit).

About the 5V power
This is really, really essential for stability, moreover a time-consuming task to do first.
Do not rely on your USB computer or any cheap power.
Refer to the Parallella manual (REV 14.09.09): you need a « high quality 2000mA rated 5V DC power supply with 5.5mm OD / 2.1mm ID center positive polarity plug.»
This hardware is exotic and hard to find ; I got mine from this provider: VHBW.
www.vhbw.fr ; SKU OS4-C-61 ; only 10 € ; fully functional, zero problem after 10 days.
Other examples for your country are on the forum, I suggest searching « polarity plug» or «5.5mm» on the forum:
https://parallella.org/forums/
https://parallella.org/forums/viewtopic.php?f=12&t=1319&p=8222&hilit=polarity+plug#p8222

About the micro SD card reader
This format is not common either. Make sure your multi card reader can read micro SD cards.
It is about 10 millemeters wide, near the µ-HDMI connector ; please refer to above Picture 1 link.

About the 8GB micro SD card
This is so cheap you can afford 16 or 32GB if you like.
As we use a «headless» configuration, even 4 GB would be enough - just enough indeed.
«Headless» is lightweight on the contrary to the more user-friendly «HDMI» config, that required a 16 GB micro SD card last April (2016):
- no X client running on the host Linux
- no dedicated ZYNQ FPGA resources for the HDMI
Besides, this is more up to date, dedicated to performance, not the same audience as the «HDMI» also called «Desktop» version.
With our «headless» configuration we are definitely NOT blind since we'll use SSH from our favourite Linux computer.

About the standard Ethernet cable
Some people might say you need a crossover cable.
It is true... if you work with really old hardware.
But for a few years now, any basic Ethernet card can safely manage both connections: crossover or standard Ethernet cable. A computer technician told me that.
So I took a crossover cable, tested it... fine.
I pinged the connection... 160 ms not a very good latency but enough for our purpose.
So I took a standard Ethernet cable, tested it... fine.
I pinged the connection... 160 ms again. No difference in any kind.
IMO you can buy what you want.
CAT5/5E/6/7 are speed norms ; CAT5E tends to replace CAT5, better transfer rates.
CAT6 and 7 families are even better.
But for transferring a few C files while debugging, CAT5 should be enough ; I personally use a CAT6 Ethernet cable.

b) Downloads
I repeat: this is my Fedora point of view, the syntax might differ slightly if you have another Linux.

The SD card image
Follow the «quick steps» here:
https://www.parallella.org/quick-start/
to burn an SD image, this is all good.

Find a «headless» image for Kickstarter... so for Xilinx 7020.
I personally used ubuntu-15.04-esdk-2016.3-headless-z7020.img.gz
You should find it under this stable URL:
http://www.parallella.org/create-sdcard/

Plug your SD card reader to the computer.
You can use «df -h» or Gigolo or gparted to see what partitions are used.
Mine is /run/media/<myself>/sdc, with 2 partitions sdc1/sdc2 because previously used.
<!> Take care ! Better remove any USB key before... and make sure you don't format any hard disk lol For a 8 Gb SD card you should see something around 7.5 G...

Here are a few hints during installation:
#flash SD CARD 2016/05/13
  sudo su...
  gunzip -d ubuntu-15.04-esdk-2016.3-headless-z7020.img.gz
  df -h
#unmount the SD card ; better done through the file manager :)
#maybe umount /run/media/<you>/<your_card_partition(s) under a terminal
#burn the image to the SD card ; it will result in a fat16 partition with 'boot' flag + a ext4 partition with all the Parallella coding material inside
  dd bs=4M if=ubuntu-15.04-esdk-2016.3-headless-z7020.img of=/dev/sdc
#as for the 2015.1 image it took about 6 minutes to complete ; it's a cheap class 10 µSD card
931+1 enregistrements lus
931+1 enregistrements écrits
3905945600 octets (3,9 GB) copiés, 359,75 s, 10,9 MB/s
  sync
  exit

Extra downloads
Make sure you have SSL.
And download putty 0.67: we'll use its utility psftp for file transfers.

c) SSL, putty... Parallella boots !
It's probably not the best way (some people use DHCP for example), but it works for me.
Here is my context: I have no internet box, no switch... no nothing at home. Only Rossinante.
So I guarantee nothing at all - just be brave enough, or resign now.
You'll certainly lose the internet connection while you «talk» to Parallella.
Make a backup of your internet connection ; create a dedicated connection for Parallella... we'll see how.
And when you're done with Parallella, unplug/plug the ethernet cable, select the internet connection again... that's all.

<!> There’s a tiny difference between the 2015.1/2016.3.1 images: rootfs -> root
<!> And if you’re like me - 2nd image test, better flush ~/.ssh folder (Ctrl H to see the hidden folders ; ‘~’ stands for your Linux ‘home’)

We edit the SD image, /<you & your SD card>/root/etc/network/interfaces.d/eth0 :
#save it -- auto eth0
#save it -- iface eth0 inet dhcp
  allow-hotplug eth0
  iface eth0 inet static
  address 10.0.2.15
  netmask 255.255.255.0

and we create a specifig «network connection» through the (Fedora) menu, Preferences / Network connection, IP 4 address & netmask:
  10.0.2.1
  255.255.255.0
There’s another file to modify, on your computer this time: /etc/hosts
Add this line:
  10.0.2.15 parallella
FYI hostname is ‘parallella’

Power the fan, plug the Ethernet cable, plug the SD card, last plug the 5V/2A power.
Open a terminal ; the initial boot may be slower because of the SSH generation, other boots take about 15 s to answer the ping command :)
  ping 10.0.2.15
For ssh, remember we use the ‘parallella’ user and password, and this has no root privilege for security purpose.
You can automate many things if you like - I do ! Linux is great.
  ssh 10.0.2.15 -l parallella
  ssh parallella@10.0.2.15

You can now browse the Linux distro run by the ARM processor, give commands with your regular keyboard and your monitor to control what is done :)
For example, inside a SSH session you can run the hello example and you look at the results on your own screen. Your good old monitor.

What’s next ?
Well, you’re the oil king now ! in French «le roi du pétrole !» ;)
You are powerful:
- you edit the batch or even your source files directly -- nano ./build.sh
- or you work with your standard editor from your computer,
automatically transfer your source files to a folder under the Parallella :
  psftp parallella@10.0.2.15 -pw parallella -b psftp.txt
(psftp --help explains everything, the psftp.txt is an optional batch of commands to automate what you want ; you can put any Putty command inside ; yes... psftp and Putty are the same software)
There is a lot of open information on almost anything on this exciting Parallella platform.
It’s Andreas Olofsson’s will:
«For up to date board information, please refer to:
http://github.com/parallella/parallella-hw
All support for this board is provided via community support at
http://forums.parallella.org»

OK. You learned the hardware/software basics for booting your Parallella.
I had promised a tutorial #2 where we'd begin with a more advanced «Hello world» example.
It is delayed because the ESDK did change a lot and I myself have to learn more.
Anyway, my little project Paralle2 (a tiny Eternity II solver) works already, with a
commented source code that is/was my first basic canvas for SPMD programming.
Single Program Multiple Data is efficient and simple for multi-core programming,
I do believe.

Don Quichotte, 2016/05/14