BEAGLEBOARD MANUAL GUIDE (for dummies)

Ricardo Cañuelo Navarro
(dummy)

License

Contents

1 Introduction

This is a basic illustrated guide about how to connect successfully BeagleBoard to a PC. This guide is aimed to people with no experience in this gadgets and those who need making the board function quickly, at the first attempt and with no headaches.

For this, a pack of components available at the board's website will be used. At this way, the same kind of components will be connected, as there are serveral versions and reviews of the board that include subtle hardware changes -specially, changes related with connectivity-.

I also get a Debian GNU/Linux image ready that can be downloaded and copied onto a memory card. So, do not worry any more about placing the memory card into the board and turning it on to start experimenting with the board.

Simplicity is the final achievement. After one has played a vast quantity of notes and more notes, it is simplicity that emerges as the crowning reward of art

Frédéric Chopin

2 Components

The board used on this guide is a **Rev C2 BeagleBoard**, used at the **Tea** project[1]. Board specifications and other technical details are explained on the Technical Report 1. As this is not the main objective of this guide, let's leave the technical stuff aside.

Figura 2.1: Box label

Appart from the board, a serie of components will be used to connect the board to a PC, to supply it and to add supporting-peripheral ports. Components used are included in the kit located at http://specialcomp.com/beagleboard/RevC2.htm (figura 2.2).

[1] In fact, this guide was written to be enclosed to the project documentation.

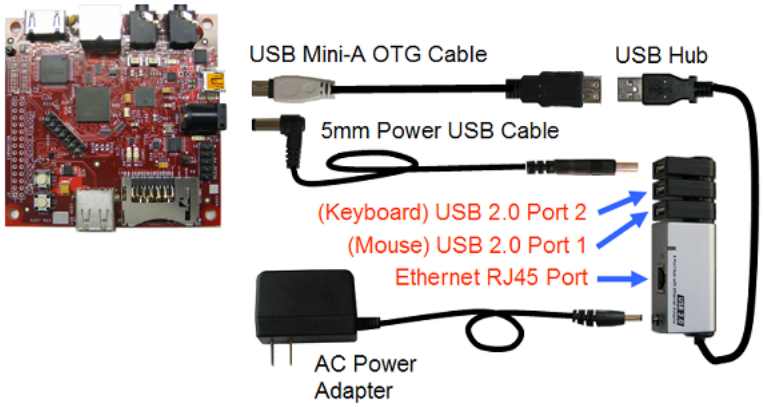

Figura 2.2: Components kit for the BeagleBoard Rev C2

In order to avoid undesirable surprises, it is recommended to get a kit, available on that website. It is well-known that this kit works.

The board will be connected to a PC via serial cable. As the board does not have a suitable interface to support it, a converter cable will be needed for the board's interface. It is available at http://search.digikey.com/scripts/DkSearch/dksus.dll?Detail?name=BBC01-ND. Now you can connect the board to a PC serial port via DE-9 null modem.

Figura 2.3: DE-9 adapter cable

Nowadays, this port is not usually included because designers think it is easier to remove this port, in order to put there 2 or 4 additional USB

ports. For this reason, you may need a USB Serial adapter cable.

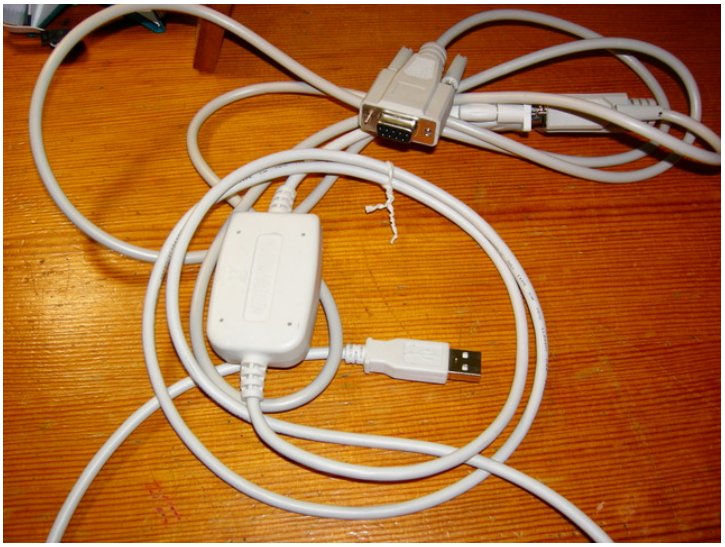

Figura 2.4: USB Serial adapter cable

A 2GB SD memory card is also needed. The bigger the memory is, the better. Actually, it is difficult to find memory cards with less capacity than 2GB. If you do not have a memory card reader in your PC, be aware that it is also necessary. These things are not included in the memory card kit.

Other things you may need are:

- An additional null modem serial cable to connect the serial converter of the board to the USB-Serial converter. In my case, I will need a female to female cable.

- An adapter to connect the power supply included in the kit because it has North American pins. An adapter like this 2.6 costs approximately €1 (picture 2.6).

- An Ethernet cable with RJ45 pins to connect the board to a modem, router or whatever.

Figura 2.5: Memory card and card reader

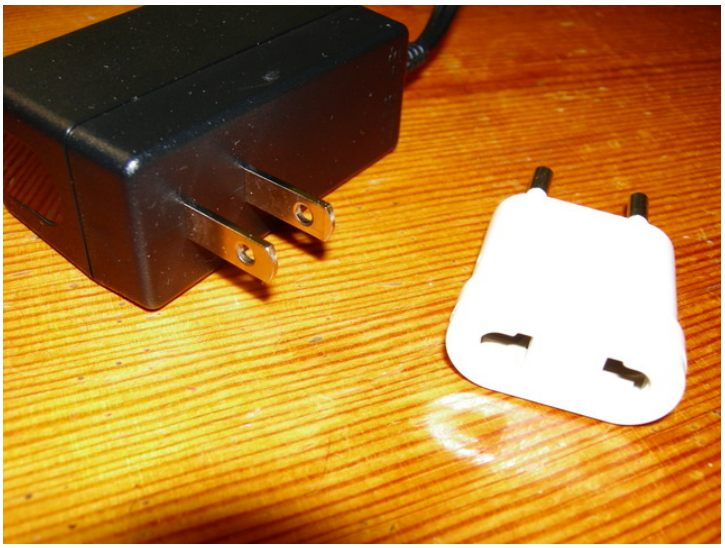

Figura 2.6: Yankee power supply and adapter

It can scarcely be denied that the supreme goal of all theory is to make the irreducible basic elements as simple and as few as possible without having to surrender the adequate representation of a single datum of experience

Albert Einstein

3 Connections

A cleared work surface near from a PC will be used. So, get it one firstly. I have been using my desk, next to my laptop (picture 3.1). This area will be full of cables soon, and everybody knows that, when many tangled cables are together in the same place, they come alive and chaos domains everything, so it is a good idea to draw up a plan for it beforehand.

(a) Not-to-work surface

(b) Adapted not-to-work surface

Figura 3.1: Work area

Here we have the box with the board inside. I'm curious and astonished. Let's open it.

Figura 3.2: Opening

At last, here is the board! The board is on my palm, so you can notice the size of the board. I have to clarify that my hands are small, this is reason of why the board seems to be bigger than what it really is.

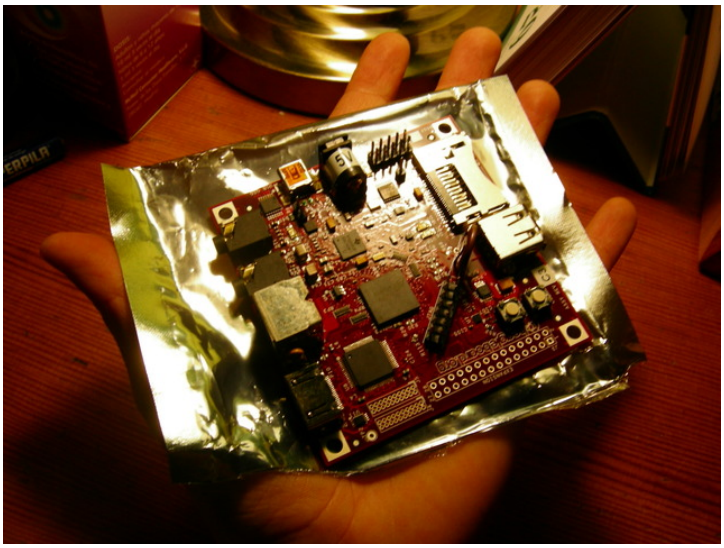

Figura 3.3: Taking the board

Handle the board with care, God knows what you have touched before. Take it along the edge or using the plastic wrapping as a tray, like in the picture.

The brand is indeed written on the board:

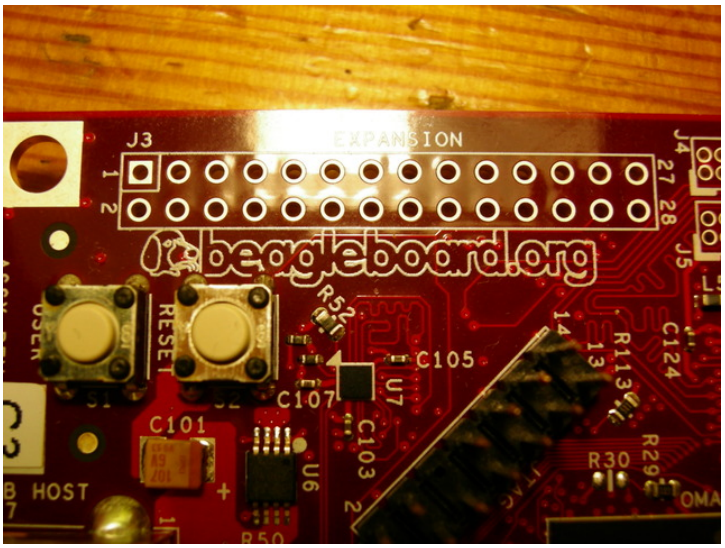

Figura 3.4: Yes, it is a BeagleBoard

Among other things, the OMAP3530 is located inside the chip marked as OMAP3:

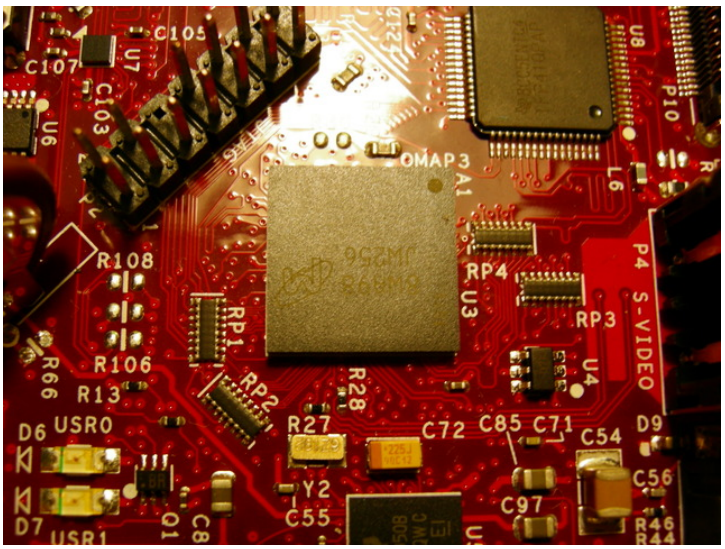

Figura 3.5: Here is the chip

Now, take the things out of the kit and mount them. Firstly, connect the DE-9 adapter to the board. Look at the picture carefully to see more

in detail where and how to connect it. Take care with the connector's possition, pay attention to the red stripe of the cable.

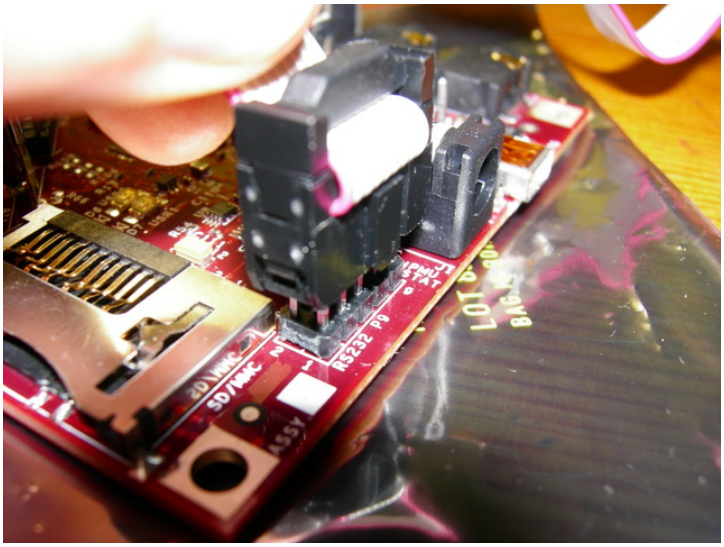

Figura 3.6: Successful connection of the adapter

Connect the hub's feeder cable to the hub. Do not mind about the posibility of getting confused and thinking that the power supply should be connected to the board, because the board's jack does not fit the power supply's slot.

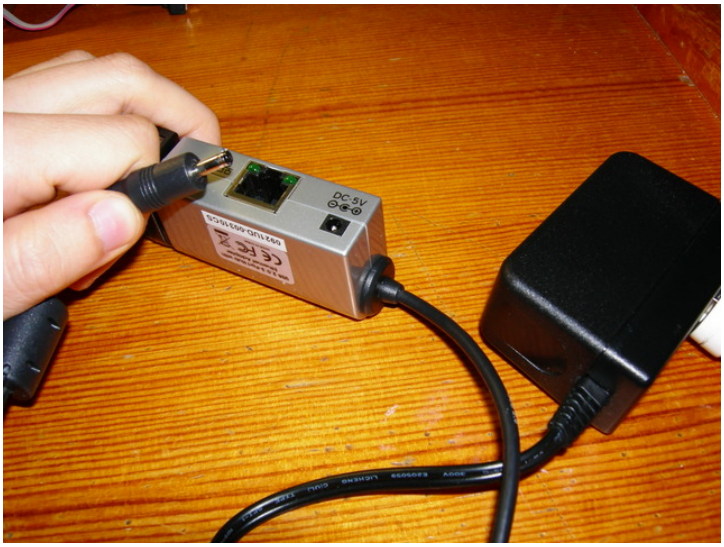

Figura 3.7: Hub's power supply and slot

Figura 3.8: Hub connected to power supply

Take the power supply cable of the board. USB A connector must be connected in one of the hub's slot, and the power supply plug to the board.

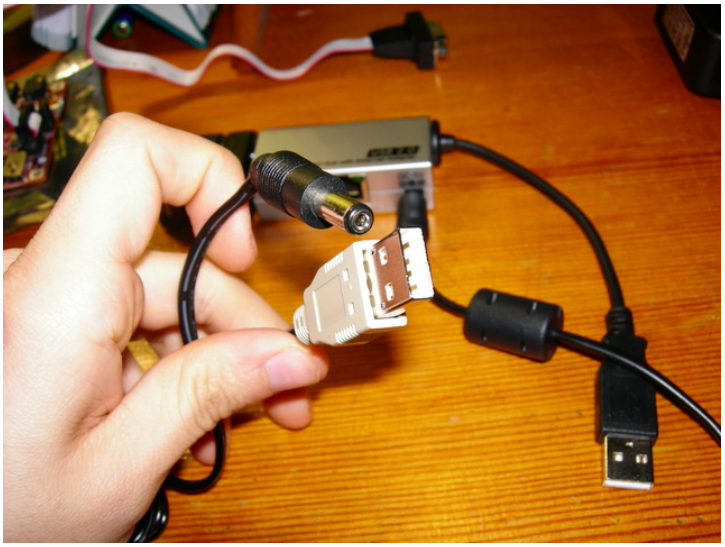

Figura 3.9: Power supply cable

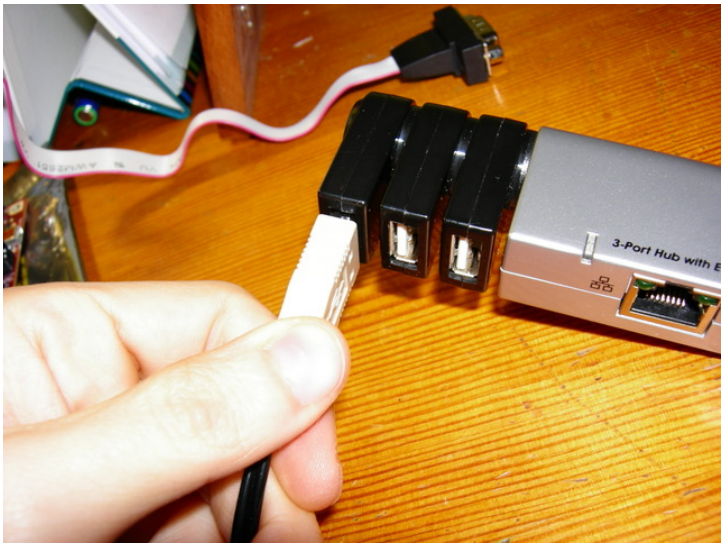

Figura 3.10: Male USB A connector with female USB A

Figura 3.11: Board power supply connection

Now you have only got the serial cable to connect. Use the USB Serial adapter. Connect both DE-9 connectors.

Figura 3.12: DE-9 connectors

Connect the other end of the USB Serial adapter to the PC.

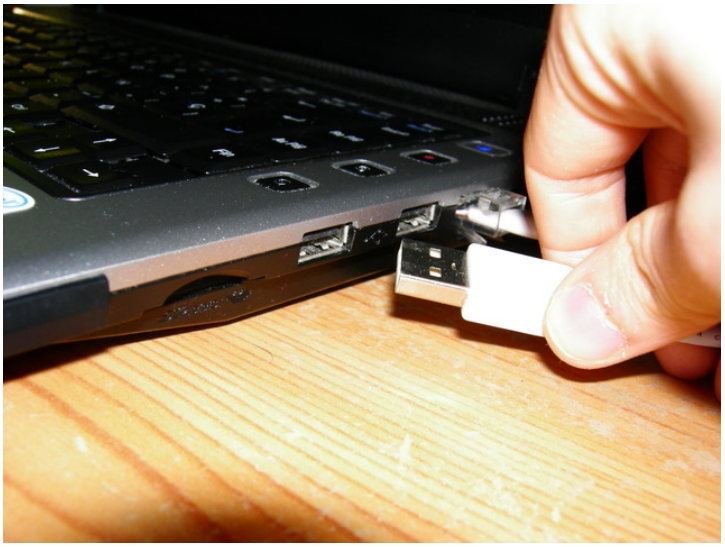

Figura 3.13: Inminent conection of the adapter to the PC

This is the result:

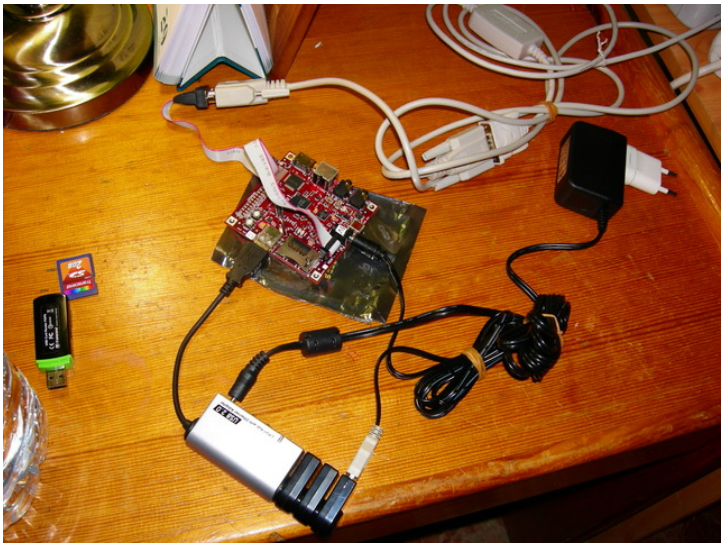

Figura 3.14: Laboratory

And is this cable spare? OK. This board has two USB ports. One of them is a regular female A connector. This port acts as an EHCI host. This means that any kind of device can be connected here to make this device be controlled by the board. The other port has a female mini-A connector (not very common). There, the board is supplied and can work as a host or as a slave. So, why do not I use it? Because, if a mini-A plug (this plug jumps a pair of pins, configuring this port as a host) is connected in this port, it should work as a host. However, when I tested it, there was no way to make it function as host. I do not know why. But it is expected to start quickly, so I recommend you not to bother about it because it can be caused by a designing effect, for example. I stopped thinking about it because it was pretty frustrating, as detailed in the picture 3.15. Despite the fact that the port detected the hub and the board was being powered, I could not make the Ethernet port work in Linux.

So, the remaining option is connecting the hub to the EHCI port, which works properly.

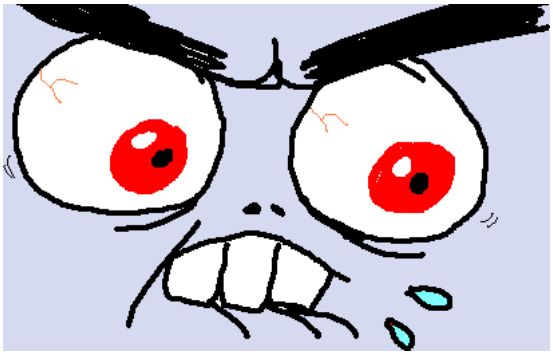

Figura 3.15: My face when I was trying to make the hub work via USB OTG port

I'll play it first and tell you what it is later
Miles Davis

4 Booting up

Everything is connected, so prepare what remains.

4.1 SD Memory Card

This memory card will be used to store the whole system (kernel + file system). For this, no modifications of the board are needed. Just place the memory card into the board and turn it on. The charger pump startup (that will be reconfigured later on) will do the rest.

Start by downloading the system zipped image. Unzip the image with:

```
$ bunzip2 sd.dd.bz2
```

As a result of this, a 2GB file will appear, and this file will be stored in the memory card.

Insert the memory card in the PC and check which device refers to it. Once the memory card has been connected, this can be seen by checking the last kernel messages:

```
$ dmesg | tail
```

For example, I have this:

```
[341483.238451] sdb: sdb1 sdb2 sdb3 < sdb5 >
[341483.342451] sd 10:0:0:0: [sdb] Attached SCSI removable disk
```

So, the memory card refers the device /dev/sdb. Then copy the system image onto the memory card with:

```
$ dd if=./sd.dd of=/dev/sdb
```

And now place it in the board.

4.2 Minicom

Minicom is the program that will be used to have access to the board serial console. For this, licenses are necessary to have acess to the PC serial device, so you must probably be logged as root.

As you did with the memory card, you can find out what kind of device must be used if the computer does not have a serial port and de USB Serial adapter must be used. Connect the adapter to a USB port and enter

```
$ dmesg | tail
```

I have this:

```
[341959.597576] usb 1-1: pl2303 converter now attached to ttyUSB0
[341959.597576] usb 1-1: New USB device found, idVendor=067b, idProduct=2303
[341959.597576] usb 1-1: New USB device strings: Mfr=1, Product=2, SerialNumber=0
[341959.597576] usb 1-1: Product: USB-Serial Controller
[341959.597576] usb 1-1: Manufacturer: Prolific Technology Inc.
```

So, the 'serial port' used for the communication would be /dev/ttyUSB0.

Then, run Minicom:

```
% minicom
```

And press Ctrl-A Z to have access to the help menu. Press O to configure Minicom and select 'Configure serial port'.

```
           +------------[Configuration]-------------+
           | Routes and files name                  |
           | File transfer protocols                |
           | Serial port configuration              |
           | Dialed modem and number                |
           | Screen and keyboard                    |
           | Save configuration as dfl              |
           | Save configuration as...               |
           | Quit                                   |
           +----------------------------------------+
```

Press 'A' to edit the device and make changes. Set the hardware flow control to none by pressing 'F' and press 'E' to scroll to another menu.

Press 'E' to select a baud rate of 115200 bps and 'Q' to restore the remaining parameters. Press Intro to return to the previous menu. The result should be a configuration like the following one:

```
  A - Serial device           : /dev/ttyUSB0
  B - Lock file location : /var/lock
  C - Callin program          :
  D - Callout program            :
  E - Bps/Parity/Bits            : 115200 8N1
```

```
  F - Hardware flow control: No
  G - Software flow control: No
```

```
Return by pressing Esc and selecting 'Quit'. Press Ctrl-A m to restart the conection, now you
     can connect the board to the electric current. Several charger pump startup messages
     should be displayed.\\

If this is the first time the board is booted, stop it by pressing any key when the u-boot
     countdown is displayed and enter:

\begin{lstlisting}[style=consola, numbers=none]
setenv bootcmd 'mmcinit; fatload mmc 0:1 0x80300000 uImage; bootm 0x80300000'
setenv bootargs 'console=ttyS2,115200n8 root=/dev/mmcblk0p2 rootwait rootfstype=ext3 ro'
```

WARNING, these orders change the bootloader configuration in order to make the board boot from the memory card. It is necessary to configure the board again if you want the board to boot from the NAND memory

These orders configure the bootloader to boot from the image located in the memory card according with several parameters. The parameter 'console' let us see the console output via serial port.

Restart the board and, if everything is OK, you should be watching how the system boots after the u-boot countdown. Congrats! You have a whole ready-to-use system. By the way, the root password is 'beagle'.