THIS IS A TUTORIAL TO HELP YOU

PORT THE

“UBUNTU”

OPERATING SYSTEM

ON THE BEAGLEBOARD xM

Kaushik Kanetkar
Porting UBUNTU on the BeagleBoard:

Following are the steps to port another operating system, Ubuntu, on the BeagleBoard Xm.

Steps:

- Make your home directory to store all the files for the Ubuntu operating system. (root files)
- Plug in your micro SD card onto your host laptop system.

- Now we need to get the armel:

  ```
  wget http://ynezz.ibawizard.net/beagleboard/natty/ubuntu-11.04-r1-minimal-armel.tar.xz
  md5sum ubuntu-11.04-r1-minimal-armel.tar.xz
  tar -xvf ubuntu-11.04-r1-minimal-armel.tar.xz
  ```

  - Instead of the above commands, you can use the files in my folder ‘Ubuntu-make’.

- Now, enter the following:

  ```
  cd ubuntu-11.04-r1-minimal-armel.tar.xz
  sudo ./setup_sdcard.sh --mmc /dev/sdb --uboot beagle_bx --rootfstype ext3
  ```
This process includes formatting your SD card, transferring boot files into the boot partition and the root filesystem onto the other partition of the memory card.
- When it is done, remove the SD card, and plug it back after 2-3 seconds.
- You will see two partitions, as listed below.

```
root@kaushik-Satellite-L745:/media
sh' /tmp/fix_zippy2.sh' -> '/tmp/QoYhbj8qFJ/disk/tools/fix_zippy2.sh'
/tmp/latest_kernel.sh' -> '/tmp/QoYhbj8qFJ/disk/tools/latest_kernel.sh'
/tmp/minimal_xfce.sh' -> '/tmp/QoYhbj8qFJ/disk/tools/minimal_xfce.sh'
/tmp/get_chrome.sh' -> '/tmp/QoYhbj8qFJ/disk/tools/get_chrome.sh'
Finished populating Boot Partition
0 / 9: Populating rootfs Partition
Be patient, this may take a few minutes
279MB 0:01:36 [2.89MB/s] [--------------------->] 100%
Finished populating rootfs Partition
9 / 9: setup_sdcard.sh script complete
root@kaushik-Satellite-L745:/home/kaushik/ubuntu-make/ubuntu-11.04-r2-minimal-armel# cd /media/
root@kaushik-Satellite-L745:/media# ls
root@kaushik-Satellite-L745:/media# cd /media/
root@kaushik-Satellite-L745:/media# ls
boot  rootfs
root@kaushik-Satellite-L745:/media#`
```

- Remove the SD card from the card holder and plug it in the Beagleboard and switch on the power.
- Connect the Serial (BeagleBoardxM) to USB(Host PC) converter and configure the minicom connection.

- Start minicom on your host machine in configuration mode. As root: `#minicom -o -s -w`
  - A menu of configuration should appear. Use the Down-arrow key to scroll down and select the Serial port setup option, and press Enter.
  - Verify that the listed serial port is the same one that is connected to the target board. If it is not, press A, and enter the correct device. This is /dev/ttyUSB0 on most Linux distributions.
· Set the Bps/Par/Bits option by pressing the letter E and using the next menu to set the appropriate values. You press the key that corresponds to the value 115200, and then press Enter.

· Set Hardware flow control to No using the F key.

· Set Software flow control to No using the G key.

· Press Enter to return to the main configuration menu, and then press Esc to exit this menu.

• Reset the board, and wait for a moment. If you do not see output from the board, press Enter several times until you see the prompt. If you do not see any output from the board, and have verified that the serial terminal connection is setup correctly, contact your board vendor.

• You will see something like this when the kernel boots.

Once, it has booted, it will ask for a login, and a password.

The login is ‘ubuntu’, and the password is ‘temppwd’.
- It is necessary that you update your kernel using the following command.

```
sudo apt-get update
```

- Use an Ethernet LAN cable to connect the Beagle board to a LAN port.
- To get the Ethernet working, enter the following command.

```
sudo dhclient eth0
```

- You can check your assigned ip-address using `ifconfig -a`
- Now you can run the following commands.

```
sudo apt-get install build-essential
apt-get install gcc
sudo apt-get install openssh-server
sudo apt-get install openssh-client
```
Options for compiling:

1. On the Host:
   - You can compile your code on your host system (laptop) by:
     
gcc sample.c
   
Transfer the a.out file to the target. (Using scp as explained later)

On the target:

chmod 777 a.out

./a.out -> to execute it.

2. On the target:
   - You can compile your code on the target using:
     
gcc sample.c (OR)

arm-angstrom-linux-gnueabi-gcc sample.c

./a.out -> to execute it.

3. Code Sourcery:

http://www.codesourcery.com/sgpp/lite/arm/portal/release1600
Transferring files using SCP:

Firstly, configure ip addresses of both host and target:

```
ifconfig eth0 192.168.1.XX
```

You can check it by running

```
ifconfig -a
```
and ping on both sides.

To transfer a file from a host to a target (for example), run the following command on the host:

```
scp a.out ubuntu@192.168.1.14:/home/ubuntu
```
**To get a GUI:**

Run the following command to download the packages.

```
sudo apt-get install xfce4 gdm xubuntu-gdm-theme xubuntu-artwork xserver-xorg-video.omap3 network-manager
```

Now you need a DVI cable to connect it from your Beagle board to a monitor (which has a DVI-port).

Restart your Beagle to get a GUI on the monitor.

#Note: If the terminal window doesn’t come up in the GUI mode, try installing all the packages just to update your OS using software center. Also, you could just run the console OS (not GUI mode) on your laptop and try doing an update, build-essential, or `sudo dpkg –configure-a`