

The Embedded Linux Quick Start Guide
Using the LPC3250-Stick

Chris Simmonds

Embedded Linux Conference Europe 2010

Copyright © 2010, 2net Limited

Overview

- Description of the hardware
- Installing the tool chain
- Compiling the kernel
- Configuring the U-Boot environment
- Booting up
- What to do next

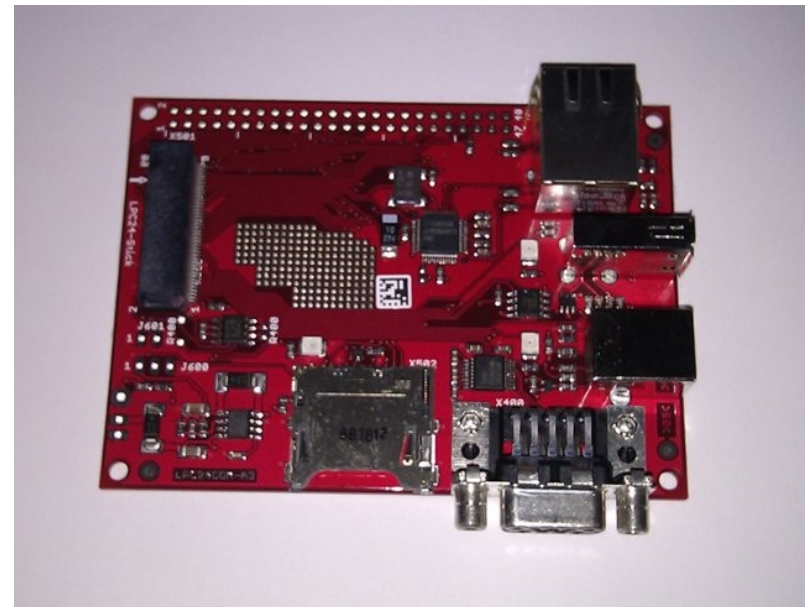
The LPC3250-Stick

- ARM926EJS, 266 MHz
- 32 MiB SDRAM
- 128 MiB NAND flash
- Ethernet
- USB 2.0 host and device
 - serial and JGAG over USB
- LCD controller

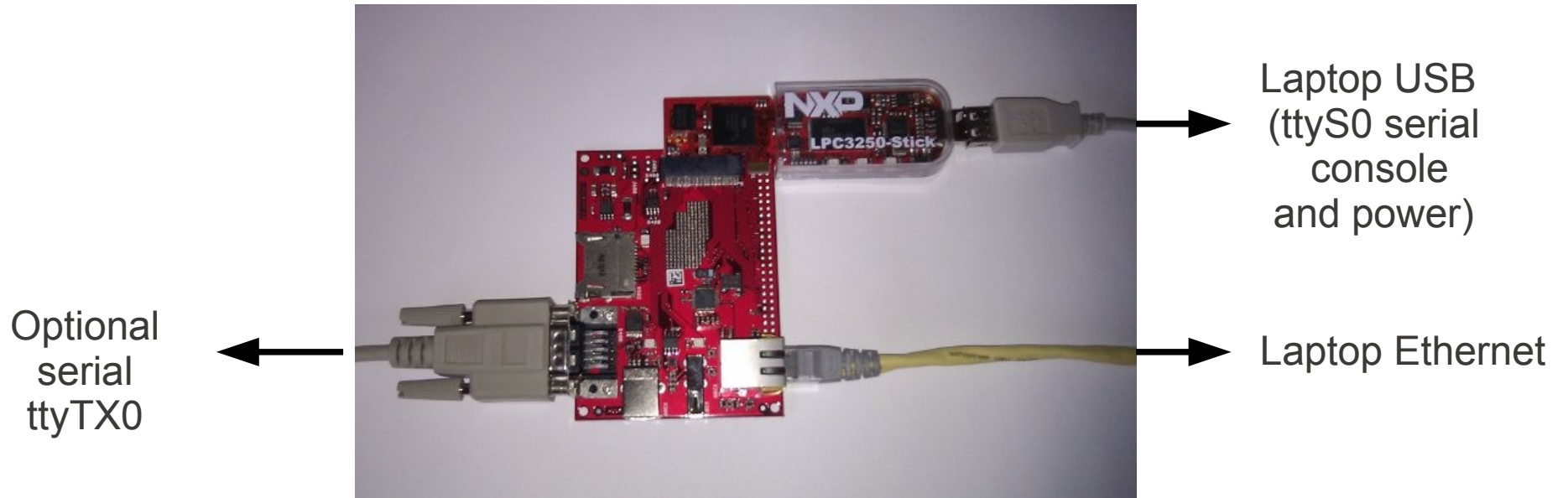


The COM board

- Connectors and interface logic for
 - RS-232
 - Ethernet
 - USB host and device
 - Mini-SD card



Connections



Tool chain: Ångström

- ARM926EJS implements arm v5 instructions including thumb (16-bit)
- So, we need an armv5t tool chain
- Pre-built tool chains available from
 - <http://www.angstrom-distribution.org/toolchains/>
- Or, build one from source following these instructions
 - <http://www.angstrom-distribution.org/building-angstrom>
 - Set machine to any ARM926EJS core, e.g. at91sam926ek

What goes where?

Base directory:

`/usr/local/angstrom/arm`

Binaries (cross compiler)

`/usr/local/angstrom/arm/bin`

Headers

`/usr/local/angstrom/arm/arm-angstrom-linux-gnueabi/usr/include`

Libraries (development and run-time)

`/usr/local/angstrom/arm/arm-angstrom-linux-gnueabi/lib`

`/usr/local/angstrom/arm/arm-angstrom-linux-gnueabi/usr/lib`

Trying it out

The Hello World program

```
#include <stdio.h>
int main(void) {
    printf ("Hello, ARM\n");
    return 0;
}
```

```
$ export PATH=/usr/local/angstrom/arm/bin:$PATH
$ arm-angstrom-linux-gnueabi-gcc hello-arm.c -o hello-arm
```

```
$ file hello-arm
hello-arm: ELF 32-bit LSB executable, ARM, version 1 (SYSV),
dynamically linked (uses shared libs), for GNU/Linux 2.6.16, not stripped
```


Serial devices

- The LPC3250-stick uses the usb-ftdi serial chip with non-standard vendor and product ID
- So, first load the driver like so
 - `sudo modprobe ftdi_sio vendor=0x0640 product=0x0026`
- Then plug in the USB cable
 - `/dev/ttyUSB0` (JTAG) and `/dev/ttyUSB1` (RS-232) are created
 - and they will disappear when unplugged...

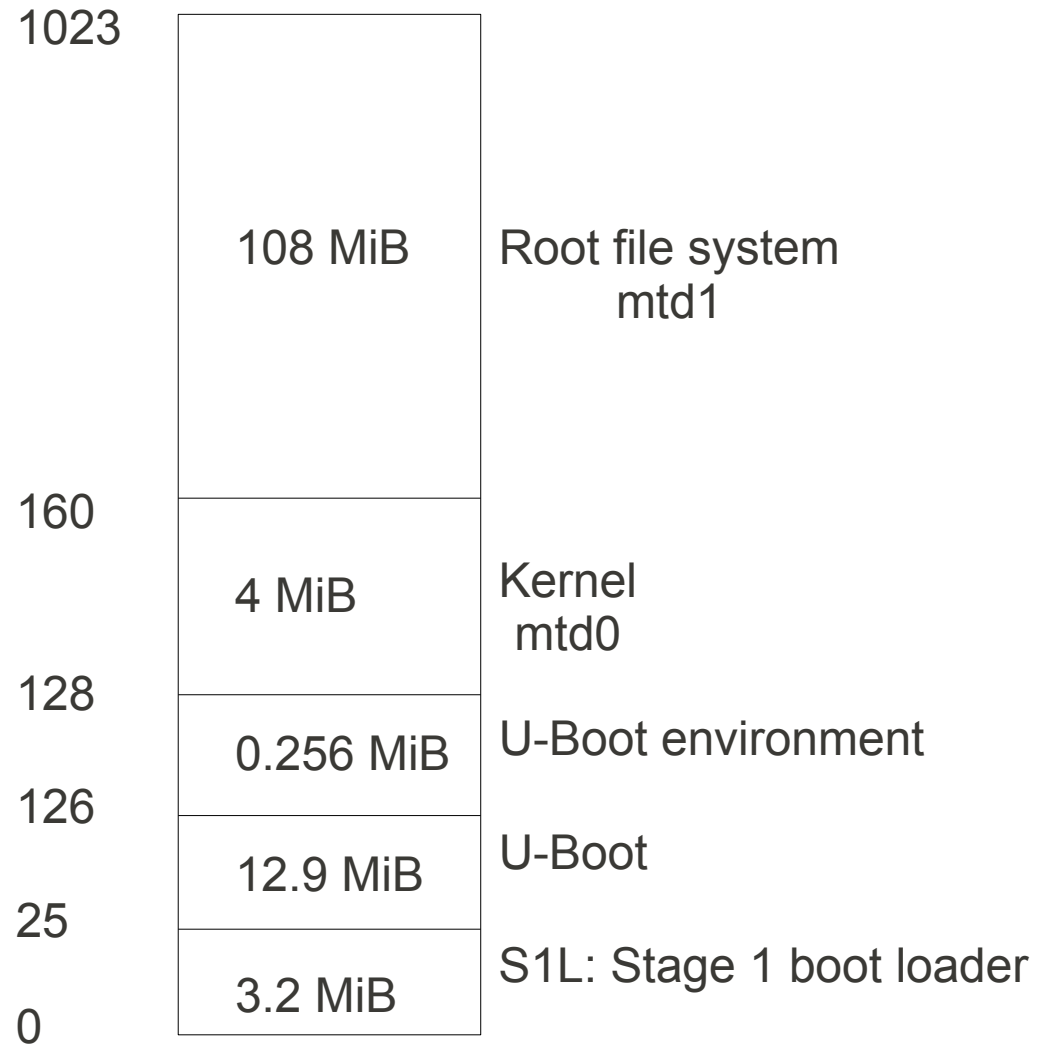
Bootloader: U-Boot

- U-Boot version 2009.03
- Serial console over USB-serial adapter
 - 115200 8n1
- Minicom (terminal emulator) settings

```
+-----+
| A -   Serial Device       : /dev/ttyUSB1
| B - Lockfile Location    : /var/lock
| C -   Callin Program     :
| D -   Callout Program    :
| E -   Bps/Par/Bits       : 115200 8N1
| F - Hardware Flow Control : No
| G - Software Flow Control : No
|
| Change which setting?
+-----+
```

Flash memory layout

- 128 MiB NAND flash
- 128 KiB erase block
- 1024 erase blocks total



Kernel: 2.6.34

- Need vanilla 2.6.34 plus LPC3250-stick BSP (a patch)
- Known to work
 - UARTs
 - Ethernet
 - NAND flash
- Not tested/known not to work
 - USB host/gadget
 - Mini SD
 - RTC

Root file system: Ångström

- Generated using on-line builder
 - <http://www.angstrom-distribution.org/narcissus/>
 - Machine at91sam926ek
 - Console-only image
 - 39 MiB when uncompressed
- Could be built from source, as with toolchain

Configuring the network

Network configuration in: `/etc/network/interfaces`

Bring interface up with: `ifup [interface | -a]`

Take down with: `ifdown [interface | -a]`

Example `/etc/interfaces`:

```
auto lo
iface lo inet loopback

auto eth0
iface eth0 inet dhcp

iface eth1 inet static
    address 192.168.0.202
    netmask 255.255.255.0
    network 192.168.0.0
    gateway 192.168.0.200
```

Ångström package manager: opkg

- Ångström has > 1000 packages
- Installing packages is less error prone than “roll-your-own”
- Handles dependencies and version numbering
- But, takes up storage space (~ 10 MiB) and less flexible than RYO

opkg examples

Update local database of packages from feeds (see /etc/opkg)

```
opkg update
```

List all packages available

```
opkg list
```

List packages installed

```
opkg list_installed
```

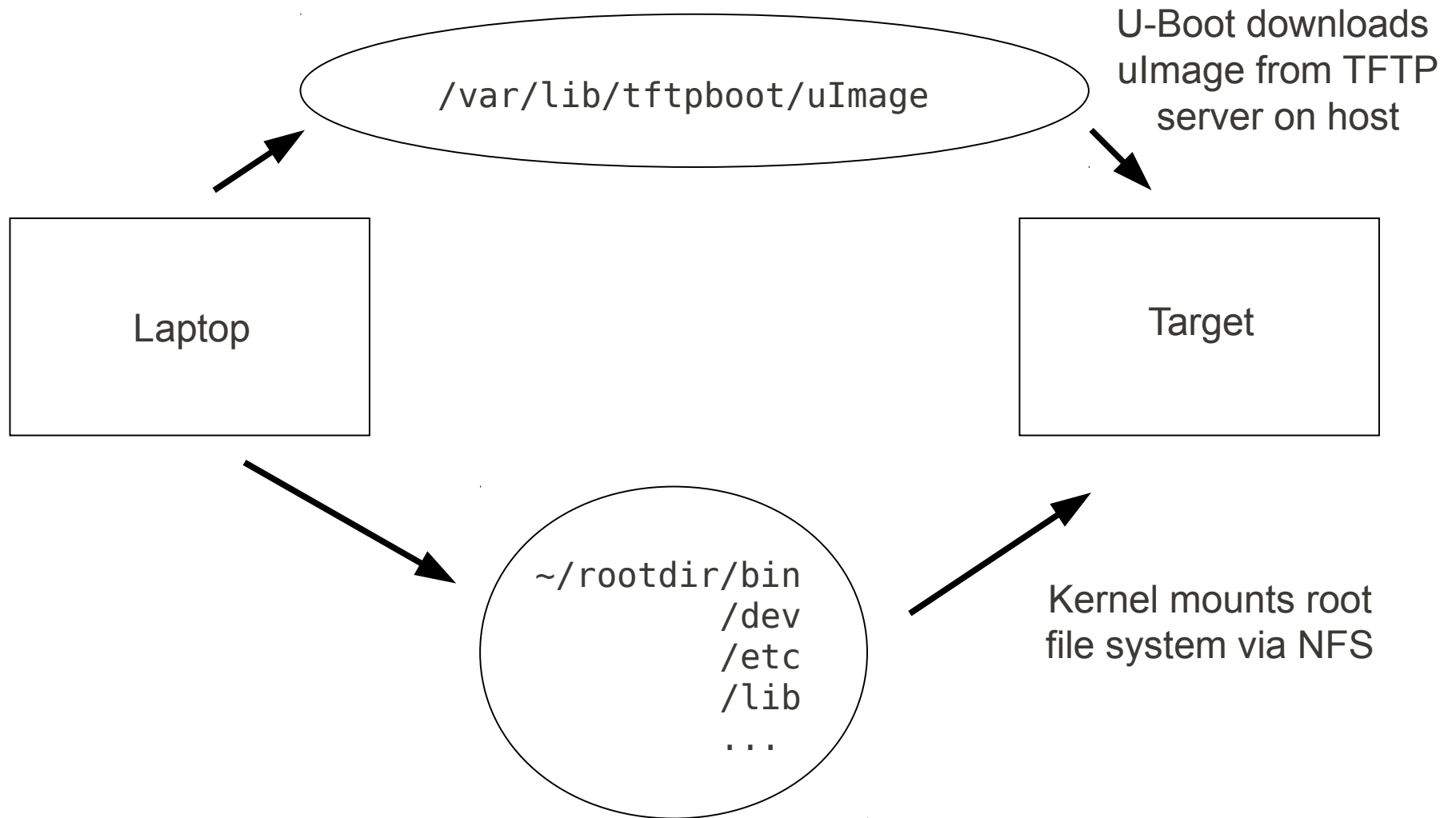
Install a package (e.g. procps)

```
opkg install procps
```

Install a specific package file (bypassing the feeds)

```
opkg install /ipk/procps_3.2.8-r9.1.5_armv5te.ipk
```


TFTP and NFS



Hands-on

- Get lab notes from
 - <http://www.embedded-linux.co.uk/downloads/elce-2010/linux-quick-start-lab-notes.pdf>
- Just follow the instructions...

Further reading

- “Inner Penguin”
 - <http://www.embedded-linux.co.uk/>
- Embedded Linux Wiki
 - http://elinux.org/Main_Page
- Embedded Linux Primer
 - Christopher Hallinan, Prentice Hall
- Building Embedded Linux Systems, 2nd Edition
 - Karim Yaghmour et al, O'Reilly & Associates

Thank you for listening

Happy hacking*

(*) RFC 1392 "Internet Users' Glossary"

hacker

A person who delights in having an intimate understanding of the internal workings of a system, computers and computer networks in particular. The term is often misused in a pejorative context, where "cracker" would be the correct term. See also: cracker.