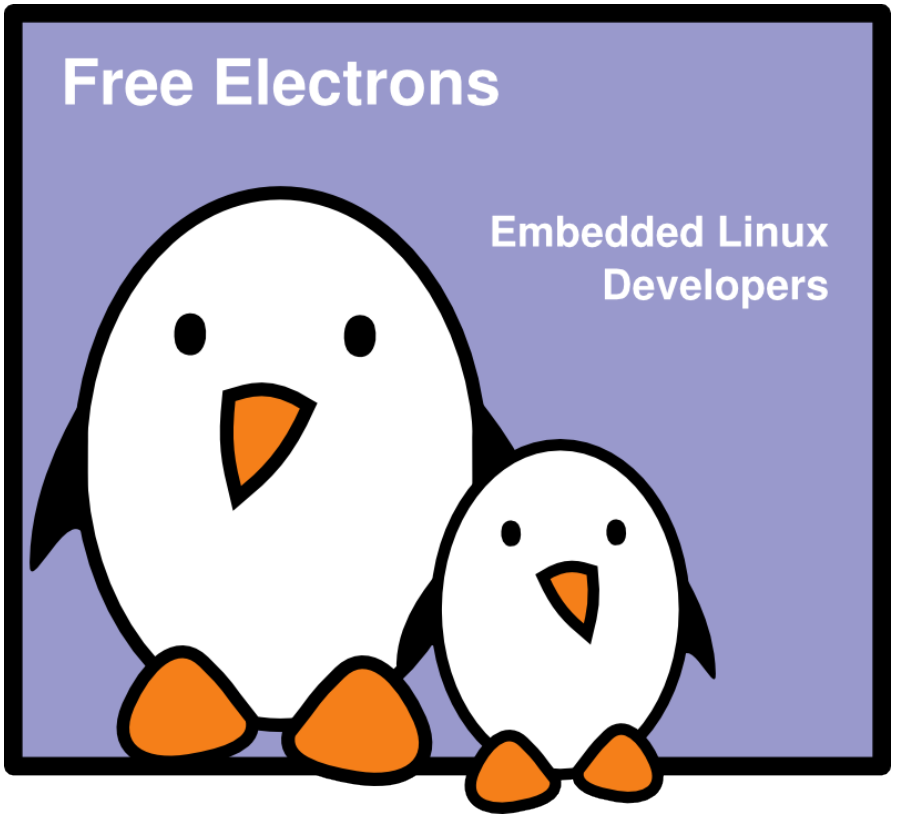
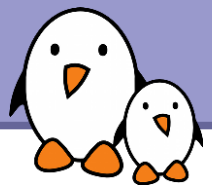
Porting U-boot

Michael Opdenacker
Free Electrons





Rights to copy

© Copyright 2004-2009, Free Electrons
feedback@free-electrons.com

Document sources, updates and translations:
http://free-electrons.com/docs/porting-u-boot

Corrections, suggestions, contributions and translations are welcome!

Latest update: Sep 15, 2009

creative commons
COMMONS DEED

Attribution – ShareAlike 3.0

You are free

- to copy, distribute, display, and perform the work
- to make derivative works
- to make commercial use of the work

Under the following conditions

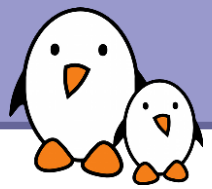
Attribution. You must give the original author credit.

Share Alike. If you alter, transform, or build upon this work, you may distribute the resulting work only under a license identical to this one.

- For any reuse or distribution, you must make clear to others the license terms of this work.
- Any of these conditions can be waived if you get permission from the copyright holder.

Your fair use and other rights are in no way affected by the above.

License text: http://creativecommons.org/licenses/by-sa/3.0/legalcode

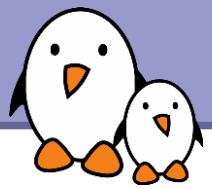


Porting U-boot (1)

It is relatively easy to port U-boot to a new board.
This explains its success.

- Look for an already supported board which is as close as possible to yours.
- Create a new file in `include/configs`.
  For example, have a look at the `adsvix.h` file (pxa270 processor).

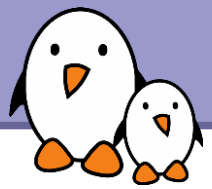
Free Electrons. Kernel, drivers and embedded Linux development, consulting, training and support. **http//free-electrons.com**



Porting U-boot (2)

- Then create support code in the `board` directory
- Example: have a look at `board/adsvix`:
  `adsvix.c`
  `config.mk`
  `lowlevel_init.S`
  `Makefile`
  `pcmcia.c`
  `pxavoltage.S`
  `u-boot.lds`
- You will find many similarities with `board/lubbock`.

Free Electrons. Kernel, drivers and embedded Linux development, consulting, training and support. **http//free-electrons.com**



Porting U-boot (3)

Then, add your board to the following files:

- `MAKEALL`:

```
LIST_pxa="      \
        adsvix          cerf250         cradle          csb226          \
        delta           innokom         lubbock         pleb2           \
        pxa255_idp      wepep250        xaeniax         xm250           \
        xsengine        zylonite                                        \
```

- `Makefile`:

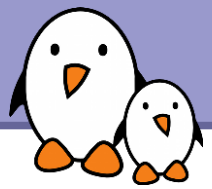
```
adsvix_config   :       unconfig
        @$(MKCONFIG) $(@:_config=) arm pxa adsvix
```

- `include/asm-arm/mach-types.h`:

```
#define MACH_TYPE_ADSVIX                620
```

(originating from Linux headers, should already be present)

Free Electrons. Kernel, drivers and embedded Linux development, consulting, training and support. **http//free-electrons.com**



Other U-boot implementation details

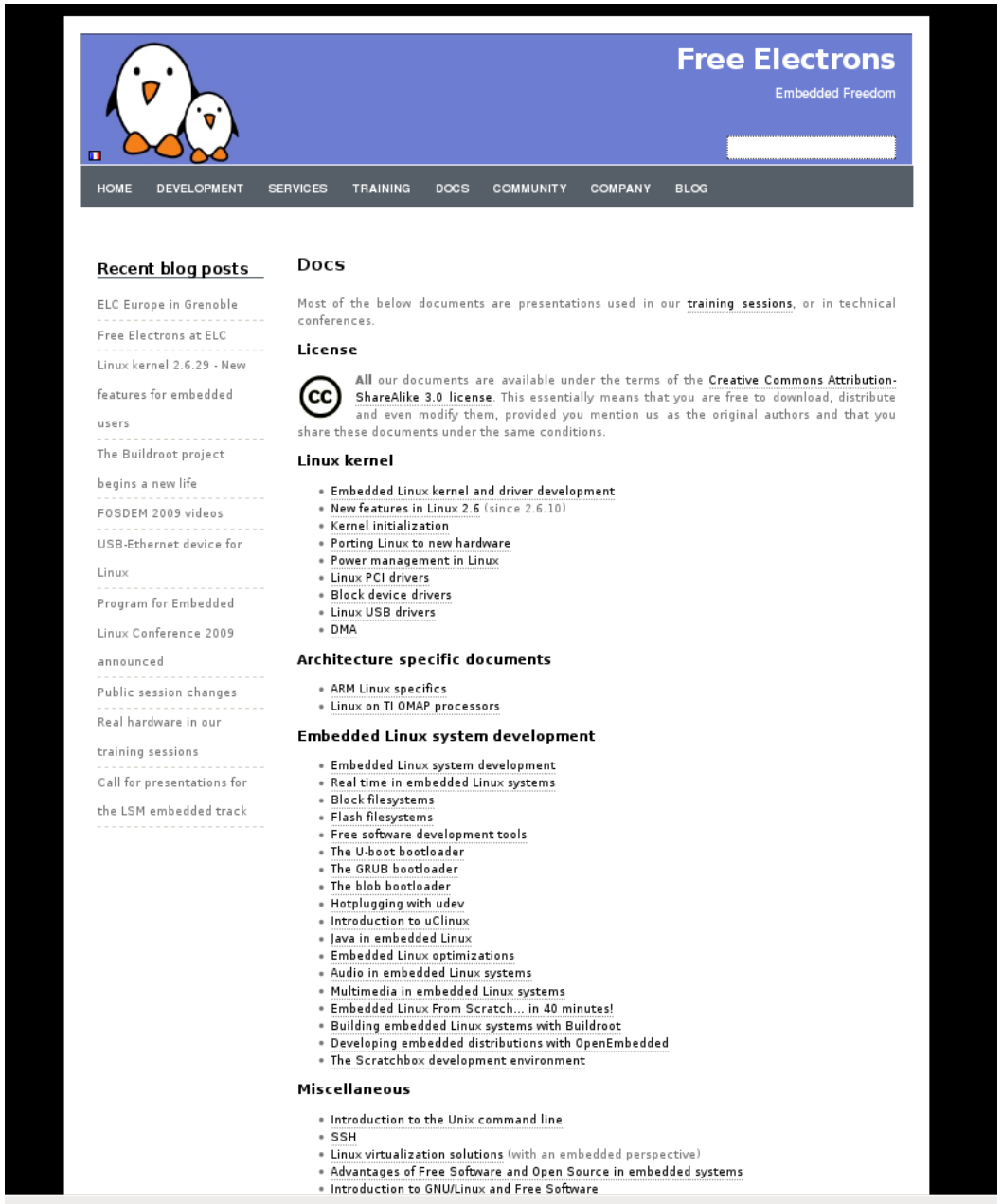
- CPU support implemented in the `cpu/` directory

```
74xx_7xx   arm926ejs   i386         mips      mpc8260   nios      sa1100
arm1136    arm946es    ixp          mpc5xx    mpc83xx   nios2
arm720t    arm_intcm   lh7a40x      mpc5xxx   mpc85xx   ppc4xx
arm920t    at32ap      mcf52x2      mpc8220   mpc86xx   pxa
arm925t    bf533       microblaze   mpc824x   mpc8xx    s3c44b0
```

- Device drivers implemented in `drivers/`
  A lot of reuse from Linux drivers!
- Headers implemented in `include/`
  and in `include/asm-<arch>/` (mostly Linux headers)

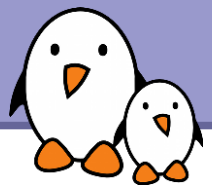
Free Electrons. Kernel, drivers and embedded Linux development, consulting, training and support. **http//free-electrons.com**

Related documents



All our technical presentations on http://free-electrons.com/docs

- Linux kernel
- Device drivers
- Architecture specifics
- Embedded Linux system development



How to help

You can help us to improve and maintain this document...

- By sending corrections, suggestions, contributions and translations
- By asking your organization to order development, consulting and training services performed by the authors of these documents (see http://free-electrons.com/).
- By sharing this document with your friends, colleagues and with the local Free Software community.
- By adding links on your website to our on-line materials, to increase their visibility in search engine results.

Linux kernel

Linux device drivers
Board support code
Mainstreaming kernel code
Kernel debugging

Embedded Linux Training

All materials released with a free license!

Unix and GNU/Linux basics
Linux kernel and drivers development
Real-time Linux, uClinux
Development and profiling tools
Lightweight tools for embedded systems
Root filesystem creation
Audio and multimedia
System optimization

Free Electrons

Our services

Custom Development

System integration
Embedded Linux demos and prototypes
System optimization
Application and interface development

Consulting and technical support

Help in decision making
System architecture
System design and performance review
Development tool and application support
Investigating issues and fixing tool bugs