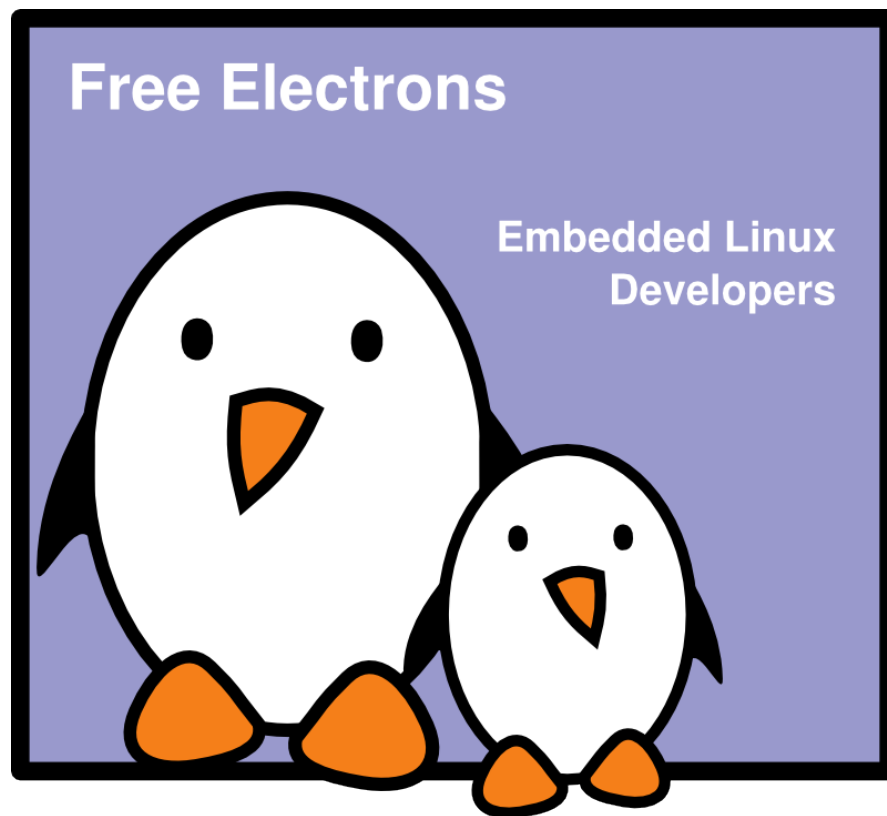




Porting U-boot

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



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).



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`.



Other U-boot implementation details

- ▶ CPU support implemented in the `cpu/` directory

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

- ▶ Device drivers implemented in `drivers/`

A lot of reuse from Linux drivers!

- ▶ Headers implemented in `include/`

and in `include/asm-<arch>/` (mostly Linux headers)



Related documents

Free Electrons
Embedded Freedom

HOME DEVELOPMENT SERVICES TRAINING DOCS COMMUNITY COMPANY BLOG

Recent blog posts

- ELC Europe in Grenoble
- Free Electrons at ELC
- Linux kernel 2.6.29 - New features for embedded users
- The Buildroot project begins a new life
- FOSDEM 2009 videos
- USB-Ethernet device for Linux
- Program for Embedded Linux Conference 2009 announced
- Public session changes
- Real hardware in our training sessions
- Call for presentations for the LSM embedded track

Docs

Most of the below documents are presentations used in our [training sessions](#), or in technical conferences.

License

All our documents are available under the terms of the [Creative Commons Attribution-ShareAlike 3.0 license](#). This essentially means that you are free to download, distribute and even modify them, provided you mention us as the original authors and that you share these documents under the same conditions.

Linux kernel

- [Embedded Linux kernel and driver development](#)
- [New features in Linux 2.6](#) (since 2.6.10)
- [Kernel initialization](#)
- [Porting Linux to new hardware](#)
- [Power management in Linux](#)
- [Linux PCI drivers](#)
- [Block device drivers](#)
- [Linux USB drivers](#)
- [DMA](#)

Architecture specific documents

- [ARM Linux specifics](#)
- [Linux on TI OMAP processors](#)

Embedded Linux system development

- [Embedded Linux system development](#)
- [Real time in embedded Linux systems](#)
- [Block filesystems](#)
- [Flash filesystems](#)
- [Free software development tools](#)
- [The U-boot bootloader](#)
- [The GRUB bootloader](#)
- [The blob bootloader](#)
- [Hotplugging with udev](#)
- [Introduction to uClinux](#)
- [Java in embedded Linux](#)
- [Embedded Linux optimizations](#)
- [Audio in embedded Linux systems](#)
- [Multimedia in embedded Linux systems](#)
- [Embedded Linux From Scratch... in 40 minutes!](#)
- [Building embedded Linux systems with Buildroot](#)
- [Developing embedded distributions with OpenEmbedded](#)
- [The Scratchbox development environment](#)

Miscellaneous

- [Introduction to the Unix command line](#)
- [SSH](#)
- [Linux virtualization solutions \(with an embedded perspective\)](#)
- [Advantages of Free Software and Open Source in embedded systems](#)
- [Introduction to GNU/Linux and Free Software](#)

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

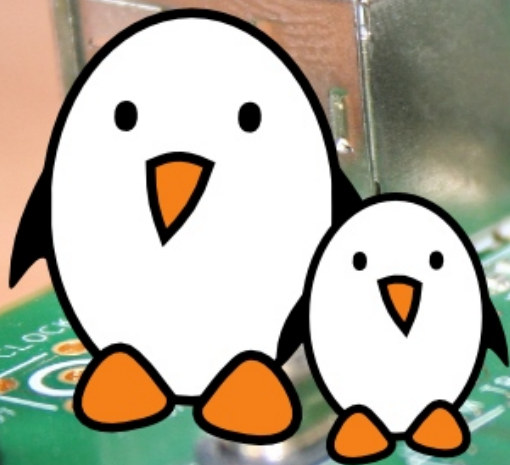
Our services

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



Free Electrons
Embedded Linux Experts

<http://free-electrons.com>