

Haute école d'ingénierie et d'architecture Fribourg
Hochschule für Technik und Architektur Freiburg

MA-CSEL1 – Construction Systèmes Embarqués sous Linux

Introduction

HES-SO//Master TIC/TIN 2020-21



Contenu

- ▶ **Bienvenue**
- ▶ **Objectifs du module**
- ▶ **Contenu du module**
- ▶ **Déroulement des cours**
- ▶ **Rapport de laboratoire**
- ▶ **Programme des cours**
- ▶ **Communication**
- ▶ **Evaluation**
- ▶ **Littérature et liens utiles**



Bienvenue

- ▶ **Module : MA-CSEL1**
- ▶ **Titre : MA-CSEL1 - Construction de Systèmes Embarqués sous Linux**
- ▶ **Intervenant**
 - ❑ Daniel Gachet (HEIA-FR / ISC)
- ▶ **24 étudiants inscrits**
 - ❑ 15 étudiants TIC
 - ❑ 9 étudiants TIN
 - ❑ 8 plein temps
 - ❑ 16 temps partiel



Communication

► Communication e-mail

- ❑ daniel.gachet@hefr.ch

► Support de cours

- ❑ Transparents et documents sont disponible en version PDF sous Moodle <http://moodle.msengineering.ch>
 - *Lausanne*
 - *Modules d'approfondissement*
 - *S1 -20120-2021*
- ❑ Cours : **CSEL1 - Construction de systèmes embarqués sous Linux**
- ❑ Code : **csel1a**
- ❑ Lien: <https://moodle.msengineering.ch/course/view.php?id=1482>



Groupes

Groupe A

1. Arrigo Marc-Olivier
1. Walpen Elliott
2. Sampiemon Thibault
2. Bach Jacques
3. Willy Sylvain
3. Besançon Matteo
4. Florek Kamil
4. Hayoz Amanda (+ codeuse)
5. Moullet Tobias
5. Rapin Quentin

Groupe B

6. Andrey Lucas
6. Bétend Loan
7. Charbon Yann
7. Spinelli Isaïa
8. Dupuis Raoul
8. Geier Phillip
9. Compagnon Dimitri
9. Schneuwly Vincent
10. Gonzalez Puertas Antonio
10. Baldo Fabio
11. Pothula Anuraag
11. Almou Samir El Fatmi
12. Sahli Michel
12. Cippà Celso



Programme des cours

Semaine	Date	Groupe	Salle	Horaire	Thème	
38	SA-1	18.09.2020	tous	B30:20	08:45-14:30	Environnement Linux embarqué
39	SA-2	25.09.2020	A	C00:16	08:45-14:30	Programmation noyau Linux – 1 ^{ère} partie
40	SA-3	02.10.2020	B	B30:20		
41	SA-4	09.10.2020	A	C00:16	08:45-14:30	Programmation noyau Linux – 2 ^{ème} partie
42	SA-5	16.10.2020	B	B30:20		
43	SA-6	23.10.2020	A	C00:16	08:45-14:30	Programmation système Linux – 1 ^{ère} partie
44	SA-7	30.10.2020	B	B30:20		
45	SA-8	06.11.2020	A	C00:16	08:45-14:30	Programmation système Linux – 2 ^{ème} partie
46	SA-9	13.11.2020	B	B30:20		
47	SA-10	20.11.2020	A	C00:16	08:45-14:30	Optimisation système Linux
48	SA-11	27.11.2020	B	B30:20		
50	SA-13	11.12.2020	tous	C00:16	08:45-14:30	Mini projet intégré (distanciel)



Horaires

▶ **1^{ère} partie : 08h45 – 09h30**

→ *pause café*

▶ **2^{ème} partie : 09h45 – 11h30**

→ *pause de midi*

▶ **3^{ème} partie : 12h30 – 14h30**



Déroulement des cours

▶ Déroulement

- ❑ Travail personnel de préparation avant le cours
- ❑ Cours combiné: théorie et travail pratique
- ❑ Travail personnel après le cours
- ❑ Rédaction d'un rapport

▶ Participation et collaboration

- ❑ Ne pas hésiter à participer et rendre le cours plus interactif et plus intéressant
- ❑ Ne pas hésiter à poser des questions
- ❑ Ne pas hésiter à partager vos propres expériences
- ❑ Ne pas hésiter à aider vos collègues
- ❑ Ne pas hésiter à expliquer ce que vous avez compris à vos collègues
- ❑ Ne pas hésiter à notifier des bugs aux professeurs
- ❑ Ne pas hésiter à regarder des solutions sur Internet



Objectifs du module

- ▶ **Au terme de ce module, les étudiant-e-s seront capable de :**
 - ❑ Choisir l'architecture (HW et SW) pour des systèmes embarqués pour Linux
 - ❑ Mettre en place un environnement de développement pour Linux
 - ❑ Choisir et configurer un boot-loader pour une cible donnée
 - ❑ Choisir et configurer un système de fichiers pour Linux
 - ❑ Choisir et configurer une distribution Linux pour une cible donnée
 - ❑ Concevoir et développer des device drivers complexes sous Linux embarqué
 - ❑ Concevoir et développer des applications sous Linux embarqué



Contenu du module (I)

▶ Environnement Linux embarqué

- ❑ Architecture générale de systèmes embarqués pour Linux
- ❑ Environnements de développement
- ❑ Outils de développement
- ❑ Création, génération et débogage d'applications
- ❑ Survol des boot-loaders
- ❑ Survol et configuration des systèmes de fichiers
- ❑ Survol des distributions Linux embarqué

▶ Programmation noyaux Linux

- ❑ Programmation de modules noyau
- ❑ Développement et débogage d'un module noyaux
- ❑ Types de pilotes de périphériques
- ❑ Développement et débogage d'un pilote de périphériques
- ❑ Traitement des interruptions
- ❑ Paramétrage et configuration d'un pilotes de périphériques



Contenu du module (II)

▶ **Programmation système Linux**

- ❑ Interface noyau (system calls et bibliothèques)
- ❑ File I/O
- ❑ Gestions des processus et threads
- ❑ Communication inter processus (IPC, signaux, ...)
- ❑ Mémoire virtuelle et accès aux périphériques par UIO
- ❑ Unix domain sockets
- ❑ IPC system V
- ❑ Caractéristiques de l'ordonnanceur
- ❑ Mesure de performance (profiling)
- ❑ Aspects temps-réel (processus/thread)

▶ **Mini-projet**

- ❑ Mise en œuvre des concepts étudiés durant les 6 premières semaines



Rapport de laboratoire

► Rapport

- ❑ Rapports de laboratoire par groupe de 2 étudiants, à rendre
 - ❖ 30.10.2020 : cours 1+2+3
 - ❖ 11.12.2020 : cours 4+5+6
 - ❖ 08.01.2021 : cours 7
- ❑ Format : fichier PDF avec les annexes dans un fichier .tar
- ❑ Langue : français, allemand ou anglais

► Contenu (*indicatif*)

- ❑ En-tête (logos, établissement, titre, auteur, lieu et date, ...)
- ❑ Par journée de cours (sujet / thème) entre 1 et 4 pages
 - ❖ *Résumé du laboratoire*
 - ❖ *Réponse aux questions*
 - ❖ *Synthèse sur ce qui a été appris/exercé*
 - » *Non acquis*
 - » *Acquis, mais à exercer*
 - » *Parfaitement acquis*
 - ❖ *Remarques et choses à retenir*
 - ❖ *Feedback personnel sur le laboratoire*
- ❑ Annexes (références/littérature, code source/patches dans fichier tar)



Evaluation

▶ Examen théorique

- ❑ Sous forme d'examen oral
 - ❖ *Deux thèmes tirés au sort*
 - ❖ *20' de préparation (un résumé de 5 feuilles A4 recto-verso autorisé)*
 - ❖ *20' de présentation au rétroprojecteur sur transparents avec questions des professeurs (le résumé n'est plus autorisé)*
- ❑ Experts: le professeur du module + un professeur du domaine

▶ Note finale du cours

- ❑ Examen théorique oral 65%
- ❑ Le rapport de laboratoire 35%



Littérature – first reading (I)

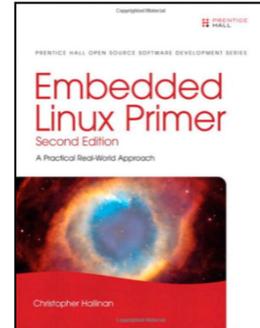
Embedded Linux Primer, Second Edition

A Practical Real-World Approach

By Christopher Hallinan

November, 2010, 656 pages

ISBN-10: 0137017839 | ISBN-13: 978-0137017836



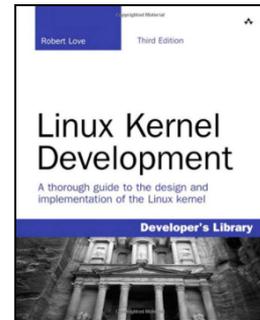
Linux Kernel Development

A thorough guide to the design and implementation of the Linux kernel

By Robert Love

July, 2010, 440pages

ISBN-10: 0672329468 | ISBN-13: 978-0672329463



Embedded Linux Primer, Second Edition

Probably the most wide ranging and complete Linux device driver book...

By Sreekrishnan Venkateswaran

April, 2008, 744 pages

ISBN-10: 0132396556 | ISBN-13: 978-0132396554





Littérature – first reading (II)

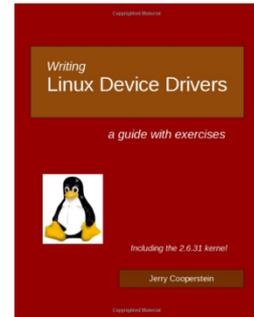
Writing Linux Device Drivers

A guide with exercices

By Jerry Cooperstein

September, 2009, 394 pages

ISBN-10: 1448672384 | ISBN-13: 978-1448672387



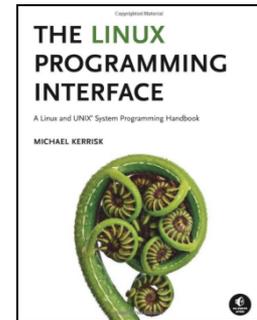
The Linux Programming Interface:

A Linux and UNIX System Programming Handbook

By Michael Kerrisk

October, 2010, 1552 pages

ISBN-10: 1593272200 | ISBN-13: 978-1593272203





Littérature – second reading (I)

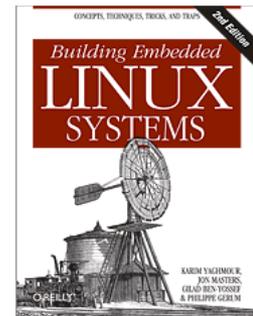
Building Embedded Linux Systems, Second Edition

Concepts, techniques, tricks, and traps

By Philippe Gerum, Karim Yaghmour, Jon Masters, Gilad Ben-Yossef

August 2008, 462 pages

ISBN 10: 0-596-52968-6 | ISBN 13: 9780596529680



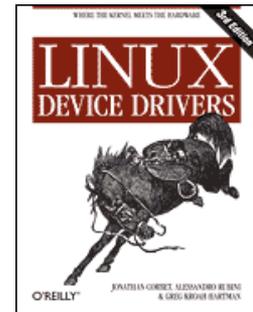
Linux Device Drivers, Third Edition

By Jonathan Corbet, Alessandro Rubini, Greg Kroah-Hartman

February 2005, 636 pages

ISBN 10: 0-596-00590-3 | ISBN 13: 9780596005900

<http://lwn.net/Kernel/LDD3/> (as PDF)

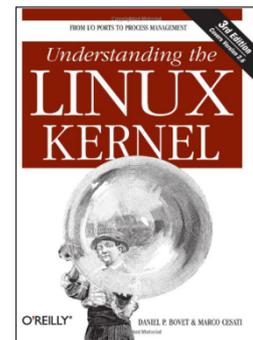


Understanding the Linux Kernel, Third Edition

By Daniel P. Bovet & Marco Cesati

November 2005, 944 pages

ISBN 10: 0-596-00565-2 0-596-00590-3 | ISBN 13: 978-0596005658





Littérature – second reading (II)

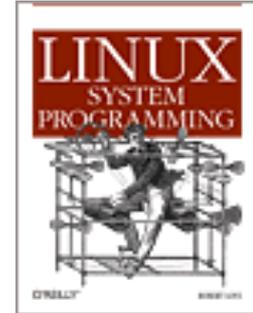
Linux System Programming

Talking Directly to the Kernel and C Library

By Robert Love

September 2007, 388 pages

ISBN 10: 0-596-00958-5 | ISBN 13: 9780596009588

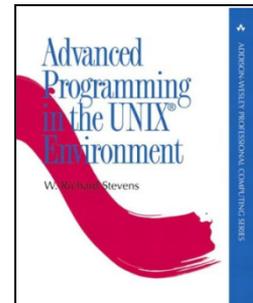


Advanced Programming in the UNIX Environment

By W. R. Stevens, Stephen A. Rago

January 1992, 768 pages

ISBN-10: 0201563177 | ISBN-13: 978-0201563177

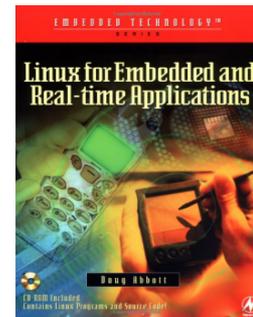


Linux for Embedded and Real-time Applications

By Doug Abboot

March, 2003, 254 pages

ISBN-10: 0750675462 | ISBN-13: 978-0750675468





Littérature – second reading (III)

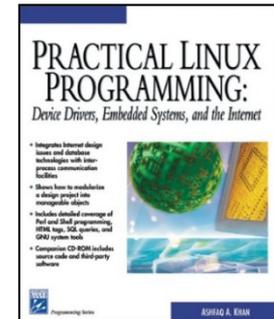
Practical Linux Programming

Device Drivers, Embedded Systems, and the Internet

By Ashfaq A. Khan

February, 2002, 420 pages

ISBN-10: 1584500964 | ISBN-13: 978-1584500964



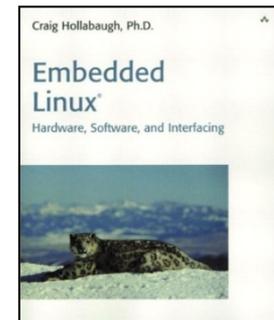
Embedded Linux

Hardware, Software, and Interfacing

By Craig Hollabaugh

March 2002, 432 pages

ISBN-10: 0672322269 | ISBN-13: 978-0672322266





Liens utiles

▶ Noyau Linux

www.tldp.org

Linux documentation

lwn.net

Linux weekly news

www.tux.org/lkml/

Linux kernel mailing list (FAQ)

kernelnewbies.org

Kernel Newbies (glossaire, articles, présentations, ...)

▶ Projet ARM

www.arm.linux.org.uk ARM Linux project

▶ Consultants et fabricants

www.denx.de

Denx software Engineering (U-Boot / ELDK)

bootlin.com

bootlin (Experts Linux embarqué)