Mastering Embedded Linux Programming

Third Edition

Create fast and reliable embedded solutions with Linux 5.4 and the Yocto Project 3.1 (Dunfell)

Frank Vasquez

Chris Simmonds

Packt

BIRMINGHAM—MUMBAI

Mastering Embedded Linux Programming

Third Edition

Group Product Manager: Wilson Dsouza
Publishing Product Manager: Sankalp Khattri
Senior Editor: Rahul Dsouza
Content Development Editor: Sayali Pingale
Technical Editor: Sarvesh Jaywant
Copy Editor: Safis Editing
Project Coordinator: Neil Dmello
Proofreader: Safis Editing
Indexer: Tejal Soni
Production Designer: Nilesh Mohite

First published: December 2015
Second edition: June 2017
Third edition: March 2021
Production reference: 0140421

Published by Packt Publishing Ltd.
Livery Place
35 Livery Street
Birmingham
B3 2PB, UK.

ISBN 978-1-78953-038-4
www.packt.com

To the open source software community (especially the Yocto Project) for welcoming me in wholeheartedly. And to my wife Deborah for putting up with the late-night hardware hacking. The world runs on Linux.

– Frank Vasquez

Contributors

About the authors

Frank Vasquez is an independent software consultant specializing in consumer electronics. He has over a decade of experience designing and building embedded Linux systems. During that time, he has shipped numerous devices including a rackmount DSP audio server, a diver-held sonar camcorder, and a consumer IoT hotspot. Before his career as an embedded Linux engineer, Frank was a database kernel developer at IBM, where he worked on DB2. He lives in Silicon Valley.

Chris Simmonds is a software consultant and trainer living in southern England. He has almost two decades of experience in designing and building open source embedded systems. He is the founder and chief consultant at 2net Ltd, which provides professional training and mentoring services in embedded Linux, Linux device drivers, and Android platform development. He has trained engineers at many of the biggest companies in the embedded world, including ARM, Qualcomm, Intel, Ericsson, and General Dynamics. He is a frequent presenter at open source and embedded conferences, including the Embedded Linux Conference and Embedded World.

About the reviewers

Ned Konz is an autodidact who believes in Sturgeon's law and tries to work on the other 10% of everything. His work experience over the last 45 years includes software and electronics design for industrial machines, and consumer and medical devices, as well as doing user interface research with Alan Kay's team at HP Labs. He has embedded Linux in devices including high-end SONAR systems, inspection cameras, and the Glowforge laser cutter. As a senior embedded systems programmer at Product Creation Studio in Seattle, he designs software and electronics for a variety of client products. In his spare time, he builds electronics gadgets and plays bass in a rock band. He has also done two solo bicycle tours of over 4,500 miles each.

I'd like to thank my wife Nancy for her support, and Frank Vasquez for recommending me as a technical reviewer.

Khem Raj holds a bachelor's degree (Hons) in electronics and communications engineering. In his career spanning 20 years in software systems, he has worked with organizations ranging from start-ups to Fortune 500 companies. During this time, he has worked on developing operating systems, compilers, computer programming languages, scalable build systems, and system software development and optimization. He is passionate about open source and is a prolific open source contributor, maintaining popular open source projects such as the Yocto Project. He is a frequent speaker at open source conferences. He is an avid reader and a lifelong learner.

Table of Contents

3
All about Bootloaders

4
Configuring and Building the Kernel

5
Building a Root Filesystem

6
Selecting a Build System

7
Developing with Yocto

8
Yocto Under the Hood

Section 2: System Architecture and Design Decisions

9
Creating a Storage Strategy

10
Updating Software in the Field

11
Interfacing with Device Drivers

12
Prototyping with Breakout Boards

13
Starting Up – The init Program

14
Starting with BusyBox runit

15
Managing Power

Section 3: Writing Embedded Applications

16
Packaging Python

17
Learning about Processes and Threads

18
Managing Memory

Section 4: Debugging and Optimizing Performance

19

Debugging with GDB

20

Profiling and Tracing

21
Real-Time Programming