

Linux em Sistemas Embarcados

Marcelo Veiga Neves
marcelo.veiga@gmail.com

SACTA 2010
Alegrete, 25 de junho de 2010

Roteiro

- Introdução
- Projeto de Linux Embarcado
- Demonstração
- Referências

Roteiro

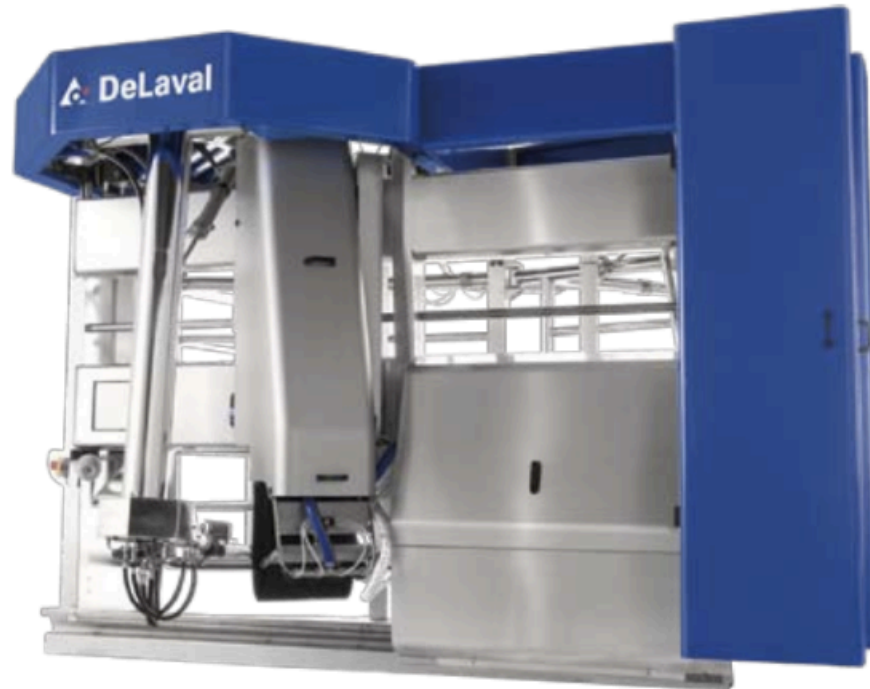
- Introdução
 - Sistemas Embarcados
 - Linux Embarcado
 - Exemplos de Sistemas Embarcados com Linux
 - Inicialização do Linux em Sistemas Embarcados
- Projeto de Linux Embarcado
- Demonstração
- Referências

Introdução

- Objetivos:
 - Conhecer a área de Linux para Sistemas Embarcados
 - Compreender o funcionamento de um Sistema de Linux Embarcado
 - Quais os componentes de software precisam ser preparados para criar um Linux Embarcado completo
- Pré-requisitos:
 - Compreensão do papel e funcionamento do Kernel do Linux
 - Familiaridade com ambiente de desenvolvimento Linux

Sistemas Embarcados

- Definição de Sistema Embarcado:
 - “Um sistema embarcado é um computador que é completamente encapsulado ou dedicado ao dispositivo ou sistema que ele controla.” (Wikipedia)
- Requisitos/Características de Sistemas Embarcados
 - Recursos reduzidos (CPU, memória, energia, etc.)
 - Armazenamento de estado sólido (flash)
 - Operação “Head less” (sem display)
 - Etc.



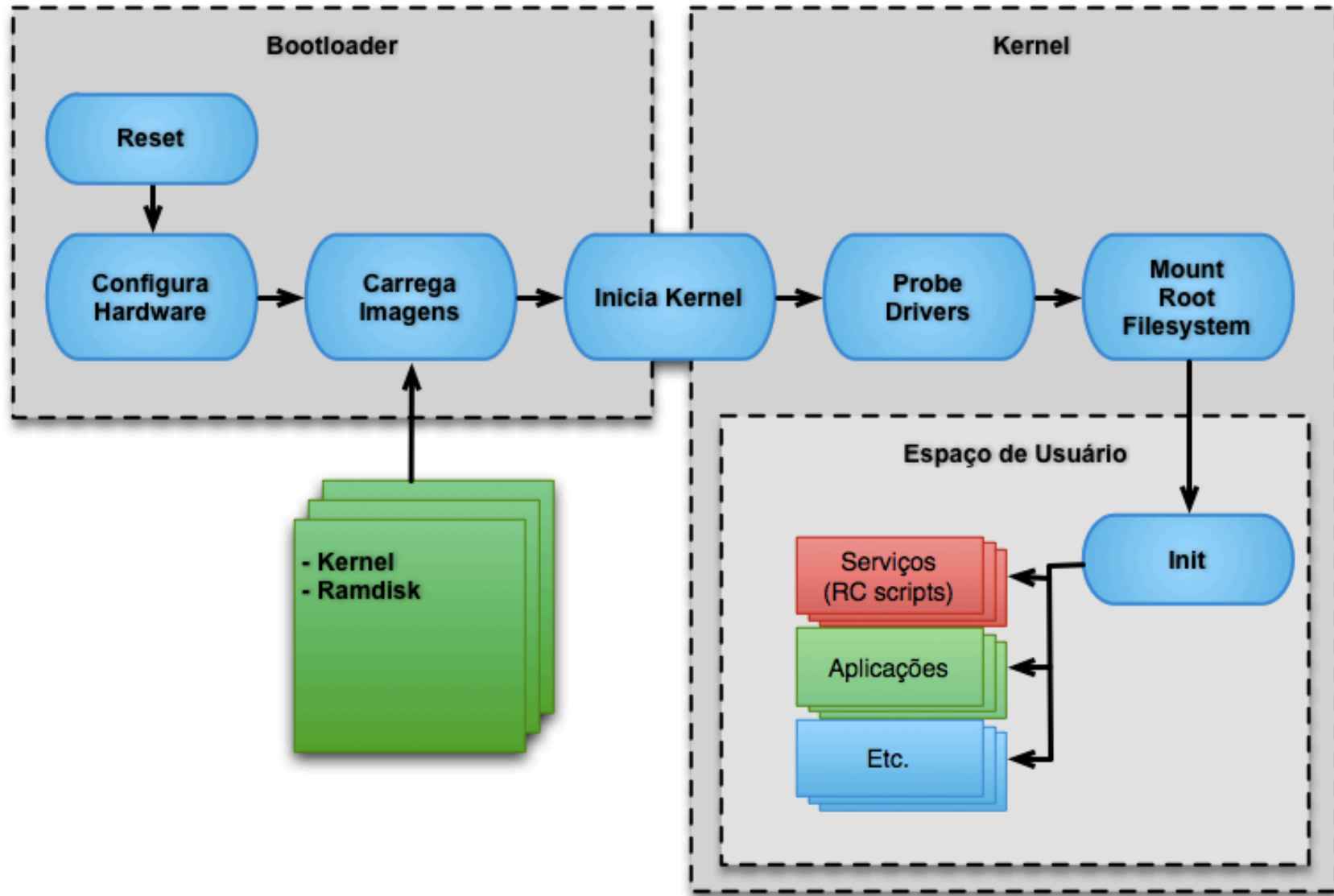
Oportunidades em Linux Embarcado

- Oportunidade para:
 - Nerds que gostam de Linux e Software Livre
 - Quem quer desenvolver em nível de Kernel
 - Porte do Linux para novas plataformas
 - Criação e adaptação de drivers de dispositivos
 - Implementação de protocolos de rede
 - Quem prefere programar em C em ambiente Linux (também tem lugar para quem gosta de Java, C++, Python, etc.)
- Mercado está aquecido e não existe profissionais
- Oportunidades também para empreendedores

Linux Embarcado

- Todo o software necessário para o Linux funcionar em um Sistema Embarcado
 - Aplicações para prover funcionalidades específicas do sistema
 - Kernel com configuração específica
 - Bootloader para configurar o hardware e iniciar o sistema operacional

Inicialização do Linux



O que precisa ser preparado?

O que precisa ser preparado?



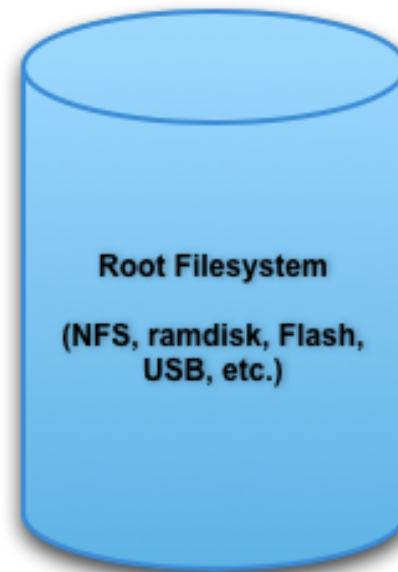
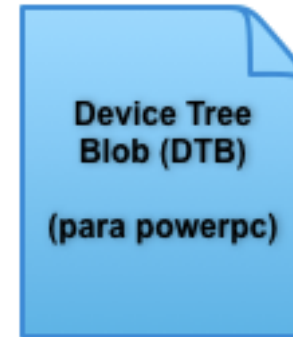
O que precisa ser preparado?



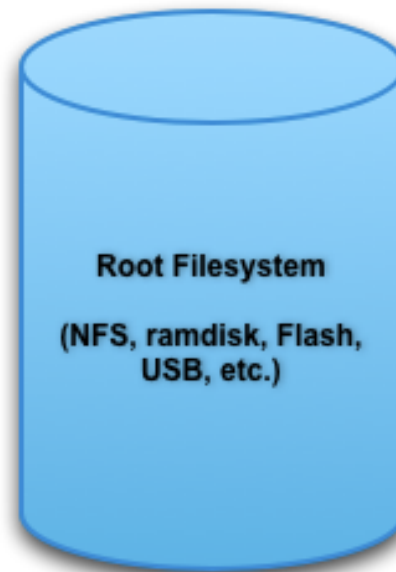
O que precisa ser preparado?



O que precisa ser preparado?



O que precisa ser preparado?



Roteiro

- Introdução
- Projeto de Linux Embarcado
 - Toolchain
 - Bootloader
 - Kernel
 - Root Filesystem
- Demonstração
- Considerações Finais

Toolchain

- Conjunto de ferramentas necessárias para compilar (ou cross-compile) um sistema Linux
- Componentes:
 - Binutils (as, ld, ar, rambld, strip, etc.)
 - Kernel headers (#include <linux/*>, <asm/*>, etc.)
 - Compilador C (GCC)
 - Biblioteca C (glibc, uClibc, dietlibc, etc.)
- Opções para geração de toolchain:
 - Gerar manualmente (compilando cada pacote)
 - Obter uma cópia pré-compilada
 - Utilizar uma ferramenta de geração (crosstool-ng, buildroot, PTXdist, OpenEmbedded, etc.)

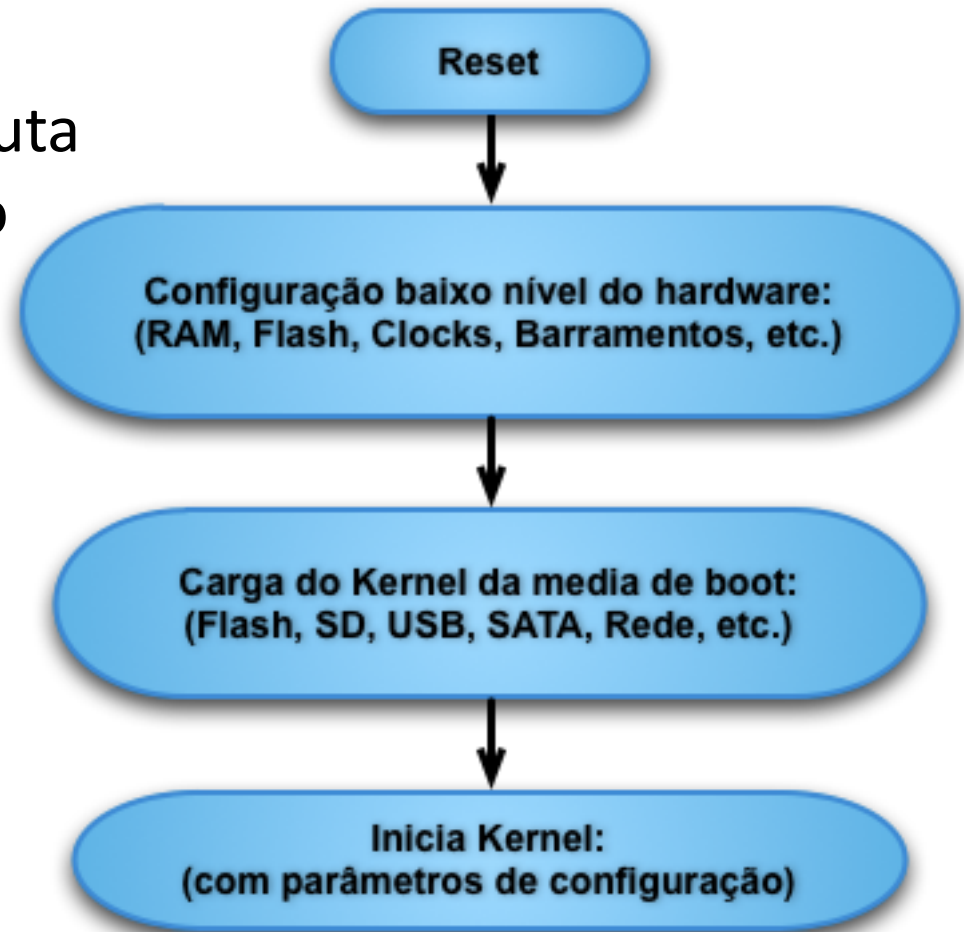


Roteiro

- Introdução
- Projeto de Linux Embarcado
 - Toolchain
 - Bootloader
 - Kernel
 - Root Filesystem
- Demonstração
- Referências

Bootloader

- Primeiro software que executa quando o hardware é ligado
- Responsável por carregar e transferir o controle para o Kernel
- Para isso:
 - Configura o hardware
 - Oferece comandos para configuração e diagnóstico



Bootloaders

List of bootloaders

Bootloader	ARM	BFIN	MIPS	PPC	SH	x86	remarks
APEX	y	-	?	?	?	?	
Blob	y	-	-	-	-	-	
CFE ↗	-	-	y	-	-	-	for specific BroadCom chipsets
coreboot (LinuxBIOS) ↗	-	-	-	-	-	y	Conference talk by Peter Stuge at Embedded Linux Conference Europe 2008, video ↗
Das U-Boot	y	y	y	y	y	y	allows networked setup
U-Boot-v2 ↗	y	y	-	y	-	-	allows networked setup, integrated editor and scripting
Grub	?	-	?	y	?	y	
Lilo	-	-	-	-	-	y	x86 only, requires nasm to build
MicroMonitor ↗	y	y	?	y	y	?	
PMON 2000 ↗	-	-	y	-	-	-	
RedBoot	y	?	y	y	y	y	allows networked setup
Syslinux	-	-	-	-	-	y	variants (including) isolinux are very flexible for booting x86
Yaboot ↗	-	-	-	y	-	-	
YAMON ↗	-	-	y	-	-	-	

Legend: ?:Unknown -:Not supported

Fonte: <http://www.elinux.org/Bootloader>, 13/11/09

Bootloaders

List of bootloaders

Bootloader	ARM	BFIN	MIPS	PPC	SH	x86	remarks
APEX	y	-	?	?	?	?	
Blob	y	-	-	-	-	-	
CFE ↗	-	-	y	-	-	-	for specific BroadCom chipsets
coreboot (LinuxBIOS) ↗	-	-	-	-	-	y	Conference talk by Peter Stuge at Embedded Linux Conference Europe 2008, video ↗
Das U-Boot	y	y	y	y	y	y	allows networked setup
U-Boot-v2 ↗	y	y	-	y	-	-	allows networked setup, integrated editor and scripting
Grub	?	-	?	y	?	y	
Lilo	-	-	-	-	-	y	x86 only, requires nasm to build
MicroMonitor ↗	y	y	?	y	y	?	
PMON 2000 ↗	-	-	v	-	-	-	
RedBoot	y	?	y	y	y	y	allows networked setup
Sylinux	-	-	-	-	-	y	variants (including) isolinux are very flexible for booting x86
Yaboot ↗	-	-	-	y	-	-	
YAMON ↗	-	-	y	-	-	-	

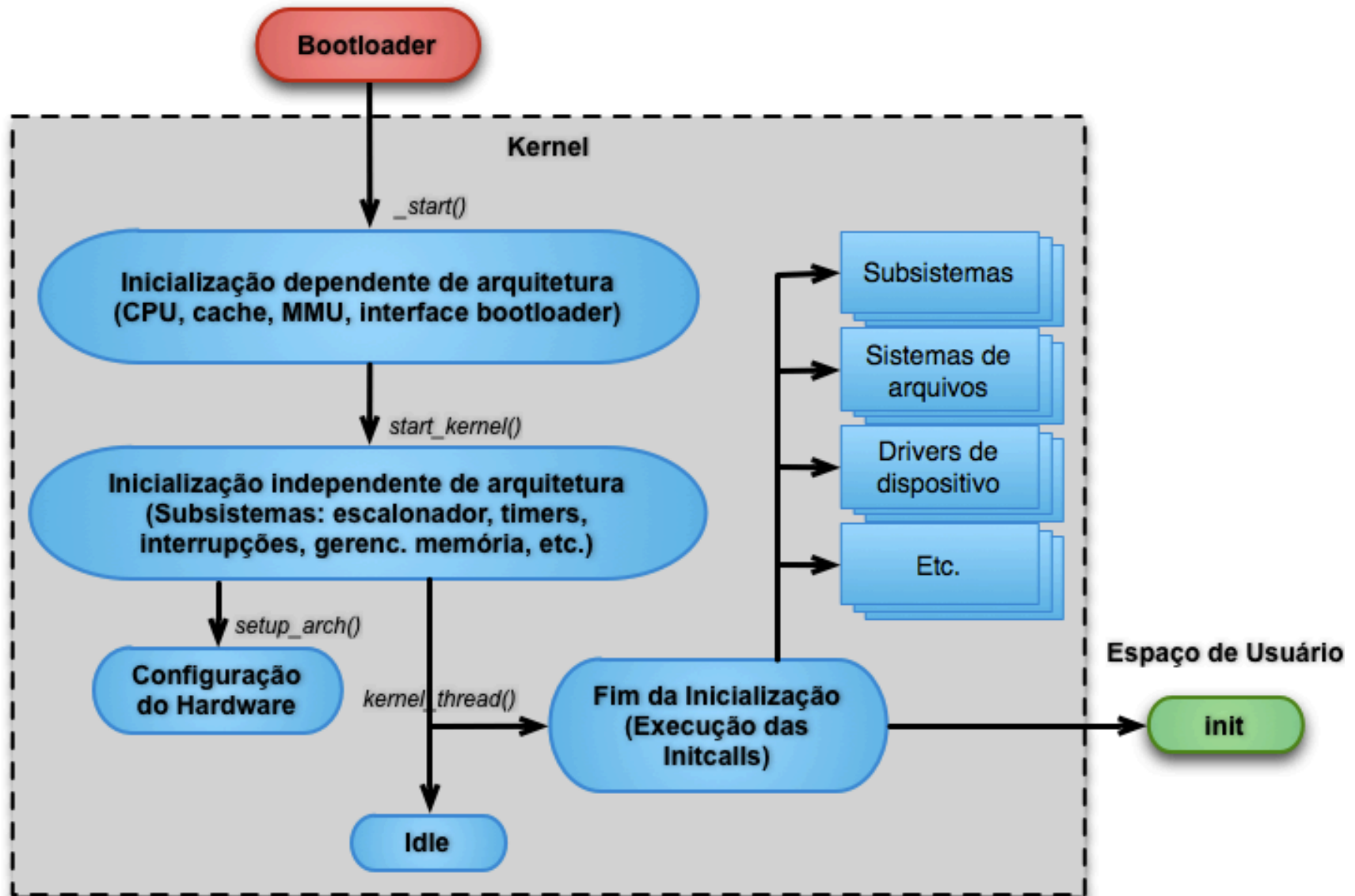
Legend: ?:Unknown -:Not supported

Fonte: <http://www.elinux.org/Bootloader>, 13/11/09

Roteiro

- Introdução
- Projeto de Linux Embarcado
 - Toolchain
 - Bootloader
 - Kernel
 - Root Filesystem
- Demonstração
- Referências

Kernel do Linux



Exemplo: Linux para PowerPC

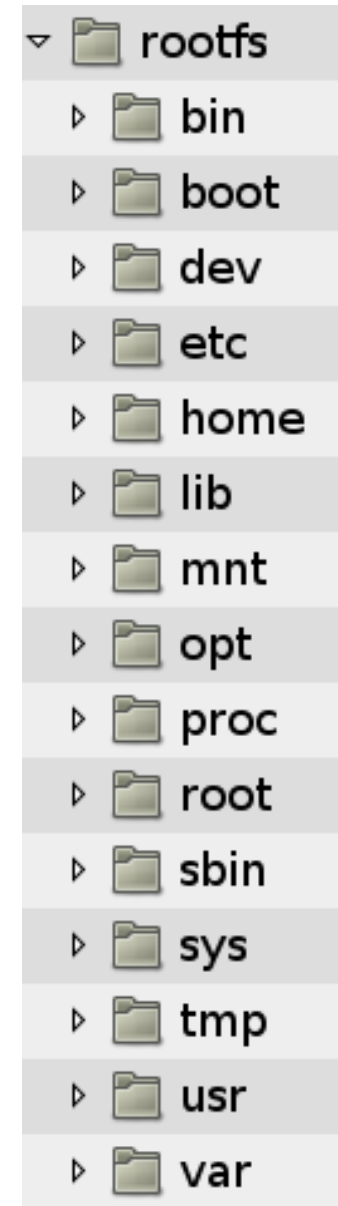
- Porte para uma nova plataforma
- PowerPC:
 - Device Tree Blob (DTB)
 - Arquivo que descreve o hardware
 - Também utilizado pelo bootloader
- Outras arquiteturas:
 - Modificar o código fonte
 - Ex: ARM
 - Modificar/criar arch/arm/mach-[plataforma]/

Roteiro

- Introdução
- Projeto de Linux Embarcado
 - Toolchain
 - Bootloader
 - Kernel
 - Root Filesystem
- Demonstração
- Referências

Root Filesystem

- Árvore de diretórios contendo bibliotecas, scripts, aplicações, etc.
- Opções para geração de roots:
 - Manual (compilando e instalando cada pacote)
 - Utilizando um sistema de build
- Tipos de sistemas de arquivos:
 - Ext2/3
 - Em flash:
 - SquashFS (read-only, rápido)
 - JFFS2 (mais utilizado atualmente)
 - UBIFS (sucessor do JFFS2, rápido)
 - cramfs, initramfs, etc.



Root Filesystem

Ferramentas para geração	License	Small systems	glibc (G) uClibc (U)	Reproducibility / Leverage	Popularity	Actively maintained
Buildroot	Free	Yes	G	Good	Good	Yes
Scratchbox	Free	Yes	G + U	Poor	Low	Yes
OpenEmbedded	Free	No	G + U	Good	Very good	Definitely
LTIB	Free	Yes	G + U	Good	Low	Yes
PTXdist	Free	Yes	G	Good	Low	Yes
Gentoo embedded	Free	No?	G (U?)	?	Low?	Yes
Firmware Linux	Free	Yes	?	Poor	Low	Rob never sleeps
Vendor tools	Closed	?	G + U	Good	N/A	Yes
Home made	Closed	?	?	?	N/A	?

Root Filesystem

Ferramentas para geração	License	Small systems	glibc (G) uClibc (U)	Reproducibility / Leverage	Popularity	Actively maintained
Buildroot	Free	Yes	G	Good	Good	Yes
Scratchbox	Free	Yes	G + U	Poor	Low	Yes
OpenEmbedded	Free	No	G + U	Good	Very good	Definitely
LTIB	Free	Yes	G + U	Good	Low	Yes
PTXdist	Free	Yes	G	Good	Low	Yes
Gentoo embedded	Free	No?	G (U?)	?	Low?	Yes
Firmware Linux	Free	Yes	?	Poor	Low	Rob never sleeps
Vendor tools	Closed	?	G + U	Good	N/A	Yes
Home made	Closed	?	?	?	N/A	?

Fonte: Embedded building tools BOF, Thomas Petazzoni (ELC 2009) 27

Roteiro

- Introdução
- Projeto de Linux Embarcado
- **Demonstração**
- Referências

Demonstração na Placa



- Roteador 3G da Digitel – NRX 5100
- Processador PowerPC de 333 MHz
- 64 MB de memória RAM e 32 MB de Flash

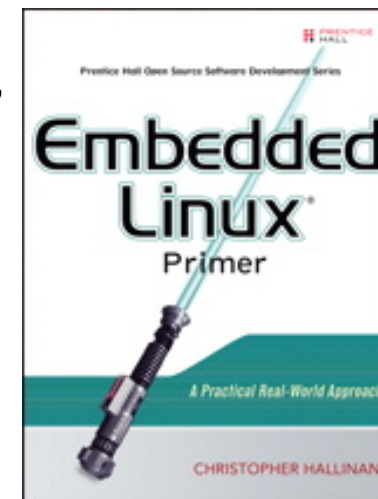
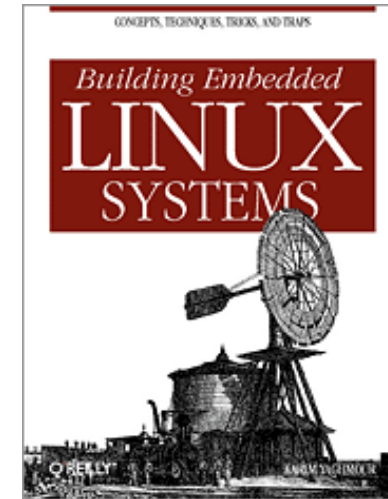
- Demonstração:
 - Boot do Linux Embarcado via rede (protocolo TFTP)
 - Configuração da rede e dos parâmetros de boot do Kernel:
 - Acesso ao equipamento, servidor Web, sistemas de arquivos, etc.

Referências

- Embedded Linux Conference
 - <http://www.embeddedlinuxconference.com/>
 - Apresentações (ELC 2009):
 - It's Alive! - Linux on Embedded PowerPC porting guide, Grant Likely
 - Update on filesystems for flash storage, Michael Opdenacker
 - Embedded Linux and Mainline Kernel, David Woodhouse

Referências

- Livros:
 - Building Embedded Linux Systems, Karim Yaghmour
 - Embedded Linux Primer: A Practical Real-World Approach, Christopher Hallinan



Referências

- Embedded Linux Wiki
 - <http://www.elinux.org>
- Free Electrons
 - <http://free-electrons.com/docs/>

Linux em Sistemas Embarcados

Marcelo Veiga Neves
marcelo.veiga@gmail.com

SACTA 2010
Alegrete, 25 de junho de 2010