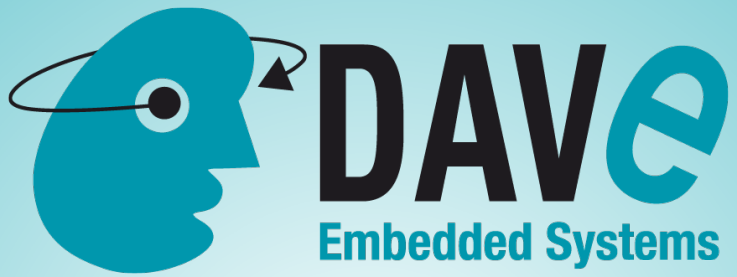


OSD 2014
Sviluppare con Linux Embedded

RELATORE: Andrea Scian
andrea.scian@dave.eu

DATA: 29 Novembre 2014

Chi vi parla

Biografia

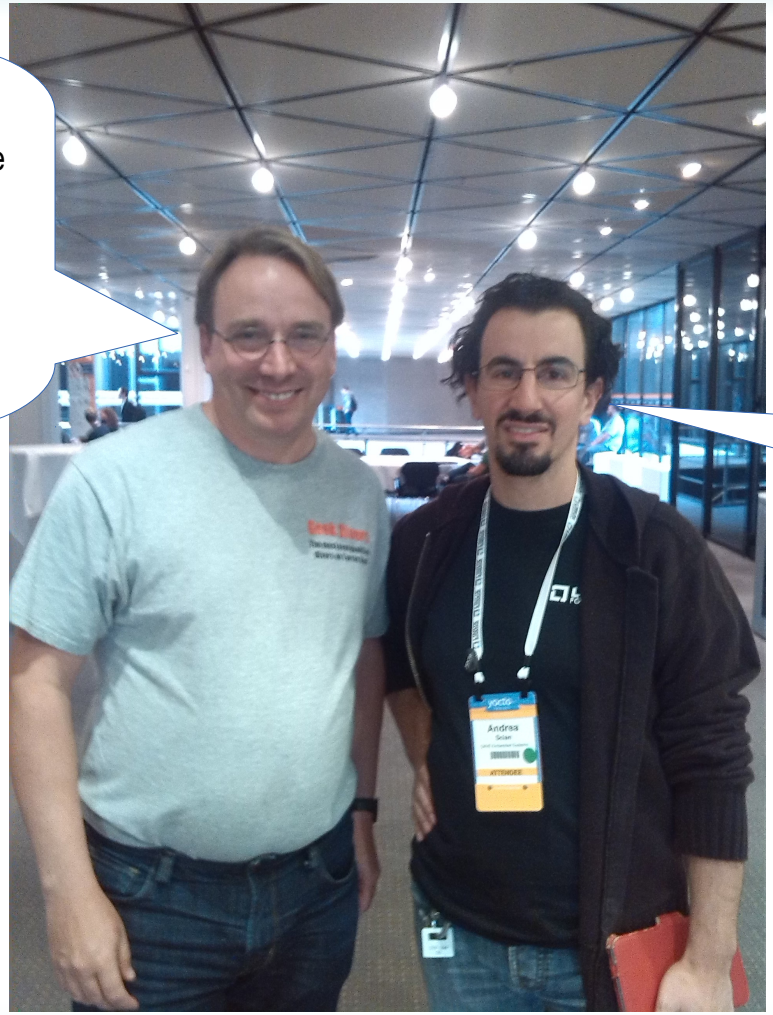
- Perito informatico all'ITIS Kennedy (Pordenone, 1998)
- Laureto in Ingegneria Informatica a Padova (2003)
- Tesi sull'utilizzo di Java nel mondo Linux Embedded (ARM7TDMI, NOMMU, 16MB RAM, 4MB NOR Flash, 32MB NAND Flash)
- Sviluppo software per Linux (e non solo) Embedded (e non solo) dal 2003
- Principale esperienza in architetture ARM (ARMv7) ma anche PowerPC (e x86)

Ora responsabile degli sviluppi software e responsabile IT c/o DAVE Embedded Systems
Appassionato del mondo Linux e FLOSS
Socio PNLUG (www.pnlug.it)

DAVE
Embedded Systems

Con un certo tipo di conoscenze...

I still really despise the absolute incredible sh*t that is non-discoverable buses, and I hope that ARM SoC hardware designers all die in some incredibly painful accident.



I totally agree with you, Linus (I'm a software guy)

DAVE Embedded Systems

Keyword

Realtime

Terminal Emulator

i.MX6

Root file system

Linux Embedded

JTAG

TFTP

Bridged Network

CortexA9-MPCore

Cross-Toolchain

GIT

SSH/SFTP

GDB

AMP

Yocto

Virtual Machine

NFS

Eclipse IDE

DAVE Embedded Systems

Agenda

- Linux Embedded
  - Cos'è?
  - Componenti Software
    - Target
    - Sistema di sviluppo
- (Sistemi Linux AMP)

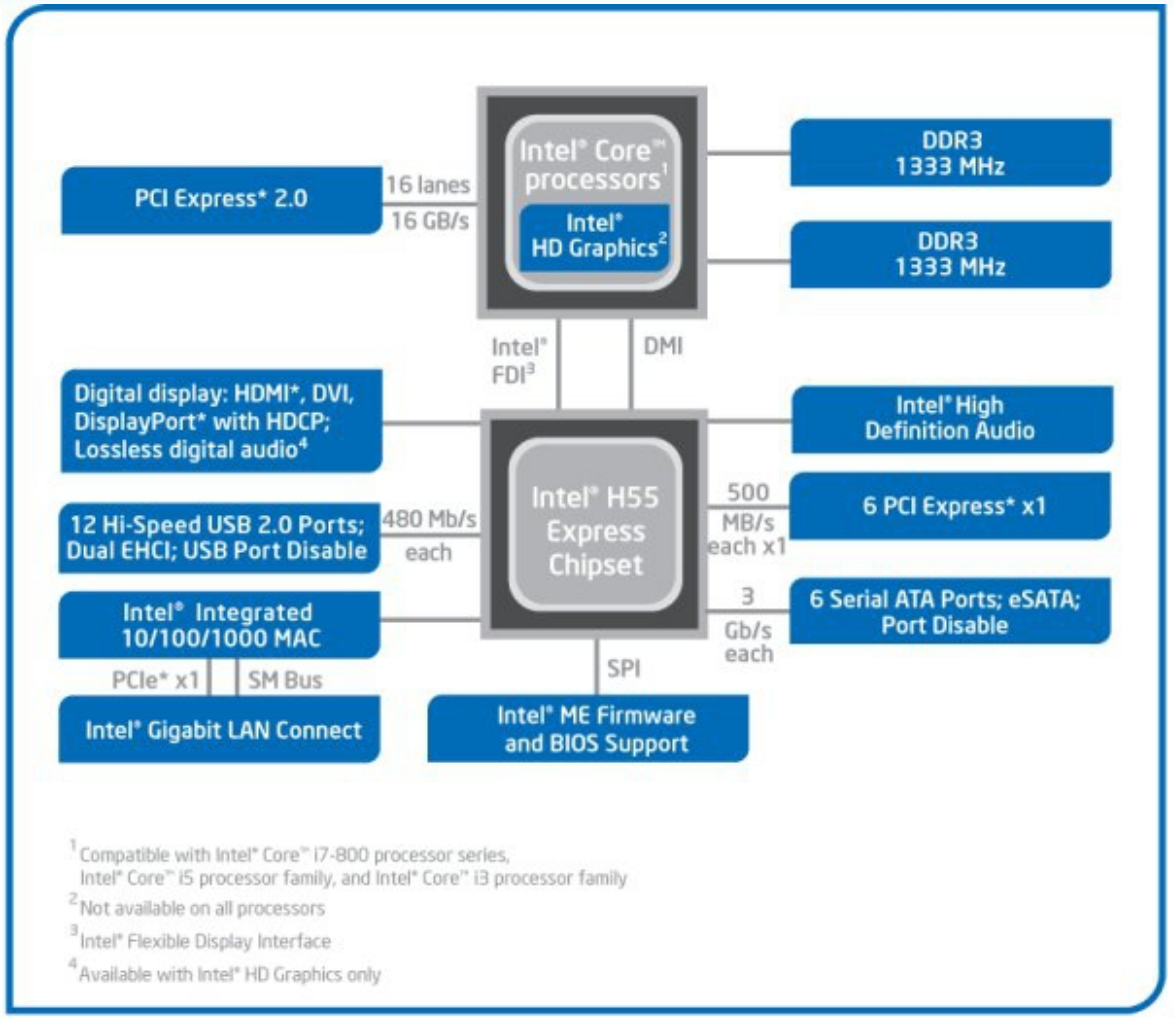
DAVE Embedded Systems

Sistema Embedded: cos'è?

- General purpose vs embedded
- Ha un compito predefinito già durante la progettazione
- SOC non x86
- Consumi, dimensioni, costo, range temperatura, immunità ai disturbi, durata, reperibilità ...

DAVe Embedded Systems

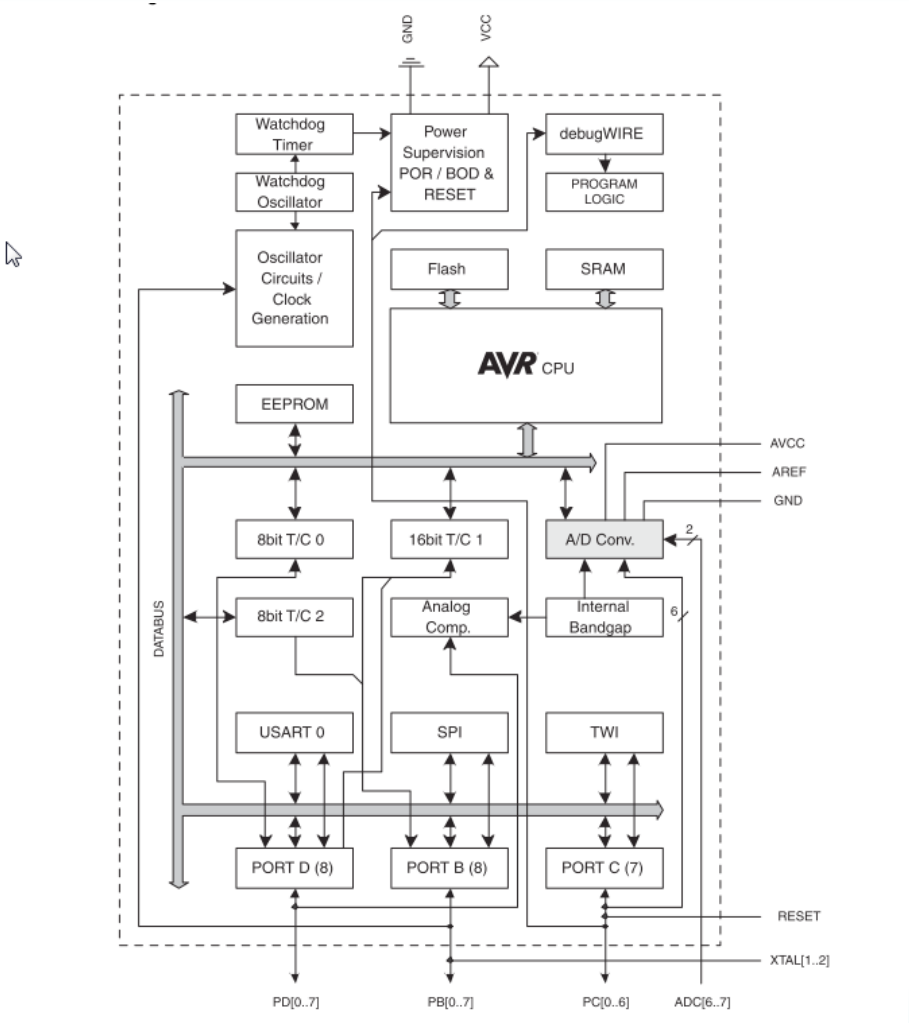
Cosa NON è un SOC



Intel® H55 Express Chipset Platform Block Diagram

DAVE Embedded Systems

Cosa è un SOC (1)



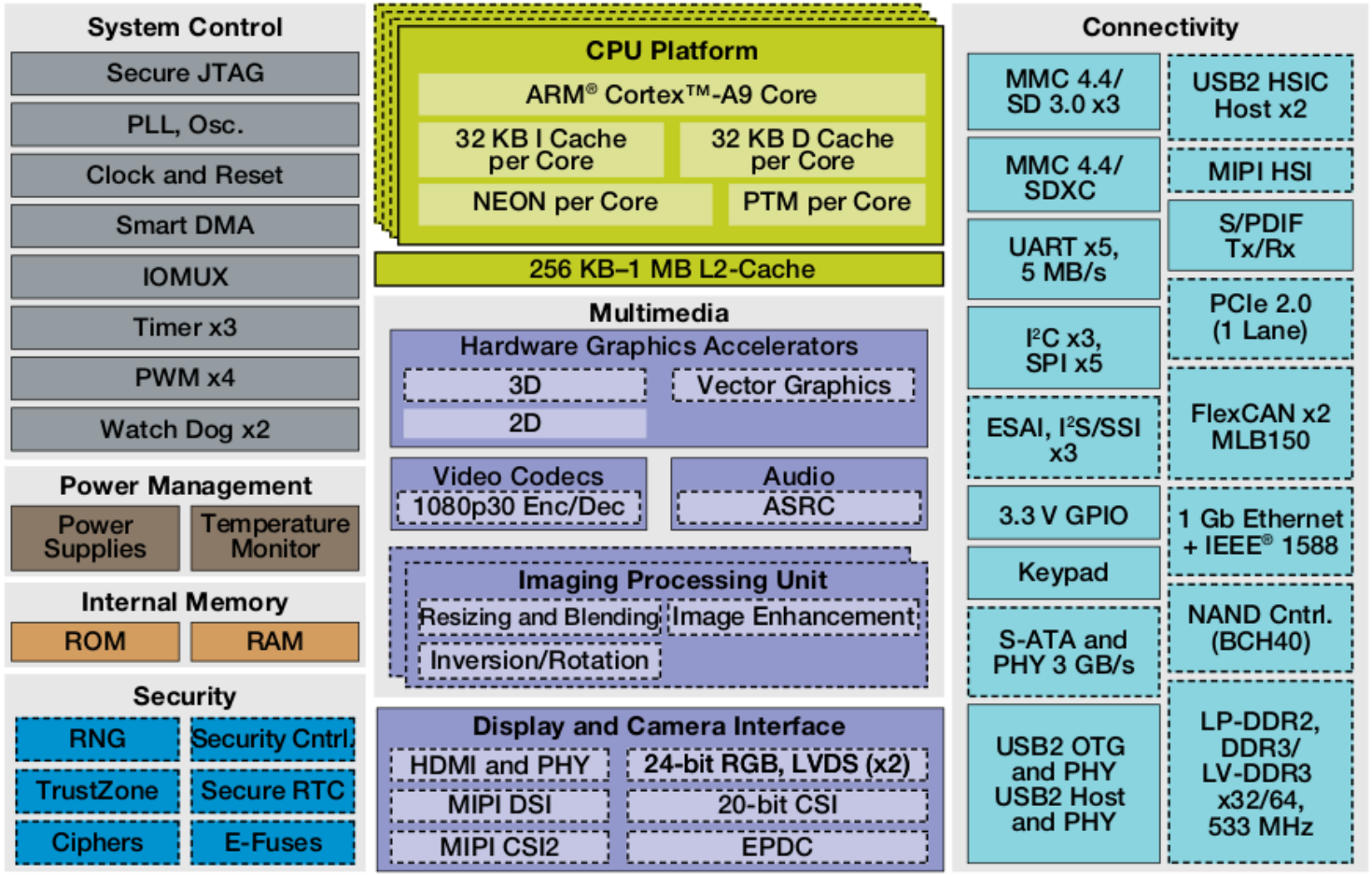
ARDUINO

ATMEL

http://www.atmel.com/devices/ATMEGA328P.aspx

DAVE Embedded Systems

Cosa e' un SOC (2)

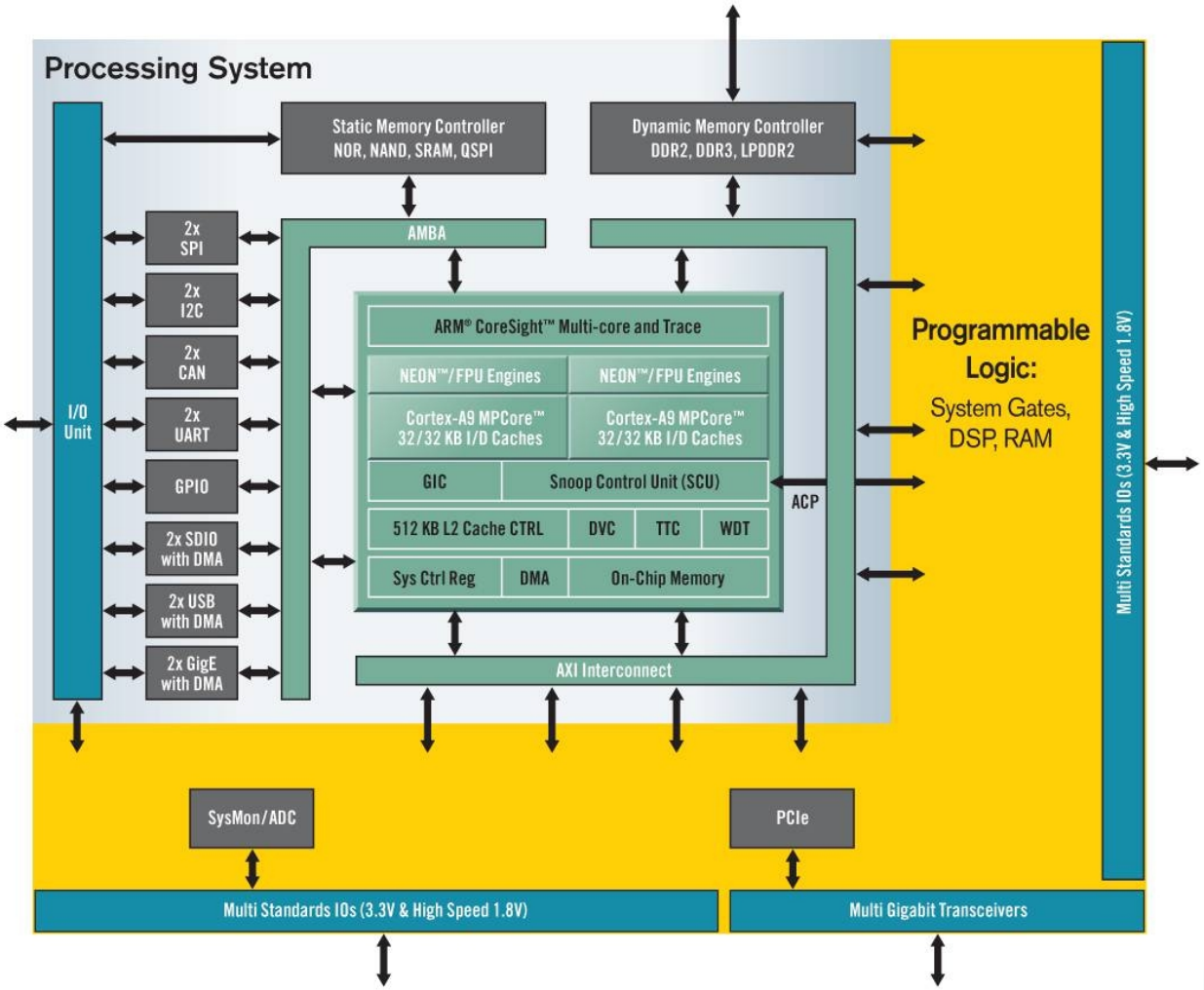


freescale™ semiconductor

DAVE Embedded Systems

www.freescale.com/webapp/sps/site/taxonomy.jsp?code=IMX6X_SERIES

Cosa è un SOC (3)



XILINX
ALL PROGRAMMABLE™

http://www.xilinx.com/products/silicon-devices/soc/zynq-7000.html

DAVE Embedded Systems

Periferiche

- Strettamente embedded
  - I2C, SPI, GPIOs, UART, PWM
  - Usabili anche in baremetal: un driver per pilotare una manciata di registri (easy)
- Periferiche “general purpose” (ma comunque usate)
  - USB, SATA, PCIe, Ethernet, Wi-Fi/Bluetooth
  - È necessario avere una base software di partenza molto più ampia dei semplici driver (già complessi di suo) forniti dal produttore (leggi: sistema operativo)
- Per non parlare poi di:
  - Storage/file systems, Networking, GUI evolute, 3D, video streaming/encoding/decoding Full-HD, audio processing, multitouch capacitivi

DAVE Embedded Systems

Linux Embedded

- Embedded Linux: utilizzo di Linux in sistemi embedded
  - Vantaggi: qualità, riuso del software, community, controllo (OSS)
  - Svantaggi: OS complesso, setup ambiente di sviluppo, mancanza di uniformità

DAVE Embedded Systems

Componenti Software

- ROM (fuso nel SOC)
- Bootloader (U-Boot)
  - Inizializzazione di base (DDRx, PLL, GPIO)
  - Preleva il kernel (NOR/NAND/TFTP) e gli cede il controllo
- Kernel (vendor dependent)
  - Scheduler, gestione memoria, device drivers ecc
- Filesystem
  - Contiene tutti gli applicativi/file di configurazione ecc
  - NON è possibile farne a meno!
  - Ramdisk, NOR/NAND, SSD, HDD
  - Ext2-3-4, JFFS2/UBIFS, F2FS, ROMFS, NFS

Componenti Software

- Bootloader (U-Boot)
  - Inizializzazione di base (DDRx, PLL, GPIO)
  - Preleva il kernel (NOR/NAND/TFTP) e gli cede il controllo
- Kernel (vendor dependent)
  - Scheduler, gestione memoria, device drivers ecc
- Filesystem
  - Contiene tutti gli applicativi/file di configurazione ecc
  - NON è possibile farne a meno!
  - Ramdisk, NOR/NAND, SSD, HDD
  - Ext2-3-4, JFFS2/UBIFS, F2FS, ROMFS, NFS

Das U-Boot

- http://www.denx.de/wiki/U-Boot
- Iniziato a sviluppare per PowerPC a fine 1999
- 68k, ARM, AVR32, Blackfin, MicroBlaze, MIPS, Nios, PPC and x86
- Karim Yaghmour: "Though there are quite a few other bootloaders, 'Das U-Boot,' the universal bootloader, is arguably the richest, most flexible, and most actively developed open source bootloader available."
- http://www.barebox.org/ (formely known as u-boot-v2)

DAVE Embedded Systems

Cosa fa il bootloader?

- Inizializzazione di base (DDRx, PLL, GPIO)
- Preleva il kernel (da qualche parte...) e gli cede il controllo

TUTTO QUI?

Das U-Boot: the full story

- Dual stage (SPL)
- Stack TCP/IP
- File system/Partitioning (ext, fat32, jffs2, ubifs)
- Stack USB (host/device)
- SD
- I2C
- Serial/net console
- Command line scripting
- Environment (redundand)
- NAND, NOR, SPI NOR, EEPROM
- Update di tutte le immagini

```
v2013.04
1.774.449 linee di codice in 6046 file
```

DAVE Embedded Systems

Componenti Software

- Bootloader (U-Boot)
  - Inizializzazione di base (DDRx, PLL, GPIO)
  - Preleva il kernel (NOR/NAND/TFTP) e gli cede il controllo
- Kernel (vendor dependent)
  - Scheduler, gestione memoria, device drivers ecc
- Filesystem
  - Contiene tutti gli applicativi/file di configurazione ecc
  - NON è possibile farne a meno!
  - Ramdisk, NOR/NAND, SSD, HDD
  - Ext2-3-4, JFFS2/UBIFS, F2FS, ROMFS, NFS

DAVe Embedded Systems

Linux kernel

- http://en.wikipedia.org/wiki/Linux_kernel
- Creato nel 1991 da uno studente finlandese (Linus Torvalds)
  - “I'm doing a (free) operating system (just a hobby, won't be big and professional like gnu)[...]. It is NOT portable (uses 386 task switching etc), and it probably never will support anything other than AT-harddisks, as that's all I have.”
- 0.01: 10.239 LOC
- Clone di Minix (Tanenbaum) GPL per i386
- Dalla release 0.11 pote' compilare se stesso
- Aggiunti i tool GNU (1983) divenne un vero e proprio OS
- Kernel + GNU + packaging => **distro**

Linux kernel: i numeri

Version	Date	CSets	Devs	Lines (thousands)		
				Added	Removed	Delta
3.15	Jun 8	13,722	1492	1066	707	360
3.16	Aug 3	12,804	1478	578	329	249
3.17	Sep 28*	12,153	1408	692	708	-16
3.15–17		38,679	2546	2336	1744	593

http://lwn.net/Articles/613006/

```
bash# git clone →
git://git.kernel.org/pub/scm/linux/kernel/git/torvalds/linux.git
bash# cd linux
bash# find ./ -name '*.[hcS]' | wc -l
38.367
bash# find . -name "*.[hcS]" -not -regex '\./\.git.*' | xargs cat | wc
-l
17.588.409
```

DAVE Embedded Systems

Linux kernel: cosa fa?

- Device driver
- Scheduler
- Memory Management
- File system
- I/O
- ...

DAVE Embedded Systems

Linux kernel (embedded)

- Board files (historical)
  - Linus Torvalds, March 2011: “Gaah. Guys, this whole ARM thing is a f*cking pain in the ass”
  - Passaggio a device tree (e' un bene?)
- No MMU (uCLinux)
- Build-in vs modules
- Supporto bus/dispositivi particolari (e.g. MTD)
- No virtualizzazione

DAVE Embedded Systems

Componenti Software

- Bootloader (U-Boot)
  - Inizializzazione di base (DDRx, PLL, GPIO)
  - Preleva il kernel (NOR/NAND/TFTP) e gli cede il controllo
- Kernel (vendor dependent)
  - Scheduler, gestione memoria, device drivers ecc
- Filesystem
  - Contiene tutti gli applicativi/file di configurazione ecc
  - NON è possibile farne a meno!
  - Ramdisk, NOR/NAND, SSD, HDD
  - Ext2-3-4, JFFS2/UBIFS, F2FS, ROMFS, NFS

Root file system

- Storage
  - HDD, USB
  - NAND, NOR
- File system type
  - EXT, UBIFS, SQUASHFS, ROMFS, ramdisk
- Contiene almeno 10-100 tools
- Tools di base (ls, cat, dd.. bash!): BusyBox
- File di configurazione

DAVE Embedded Systems

Come crearlo

- DIY (Do It Yourself)
- Buildroot
- Yocto
- (Android)
- Ma anche:
  - Ubuntu, Gentoo, Arch →non per Embedded (IMHO)

DAVE Embedded Systems

DIY

- Creazione struttura RFS “a mano”
  - ok ... scripting!
- Cross-compilazione singoli package
- Totale controllo del risultato (dimensione, velocità di boot)
- Inapplicabile (IMHO) per sistemi complessi (gui, video)

DAVE Embedded Systems

Buildroot

- http://buildroot.uclibc.org/
- Semplice e leggero
- Manca di funzionalità avanzate (QA, gestione licenze, autobuilder)

DAVE Embedded Systems

Yocto Project

- “It's not an embedded Linux distribution – it creates a custom one for you” - http://yoctoproject.org/
- Plus:
  - Tool avanzati (ADT per sviluppo applicazioni in Eclipse)
  - Packaging
  - QA, licensing, SDK...
- Minus
  - Complesso da maneggiare (BSP)
  - Size/tempo di build (10-100GiB, 2-10 ore)

More on this later...

DAVE Embedded Systems

Android (AOSP)

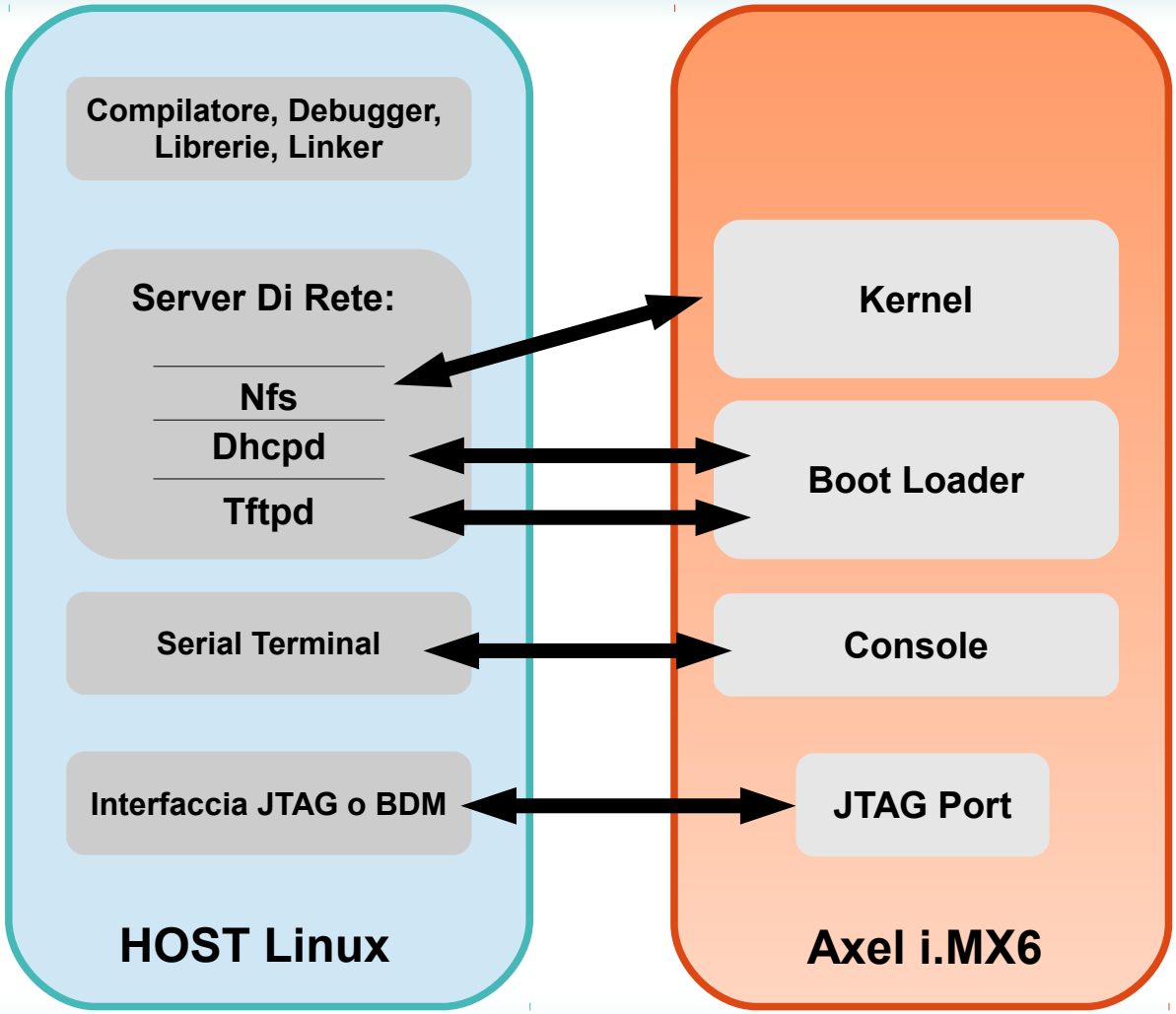
- https://source.android.com/
- Più che un sistema di build, è una distribuzione ad-hoc
- Plus:
  - sviluppo applicativi in Java indipendenti dalla piattaforma
- Minus:
  - Se manca il supporto del silicon vendor ...
  - dispositivi non contemplati (Ethernet!) richiedono modifiche a (praticamente) tutti i livelli dello stack
  - cambiano molte cose (bionic vs libc, missing busybox...)
- Google (Plus? Minus?)

Componenti Software (devel)

- Text Editor
- (Cross) Toolchain
- Debugger
  - Gdb/Gdbserver, JTAG
- IDE (Eclipse)
- Sistema di build (DIY, Buildroot, Yocto, Android)
- Ethernet/Seriale
- **TFTPD, NFSD**, [Samba, SSH]
- **Linux based workstation**

DAVE Embedded Systems

Ambiente di sviluppo tipico



DAVE Embedded Systems

Set up del sistema di sviluppo

- Una lunga lista di attività necessarie prima di poter essere operativi...
  - [x] Installare la distribuzione Linux raccomandata dal silicon vendor (per non incorrere in problemi di compatibilità)
  - [x] Installare tutti i pacchetti software richiesti (tool, utility, librerie, ...)
  - [x] Installare la cross-toolchain
  - [x] Installare e configurare i servizi di rete (tftp, dhcp, NFS, ...)
  - [x] Scaricare i tree dei sorgenti
  - [x] Configurare le variabili di ambiente
  - [ ] ...
- ... che richiede tempo, competenze e risorse.

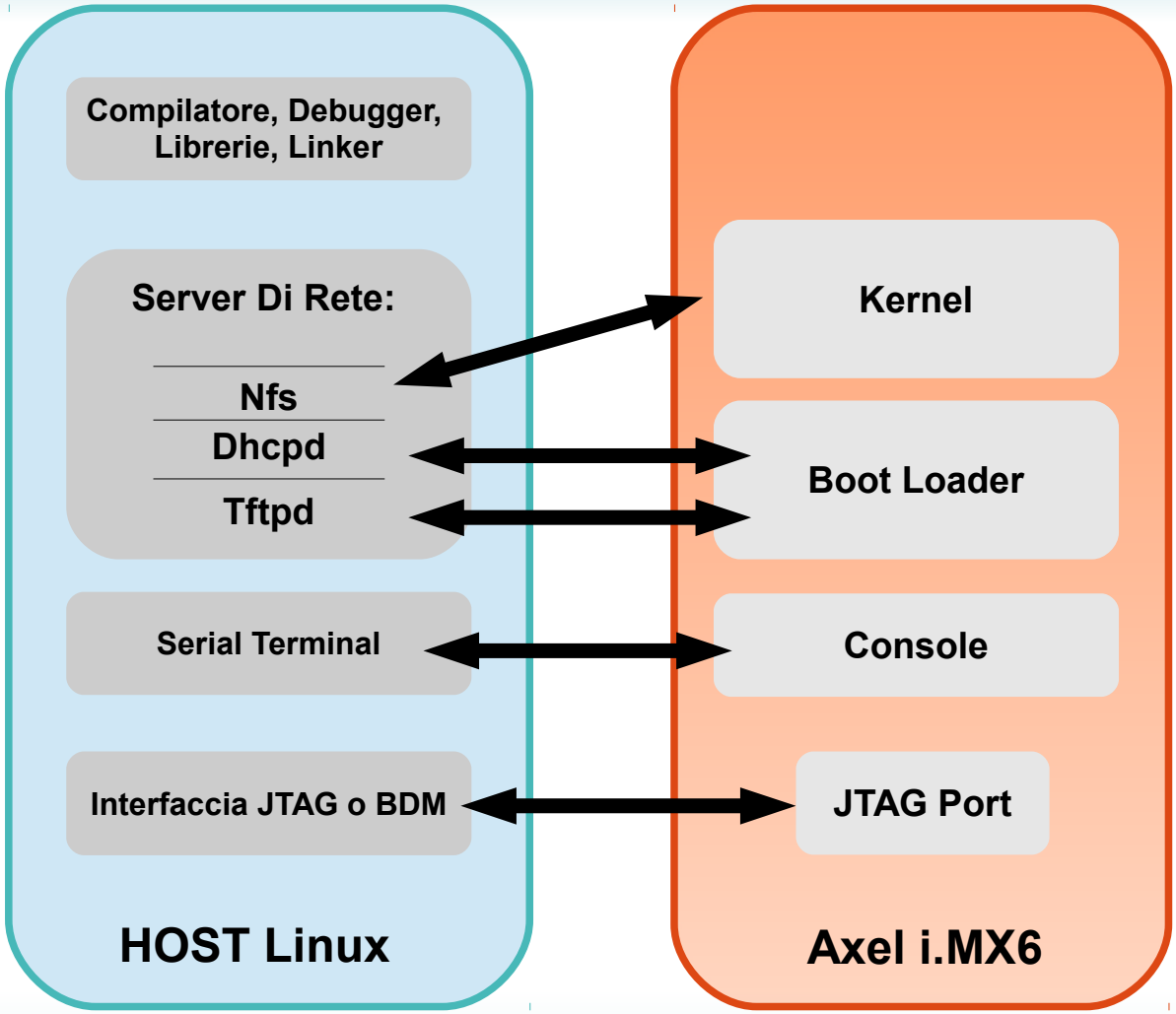
DAVE Embedded Systems

DVDK - DAVE Virtual Development Kit

- Cos'è?
  - VM Virtual Box con il development kit pre-installato
  - Setup mimino (networking)
  - Utilizzo da Windows/OSX/Linux (indipendenza dalla distribuzione/setup)
  - Obiettivo: minimizzare gli sforzi del cliente per il setup dell'ambiente di sviluppo
- Git preconfigurato per aggiornamenti → trasferimenti in rete ridotti

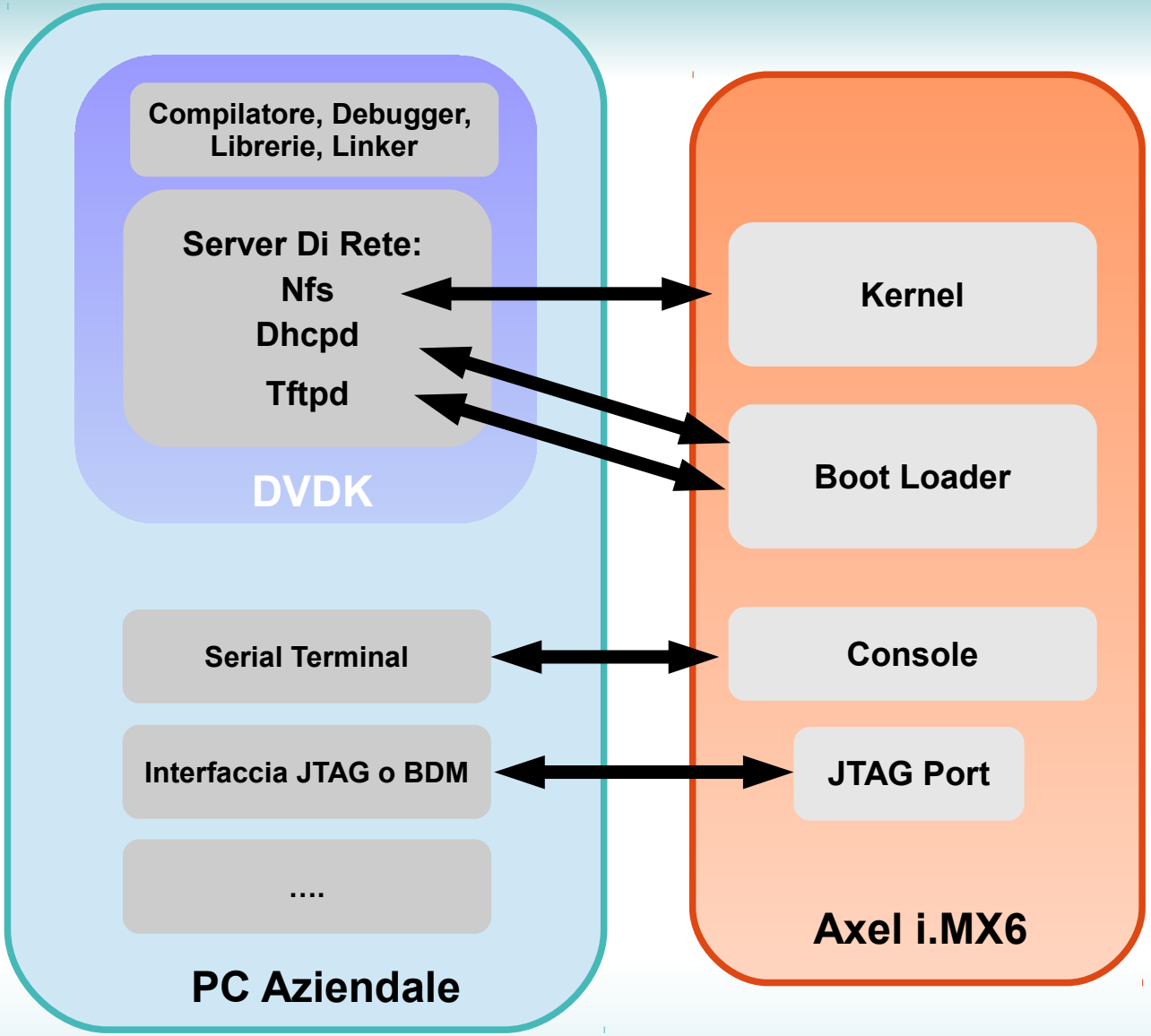
DAVE Embedded Systems

Senza DVDK



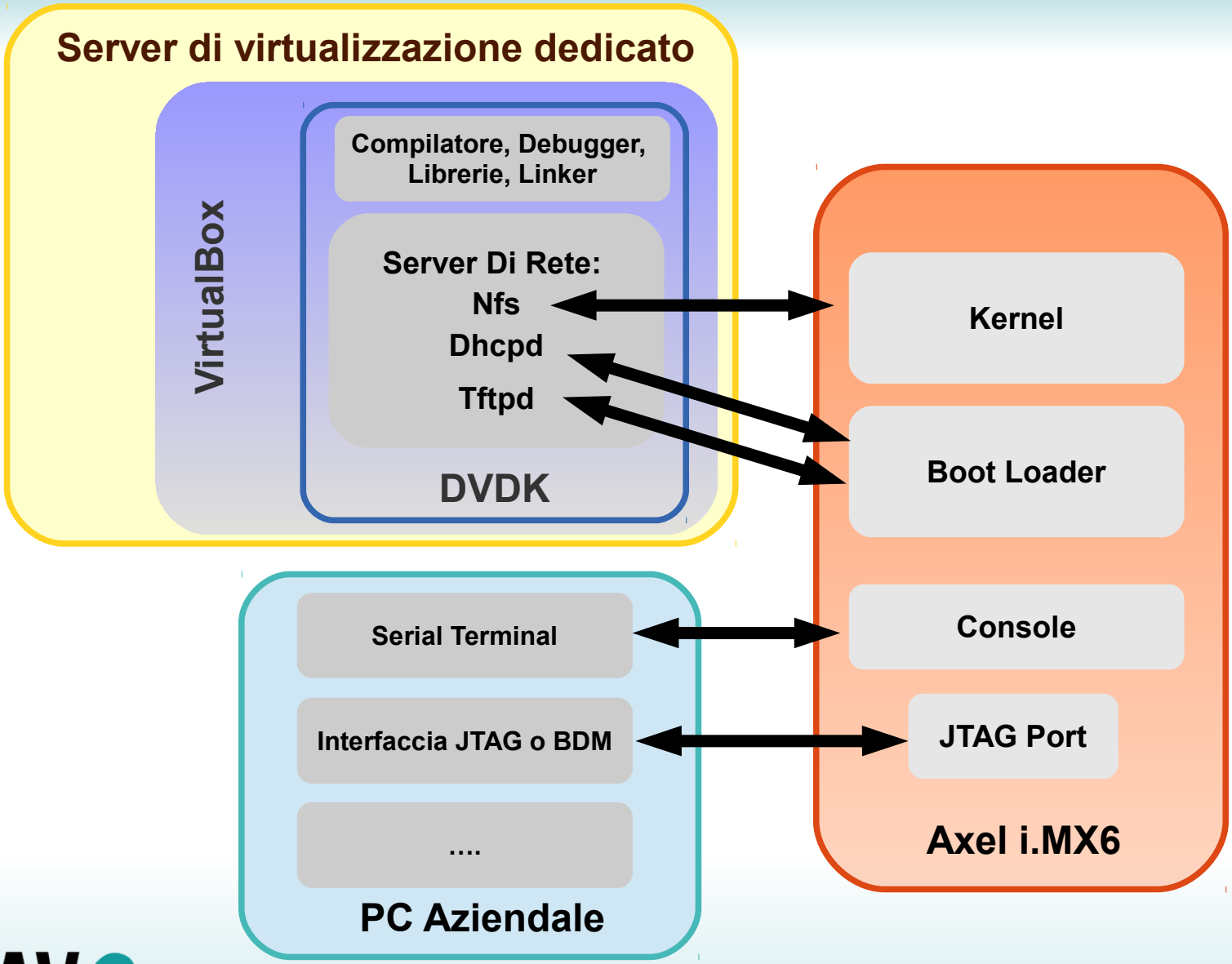
DAVE Embedded Systems

Ambiente di sviluppo DVDK



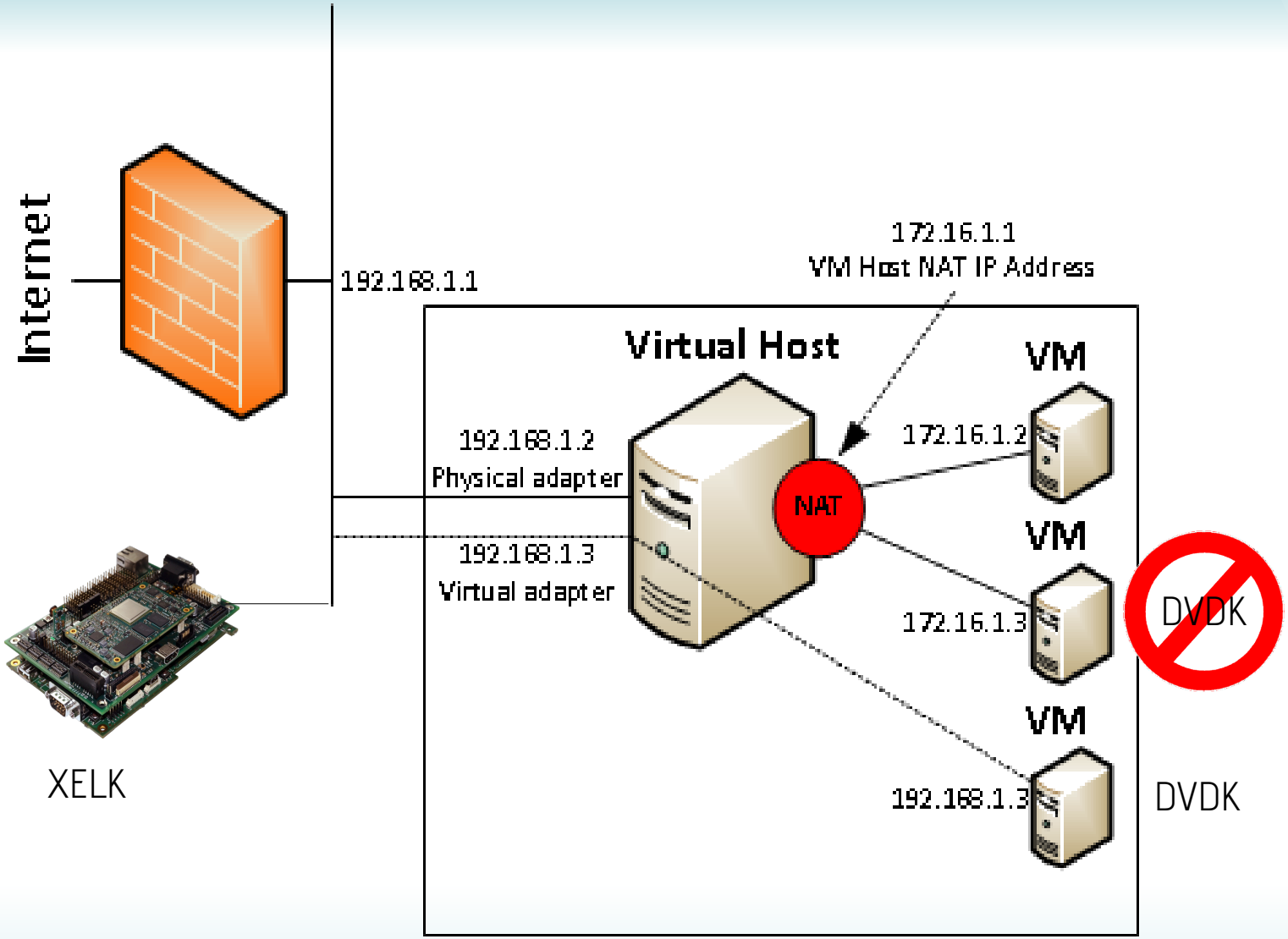
DAVE Embedded Systems

Ambiente di sviluppo DVDK (avanzato)



DAVE Embedded Systems

NAT vs Bridge



DAVE Embedded Systems

References

- http://wiki.dave.eu
- http://wiki.dave.eu/index.php/Category:Linux
- http://free-electrons.com/docs
- Training http://free-electrons.com/doc/training/embedded-linux/
- http://elinux.org
- http://lwn.net (if you're not subscriber, please do so!)
- Building Embedded Linux Systems, By Karim Yaghmour, O'Reilly Media

DAVE Embedded Systems

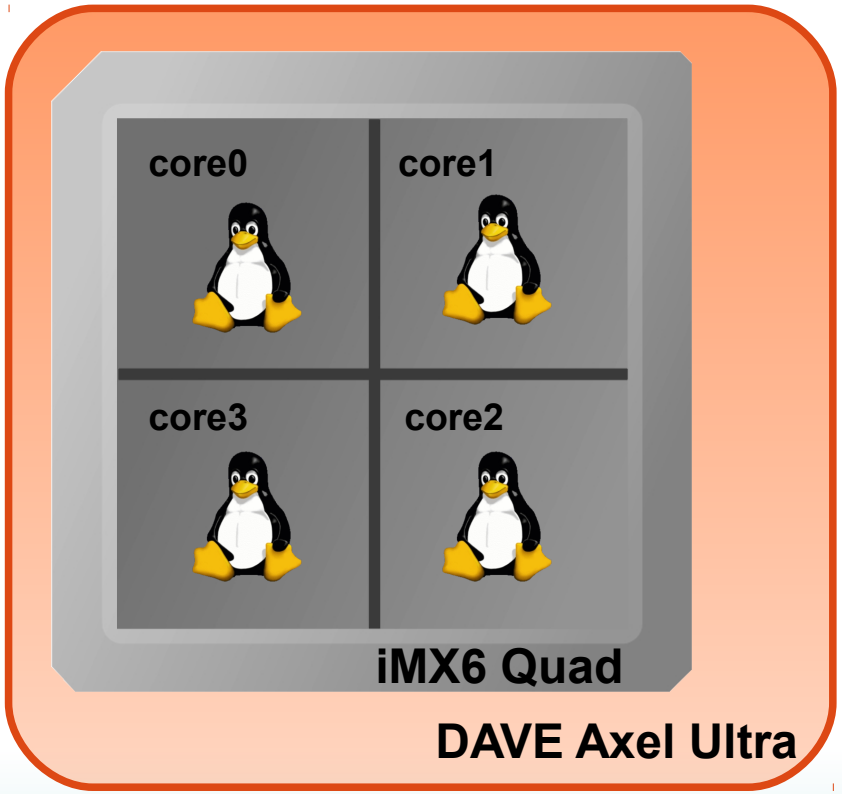
Linux AMP

- SMP vs AMP
  - SMP: Simmetric MultiProcessing
    - 2+ processori omogenei sotto il controllo di un unico OS
    - Risorse condivise
    - I processi (tipicamente) girano indifferentemente su uno dei core (taskset)
  - AMP: Asymmetric MultiProcessing
    - 2+ processori, anche eterogenei (bit.LITTLE)
    - Risorse condivise parzialmente (TrustZone)
    - Possibilità di avere piu' di un OS (o OS + baremetal)
    - Hypervisor/Monitor

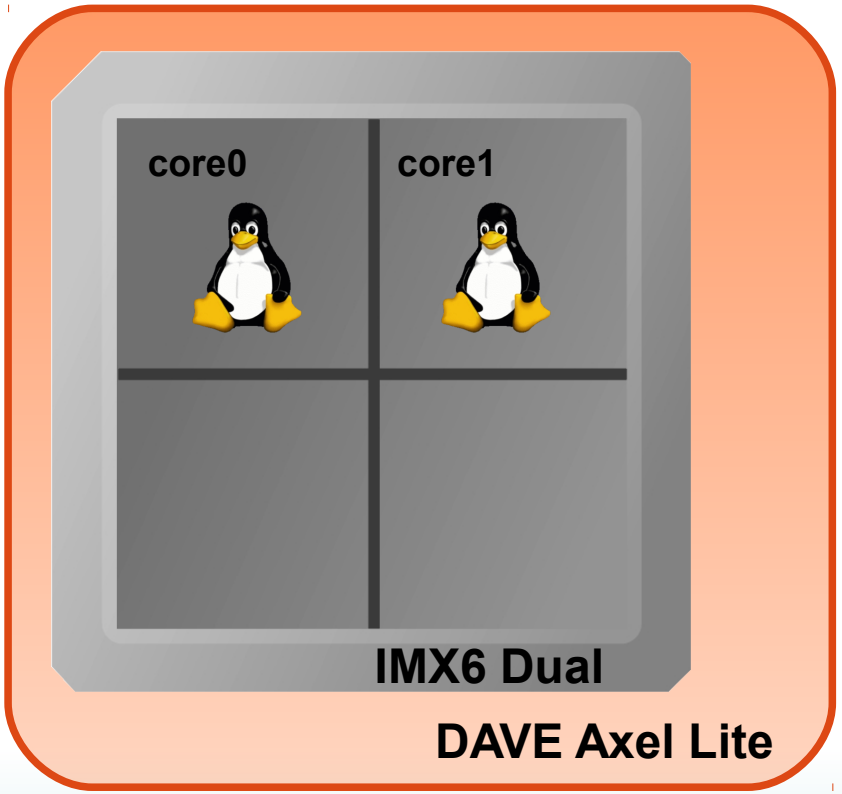
DAVE Embedded Systems

Linux SMP

Linux in esecuzione su 4 core Omogenei



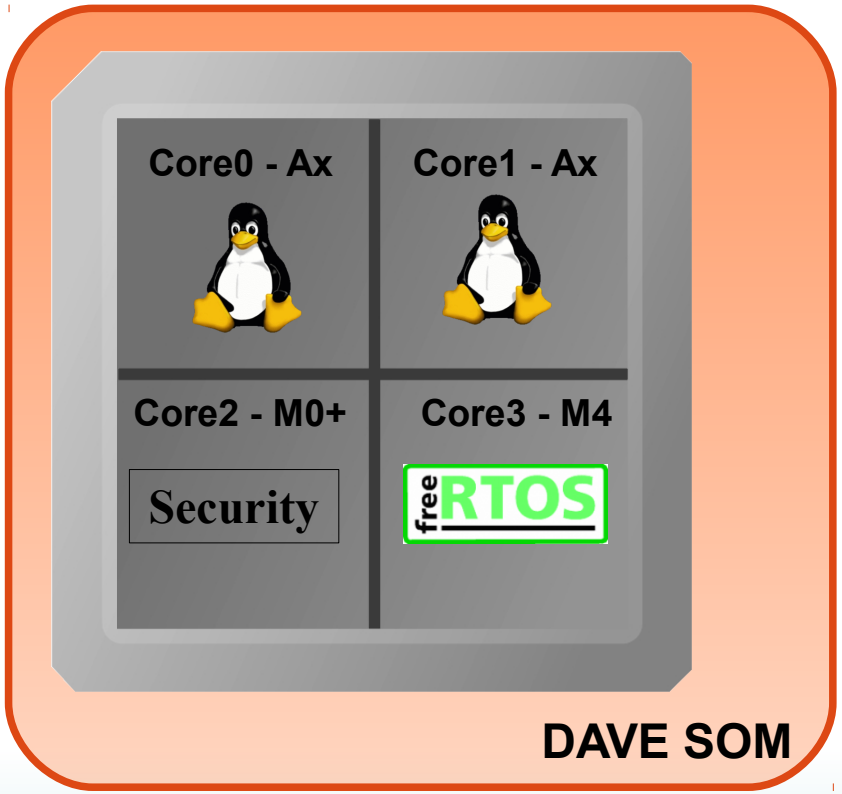
Linux in esecuzione su 2 core Omogenei



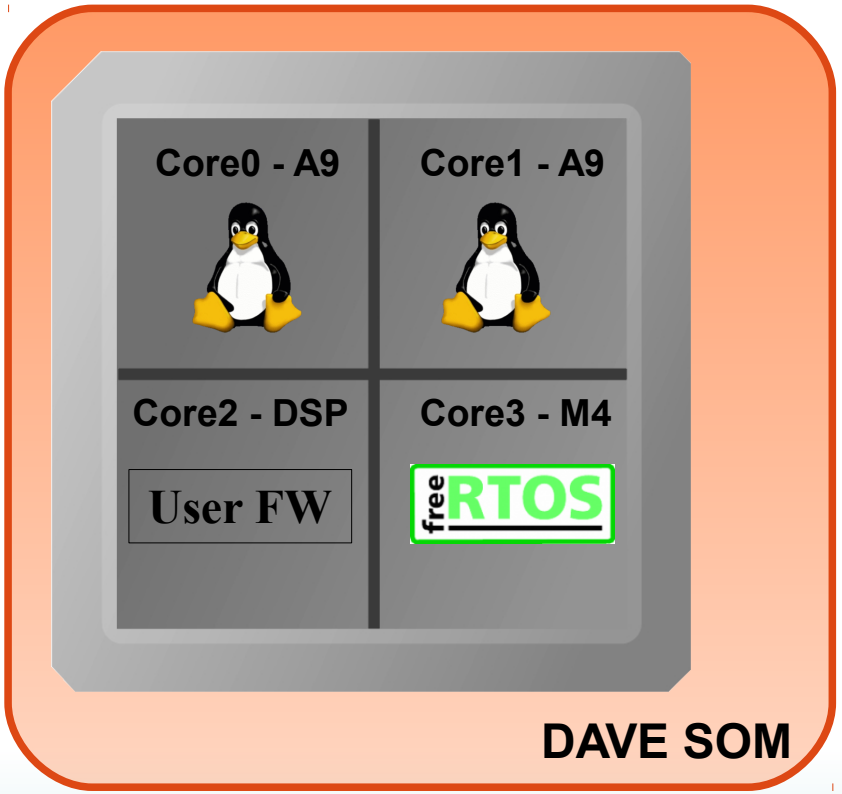
DAVE Embedded Systems

Linux AMP – Core Eterogenei

Linux in esecuzione su 2 core Ax
RTOS M4, Security FW su M0+



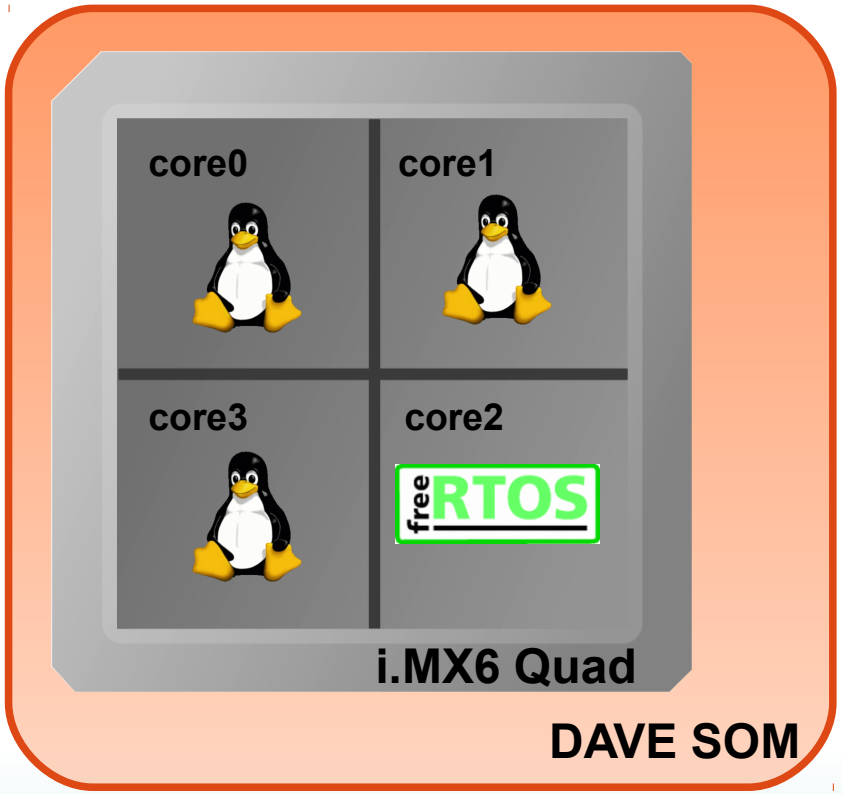
Linux in esecuzione su 2 core
Omogenei (A9) – custom firmware
Su DSP



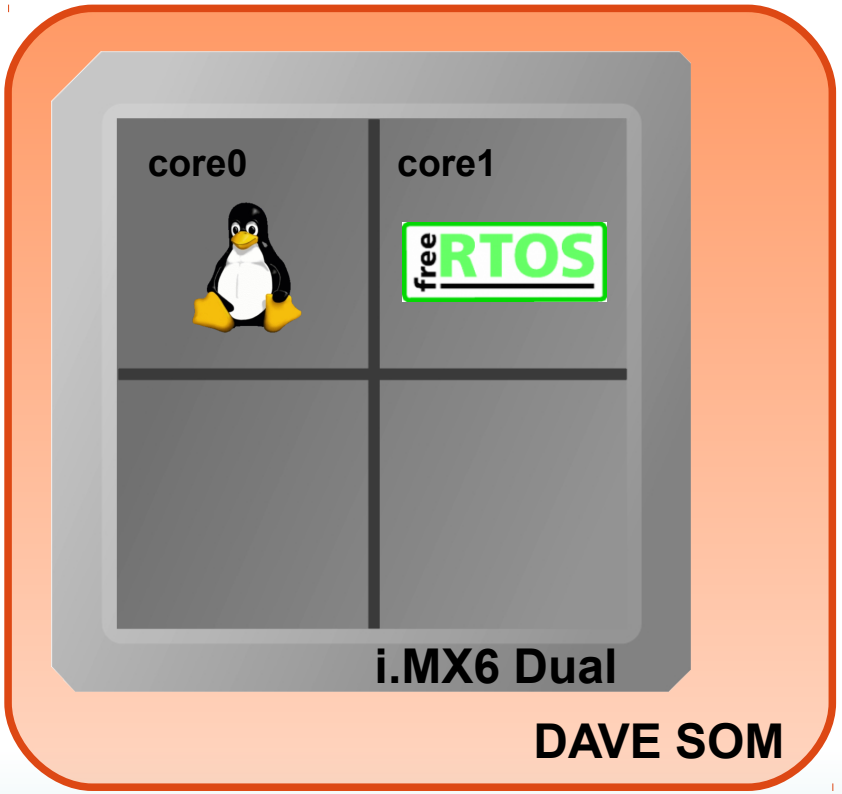
DAVE Embedded Systems

Linux AMP – Core Omogenei

Linux in esecuzione su 3 core
- Omogenei – RTOS in esecuzione
- Su 1 core



Linux in esecuzione su 1 core
Omogenei – RTOS in esecuzione
Su 1 core



DAVE Embedded Systems

Soluzioni per Linux-RT

- Linux NON è un OS real-time
- Soluzioni
  - Supporto RT kernel “recenti” (>2.6.x, preemption – soft RT)
  - Xenomai
  - Affiancare un microcontrollore esterno (bus, synch)
  - AMP: Linux su uno (o più) core, RTOS su uno (o più) core

DAVE Embedded Systems

AMP: pro e contro

- Vantaggi
  - Comunicazione tramite shared memory
  - Flessibilità
- Svantaggi
  - Risorse condivise (bus interno, DDR, Cache L2, periferiche, clock/governor)
  - → Linux può influenzare (pesantemente!) il comportamento dell'RTOS anche se è su un core diverso!
  - Debugging (demo a seguire)

DAVE Embedded Systems

Implementazioni AMP (1)

- Riduzione risorse Linux
  - numcpus=n-1/mem=xxM
  - Core n non inizializzato e disponibile per RTOS
  - Range di memoria >xxMiB disponibile per RTOS
  - Gestione IPC “custom”

DAVE Embedded Systems

Implementazioni AMP (2)

- Hypervisor
- Monitor
  - SaveG http://www.toppers.jp/en/safeg.html
  - Xen ARM
    - Supporto limitato (Cortex-A15)
    - Target principale: consumer/aziendale
  - JailHouse https://github.com/siemens/jailhouse
    - X86 only (ARM in progress), lightwigth, target industriale
    - Pubblicato 19 Novembre 2013 (http://lwn.net/Articles/574273/)

DAVE Embedded Systems

Implementazioni AMP (3)

- Linux RPMS/remoteproc
  - Inizio sviluppo in kernel 3.0.x
  - Sviluppato da TI (DSP)
  - Dichiarato ancora “instabile”
  - Facilità
    - Startup dei core secondari (spegnimento/accensione a runtime!)
    - Assegnazione risorse (memoria riservata al boot e mappata a runtime)
    - Comunicazione standard (RPMS)

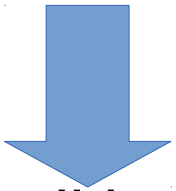
AMP JTAG Debugging

- SMP: un solo debugger controlla tutti i core
- AMP: sono necessarie più istanze del debugger
  - Risorse diverse (Cache L1, MMU, Performance counter)
  - Break sync/async
  - Simboli Debug/Awarness (Linux vs RTOS)
- È comunque sempre necessario correlare gli eventi nelle due istanze del debugger!

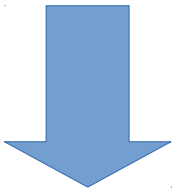
DAVE Embedded Systems

AMP JTAG Debugging (2)

- Il sistema parte sempre con 1 solo core (core0)
- Gli altri core sono attivati a runtime



- Il primo debugger controlla il boot
- Il secondo si “aggancia” quando parte il relativo core



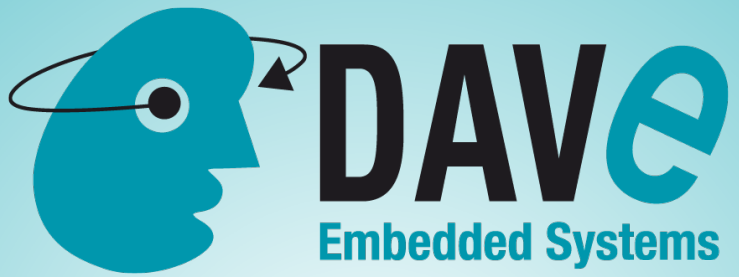
- Script – Script – Script!

DAVE Embedded Systems

References

- http://wiki.dave.eu
- http://wiki.dave.eu/index.php/Category:Linux
- http://free-electrons.com/docs
- Training http://free-electrons.com/doc/training/embedded-linux/
- http://elinux.org
- http://lwn.net (if you're not subscriber, please do so!)
- Building Embedded Linux Systems, By Karim Yaghmour, O'Reilly Media

DAVE Embedded Systems



Sviluppare con Linux Embedded

DAVE S.r.l.

Via Talponedo, 29/A
I-33080, Porcia (PN) Italy

Tel +39 0434 921215
Fax +39 0434 1994030

OSD2014
Polo Scientifico dell'Università degli Studi di Udine

www.dave.eu
info@dave.eu
wiki.dave.eu