Linux in TV
Going from prototype to product

Klaas de Waal
November 3, 2007
Content

- TV520 Introduction
- Why Linux?
- Stability and maturity of Linux
- Change to Virtual memory
- Boot time reduction
- Next Steps
- Conclusion
- Where is all the software?
TV520 product: Philips Aurea
TV520 Program
TV520 System Overview
TV520 Software Overview

- PNX8535
- MIPS 4KEC
  - Applications
  - Middleware
  - UI
  - Graphics Mgr
- Hybrid FE Control
  - CA
  - Analog Input
  - Analog Av Decoding
  - Hdmi
- Audio Processing
  - Video Featuring
- BE Control
- 8051 uP
  - Power Mgt
  - Stby/watch dog
  - RC P50
  - LKB
  - ADC
  - PWM
  - Boot Script
- TriMedia 2270
  - Demux
  - Spdif input
  - Conn input
  - Analog Av Capturing
  - Digital AV Decoding
  - Video Processing
  - Audio/Spdo Rendering
  - pSOS
Content

- TV520 Introduction
- Why Linux?
- Stability and maturity of Linux
- Change to Virtual memory
- Boot time reduction
- Next Steps
- Conclusion
- Where is all the software?
Why Linux?

- Brings rich set of “computing” features to CE products
  - Networking
  - Connectivity
  - File Systems

- Royalty free
  - Important for our customers

- Leverage large ecosystem
  - Linux is supported by a vast community of software developers

- Development cost
  - Engineering efficiency
When & How?

- Two Step Approach:

  1- Replace RTOS with Linux, leave TV-application as is
    - Reduce risk

  2- Change architecture towards Linux
    - Have all the benefits
What’s New

- Linux results in major changes in MIPS SW Infrastructure:
  - Linux drivers for I2C, GPIO, XIO, Shared Memory control, …
  - Flash file system replaced by JFFS2
  - USB stack from Linux distribution
  - OS abstraction layer implementation on top of Linux kernel / Linux drivers
  - Changed boot sequence

- Attention needed for:
  - Memory Footprint
  - Boot Time
  - RFS (Root File System) size
Content

- TV520 Introduction
- Why Linux?
- Stability and maturity of Linux
- Change to Virtual memory
- Boot time reduction
- Next Steps
- Conclusion
- Where is all the software?
Stability and maturity of Linux

- Linux distribution from a major vendor
  - Risk reduction
  - Support of our customers

- Based on Linux-2.6.10
  - That was already rather old…
  - but quite new if you once used Linux version 0.99

- Problems encountered:
  - JFFS2:
    - lots of patches needed
  - USB bug:
    - Was fixed in 2.6.18
    - USB fixes backported

- Problems in tooling:
  - Symbolic debugger tool did not work properly
    - …application too big?
    - …too many threads? (only 40)
Case: USB

Use case: playback JPG / MP3 from card reader

- Requirement: Detect cards automatically
- PC behaviour: not acceptable

Solution:
- Polling

Issues:
- Stability
- Back porting latest USB changes
Change to Virtual memory

- **Starting point**
  - Memory layout spreadsheet
    - Exact number of bytes for each application specified
    - Exact number of bytes for the OS specified
  - TV application fixed in memory
  - Linux did not fit in the allotted space
  - Not enough memory for everything

- **Solution**
  - Linux supports MMU and demand paging (this is not uCLinux)
  - Remove code to fix the application in memory
  - Let Linux decide which parts to keep in memory

- **Biggest challenge:**
  - Convince the customer that this is really sound engineering
Case: Virtual Memory helps

- Top requirement:
  - Avoid frequent reading from NAND flash memory
  - Flash memory sensitive to reading

- Customer expected:
  - This will never work

- Actual proof by monitoring page faults

- Result:
  - size of application plus Linux together as small as application plus VxWorks together

Page fault rate vs. Memory size diagram

Boot time reduction

- A TV has to start really quick after the user presses the On/Off button…
- At boot a JFFS2 partition is mounted
- This takes rather long…
  - …people who want to watch TV are not interested in the why..
- JFFS2 mount starts a garbage collection kernel thread
- Solution as implemented:
  - Delay garbage collection about 30 seconds
Content

- TV520 Introduction
- Why Linux?
- Stability and maturity of Linux
- Change to Virtual memory
- Boot time reduction
- Next Steps
- Conclusion
- Where is all the software?
Next Steps

- Hardware access
  - Shift AV drivers from user mode → kernel mode
  - Shorter development time

- Improving the TV application:
  - Split into separate Linux processes
  - Make better use of Linux facilities

- We’re just at the beginning…
Conclusion

- TV520 is a commercial success!
  - All Philips TV’s featuring FullHD/1080p are based on TV520 and Linux
  - Number of design-in for new customers

- Trade-off: Driver Reuse vs. Linux philosophy

- Optimization has taken quite some time

- Expertise is available externally
  - And you will need it, but:

- There Ain't No Such Thing As A Free Lunch
  - No development cost reduction seen yet

- YMMV
Where is all the software?

- NXP delivers everything to it’s customers, the TV-set makers
  - They have to comply with GPL
  - And make the GPL’ed source code available to their customers

- Interested in the TV520 source code including all patches?
  - [http://www.p4c.philips.com/cgi-bin/dcbint/cpindex.pl?ctn=52PFL9632D/10](http://www.p4c.philips.com/cgi-bin/dcbint/cpindex.pl?ctn=52PFL9632D/10)
  - Select “Software & drivers”
Our biggest Linux guru …