State of Multimedia in 2010’s Embedded Linux Devices

Benjamin Zores

ELCE 2010 - 27th October 2010 - Cambridge, UK
State of Multimedia in 2010’s Embedded Linux Devices

Presentation Objectives

- Find out about multimedia status on today's embedded devices.

- Help you make the best choices for your next device's design:
  - What is the best hardware for you?
  - Which OS can you run on top of it?
  - How to get the best out of your SoC?
  - How to write embedded applications?

- Find out more about "OpenSource" compatibility.
State of Multimedia in 2010’s Embedded Linux Devices

About Myself ...

Software Architect at Alcatel-Lucent

- Expert on Open source software.
- 7y experience on various embedded devices design.
- From low-level BSP to global software architecture.

Open Source projects founder, leader and / or contributor for:

- **OpenBricks** embedded Linux framework.
- **GeeXboX** embedded multimedia distribution.
- **Enna** EFL Media Center.
- **uShare** UPnP and DLNA Media Server.
- **MPlayer** media player application.
Hardware Selection
Which SoC for which target?
State of Multimedia in 2010’s Embedded Linux Devices
Hardware Selection - Which SoC for which target?

- A few simple rules to follow:
  - #1: Think about what your customers really need.
  - #2: Now think about features and product's lifespan.
  - #3: Only by then, think about shopcost.

- Always remember:
  - A SoC is not only a CPU.
  - Define your own criterias.
  - The most powerful SoC is not necessarily the good one for your product.

- PS: I'm neither affiliated to any vendor nor related to any ARM shareholder :-)

- PS2: All logos are trademark of their respective owners.
# State of Multimedia in 2010’s Embedded Linux Devices

## Hardware Selection - Which SoC for which target?

<table>
<thead>
<tr>
<th>Criteria</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPU</td>
<td><em>Main Processing Capabilities (Raw Speed, MHz ...)</em></td>
</tr>
<tr>
<td>SMP</td>
<td><em>Multi-Core Capabilities</em></td>
</tr>
<tr>
<td>SIMD</td>
<td><em>Instruction Set Extensions (MMX, SSE, VFP, NEON ...)</em></td>
</tr>
<tr>
<td>PSU</td>
<td><em>Power Consumption and Management</em></td>
</tr>
<tr>
<td>NET</td>
<td><em>Networking Capabilities (FE, GbE, WiFi, Bluetooth, 3G ...)</em></td>
</tr>
<tr>
<td>CONNECT</td>
<td><em>Extra Peripherals Connectivity Capabilities (USB, (m)PCI(e) ...)</em></td>
</tr>
<tr>
<td>STORAGE</td>
<td><em>Storage Capabilities (NOR, NAND, SDHC, (e)SATA ...)</em></td>
</tr>
<tr>
<td>2D</td>
<td><em>2D Graphics Capabilities</em></td>
</tr>
<tr>
<td>3D</td>
<td><em>3D Graphics Capabilities</em></td>
</tr>
<tr>
<td>VIDEO</td>
<td><em>Video Decoding/Encoding Capabilities</em></td>
</tr>
</tbody>
</table>
State of Multimedia in 2010’s Embedded Linux Devices
Hardware Selection - Which SoC for which target?

**MIPS32 SoCs**
- Usually low-end CPUs (< 300 MHz)
- Mostly used with wired equipments.
- Legacy Ethernet networking.
- Good for A/V output.
- Focus on 2D and Video processing.
- Usually comes with regular connectivity: USB, SATA...

**SoC Examples**
- Sigma Designs

**Typical Usage**
- Set-Top-Box, Bluray / DVD Players
State of Multimedia in 2010’s Embedded Linux Devices

Hardware Selection - Which SoC for which target?

**ARM9 SoCs**

- Old mobile phones SoCs (< 300 MHz)
- Still in the wild, clocked up to 1.5+ Ghz.
- Mostly used with wired equipments.
- Usually comes with enhanced Eth networking.
- Good for storage: usually USB / PATA / (e)SATA.

**SoC Examples**

- TI OMAP 1, Freescale i.MX2x, Marvell Kirkwood

**Typical Usage**

- NAS, Routers, Network equipments.
State of Multimedia in 2010’s Embedded Linux Devices

Hardware Selection - Which SoC for which target?

**ARM11 SoCs**
- Mid-end CPUs (400-700 Mhz)
- Mostly seen with 2007+ smartphones.
- Used with both wired and mobile equipments.
- Focus on multimedia with 2D/3D features.
- Focus on networking capabilities.

**SoC Examples**
- TI OMAP 2, Freescale i.MX3x,
- Qualcomm MSM72xx, Broadcom BCMring

**Typical Usage**
- Telecommunication Industry Smartphones and wired phones with low-end multimedia.
ARM Cortex-A8 SoCs

- High-end CPUs (600 Mhz - 1 GHz)
- Mostly seen with 2009+ smartphones.
- Designed for mobile equipments only.
- Introduced NEON instructions optimizations.
- Enhanced multimedia 2D / 3D / Video features.
- Mobile-only networking capabilities, lack of Ethernet and external storage.

SoC Examples

- TI OMAP 3, Freescale i.MX5x, Apple A4, Marvell ARMADA, Qualcomm SnapDragon QSD86xx.

Typical Usage

- High-end smartphones, Tablet PCs.
ARM Cortex-A9 SoCs

- Ultra high-end CPUs (700 Mhz - 1.5 GHz)
- Mostly seen with Q4 2010+ smartphones.
- Designed for mobile and wired equipments.
- Introduced SMP Optimizations: 1-4 Cores.
- Optional SIMD instructions
- Ultra high-end multimedia 2D/3D/Video features.
- Enhanced networking capabilities, mini-PCIe.

SoC Examples

- TI OMAP 4, nVidia Tegra 2, Qualcomm SnapDragon MSM86xx and QSD86xx.

Typical Usage

- High-end smartphones, Set-Top-Box, Connected TVs.
State of Multimedia in 2010’s Embedded Linux Devices

Hardware Selection - Which SoC for which target?

**Intel Embedded ATOM SoCs**

- Ultra high-end CPU (1200 MHz)
- Introduced with 2010 Set-Top-Boxes.
- Designed for wired equipments.
- Consequent power consumption (7W).
- Single-Core, x86 instructions set and large SIMD optimizations.
- Ultra high-end multimedia 2D / 3D / Video features.

**SoC Examples**

- Intel CE4100 and CE4200.

**Typical Usage**

- Set-Top-Box (Boxee), Connected TVs (Google TV).
State of Multimedia
State of Multimedia in 2010’s Embedded Linux Devices
State of Multimedia - 2D Raster Graphics

- **2D Raster Graphics**
  - Usually raw basic kernel framebuffer driver support.
  - Occasionally, DirectFB drivers:
    - Limited hardware acceleration for framebuffer.
    - Rarely supported by semiconductor vendors.
  - May come with proprietary X11 driver:
    - Basic implementation
    - Restricted to a given X.Org ABI (i.e deal with it!)
    - Why would you need X11 for embedded device??
  - Most of the time implemented through OpenGL|ES.
  - Hopefully addressed by application framework, when supported ...
2D Vector Graphics

- Useful for Flash and SVG rendering.
- Though rarely supported at all!!
- Most of the time implemented through OpenVG hardware acceleration framework.
- Mostly rely on proprietary drivers and libs.
- Supported by some application frameworks:
  - Android
  - MeeGo (Qt)
  - Cairo
  - Adobe Flash
State of Multimedia in 2010’s Embedded Linux Devices

State of Multimedia - 3D Graphics

- **3D Graphics: OpenGL|ES**

  - Complete hardware graphics acceleration.
    - Limited by GPU capabilities.
    - Sometimes slower than software rendering.
    - Available through proprietary drivers and libs only.

  - Relies on EGL:
    - Usually comes through vendor-specific implementation.
    - And many vendor-specific extensions.

  - Applications support heavily varies:
    - Imagination PowerVR SGX is best (only?) supported.
    - **No MediaPlayer support GLES as video output yet!!**
State of Multimedia in 2010’s Embedded Linux Devices

State of Multimedia - Software A/V Processing

- Audio / Video
  Software Implementation

  - Fully supported by OpenSource software:
    - FFmpeg: multi-codecs audio/video encoding/decoding library
    - Codec specific libs: libvpx, libmad, libvorbis, libfaad, libmPEG2 ...

  - Various proprietary software vendors.

  - Limited by CPU processing only and wide A/V codecs range support.

  - Optimized for VFP / NEON / SSE instructions and multi-core decoding.

  - Sometimes requires Integer-specific implementation for audio codecs (many ARM chips lack of FPU).

  - May have software patent issues in a few countries.
State of Multimedia in 2010’s Embedded Linux Devices

State of Multimedia - Hardware A/V Processing

- Audio / Video Hardware (DSP) Implementation

  - 100% CPU offloading.

  - Limited A/V codecs support.

  - Mostly closed-source:
    - Usually vendor-specific drivers and libs.
    - Never hit mainstream Linux.
    - Proprietary firmware, libs and DSP code.

  - Rarely supported by OpenSource projects.

  - Vendor-Specific DSP API (hard to support).

  - Generally available through OpenMAX IL / VA-API abstraction layer.
State of Multimedia in 2010’s Embedded Linux Devices
State of Multimedia - Hardware A/V Processing

- **OpenMAX**
  - Portable DSP abstraction API.
  - Generic implementation from MediaPlayer side.
  - Slower than native DSP access, less robust, but more portable.
  - Only supported by GStreamer and VLC OpenSource mediaplayers.

- **VA-API**
  - FreeDesktop equivalent to OpenMAX, mostly for x86.
  - Limited to X.Org video output.
  - Better support mostly due to x86 Desktop orientation.
State of Multimedia in 2010’s Embedded Linux Devices

State of Multimedia

- **State of 2D / 3D / Video Support**
  - Theoretically fully h/w supported, full CPU offloading.
  - Usually relies on proprietary implementation.
  - Limited capabilities and support, either due to h/w or s/w implementation.
  - Few commitment from semiconductor vendors to provide regular Linux mainstream support.
  - Possible h/w video decoding but semi-s/w rendering.
  - Only 100% usable on industry's mobile "standard" OS: Google Android
  - Maybe Nokia / Intel MeeGo some day?
The Underlying OS
State of Multimedia in 2010’s Embedded Linux Devices

The Underlying OS

- **Half-Commercial OS:**
  - WindRiver, Montavista …
  - Android, MeeGo, Ubuntu …

- **Homebrew OS:**
  - OpenEmbedded, OpenBricks
  - OpenWrt, Buildroot
  - …

- **Linaro Foundation**
  - Not really an OS but helps in making your devices works with Open Source Software.
State of Multimedia in 2010’s Embedded Linux Devices

The Underlying OS

- **Half-Commercial OS:**
  - Based on OSS but with deep commercial tights.
  - Take it the way it is:
    - Follow the project / product's philosophy.
    - Follow its lifecycle and roadmap.
    - Hard to change the overall software architecture.
    - Adapt your apps and skins to existing framework.

- **PROS:**
  - Potentially impressive Time-To-Market (TTM) and Long-Term-Support (LTS)
  - Good for rapid product deployment and basic applications development.

- **CONS:**
  - May not be adapted to custom and very specific apps.
  - Vendor roadmap follow-up may imply OS upgrades and API changes.
State of Multimedia in 2010’s Embedded Linux Devices
The Underlying OS

- **Homebrew OS:**
  - Fully based on OSS with barely no support.
  - Take it as your next design's framework:
    - Fine-tuning for on-demand custom OS creation.
    - You're on your own: fix it, debug it, adapt it.
    - Easy to create the software architecture you want.
    - Pray for your whole hardware to be fully supported in upstream Linux kernel :-(

- **PROS:**
  - Complete control over your product’s global software roadmap.
  - Good for autonomous system and application deployment.

- **CONS:**
  - You're on your own from the very beginning to the end (no or very bad LTS).
  - Security threat and fixes workload
Embedded Applicative Framework
State of Multimedia in 2010’s Embedded Linux Devices
Embedded Applicative Framework

- How to write applications and user interfaces?
  - Google’s Android native applications framework.
  - Nokia’s Qt (MeeGo, KDE …)
  - GTK+ (Gnome)
  - Enlightenment Foundation Libraries (EFL)
  - Simple Direct Media Layer (SDL)
  - Web Technologies
State of Multimedia in 2010’s Embedded Linux Devices
Embedded Applicative Framework

**Questions you may ask yourself:**

- Has it to be a tradeoff between code efficiency and programming convenience?
- Native compiled code or interpreted one?
- Is my code meant to run on multiple devices?
- MVC Approach: should I distinct middleware / core from user interface?
- SDK or Open Source collaboration: what if I want to gather developers around my project?
## State of Multimedia in 2010’s Embedded Linux Devices

Embedded Applicative Framework (based on personal feedback)

<table>
<thead>
<tr>
<th></th>
<th>Android</th>
<th>Qt</th>
<th>GTK+</th>
<th>EFL</th>
<th>SDL</th>
<th>Web</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Language</strong></td>
<td>Java</td>
<td>C++</td>
<td>C</td>
<td>C</td>
<td>C</td>
<td>JS, CSS, HTML</td>
</tr>
<tr>
<td><strong>Maturity</strong></td>
<td>Average</td>
<td>Good</td>
<td>Good</td>
<td>Weak</td>
<td>Good</td>
<td>Average</td>
</tr>
<tr>
<td><strong>Portability</strong></td>
<td>Weak</td>
<td>Good</td>
<td>Average</td>
<td>Average</td>
<td>Good</td>
<td>Average</td>
</tr>
<tr>
<td><strong>Footprint</strong></td>
<td>N.A</td>
<td>Weak</td>
<td>Average</td>
<td>Good</td>
<td>Good</td>
<td>Good</td>
</tr>
<tr>
<td><strong>Performance</strong></td>
<td>Good</td>
<td>Average</td>
<td>Average</td>
<td>Good</td>
<td>Good</td>
<td>Weak</td>
</tr>
<tr>
<td><strong>System-Wide</strong></td>
<td>Good</td>
<td>Good</td>
<td>Weak</td>
<td>Average</td>
<td>Weak</td>
<td>Weak</td>
</tr>
<tr>
<td><strong>Community Followers</strong></td>
<td>Good</td>
<td>Good</td>
<td>Average</td>
<td>Weak</td>
<td>Average</td>
<td>Good</td>
</tr>
<tr>
<td><strong>Multimedia Integration</strong></td>
<td>Good</td>
<td>Good</td>
<td>Average</td>
<td>Average</td>
<td>Average</td>
<td>Good</td>
</tr>
<tr>
<td><strong>OpenGL ES</strong></td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td><strong>OpenVG</strong></td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td><strong>MVC Mode</strong></td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
</tr>
</tbody>
</table>
Conclusion
State of Multimedia in 2010’s Embedded Linux Devices

Designing your new device: Questions to be raised ...

- Questions to be raised:
  - Which features do you really need?
  - What is your project's expected lifespan?
  - Do you target TTM or LTS?
  - Do you need complete code mastership?
  - Do you need SDK / external apps openness?
  - Is regular upstream Linux support mandatory for your application?
State of Multimedia in 2010’s Embedded Linux Devices

Conclusion - State of Embedded Linux

- Multimedia on embedded Linux rocks!
  - Maybe even more than on desktop Linux ...
  - But unfortunately mostly relies on 100% proprietary software.

- Linux now supports so many SoCs ...
  - But h/w vendors rarely contribute upstream.
  - Mostly comes with Android-only software.
    - Usually provided under binary form only
    - You have to stick to first (and last) BSP release.

- Need to sponsor MeeGo / Linaro-like initiatives to upstream SoC support on Linux.
State of Multimedia in 2010’s Embedded Linux Devices
Annex - Some references

- DirectFB: [http://directfb.org/](http://directfb.org/)
- OpenVG: [http://www.khronos.org/opengv/](http://www.khronos.org/opengv/)
- OpenGL|ES: [http://www.khronos.org/opengles/](http://www.khronos.org/opengles/)
- OpenMAX: [http://www.khronos.org/openmax/](http://www.khronos.org/openmax/)
State of Multimedia in 2010’s Embedded Linux Devices
Annex - Some references

- MeeGo: [http://www.meego.com/](http://www.meego.com/)