The Use of Hardware Simulation in Smartphone Projects

Author: Joonas Kolmonen
Supervisor: Professor Heikki Hämmäinen
Instructor: Timo Pasonen
Contents

- Background
- Research Problem
- Research Methods
- S60 Reference Implementations
- Virtual Platforms
- Virtio VPOM-2420 Virtual Platform
- Results
- Conclusions
Background

- S60 smartphones are feature-rich mobile phones

The S60 platform
- An open mobile software platform by Nokia
- Built on top of Symbian OS
- Licensed by several mobile phone manufacturers
- The leading smartphone platform in the world
Research Problem

- Time-to-market is too long
  - Smartphones are complex embedded systems
  - Hardware available relatively late
  - The risk of delays is comparatively high

- Virtual platforms
  - Could they speed up the overall process?
  - Could they reduce the risk of missing the intended market window?
  - What can be developed/tested on virtual platforms?
  - How reliable are the results?
Research Methods

- Investigating the feasibility of virtual platforms by
  - Creating an S60 environment on
    - Hardware (OMAP2420 Software Development Platform)
    - Virtual platform (Virtio VPOM-2420 Virtual Platform)
  - Defining and executing several use cases
    - Performance measurements
      - Boot up time
      - Disk/memory write and read
    - Functionality tests
S60 Reference Implementations

- Model designs for developing S60 devices
- Pre-integrated products
- Roughly 50 – 80 % of the complete product
S60 Reference Implementations

- Opening up a possibility to
  - Reduce the development costs
  - Shorten the time-to-market

- Reducing the risk of missing the intended market window

- Increasing the competitiveness of the chipset vendor
Virtual Platforms

- Emulator environments, emulating embedded development boards / devices on a PC
  - Providing target binary compatible environments
    - Advantage over the WINS emulator
  - Available before the actual hardware exists
    - SW development can be started in parallel with the hardware design
Virtio VPOM-2420 Virtual Platform

- Models OMAP2420 Software Development Platform
- Real world connectivity via the Windows OS
- Advanced debugging capabilities
- Online distribution of hardware models
Limitations of VPOM-2420

- Instruction accurate instead of cycle accurate
  - Clock cycle details not modeled
  - May cause problems in timing critical device driver development and performance optimization

- Only a subset of the hardware functionality and peripherals modeled
  - E.g. no modem, WLAN, Bluetooth or IrDA
  - Limited power management modeling
Results - General

- Relatively large number of bugs found from VPOM-2420 Virtual Platform
  - A lot of debugging was required
- Easy to debug
- Virtio support excellent
Exactly the same S60 software booted up on both environments

Tests did not reveal any significant differences in functionality

- Corrections to the existing models needed to achieve this result
- Problems with some hardware drivers on the virtual platform

<table>
<thead>
<tr>
<th>Test case</th>
<th>Result VPOM-2420</th>
<th>Result OMAP2420</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Phonebook</td>
<td>OK</td>
<td>OK</td>
<td></td>
</tr>
<tr>
<td>Clock</td>
<td>OK</td>
<td>OK</td>
<td>VPOM-2420: The system time running about three times slower than real time</td>
</tr>
<tr>
<td>Camera</td>
<td>OK</td>
<td>OK</td>
<td>VPOM-2420: Camera application can be launched only once</td>
</tr>
<tr>
<td>RealOne Player</td>
<td>FAIL</td>
<td>FAIL</td>
<td>Similar error on both environments</td>
</tr>
<tr>
<td>Memory Card</td>
<td>OK</td>
<td>OK</td>
<td>VPOM-2420: Parameter tweaking required to get the memory card to work</td>
</tr>
<tr>
<td>Application Installer</td>
<td>OK</td>
<td>OK</td>
<td>VPOM-2420: Parameter tweaking required to get the memory card to work</td>
</tr>
<tr>
<td>Basic Call</td>
<td>FAIL</td>
<td>FAIL</td>
<td>Similar error on both environments</td>
</tr>
</tbody>
</table>
Results - Performance

- VPOM-2420 performance very poor with the default settings

- Performance could be brought up to a relatively good level with model updates and parameter tweaking
  - The virtual platform approximately two to three times slower than the actual hardware
  - In some functionality areas more significant differences were measured

- The state of the Windows OS affects the performance of the virtual platform
## Performance Results Example

### S60 Boot Up Time

<table>
<thead>
<tr>
<th>Environment</th>
<th>Average in seconds</th>
<th>Standard Deviation</th>
</tr>
</thead>
<tbody>
<tr>
<td>VPOM-2420 - first boot</td>
<td>499.0</td>
<td>5.21</td>
</tr>
<tr>
<td>OMAP2420 - first boot</td>
<td>228.5</td>
<td>0.19</td>
</tr>
<tr>
<td>VPOM-2420 - subsequent boot</td>
<td>207.1</td>
<td>1.08</td>
</tr>
<tr>
<td>OMAP2420 - subsequent boot</td>
<td>107.2</td>
<td>0.37</td>
</tr>
</tbody>
</table>

![Bar chart showing S60 Boot Up Time](chart.png)
Conclusions

- Advanced debugging capabilities
  - Improve efficiency especially in low-level software development
  - Possible to find hardware design faults

- If used in an actual smartphone project, the virtual platform has to be reliable
  - A close co-operation between the virtual platform provider and the hardware manufacturer is essential
  - Updates needed if the hardware is changed
Conclusions

- Suitable for smartphone development
  - Prior to silicon
  - When the actual hardware already exists

- The use of a virtual platform could be particularly beneficial in Reference Implementation projects
Questions?

Thank You!