Multi Core for Mobile Phones
– the sky is the limit?

MPSoC 2010, Gifu, June 30

Kees van Berkel

fellow

TU/e

ST ERICSSON
Multi Core for Mobile Phones

The sky is the limit?
That’s so old-skoool architecting!

Saint Rumbold's Tower, Mechelen, Belgium
... the sky’s sky is the limit!

... a lot to explore!

Cosmic Microwave Background Radiation (CMBR),
Wilkinson Microwave Anisotropy Probe (WMAP)
The Square Kilometer Array (SKA)

... the ultimate exploration tool
... and the ultimate software defined radio
The Square Kilometer Array (SKA)

- antenna surface: $1\ km^2$ (sensitivity $50\times$)
- large physical extent ($3000+$ km)
- wide frequency range: $70\ MHz - 30\ GHz$
- full design by 2010; phase 1: 2017; phase 2: 2022

- $1000-1500$ dishes (15m) in the central $5\ km$ ($2000-3000$ total)
- + dense and/or sparse aperture arrays
- connected to a massive data processor by an optical fibre network

**Software Defined Radio Astronomy**

- computational load (on-line) $\approx 1\ \text{exa MAC} \ (10^{18}\ \text{MAC/s})$
- power budget = $30\ MW$
- $\approx 30\ \text{pJ/MAC} \ "all-in"$
Low-end GSM phone: power dissipation [indicative]

idle mode [5mW]

- Miscellaneous: display, oscillator, power conversion, interfaces, ...

- DSP + CPU + memories: only 5%, in both idle and talk modes!

talk mode [0.8W]
High-end handset: e.g. Nokia N95

- GSM/GPRS/EDGE/UMTS/HSPA (6 bands)
- 802.11 b/g, GPS, BT, FM
- MP3/WMA/AAC+/eAAC+/M4A/Real Audio
- MPEG-4, H.263, H.264, ...
- 2D/3D– Graphics
- ARM11@ 332MHz
- 2.6" QVGA display
- 5Mpix camera
- up to 8GB internal storage

High-end handset power consumption

Source: Neuvo 2004 [Nokia]
Outline

• Power budget for “computing” (incl, memories): 1W:
  • other 2W for display backlight, power amp, RF, interfaces, ....

• Workload (highly diverse operations): 100 GOPS:
  • characterization needed: applications, radios, audio, video, gfx, ...

• 100 GOPS in 1W ⇒ heterogeneous multi-core:
  • core specialization, memory hierarchy, aggressive power management

• Trends: towards 1 TOPS in 1W:
  • multi-core application processors
  • self-contained subsystems
  • Software Defined Radio (SDR)
  • Long-Term Evolution (LTE, LTE-A)
Workload characterization

• “Algorithmic GOPS” [Giga Operations Per Second]
  • typically multiply, add (not load, store)

• Large diversity in:
  • data types: mostly 4–16 b integer/fixpoint
  • energy/operation (16b integer multiply vs 8b Galois addition)

• Main computational loads [DATE’09 paper]:
  • application processing
  • radio processing
  • audio/video processing
  • graphics processing
Workload: application processing: 0.5 GOPS

- 0.5 GOPS
- Example: ARM11 [2008, 65 nm CMOS]: 772 MHz, 0.2 mW/MHz
- ⇒ 0.2 mW/MHz for phone PCs versus > 20 mW/MHz for CPU PCs

(sources: ITRS roadmap, ARM web site)
Workload: radio processing: 40 GOPS

HSDPA receiver
- digital front-end 10 GOPS
- demodulation 5 GOPS
- decoding 25 GOPS

Total 40 GOPS

- front-end: filtering, cordic, gain control, etc
- demodulation [16b]: equalization, channel estimation, interference cancellation, synchronization, …
- decoding [5b]: turbo (!), viterbi, de-interleaving,
Workload: overall: about 100 GOPS
How to realize 100 GOPS \textit{(algorithmic)} in a 1W power budget?

[10 pJ/operation* in 65nm CMOS]

* "all–in" = incl. memories, buses, control, idle cycles
EVP: VLIW $\times (16 \times \text{SIMD} \parallel \text{scalar}) \times 16\text{b}$

- First EVP-based product: TD-SCDMA modem for Chinese Market
- 100 “machine ops”/cycle $\approx 30$ algorithmic ops/cycle \textit{(typical)}
- 0.6 mW/MHz $\approx 20$ pJ/algorithmic operation $\approx 30$ pJ/MAC
16–SIMD is optimal (appl. of Amdahl’s Law to SIMD)

Definitions:

• $\text{Area}_{\text{non-lane}} = \text{Area}_{\text{instruction-decoding}} + \text{Area}_{\text{scalar}} + \text{Area}_{\text{addressing-units}}$

• $P =$ vector length = number of 16–bit lanes (16 for EVP32042)

• $f =$ fraction of work that can be vectorized

\[
\text{Area} = \left[ \frac{\text{Area}_{\text{non-lane}}}{\text{Area}_{\text{lane}}} + P \right] \times \text{Area}_{\text{lane}}
\]

\[
\text{Time} = \left[ \frac{f}{P} + (1 - f) \right] \times \text{Time}_{\text{sequential}}
\]

• EVP: $\text{Area}_{\text{non-lane}} \approx 6 \times \text{Area}_{\text{lane}}$

• EVP applications: $f = 0.98$ 
  $\approx$ vector operation $\frac{3}{4}$ cycles

optimum for $P = 17.1$
FFT energy: EVP vs dedicated hardware

\[ 1.8 \leq \frac{E_{\text{norm}}(\text{FFT\_on\_EVP})}{E_{\text{norm}}(\text{FFT\_in\_HW})} \leq 4.5 \]

\[
E_{\text{norm}} = \frac{\lambda_{\text{ref}}}{\lambda} \times \left( \frac{V_{\text{dd, ref}}}{V_{\text{dd}}} \right)^2 \times \frac{2/3 W_{\text{ref}} + 1/3 W_{\text{ref}}^2}{2/3 W + 1/3 W^2} \times \frac{\text{Energy\_per\_block}}{\#\text{butterflies\_per\_block}}
\]

[Baas, 1999]
100 GOPS in 1W

Cost of flexibility: energy/operation [ball-park figures] (2 orders of magnitude!):

<table>
<thead>
<tr>
<th></th>
<th>Energy pJ/op</th>
<th>example</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPU</td>
<td>200</td>
<td>ARM11</td>
</tr>
<tr>
<td>DSP</td>
<td>20</td>
<td>EVP</td>
</tr>
<tr>
<td>hardware</td>
<td>2 .. 5</td>
<td>turbo decoder .. FFT</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Load_{ARM} [GOPS]</th>
<th>Load_{DSP} [GOPS]</th>
<th>Load_{HW} [GOPS]</th>
<th>power [mW]</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>0</td>
<td>0</td>
<td>20000</td>
</tr>
<tr>
<td>0</td>
<td>100</td>
<td>0</td>
<td>2000</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>100</td>
<td>200 .. 500</td>
</tr>
<tr>
<td>0.9</td>
<td>9</td>
<td>90</td>
<td>600 .. 900</td>
</tr>
</tbody>
</table>

CPU/DSP/hardware load partitioning must satisfy:

- \( Load_{ARM} \times 200pJ + Load_{DSP} \times 20pJ + Load_{HW} \times 2pJ \leq 1W \)
- \( Load_{ARM} + Load_{DSP} + Load_{HW} = 100 \text{ GOPS} \)
Heterogeneous multi-core: trade-offs

- pJ/op, incl. local memory
- load: algorithmic GOPS
- DSP, incl. vector DSP
- partition based on a: “plausible example”
- load-energy distribution resembles a power law
Heterogeneous multi-core

- How to realize 100 GOPS (algorithmic) in a 1W power budget? [10 pJ/operation in 65nm CMOS]
- Cost of flexibility: energy/operation [2 orders of magnitude]:
  - ARM11: 200 pJ/op
  - DSP: 20 pJ/op EVP (Embedded Vector Processor)
  - HW: 2 pJ/op Turbo decoder,
        5 pJ/op FFT
- Value of flexibility: flexible hardware can support
  - resource sharing among diverse and within complex algorithms
  - post-Si improvement (debug) of algorithms, addition differentiators
  - standard evolution (e.g. 3GPP release 4, 5, 6)
- trade-offs are complex, and evolve over time
  - e.g. today’s video decoders less flexible than yesterday’s
Heterogeneous multi-core: “value of flexibility” & trade-offs

- Configurable hardware
- Vector DSP
- DSP
- Micro-controller

Application scope, generality

Load [ops]

Code size [Bytes] →
Heterogeneous multi-core

- CPU
- L2 cache
- Memory
- L2 memory
- L3 memory (DRAM)
- Storage (flash)
- Display
- Multi-application
- Multi-media
- Multi-radio

MPSoc -- 2010, June 30
Heterogeneous multi-core: trade-offs (example)

<table>
<thead>
<tr>
<th>stage</th>
<th>dig. frontend</th>
<th>modem</th>
<th>codec</th>
<th>protocols</th>
</tr>
</thead>
<tbody>
<tr>
<td>typical functions/algorithms</td>
<td>various filters</td>
<td>channel estimation,</td>
<td>(de)interleaving,</td>
<td>L2-L3 MAC</td>
</tr>
<tr>
<td></td>
<td>cordic gain control</td>
<td>equalization,</td>
<td>rate matching,</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>synchronization,</td>
<td>(de)puncturing,</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>interference cancel.,</td>
<td>viterbi, turbo,</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>rake receiver,</td>
<td>reed-solomon</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>sphere decoding</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## Heterogeneous multi-core: trade-offs (example)

<table>
<thead>
<tr>
<th>stage</th>
<th>dig. frontend</th>
<th>modem</th>
<th>codec</th>
<th>protocols</th>
</tr>
</thead>
<tbody>
<tr>
<td>typical functions /algorithms</td>
<td>various filters, cordic, gain control</td>
<td>channel estimation, equalization, synchronization, interference cancel., rake receiver, sphere decoding</td>
<td>(de)interleaving, rate matching, (de)puncturing, viterbi, turbo, reed-solomon</td>
<td>L2-L3 MAC</td>
</tr>
<tr>
<td>load [GOPS]</td>
<td>10</td>
<td>5</td>
<td>25</td>
<td>0.1</td>
</tr>
<tr>
<td>complexity of algorithms</td>
<td>low</td>
<td>medium-high</td>
<td>medium</td>
<td>high</td>
</tr>
<tr>
<td>diversity among standards</td>
<td>medium</td>
<td>high</td>
<td>medium</td>
<td>high</td>
</tr>
<tr>
<td>opportunities to differentiate</td>
<td>low (increasing)</td>
<td>high</td>
<td>low</td>
<td>medium</td>
</tr>
</tbody>
</table>

- Multi-standard radio: details depends on combination of radios, and on stability/maturity of standards
## Heterogeneous multi-core: trade-offs (example)

<table>
<thead>
<tr>
<th>stage</th>
<th>dig. frontend</th>
<th>modem</th>
<th>codec</th>
<th>protocols</th>
</tr>
</thead>
<tbody>
<tr>
<td>typical functions /algorithms</td>
<td>various filters, cordic, gain control</td>
<td>channel estimation, equalization, synchronization, interference cancel., rake receiver, sphere decoding</td>
<td>(de)interleaving, rate matching, (de)puncturing, viterbi, turbo, reed-solomon</td>
<td>L2-L3 MAC</td>
</tr>
<tr>
<td>load [GOPS]</td>
<td>10</td>
<td>5</td>
<td>25</td>
<td>0.1</td>
</tr>
<tr>
<td>complexity of algorithms</td>
<td>low</td>
<td>medium-high</td>
<td>medium</td>
<td>high</td>
</tr>
<tr>
<td>diversity among standards</td>
<td>medium</td>
<td>high</td>
<td>medium</td>
<td>high</td>
</tr>
<tr>
<td>opportunities to differentiate</td>
<td>low (increasing)</td>
<td>high</td>
<td>low</td>
<td>medium</td>
</tr>
<tr>
<td>configurable HW (vector) DSP uC</td>
<td>bulk</td>
<td>bulk</td>
<td>bulk</td>
<td>none</td>
</tr>
<tr>
<td></td>
<td>some</td>
<td>bulk</td>
<td>some</td>
<td>none</td>
</tr>
<tr>
<td></td>
<td>control</td>
<td>control</td>
<td>control</td>
<td>bulk</td>
</tr>
</tbody>
</table>

*Multi-standard radio: details depends on combination of radios, and on stability/maturity of standards and algorithms*
Trends towards 2014

1. Computation load in handset will continue to increase $10\times$ in 5 years:
   • towards 1 TOPS in 1W

2. Multi-core for (downloadable) applications:
   • towards a homogeneous multi-core subsystem for applications

3. Self-contained subsystems:
   • towards self-contained subsystems with standardized service interfaces

4. Software Defined Radio (SDR):
   • towards a radio computer running multiple radios (SW entities) in ||

5. 3GPP Long-Term Evolution (LTE, LTE-A):
   • towards 1 Gb/s downlink
The zoo of wireless standards (keeps evolving)

- **Positioning**  
  GPS, Galileo

- **Distribution (broadcast)**  
  DAB, DVB-T, DVB-H, DMB-T, ISDB-T, ATSC, XM, Sirius, DRM, etc.

- **Cellular 2G**  
  GSM, IS-95, IS-136, PHS, EDGE, GPRS

- **Cellular 3G**  
  UMTS, CDMA2000, TD-SCDMA, HSDPA, HSUPA, LTE

- **Hot spot**  
  802.11 a/b/g/n, 802.16a/e, WIBRO/WIMAX

- **Personal Area Network**  
  DECT, BlueTooth, UWB
SW Defined Radio – staircase

- Radio computer
- Cognitive radio
- “Full” (true) SDR
- Multi-radio (HW sharing)
- Multi-mode combos
- Architecture convergence
- More radios with fewer ICs
- More radios on less die area
- More radios with less ownership (easier migration of radios across platforms)
- Standardize on cores: EVP, ARM, ...
- Specific radio architecture & cores
- Diversity across standards

Timeline:
- 2006
- 2008
- 2010
- 2012
- 2014
SDR: a radio computer perspective [long term]

Radio program

Radio compiler

Radio package

Radio Programming Interface

run time (in handset)

Studied by ETSI for standardization

Unified-Radio System Interface

Radio Operating System

radio-storage

loader

radio

(Multi-Radio Control Framework)

virtualization layer:
real-time multi-radio on a multi-core radio computer

Radio computer

radio-design time

radio

installer

FEM

TRx

BB/inner

BB/outer

MAC

Inter Face

Radio computer

(Multi-Radio Control Framework)

Radio Operating System

radio-storage

loader

radio

(Multi-Radio Control Framework)

virtualization layer:
real-time multi-radio on a multi-core radio computer

Radio computer

radio-design time

radio

installer

FEM

TRx

BB/inner

BB/outer

MAC

Inter Face

Radio computer
BB Software Architecture: The Problem

Platform: heterogeneous multiprocessor

App: multiple radios simultaneously active
   – different rates of operation
   – unpredictable start/stop times

Requirement: provide real-time guarantees
   – each running radio must meet deadlines

Software architecture must
   – support widest variety of radio combinations and transitions
   – allow post-design updates and add-ons
   – simplify the radio design process

Joint research project 2007–2009 among Nokia, NXP, ST-Ericsson demonstrator reported at SDR forum 2009
Software Architecture for SDR/baseband

**Compile-Time (Budgeting)**
- Radio SW Components
- Radio Graph Description
  - LIME Front-end
  - Analysis Model
    - Timing Constraints
    - Heracles
      - Radio Budget
        - Static-Ordered Clusters
          - LIME Back-end
            - SoD Tasks

**Run-Time (Admission Control)**
- Radio Budget
  - Architecture
    - Resource Alloc
    - Job Request
      - SoD Tasks
    - Resource Manager
      - New Resource Allocation

Author: Orlando Moreira
Computation: Mode-Controlled Data-flow

**MCDF**: an extension of static dataflow: allows (limited) data-dependent behavior
(Orlando Moreira)

- customized for radios
- easy to program
- monotonic
- deadlock-free
- bounded buffers
- quasi-static ordering possible
- worst-case throughput/latency known
**SDR demonstrator**

- Our SDR technology demonstrator shows:
  - that multiple radios can be dynamically installed/loaded/started/stopped/unloaded, through a unified multi-radio access interface;
  - that multiple radios can run in a coordinated fashion;
  - that individual radios respect real-time constraints;
  - and that hardware resources are shared efficiently

- **Demonstrator setup**
  - 2xLinux PC: user applications + L3/L2 proc. + multi-radio resource mgmt
  - 2xprototype board (each 2 EVPs + 3 ARMs): RT–L1 + RT baseband RM
  - board-to-board connection: ~ RF link
Releases of 3GPP specifications

- **1999**
  - GSM/GPRS/EDGE enhancements

- **2000**
  - W-CDMA

- **2001**
  - Release 4
  - 1.28 Mcps TDD

- **2002**
  - Release 5
  - HSDPA, IMS

- **2003**
  - Release 6
  - HSUPA, MBMS, IMS+

- **2004**
  - Release 7
  - HSPA+ (MIMO, HOM etc.)

- **2005**
  - ITU-R M.1457
  - IMT-2000 Recommendations

- **2006**
  - Release 8
  - LTE, SAE

- **2007**
  - Release 9
  - Small LTE/SAE enhancements

- **2008**
  - Release 10
  - LTE Advanced
Towards 4G: Long-Term Evolution (LTE)

- **100 MHz bandwidth**
- **4x4 MIMO**
- **1 Gb/s downlink (scalable)**
- **up to 2 TOPS load (scalable, adaptive)**

- LTE standard completion (3GPP Release 8): 2008 Q4
- LTE-A standard completion (3GPP Release 10): 2011 Q1
Schedule computed for 4xEVP

- exploits 4x symmetries in graph; asymmetric schedules on e.g. 3xEVP
# programmable cores in a smartphone

![Graph showing the number of programmable cores in smartphones over years.](image)

- **application processor only**
- **baseband processor only**
- **combined**

- The graph shows an increasing trend from 2000 to 2015.
- For specific years, the number of programmable cores is indicated.

---

MPSoC --- 2010, June 30
Conclusion

• Today’s handsets provide about **100 GOPS in 1W** based on:
  • advanced CMOS (45–65 nm);
  • heterogeneous multi-core architectures;
  • aggressive power management.

• This requires **thorough and smart HW–FW–SW trade-offs**:
  • a detailed characterization of the work load;
  • a good understanding of both *value* and *cost* (power!) of flexibility;
  • selecting the right kind of flexibility and the right level of flexibility.

• By 2014: **1 TOPS in 1W**:
  • heterogeneous multi-core (incl. ≈10 programmable cores)
  • towards software defined multi-radio and LTE-advanced
  • based on the classical architecture dictum “**form follows function**”.
Further reading

• Multi-Core for Mobile Phones,
  Kees van Berkel,
  DATE’09, April 2009, Nice (invited paper)

• A Multi-Radio SDR Technology Demonstrator,
  Kees van Berkel, Artur Burchard, David van Kampen, Pjotr Kourzanov,
  Orlando Moreira, ST-Ericsson, Antti Piipponen, Kalle Raiskila, Sverre
  Slotte, Marinus van Splunter, Tommi Zetterman,

• LTE-Advanced Physical Layer
  Matthew Baker,
  REV-090003r1 IMT-Advanced Evaluation Workshop, 2009, Beijing

• Square Kilometer Array
  http://www.skatelescope.org/

• Building Correlators with Many-Core Hardware:
  a look at performance, optimization and programmability
  Rob V. van Nieuwoort and John W. Romein.
  IEEE Signal Processing Magazine [Vol 27, No 2],
  special issue on MultiCore Platforms
LET'S CREATE IT

THANK YOU