Multi-Core for Mobile Phones

(Invited Paper)

C.H. (Kees) van Berkel
ST-NXP Wireless, Advanced R&D, High Tech Campus 32, 5656 AE Eindhoven, The Netherlands
Also with Technical University Eindhoven, Dept. of Computer Science and Mathematics
Email: kees.van.berkel@stnwireless.com

***Abstract*—High-end mobile phones support multiple radio standards and a rich suite of applications, which involves advanced radio, audio, video, and graphics processing. The overall digital workload amounts to nearly 100GOPS, from 4b integer to 24b floating-point operations. With a power budget of only 1W this inevitably leads to heterogeneous multi-core architectures with aggressive power management. We review the state-of-the-art as well as trends.**

I. Introduction

Early GSM handsets provided a single function: voice communication across a bi-directional link of 10kb/s. Their digital architectures are typically *dual-core*:

- a DSP with an optimized instruction set, e.g. [1], running at about 20MHz, to support most of the signal processing (speech codecs, channel codecs, viterbi equalization), and
- a micro-controller, e.g. an ARM7, to support a simple user interface and the GSM-protocol stack.

In the last decade we have witnessed a rapid proliferation of functions and features, including [2], [3]:

- multiple cellular communication transceivers, often multi-band, supporting data rates up to several Mb/s, increasingly based on the Internet Protocol (IP);
- email and internet browsing;
- other forms of connectivity, such as BlueTooth and WLAN, as well as broadcast reception, such as FM, DVB-H, and GPS;
- digital music playback, and photography;
- pixel based displays with resolution of QVGA and higher, enabling video and 3D graphics;
- downloading of applications, such as games.

For low feature, high volume markets, handsets tend towards "single-chip" solutions [4], true system-on-chips (SoCs). All digital processing, radio-frequent (RF) transceivers, and power management units (PMU), are integrated on a single die. Only a power amplifier (PA), and off-chip DRAM and Flash memories are required to complete a phone chip-set. For high-feature markets the handsets (smart phones) are typically based on multiple chips, including a baseband processor and an application processor. Only when a particular feature tends towards a high attach rate – say 40% – the corresponding function is integrated on the main die.

The digital workload of a smartphone (all control, data and signal processing) amounts to nearly 100 giga operations per second [GOPS, Sec. II]. This workload increases at a steady rate, roughly by an order of magnitude every 5 years (Tab. I).

TABLE I
Mobile phone trends in 5-year intervals.

year	1995	2000	2005	2010	2015
cellular generation	2G	2.5-3G	3.5G	pre-4G	4G
cellular standards	GSM	GPRS UMTS	HSPA	HSPA LTE	LTE LTE-A
downlink bitrate [Mb/s]	0.01	0.1	1	10	100
display pixels [×1000]	4	16	64	256	1024
battery energy [Wh]	1	2	3	4	5
CMOS [ITRS, nm]	350	180	90	50	25
PC CPU clock[MHz]	100	1000	3000	6000	8500
PC CPU power [W]	5	20	100	200	200
PC CPU MHz/W	20	50	30	30	42
phone CPU clock[MHz]	20	100	200	500	1000
phone CPU power [W]	0.05	0.05	0.1	0.2	0.3
phone CPU MHz/W	400	2000	2000	2500	3000
workload [GOPS]	0.1	1	10	100	1000
software [MB]	0.1	1	10	100	1000
#programmable cores	1	2	4	8	16

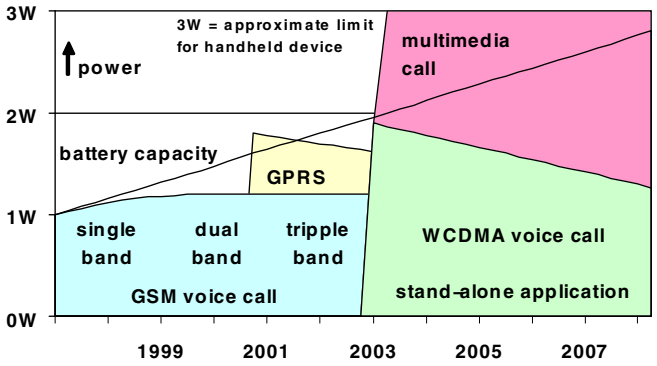

Fig. 1. Battery capacity and power consumption at maximum output power level in cellular transmitters, adapted from [3].

The most remarkable of all these trends is that most functions can run for hours on a single battery charge. Standby times of the phones are even measured in weeks. Today's Li-ion batteries provide about 800mAh at 3.7V, or nearly 3Wh.

Voice-only handsets dissipate about 1W during a call, mostly by the PA. This PA-dominance also explains why the GSM power hardly decreased over time in Fig. 1. In standby mode the RF receiver is dominant, checking with the base station for incoming calls about twice a second (paging). In both modes, all digital functions combined, including memories, consume only about 5% of the total power.

In a smartphone, even more power is required for PA and RF. Also, hundreds of mW are required for the display backlight. All this leaves only about 1W for the digital workload

978-3-9810801-5-5/DATE09 © 2009 EDAA

of 100GOPS. Accordingly, the phone's CPU has to run a full order of magnitude more power efficiently (MHz/W, Table I) than a PC CPU.

This leads to the central theorem of this paper: *the limited available battery power is the dominant determinant of the multi-core architecture for mobile phones*. In the remainder of this paper we analyze the digital workload of a mobile phone, review existing multi-core architectures, and describe several main trends.

II. Workload Characterization

Modern smartphones perform nearly 100GOPS within a power budget of only 1W. Hence, analyzing and exploiting the characteristics of this workload is absolutely essential. Unfortunately, few reports of such analysis have been published.

Load numbers are specified in *algorithmic* GOPS. These numbers provide insight in the relative importance of the many functions embedded in a smartphone. However, beware the huge variation in the energy/operation. Data types vary from 1 to 32b, with most signal processing on 4-16b data. Also, a 16b integer multiplication may consume 1-2 orders of magnitude more energy than an 8b addition in a Galois field.

A. Application processing

Typical functions include the user interface, address books, diaries, sms, java, internet browsing, email, gaming, document and spreadsheet editing, photo handling, limited forms of media processing, etc. For simplicity we equate the workload to the core's clock frequency, about 300 MHz. The amount of application software is about 10-40 MB.

B. Radio processing

The workload for the different cellular, connectivity, broadcast and positioning radios is quite well characterized. The receive path is organized in stages:

- *front-end processing*: mostly filtering and signal conditioning: 1-20GOPS of 10-16b fixed-point arithmetic;
- *demodulation*: channel estimation, equalization, demodulation, synchronization: mostly 16b fixed-point arithmetic; see Fig. 2-top;
- *decoding*: de-interleaving, de-puncturing, decoding (e.g. Viterbi, Turbo, etc. [5]), descrambling: mostly simple operations on 1-8b samples. see Fig. 2-bottom; for HSDPA, Turbo is dominant with a load up to 20GOPS of 4b ops;
- *protocol processing*: mostly control code [several MB], 10-100 of MIPS per radio.

For cellular and connectivity standards there is also a transmit path with typically lower workloads. Front-end, modem, and codec processing are all forms of digital signal processing, with lots of fine-grained, data-level parallelism intrinsically available. The amount of memory is fixed and modest: 10-200kB in total for each radio. Memory access rates are very high, typically exceeding the GOPS rates. Most of this processing is bound by tight real-time constraints: e.g. UMTS power control loops and WLAN packet retransmission have deadlines of hundreds to tens of μs, each involving both the entire receive and transmit paths!

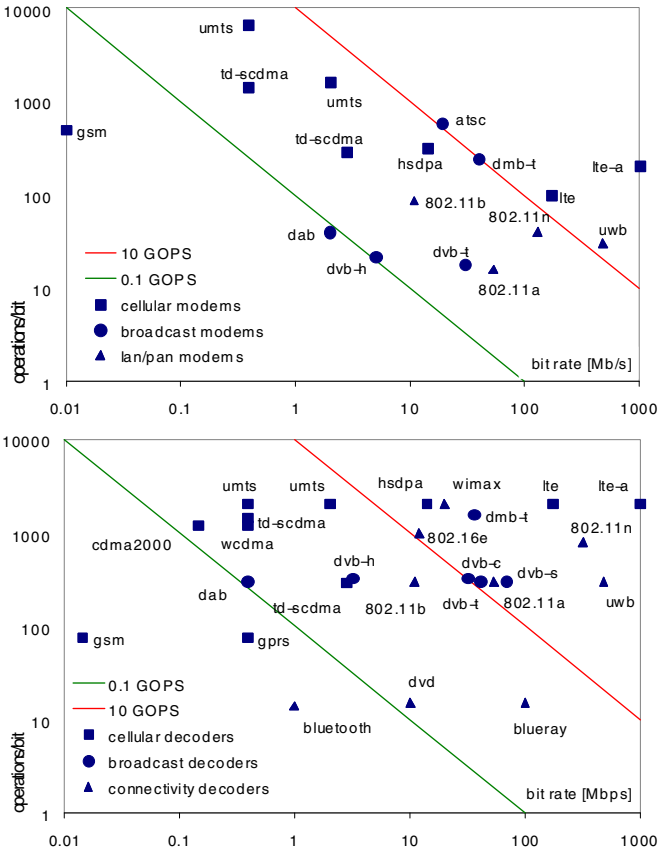

Fig. 2. Radio demodulation [top] and decoding [bottom] workloads.

C. Media Processing

Multi-media processing is equally diverse. Speech and audio codecs involve signal processing on 16b fixed-point data. Both have to deal with a rapid proliferation of codec standards and features to improve sound quality. Workloads are $<$ 0.1GOPS.

For video we assume a VGA-like display resolution ($640 \times 480 = 0.3$M pixels) at 30 frames/second. Signal processing is mostly 8-10b integer arithmetic, and includes [6], [7]:

- *display processing*: scaling, color, contrast&brightness, and sharpness enhancements, etc: 1GOPS;
- *camera processing*: de-mosaicing, color matrix, gamma introduction, etc: 0.2GOPS;
- *video decoding* (H264): entropy decoding, IDCT/iquant, vector decoding, motion compensation, image reconstruction, de-blocking: 2GOPS;
- *video coding* (H264): motion estimation, prediction, DCT/quant, entropy coding, embedded decoding: 17GOPS.

These video functions operate on multiple video frames of 0.5-1MB each, typically stored in an external SDRAM. The combined SDRAM bandwidth may exceed 1GB/s.

3D graphics for mobile phones is based on OpenGL ES, a subset of the OpenGL 3D graphics API designed for Embedded Systems [8]. For VGA at 30frames/s this involves a full suite of geometry operations (24b floating-point) at a speed of about 7M vertices/s. In addition, pixel shading/blending is

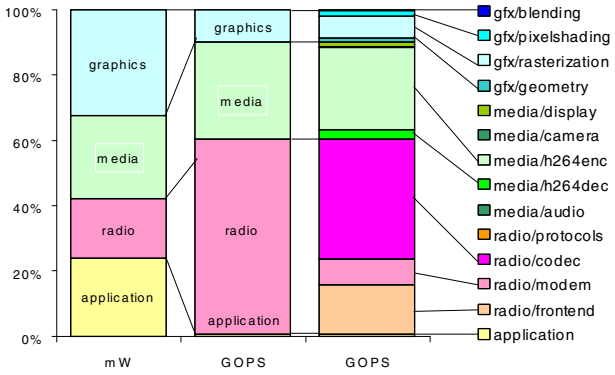

Fig. 3. 3.5G workload (power consumption) as fractions of 100GOPS (1W).

at rate of 28M pixels/s, assuming an overdraw of 3. The main graphics functions are:

- *geometry processing*: 1GOPS (floating point);
- *rasterization*: 4.2GOPS;
- *pixel shading*: 1GOPS (floating point);
- *pixel blending*: 0.3GOPS.

An effective graphics system requires 1GB/s SDRAM bandwidth.

D. Overall workload

The combined *maximum* workload for a 3G smartphone is about 100GOPS, partitioned according to Fig. 3. The power consumption levels are based on the HW architecture of Fig. 5 and includes all memory power, on-chip and SDRAM.

III. 3G Multicore Architectures

Our multicore architecture challenge is to support a workload of 100GOPS with 1W. With RISC processors that support <1 algorithmic operation per clock cycle this workload looks daunting: a single core running at 100GHz is utterly unrealistic, so are 1000 cores running at 100MHz. The only way out is a *heterogeneous* architecture based on less or more specialized programmable cores, and (configurable) function-specific hardware accelerators.

A. Programmability: value and affordability

For each function a careful study is required to decide on the type of processor or accelerator. Key questions are: where does flexibility offer value? Where can flexibility be afforded? What form/degree of flexibility?

The value of flexibility is determined by a number of factors. When applications or functions have to be programmed or downloaded after silicon manufacturing (or in the field), full programmability is a must. Likewise, differentiation of functions and features for different peripherals, products, and markets can best be supported with programmable solutions. Programmability is also valuable:

- when handsets have to support multiple (cellular, connectivity, broadcast) standards;
- when functions comprise a variety of diverse, complex algorithms;
- when changes are expected late in the design cycle (incl. post-silicon debug).

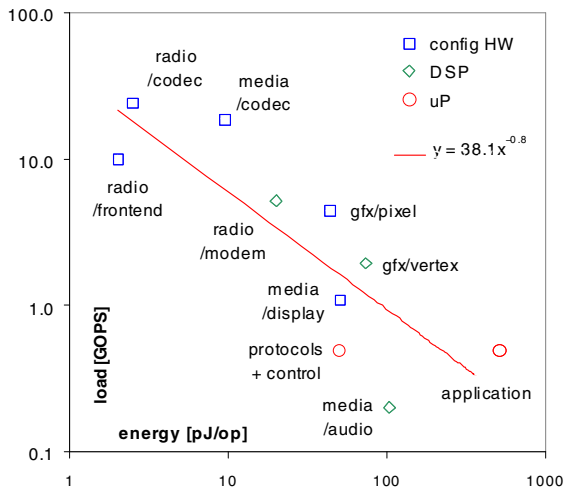

Fig. 4. Workloads [GOPS] versus energy/operation [pJ].

TABLE II
Value-affordability of programmable solutions (examples).

	radio	video	3D graphics
very high	protocol stacks		geometry proces.
high	channel estimation		pixel shading
medium	demodulation	motion estimation	
low	turbo decoder (i)fft	entropy (de)coding deblocking	
very low	filters	filters scaling	rasterization pixel blending

In contrast, programmability is of little value for fixed signal processing, based on stable standards, and on stable, simple algorithms and control.

We also have observed a correlation between code size [kB] and the required flexibility:

- 10MB for application processing;
- 1MB for control and protocol processing;
- 100kB for modem processing;
- few kB for configuring codec processing, filtering, etc.

Apparently, functions that require a high degree of flexibility represent a modest workload, and those with a high workload require little flexibility.

The affordability of flexibility is partly a cost issue (die area), but mostly an issue of power consumption. In Fig. 4, important functions are mapped onto (weakly configurable) HW, programmable DSPs, and ARM processors, in accordance with the required flexibility. Power consumption includes memory power, both on-chip and off-chip (SDRAM). Interestingly, these functions nicely fit on a straight line, following the *power law* $y = c.x^{-0.8}$. This is not a coincidence, but the result of about 15 years of mobile phone evolution, where the power consumption of a heavy function is not allowed to stick out too much. (See also Fig. 3.) Table II shows the value-affordability of the different functions in a smartphone.

B. HW architectures

High volume (low-feature) 2-2.5G handsets today are typically dual-core (Fig. 5-top). Compared to the early dual-core GSM handsets, the clock frequencies and memory sizes have

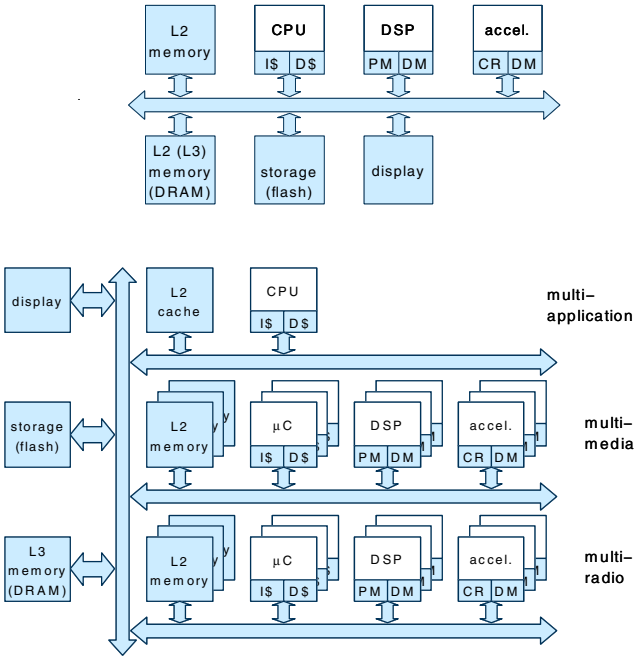

Fig. 5. A 2-2.5G dual-core architecture (top) and a 3-3.5G multi-core architecture (bottom).

increased by an order of magnitude. Communication between the CPU and the DSP is via shared memory, using a simple "mail box" in the on-chip L2 memory [1].

For high-feature 3-3.5G handsets we see more specialization *at two levels*, as schematically depicted in Fig. 5-bottom. Various programmable cores and accelerators are clustered by function category: typically application, modem, and media. There may even be a separate cluster for each radio standard, as well as separate clusters for audio, video, and graphics. Each cluster has specialized processors and accelerators, with a locally optimized memory hierarchy and interconnect. At a next level, these clusters stream data among each other and share access to the off-chip memories, incl. the display memory. Table III provides an overview along these lines of most published cell-phone ICs.

C. Core Specialization

The wide ranges in required flexibility and affordable energy per operation have led to a range of less and more specialized cores. We review them in order of decreasing flexibility and increasing power efficiency.

The application processor (300-500MHz, approaching 1GHz; Table III) handles a rapidly increasing range of applications, where a recent trend suggests that downloading of applications will become more common.

For radio and media processing, often general-purpose, simpler cores are used for handling protocols as well as controlling accelerators and local data flow. Their clock frequencies are typically in line with those of the local accelerators.

For those signal and data processing algorithms where flexibility is valuable and affordable, a wide variety of DSPs is applied. Conventional DSPs operate on 16b fixed point data, and can handle a variety of speech, audio, modem, and codec functions. Many of them have two Multiply ACumulators (MACs), and typically execute 2-4ops/cycle. A recent trend is towards SIMD DSPs with 10-50ops/cycle. See Subsection IV-B. For geometry processing and pixel shading, unified programmable shader DSPs are used [8].

When flexibility is of limited value, or simply not affordable, a wide range of (weakly) configurable accelerators has been proposed. Typical examples include: front-end filter, Turbo decoder, video scaler, pixel blender. These run at 200-300MHz, execute 10-100ops/cycle, and consume 1-10pJ/op (a few 10s incl. SDRAM power), depending on the data types.

D. Memory Hierarchy

A cell phone chip may contain 16Mb of SRAM and ROM in 250 instances [16]. Memory hierarchies follow the diversity and specialization of the cores. Most larger memories are logically unified into a single address space, where L1 memories of most accelerators are kept private.

The L1 memories typically follow the needs of the cores:

- Application processors have L1 caches for instructions and data (2×32kB). Some have an L2 cache (256kB) [15].
- Protocol/control processors often have caches for instructions and sometimes for data as well. However, real-time requirements and interrupt rates of several kHz have typically led to parallel *tightly coupled memories*. These are on-chip SRAMs with *guaranteed* access latencies of a few clock cycles used for both data and instructions.
- DSPs require a higher bandwidth to sustain 2-4ops/cycle. Accordingly, most DSPs are dual-Harvard, with separate memories for instructions (I) and data (X, Y). Hence, data bandwidth is 2 × 16b/cycle, and 64-256b/cycle for SIMD DSPs. Sometimes instructions are cached, data rarely.
- Accelerators require similar or higher bandwidths to support their internal parallelism [5]. Only I/O buffers appear in the global address map.

L2 memories are typically on-chip SRAMs, used to buffer data and to share data among cores and accelerators. Shared L2 caching has been proposed, but not seen in practice yet.

Applications and protocol processing involves many MBs of instructions and data stored in a NAND Flash with a capacity up to 8GB. The high access times of NAND Flash chips (100μs read, 1ms write) lead to architectures where programs are executed from the SDRAM (up to 256MB), after being loaded from the Flash. From this perspective, SDRAM can be seen as L3, and Flash as "L4". Some architectures employ MMUs [15] to optimize SDRAM use.

Traffic in this heterogeneous memory hierarchy is across an equally heterogeneous interconnect structure, mostly based on (multi-layer) buses and bridges. Some traffic is implicit (cached), but most is under explicit program control, using DMA controllers to transport blocks of data among memories.

[1] L1..L3 denote levels in the memory hierarchy, not necessarily cached.

TABLE III
PUBLISHED INDUSTRIAL CELL-PHONE CHIPS ("..." DENOTES A SHARED RADIO-MEDIA CORE).

year	ref	source	cmos	total	application		radio		media	
			nm	# cores	core(s)	MHz	core(s)	MHz	core(s)	MHz
1992	[1]	Philips	1000	1	n.a.		KISS-16-V2	...	...	20
2000	[2]	Infineon	250	2	CPU	78	Oak	...	...	78
2001	[9]	Samsung	180	2	ARM9		Teaklite		n.a.	
2003	[6]	Toshiba	130	3	n.a.		n.a.		3x RISC	125
2004	[3]	Nokia	130	2	CPU	50	DSP	...	...	160
2004	[10]	Qualcomm	130	3	ARM9	180	DSP	95	DSP	95
2004	[11]	Renesas	130	2	CPU	216	n.a.		DSP	216
2005	[12]	NEC	130	4	ARM9	200	n.a.		2xARM9 + DSP	200
2006	[13]	ST	130	2	ARM8	156	ST122 DSP	...	...	156
2007	[14]	Infineon	90	2	ARM9	380	TEAKlite	...	...	104
2008	[15]	Renesas et al	65	4	ARM11	500	ARM9	166	ARM11 + SHX2	500
2008	[16]	TI	45	5	ARM11	840	?		C55 + ?	480
2008	[17]	NEC	65	3	ARM11	500	ARM11	250	DSP	500
2009	[18]	Panasonic	45	4	ARM11	486	ARM11	245	2xDSP	216
2009	[19]	Renesas	65	2	CPU	500	n.a.		SHX2	500

TABLE IV
POWER MANAGEMENT KNOBS: f_C DENOTES CLOCK FREQUENCY, V_{DD} AND V_t DENOTE SUPPLY AND THRESHOLD VOLTAGE, AND P_D AND P_S DENOTE DYNAMIC AND STATIC POWER CONSUMPTION.

	knob		throughput	power
stop the clock:	$f_C \downarrow 0$	$\Rightarrow$	$\downarrow 0$	$P_D \downarrow 0$
frequency scaling (FS)	$f_C \downarrow$	$\Rightarrow$	$\downarrow$	$P_D \downarrow$
voltage scaling (VS)	$V_{DD} \downarrow$	$\Rightarrow$	$\downarrow$	$P_D \downarrow$
power down	$V_{DD} \downarrow 0$	$\Rightarrow$	$\downarrow 0$	$P_S \downarrow 0$
forward body bias (FBB)	$V_t \downarrow$	$\Rightarrow$	$\uparrow$	$P_S \uparrow$
reverse body bias (RBB)	$V_t \uparrow$	$\Rightarrow$	$\downarrow$	$P_S \downarrow$

E. Power Management

Power management is about controlling a variety of *knobs* to minimize power consumption for an increasingly diverse set of increasingly dynamic use cases. For the state of the art of this crucial technology, as applied to mobile phones ICs, see [12], [16], [17], [20]–[24]. These control knobs are described in Table IV, and apply to a dozen or so *power domains*. Many of these knobs can be applied *dynamically* and even *adaptively* by measuring the performance, and controlling the knobs accordingly, in order to achieve performance objectives. For example, AVFS: adaptive voltage-frequency scaling.

In [20] 20 power domains are defined in a hierarchical fashion. Design rules and interface circuits are required to maintain signal integrity of clocks and on inter-domain wires [20], [24]. Typically, a range of *power states* is defined for the entire ICs or for major parts, e.g. deep power down, power down, deep sleep, sleep, power save, normal, with a set of allowed state transitions [12].

IV. TOWARDS 4G

Important trends for the next 5 years *from a multi-core perspective* include: the continuation of 10×workload/5years, software-defined radio, and an increase in architecture modularity with standardized interfaces.

A. Long-Term Evolution

3GPP is preparing LTE (Long-Term Evolution), a new cellular standard that aims at higher bit rates, lower cost per bit, and a higher quality of service [25]. Given these higher bit rates and more complex signal processing algorithms [26], the work load may increase another 5×. Its successor standard LTE-a (LTE-advanced), to be defined by 2011, will likely add an additional factor 10×. Similarly, increased display resolution, up to HD [19], may lead to a corresponding increase in work load for media and graphics processing.

B. Software Defined Radio

High-end mobile phones today support multiple radio standards: e.g. GSM, EDGE/GPRS, UMTS, HSPDA/HSUPA, BlueTooth, 802.11b/g, GPS, DVB-T/H. Typically this involves 5-6 dedicated HW solutions. Predicting when, where, and which radio *combinations*, from an increasingly rich set of standards, are viable in the market place, is challenging and risky. Furthermore, as not all radios are active at the same time, there is a potential for cost saving.

Software Defined Radio (SDR) is a hot research topic[2], moving towards the market place [27] [28] [29] [30] [31]. In its most ambitious form, SDR specifies a multi-core *radio computer* [32], allowing dynamic loading, starting, pausing, and stopping of radio-application images. A radio has become a SW entity, not unlike a regular computer application, that can be compiled onto a multi-core radio computer. Real-time multi-core resource management is challenging, but doable [32]. Most SDR papers cited above propose a combination of a programmable modem controller, multiple 8-16× SIMD DSPs, and multiple configurable accelerators.

C. Self-contained subsystems

In Section III-B we already identified a two-level architecture, in which functionally related cores, accelerators, and L2 memories are clustered. With more of such clusters and with their complexity increasing, it becomes increasingly important to define simpler, more precise, and more abstract interfaces.

This would lead to well-defined subsystems, designed by specialists, each with its own life cycle, and potentially traded by competing suppliers. To simplify system integration, by simply mixing and matching subsystems, it will be important to *standardize* the subsystem interfaces, essentially according to an OSI-layered protocol stack, with a PHY (both off-chip

[2]Here we focus on the digital baseband processing; RF transceivers, front-end modules, and antennas for SDR are equally challenging.

and on-chip), a link layer, a mac layer, a transport layer, and an application layer (possibly differentiated for radio, video, graphics, display, cameras) [33].

Ideally, such subsystems are self-contained in terms of resources and (low-level) resource management. Sharing of scarce resources among subsystems, such as off-chip memories, has to be governed by protocols. From a software perspective, such subsystems are *containers of services*, where applications invoke the services provided by these subsystems.

V. Conclusion

The challenge is to provide 100GOPS within a 1W power budget. The solution has to be *multicore*, as multiple small cores at a lower clock frequency are more power efficient [34].

Asymmetric multicore also makes sense. Amdahl's Law describes the limits to the effectiveness of parallelism. Hence, a few relatively powerful cores complement the multi-core parallelism [35] [36], despite the lower area and power efficiency of these larger cores (Pollacks rule [34]). However, Amdahl's Law assumes that the sequential code and the parallel code do *not* run simultaneously. This is not true for mobile phones. E.g. the application processor runs in parallel with the radio and media processing. Also, the protocol and baseband processing work in parallel (pipelined).

Above all, mobile phones needs *heterogeneous multi-core*, exploiting specialization of cores, memory hierarchy, interconnect, and power mngt. This specialization makes mapping of tasks simple, if not trivial. However, multi-core real-time constraints makes the scheduling of these tasks challenging.

As summarized in Table I, we expect workloads to continue to increase faster than the clock frequencies, leading to more and more programmable cores. Forced by limited battery power, cell phones and cell-phone ICs have to adhere systematically and firmly to the classic architecture dictum *form follows function*.

Acknowledgment

I am indebted to John Dielissen, Bram Riemens, and Om Prakash Gangwal for their contributions on radio codecs, video processing, and graphics processing. Together with Pieter van der Wolf they also reviewed the manuscript.

References

[1] D. Weinsziehr *et al.*, "Kiss-16v2: A one-chip asic dsp solution for gsm," *IEEE J. Solid-State Circuits*, vol. 27, no. 7, pp. 1057–1066, 1992.

[2] G. Weinberger, "The new millennium: Wireless technologies for a truly mobile society," in *ISSCC Dig. Tech. Papers*, 2000, pp. 20–24.

[3] Y. Neuvo, "Cellular phones as embedded systems," in *ISSCC Dig. Tech. Papers*, 2004, pp. 32–37.

[4] M. Hammes *et al.*, "Deep submicron cmos technology enables system-on-chip for wireless communications ics," *Communications Magazine, IEEE*, vol. 46, no. 9, pp. 154–161, 2008.

[5] Dielissen *et al.*, "Multistandard fec decoders for wireless devices," *IEEE Trans. on Circuits and Systems*, vol. 55, no. 3, pp. 284–288, 2008.

[6] H. Arikida *et al.*, "A 160mw, 80na standby, mpeg-4 audiovisual lsi with 16mb embedded dram and a 5gops adaptive post filter," in *ISSCC Dig. Tech. Papers*, 2003, pp. 42–43.

[7] H.-C. Chang *et al.*, "A 7mw-to-183mw dynamic quality-scalable h.264 video encoder chip," in *ISSCC Dig. Tech. Papers*, 2007, pp. 280–281.

[8] J. Sohn, "Low-power 3d graphics processors for mobile terminals," *IEEE Communications Magazine*, vol. 43, no. 12, pp. 90–99, 2005.

[9] S. Han *et al.*, "A mobile station modem vlsi for cdma2000-1x," in *IEEE ASIC/SOC Conference*, 2001, pp. 52–56.

[10] G. Uvieghara *et al.*, "A highly-integrated 3g cdma2000 1x cellular baseband chip with gsm/amps/gps/bluetooth/multimedia capabilities and zif rf support," in *ISSCC Dig. Tech. Papers*, 2004, pp. 422–423.

[11] T. T. Kamei *et al.*, "A resume-standby application processor for 3g cellular phones," in *ISSCC Dig. Tech. Papers*, 2004, pp. 336–337.

[12] S. Torii *et al.*, "A 600mips 120mw 70μa leakage triple-cpu mobile appl. processor chip," in *ISSCC Dig. Tech. Papers*, 2005, pp. 136–137.

[13] A. Cofler *et al.*, "A reprogrammable edge baseband and multimedia handset soc w 6 mb emb. dram," *jssc*, vol. 41, no. 1, pp. 97–106, 2006.

[14] T. Lueftner *et al.*, "A 90-nm cmos low-power gsm/edge multimedia-enh. baseband processor with 380-mhz arm926 and mixed-signal extensions," *IEEE J. Solid-State Circuits*, vol. 42, no. 1, pp. 134–144, 2007.

[15] M. Naruse, "A 65nm single-chip application and dual-mode baseband processor with partial clock activation and ip-mmu," in *ISSCC Dig. Tech. Papers*, 2008, pp. 260–261.

[16] G. Gammie *et al.*, "A 45nm 3.5g baseband-and-multimedia application processor using adaptive body-bias and ultra-low-power techniques," in *ISSCC Dig. Tech. Papers*, 2008, pp. 258–259.

[17] S. Kunie *et al.*, "Low power architecture and design techniques for mobile handset lsi medity^{TM}," in *Proc. ASP-DAC*, 2008, pp. 748–753.

[18] M. Shirasaki *et al.*, "A 45nm single-chip application-and-baseband processor using an intermittent operation technique," in *ISSCC Dig. Tech. Papers*, 2009, p. 8.6.

[19] K. Iwata *et al.*, "A 342mw mobile application processor with full-hd multi-standard video codec," in *ISSCC Dig. Tech. Papers*, 2009, p. 8.7.

[20] T. Hattori *et al.*, "Hierarchical power distribution+mngt scheme for single chip mobile processor," in *ACM/IEEE DAC*, 2006, pp. 292–295.

[21] M. Ito *et al.*, "A 390mhz single chip application and dual-mode baseband processor in 90nm triple-vt cmos," in *ISSCC Dig. Tech. Papers*, 2007, pp. 274–275.

[22] H. Mair *et al.*, "A 65nm multimedia application processor with an adaptive power management scheme to compensate for variations," 2007, pp. 224–225.

[23] M. Nakajima *et al.*, "Low power techniques for mobile application socs based on integrated platform "uniphier"," *Design Automation Conference, ASP-DAC '07.*, pp. 649–653, 2007.

[24] N. Wingen, "What if you could design tomorrow's system today," in *DATE*, 2007, pp. 1–6.

[25] E. Dahlman *et al.*, *3G Evolution: HSPA and LTE for Mobile Broadband (2nd edition)*. Academic Press, 2008.

[26] Berkmann *et al.*, "On 3g lte terminal implementation - standard, algorithms, complexities and challenges," *IWCMC '08*, pp. 970–975, 2008.

[27] K. van Berkel *et al.*, "Vector processing as an enabler for software-defined radio in handheld devices," *EURASIP Journal on Advances in Signal Processing*, vol. 2005, no. 16, pp. 2613–2625, 2005.

[28] U. Ramacher, "Software-defined radio prospects for multistandard mobile phones," *IEEE Computer*, vol. 40, no. 10, pp. 62–69, 2007.

[29] B. Beheshti, "Performance analysis of the wimax-d physical layer blocks on a next generation baseband processor platform," in *WSEAS Int. Conf. on Communications*, 2008, pp. 95–100.

[30] T. Limberg, "A fully programmable 40 gops sdr single chip baseband for lte/wimax terminals," in *ESSCIRC*, 2008, pp. 466–469.

[31] Y. Lin, "Realizing software defined radio a study in designing mobile supercomputers," Ph.D. dissertation, The University of Michigan, 2008.

[32] A. Ahtiainen *et al.*, "Scheduling and resource sharing on a sw defined radio computing platform," in *SDR'08 (SDR Forum)*, 2008, pp. 1.6–2.

[33] Wikipedia, "Unipro," *http://en.wikipedia.org/wiki/Unipro#References*, 2008.

[34] S. Borkar, "Thousand core chips – a technology perspective," in *ACM/IEEE DAC*, 2007, pp. 746 – 749.

[35] M. Hill and M. Marty, "Amdahl's law in the multicore era," *Computer, IEEE*, vol. 41, no. 7, pp. 33–38, 2008.

[36] D. Woo *et al.*, "Extending amdahl's law for energy-efficient computing in the many-core era," *Computer, IEEE*, vol. 41, no. 12, pp. 24–31, 2008.

[37] L.-F. Ding *et al.*, "A 212mpixels/s 4096x2160p multiview video encoder chip for 3d/quad hdtv appls," in *ISSCC Dig. Tech. Papers*, 2009, p. 8.5.

[38] M. Hammes *et al.*, "Evolution on soc integration: Gsm baseband-radio in 0.13 μm cmos extended by fully integrated power management unit," *jssc*, vol. 43, no. 1, pp. 236–245, 2008.

[39] Y. Kanno *et al.*, "Hierarchical power distribution with power tree in dozens of power domains for 90-nm low-power multi-cpu socs," *IEEE J. Solid-State Circuits*, vol. 42, no. 1, pp. 74–83, 2007.