Energy-delay efficient data storage and transfer architectures: circuit technology versus design methodology solutions

Francky Catthoor, IMEC, Kapeldreef 75, Leuven, Belgium [1]

Abstract

Both in custom and programmable instruction-set processors for data-dominated multi-media applications, many of the architecture components are intended to solve the data transfer and storage issues. Recent experiments at several locations have clearly demonstrated that due to this fact, the main power (and largely also area) cost is situated in the memory units and the communication hardware. In this paper, the main reasons for this problem will be reviewed and a perspective will be provided on the expected near-future evolution. It will be shown that the circuit and process technology advances have been very significant in the past decade. Still, these are not sufficient to fully solve this power and area bottle-neck which has been created in the same period. Therefore, also several possible design methodology remedies will be proposed for this critical design issue, with emphasis on effective system-level memory management methodologies. These promise very large savings on energy-delay also on area for multi-media applications, while still meeting the real-time constraints.

1 Context and summary

The target domain in this paper includes all real-time data-dominated applications, which deal with large amounts of complex data types. This happens both in real-time multi-dimensional signal processing (RMSP) applications such as video and image processing (which handle indexed array signals, usually in the context of loops), and in sophisticated communication network protocols (which handle large sets of records organized in tables and pointers). Both classes of systems contain many important applications, e.g. video coding, medical image archival, multi-media terminals, artificial vision, xDSL modems, ATM networks, and LAN/WAN technology.

The top-level view of a typical heterogeneous system architecture in our target application domain is illustrated in Fig. 1. Architecture experiments have shown that 50-80% of the area cost in (application-specific) architectures for real-time multi-dimensional signal processing is due to *memory units*, i.e. single or multi-port RAMs, pointer-addressed memories, and large register files [39, 19]. The power cost is even more heavily dominated by storage and transfers for complex data types. This has been demonstrated both for custom (HW) [21, 7] (see also Fig.2) and for programmable instruction-set (SW) processors [38, 15]. The conclusion is that an off-chip data transfer consumes about 33 times more power than a typical 16-bit arithmetic operation like an addition. Experiments on video compression applications and other advanced multi-media applications indicate that the number of primitive arithmetic operations is typically only a few times higher than the number of data transfer operations to big frame signals. For instance, for a H.263 video conferencing decoder [16], this ratio is 1502K/503K for a representative stream of frames, or about a factor 3 [24]. Combined this provides at least a factor 10 in difference for the power consumption in the actual application of the original algorithm code. Similar observations can be made for modern instruction-set processors in the multi-media domain.

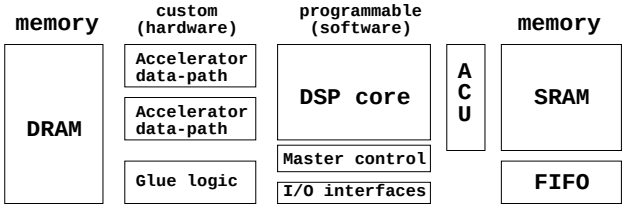

Figure 1: Typical heterogeneous system-on-a-chip architecture for multi-media applications, with custom hardware (application-specific accelerator data-paths and logic), programmable processor (DSP core and controller), and a cost-dominant distributed memory organisation.

Hence, for our target applications, the organisation of the global data transfer and storage forms the dominating factor in the system-level architecture design decisions. The required memory units and the (bus) communication hardware between them[2], are clearly power dominant in

[1] Also Professor at EE Dep. of K.U.Leuven.

[2] together with their use for a given application (which is as important)

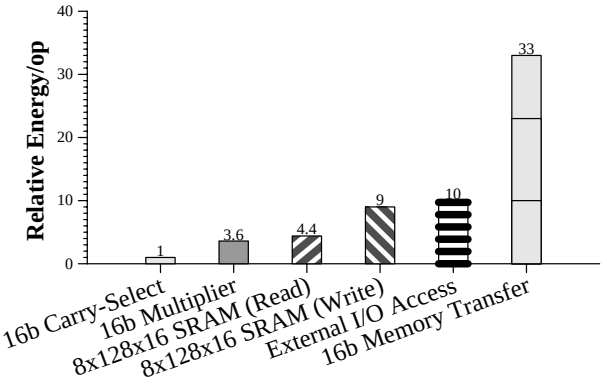

Figure 2: Demonstration of dominance of transfer and storage over data-path operations in custom HW architectures (based on [21, 35]).

the conventional approaches. Section 2 provides a summary of current design practice and a more detailed illustration of this issue. Unfortunately, in more and more cases the power consumption becomes a bottleneck, especially in mobile or embedded applications. This is due both to battery life times and restrictions on packaging cost or reliability issues. We will show in section 3 that the process and circuit technology evolution will not fully solve this problem. This holds in particular when the energy-delay product is taken into account: for a given performance (delay requirement), the necessary energy consumption is increasing very rapidly[3]. Therefore, in section 4, we will show that also design methodology innovations are required. Recently, on the HW side, several system level memory management related methodologies are being proposed which promise very large savings on power [7, 24] and also on area (e.g. [19]) while still meeting the real-time constraints. Experimental results will demonstrate that these can effectively solve the energy-delay bottleneck to a large extent.

2 Multi-media storage organisations

It is impossible to summarize all architecture solutions in the full multi-media and telecom domain. Therefore, a specific but representative subapplication domain will be selected and analyzed, namely (MPEG based) video compression. Many custom HW architectures for motion estimation and other MPEG subsystems have been proposed (see good survey in [30]). Power management and reduction is becoming a major issue for such applications [6, 9, 21].

Two examples of MPEG2 custom processors which have been presented at CICC'95 (using comparable CMOS

[3]In this paper, we will focus mostly on the energy/power issue for a given fixed delay requirement but it should be kept in mind that also then the real problem is associated with the energy-delay product.

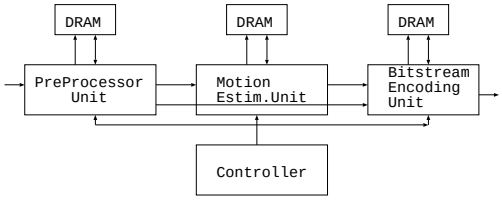

Figure 3: Global MPEG2 architecture of [2].

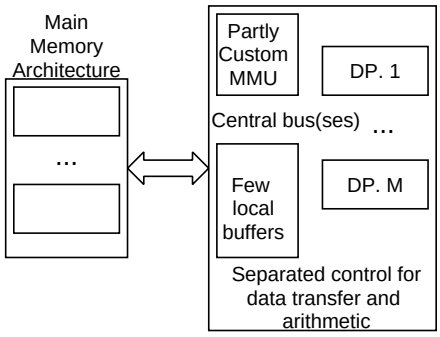

Figure 4: Current architecture model: HW

technologies) will be reviewed. It will be shown that both have a widely differing data transfer and storage organisation. The differences will illustrate an important point[4]. The first design [2] exhibits the following characteristics (see also figure 3): 4 chip set, 20W @27MHz, flexible centralized bus/memory organisation, 64 Mbit external RAM. The pros and cons are:

- Enhanced flexibility to update this design in a modular way due to the central bus structure.
- Ease of design due to the divide and conquer strategy enabled by hooking up additional modules to the bus with almost fully decoupled interaction (system level buffers).
- Larger system cost due to extra system-level buffering (resulting in extra area on-chip or extra space on the board).
- Higher power consumption due to decoupling and central bus/memory organisation (see also below).

Also today, the majority of the complex system designs follow a similar strategy (see figure 4), probably mainly driven by the attractive design time and modularity issues. They exhibit rather centralized memories and busses per processor with system-level buffers to separate the different processors. An alternative for this is to have a more

[4]These designs are only representatives of the main trends and the choice of these particular instances is only due to the fact that they can be easily compared and that they have been published with sufficient information.

distributed architecture which is more customized to the application at hand. The pros and cons are the opposite of the above list: reduced power and area/space related costs at the price of a higher design time because of the increased design complexity. This class of designs is illustrated relatively well with another MPEG2 encoder [20, 26] (see also figures 5 and 6). Its characteristics are: 3 chip set, 8.5W @8 MHz, heavily customized and distributed organisation, 44 Mbit external DRAM. As a result, the reduced flexibility and ease of design have been traded-off against a lower power budget and memory size.

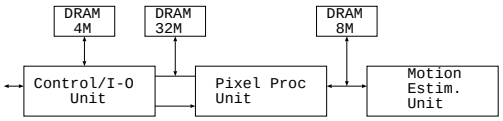

Figure 5: Global MPEG2 architecture of [20].

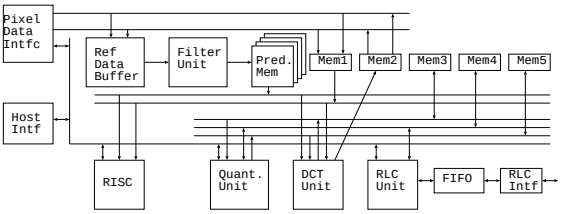

Figure 6: Global Pixel Proc. architecture of [26].

Similar observations can be made for SW processors:

1. The power consumption of processors is rising sharply and moving rapidly into storage (data and program memory) and partly also transfers (communication and I/O) [38, 15].

2. Using a simple hardware instruction cache, the microcode contribution in the power budget can be kept negligible for multi-media applications because these mainly consist of heavily nested loops with relatively large bounds. So most of the time, the link to the main program memory can be powered down. In order to fully profit from this, it is assumed however that the cache is distributed. The different partitions should be localized as close as possible to the places on the chip where the decoded control bits have to be used in the processor, i.e. close to the FU where the involved instruction field applies to. Otherwise, the distribution of a wide control bus (especially for VLIW) will still absorb too much on-chip power. Care should be exercised here to avoid unnecessary duplicate decoding overhead locally at the FUs, so a good clustering of the FUs sharing a local cache and its corresponding decoder should be aimed at.

In conclusion, the power consumption in well-designed processor architectures resides mainly in the data storage and transfer components. Actually, the same is true for the cycle overhead (performance) due to the usually quite dominant effect of the data cache stalls (see e.g. [4] and its refs).

3 Evolution of circuit and process technology solutions

Already a decade ago, the growing performance gap between the processor and the off-the-shelf (D)RAM evolution became a major issue (see overview of evolution in [10] and [31]). Also the increasing power consumption (up to several W for large DRAM and SRAM chips) became a major concern. Since then, a significant effort has been invested in better solutions at the circuit and process technology levels [17]. In order to reduce the energy-delay product, the main changes are related to the more aggressive partitioning in hierarchical memory planes[5], wide memory words to reduce the access speed [1], multi-divided arrays (both for word-line and data-line) with up to 1024 divisions in a single matrix [37], low on-chip Vdd (up to 0.5V [41]) with Vdd/2 precharge [41], special voltages on the word-line to reduce the leakage current, special NAND decoders [22], reduced bit/word-line swing (up to only 10% of the Vdd) [17], differential bus drivers and charge recycling in the I/O buffer [22].

Because of all these principles to distribute the power consumption from a few “hot spots” to all parts of the architecture, the end result is indeed a very optimized design for power where every piece consumes about an equal amount (see e.g. [35]). It is expected however that not much more can be gained because the “bag of tricks” is now containing only the more complex solutions with a smaller return-on-investment. Note however that the combination of all these approaches indicates a very advanced circuit technology which clearly outperforms the current state-of-the-art in data-path and logic circuits for low power design. Hence, It can be expected that the relative power in the non-storage parts can be more drastically reduced still towards the future (on condition that similar investments are done). Combined with the advance in process technology, all this had lead to a remarkable reduction of the DRAM related power: from several W for the 16-32 Mb generation to about 100 mW for 100 MHz operation in a 256 Mb DRAM (see e.g. [23]).

It can be concluded that modern stand-alone (S)DRAM chips already offer low power solutions but this comes at a price. Internally they contain banks and a small cache with a very wide width (see figure 7). So the low power operation per bit is only feasible when they operate in burst

[5] with usually more than 32 divisions for RAM sizes above 16Mb

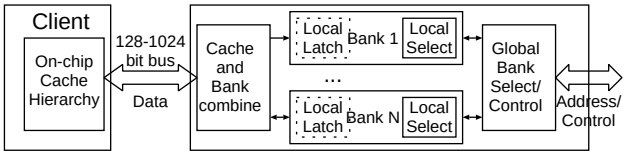

Figure 7: Off-chip memory organisation in modern (S)DRAMs.

mode with large data widths. This is not directly compatible with the actual use of the data in the processor data-paths so without a buffer to the processors, most of the bits which are exchanged would be useless (and discarded). Obviously, the effective energy consumption per useful bit would become very high in that case and also the effective bandwidth is quite low. Therefore, a hierarchical and typically much more power-hungry intermediate memory organisation is needed to match the central DRAM to the data ordering and bandwidth requirements of the processor data-paths. This is illustrated in figure 8. The decrease of the power consumption in fast random-access memories is not as advanced yet as in DRAMs but also that one is saturating, because many circuit and technology level tricks have already been applied also in SRAMs. As a result, fast SRAMs keep on consuming on the order of Watt's for high-speed operation around 500 MHz [5, 27].

From the process technology point of view this is not so surprising, especially for submicron technologies. The relative power cost of interconnections is increasing rapidly compared to the transistor (active circuit) related components. Clearly, local data-path and controllers themselves contribute little to this overall interconnect compared to the major data/instruction busses and the internal connections in the large memories. Hence, if all other parameters remain constant, the energy consumption (and also the delay or area) in the storage and transfer organisation will become even more dominant in the future, especially for deep submicron technologies. The remaining basic limitation lies in transporting the data and the control (like addresses and internal signals) over large on-chip distances, and in storing them.

One last technological recourse to alleviate the energy-delay bottleneck is to embed the memories as much as possible on-chip. The other two articles in this special session and also several recent activities at other locations (see e.g. the Mitsubishi announcements [34] and the IRAM initiative of U.C.Berkeley [29]) motivate that the option of embedding logic on a DRAM process leads to a reduced power cost and an increased bandwidth between the central DRAM and the rest of the system. This is indeed true for applications where the increased processing cost is allowed. However, it is a one-time drop after which the widening energy-delay gap between the storage and the logic will keep on progressing, due to the unavoidable evolution of the relative interconnect contributions (see above). So on the longer term, the bottleneck should be broken also by other means. In section 4, it will be shown that this is feasible with quite spectacular effects at the level of the system design methodology. The price paid there will be increased design complexity, which can however be offset with appropriate design methodology support tools.

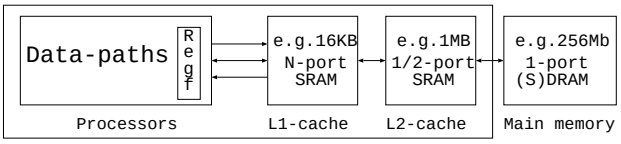

Figure 8: On-chip memory hierarchy to match DRAM to the processor data ordering and bandwidth requirements.

4 Energy-delay efficient data storage and transfer exploration

In order to achieve a significant drop of the actual energy consumed for the execution of a given application, we have to look again into the real power formula: $F_{real}.V_{dd}.V_{swing}.C_{load}$. Note that the real access rate F_{real} should be provided and not the maximum frequency F_{cl} at which the RAM can be accessed. The maximal rate is only needed to determine whether enough bandwidth is available for the investigated array signal access. Whenever the RAM is not accessed (and is idle), it should be in power-down mode. Modern memories support this low-power mode (see e.g. [35]).

We can now reduce both F_{real} and the effective C_{load} by the way we use the storage and interconnect devices. The goal should be to arrive at a memory and transfer organisation with the following characteristics:

1. Reduce the redundancy in data transfers.
2. Introduce more locality in the accesses so that more data can be retained in registers local to the data-paths.
3. Use a hierarchical memory organisation where the smaller memories (with reduced C_{load}) are accessed the most and the larger ones are accessed the least.
4. Avoid the use of N-port memories if 1-port alternatives with a reasonable cost can be used instead, because more area but also more interconnect (and hence energy-delay) is required for these multi-port components.

Also the paper by N.Wehn and S.Hein in this special paper session for motivates several of these issues.

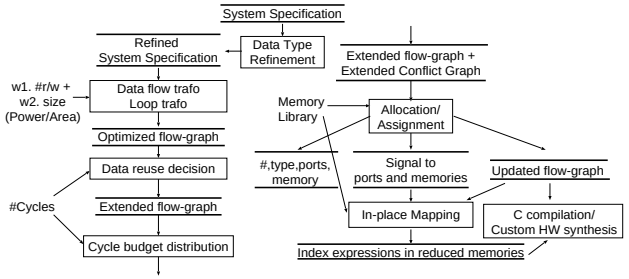

Figure 9: Methodology for energy-delay efficient system design exploration for data storage and transfers in multimedia applications.

These principles can be enabled by applying an appropriate system design methodology. General principles are proposed in papers like [32] but for our particular situation we require a more customized approach. A script for supporting our objective has been developed at IMEC for this purpose [3, 7, 8, 40].
The main steps involve (see also figure 9):

1. Reducing the required data bit-widths for storage by applying optimizing algorithmic transformations. These can change the actual algorithm choice, e.g. in digital filters or in the choice DFT versus FFT. Much can be done also in the step from floating-point data types to fixed-point types (see e.g. [8]).

2. Reducing transfer redundancy by data-flow transformations which do not modify the global input-output functionality [8].

3. Increasing the locality by global loop transformations [12, 18].

4. Exploit memory hierarchy better by an optimized decision on reuse of multi-dimensional array data [14].

5. Balanced distribution of the globally available cycle budget over the loop nests such that the maximally required band-width is reduced [36]. This ensures that fewer multi-port memories are required and also fewer memories overall.

6. Allocate a more distributed memory organisation [19] and assign the array signals cost-effectively to the different memories at each hierarchical level [36].

7. Exploit the restricted life-times of parts of the array signals to overlap them in the address space (in-place mapping) [13]. This enables a better exploitation of the caches and reduces the overall size of the required memories (and hence C_{load}). Also the offsets of data in caches can be affected at this level to reduce the number of stalls [28].

Using this methodology, surprising results can be obtained. We have demonstrated a factor 9 reduction in maximal power for the worst-case mode of a H.263 video conferencing decoder [24] by exploiting only steps 2-4. In addition, three orders of magnitude of maximal power reduction in the storage architecture have been obtained for mapping a quad-tree structured DPCM video codec on a set of parallel processors (see [11] for partial results) based on a script with steps 2-7 above combined with a better parallelisation strategy. This reorganisation has a positive impact on all cost parameters. The only real penalty is the increased design complexity which heavily impacts the design time without appropriate tool support. It should be stressed however that the work has only started and that much more research is required in this area to arrive at usable results which can be applied in different application domains. This is a real opportunity for interesting academic research with relevant industrial impact potential.

Acknowledgments: the discussions with colleagues in IMEC-VSDM are gratefully acknowledged and especially the contributions of L.Nachtergaele, S.Wuytack, E.De Greef, K.Danckaert, C.Kulkarni, S.Janssens, M.Miranda.

References

[1] B.Amrutur, M.Horowitz, "Techniques to reduce power in fast wide memories", *Symp. on Low-power Electronics*, 1994.

[2] J.Armer et al., "A chip set for MPEGF2 video encoding", *Proc. IEEE Custom Integrated Circuits Conf.*, Santa Clara CA, pp.401-404, May 1995.

[3] More information and related papers are available at the memory management related http: web site of VSDM: `//www.imec.be/vsdm/projects/atomium/index.html`.

[4] P.Balietto, M.Maresca, M.Migliardi, N.Zingirian, "Image processing on high-performance RISC systems", *Proc. of the IEEE*, invited paper, Vol.84, No.7, pp.917-930, July 1996.

[5] G.Braceras, D.Evans, J.Sousa, J.Connor, "A 350 MHz 3.3V 4Mb SRAM fabricated in a 0.3 μm CMOS process", *Proc. IEEE Int. Solid-State Circ. Conf.*, San Francisco CA, pp.404-405, Feb. 1997.

[6] R.W.Brodersen, "The network Computer and its Future," *Proc. IEEE Int. Solid-State Circ. Conf.*, San Francisco CA, pp.32-36, Feb. 1997.

[7] F.Catthoor, F.Franssen, S.Wuytack, L.Nachtergaele, H.De Man, "Global communication and memory optimizing transformations for low power signal processing systems", *IEEE workshop on VLSI signal processing*, La Jolla CA, Oct. 1994. Also in *VLSI Signal Processing VII*, J.Rabaey, P.Chau, J.Eldon (eds.), IEEE Press, New York, pp.178-187, 1994.

[8] F.Catthoor, S.Wuytack, E.De Greef, F.Franssen, L.Nachtergaele. H.De Man, "System-level transformations for low power data transfer and storage", in paper collection on "Low power CMOS design" (eds. A.Chandrakasan, R.Brodersen), IEEE Press, pp.609-618, 1998.

[9] P.Chatterjee (President Personal Productivity Products, Texas Instruments), "Gigachips: deliver affordable digital multi-media for work and play via broadband network and set-top box", Plenary paper in *Proc. IEEE Int. Solid-State Circ. Conf.*, San Francisco CA, pp.26-30, Feb. 1995.

[10] R.Comerford, G.Watson (eds.), "Memory catches up", *IEEE Spectrum*, pp.34-57, Oct. 1992.

[11] K.Danckaert, F.Catthoor, H.De Man, "System level memory optimization for hardware-software co-design", *Proc. IEEE Intnl. Workshop on Hardware/Software Co-design*, Braunschweig, Germany, pp.55-59, March 1997.

[12] E.De Greef, F.Catthoor, H.De Man, "Memory organization for video algorithms on programmable signal processors", *Proc. IEEE Int. Conf. on Computer Design*, Austin TX, pp.552-557, Oct. 1995.

[13] E.De Greef, F.Catthoor, H.De Man, "Memory Size Reduction through Storage Order Optimization for Embedded Parallel Multimedia Applications", *Intnl. Parallel Proc. Symp.(IPPS)* in Proc. Workshop on "Parallel Processing and Multimedia", Geneva, Switzerland, pp.84-98, April 1997.

[14] J.P.Diguet, S.Wuytack, F.Catthoor, H.De Man, "Formalized methodology for data reuse exploration in hierarchical memory mappings", *Proc. IEEE Intnl. Symp. on Low Power Design*, Monterey, pp.30-35, Aug. 1997.

[15] R.Gonzales, M.Horowitz, "Energy dissipation in general-purpose microprocessors", *IEEE J. Solid-state Circ.*, Vol.SC-31, No.9, pp.1277-1283, Sep. 1996.

[16] Digital Video Coding at Telenor R & D, "Telenor's H.263 Software", Version 3.1, *http://www.nta.no/brukere/DVC/h263_software/* .

[17] K.Itoh, K.Sasaki, Y.Nakagome, "Trends in low-power RAM circuit technologies", special issue on "Low power electronics" of the *Proceedings of the IEEE*, Vol.83, No.4, pp.524-543, April 1995.

[18] D.Kolson, A.Nicolau, N.Dutt, "Minimization of memory traffic in high-level synthesis", *Proc. 31st ACM/IEEE Design Automation Conf.*, San Diego, CA, pp.149-154, June 1994.

[19] P.Lippens, J.van Meerbergen, W.Verhaegh, A.van der Werf, "Allocation of multiport memories for hierarchical data streams", *Proc. IEEE Int. Conf. Comp. Aided Design*, Santa Clara CA, Nov. 1993.

[20] T.Matsumura et al., "A chip set architecture for programmable real-time MPEG2 video encoder", *Proc. IEEE Custom Integrated Circuits Conf.*, Santa Clara CA, pp.393-396, May 1995.

[21] T.H.Meng, B.Gordon, E.Tsern, A.Hung, "Portable video-on-demand in wireless communication", special issue on "Low power electronics" of the *Proceedings of the IEEE*, Vol.83, No.4, pp.659-680, April 1995.

[22] H.Morimura, N.Shibata, "A 1V 1MB SRAM for portable equipment", *Proc. IEEE Intnl. Symp. on Low Power Design*, Monterey, pp.61-66, Aug. 1996.

[23] K.Murakami, S.Shirakawa, H.Miyajima, "Parallel processing RAM chip with 256 Mb DRAM and quad processors", *Proc. IEEE Int. Solid-State Circ. Conf.*, San Francisco CA, pp.228-229, Feb. 1997.

[24] L.Nachtergaele, F.Catthoor, B.Kapoor, D.Moolenaar, S.Janssens, "Low power storage exploration for H.263 video decoder", *IEEE workshop on VLSI signal processing*, Monterey CA, Oct. 1996. Also in *VLSI Signal Processing IX*, W.Burleson, K.Konstantinides, T.Meng, (eds.), IEEE Press, New York, pp.116-125, 1996.

[25] L.Nachtergaele, D.Moolenaar, B.Vanhoof, F.Catthoor, H.De Man, "System-level power optimization of video codecs on embedded cores : a systematic approach", accepted for special issue on *Future directions in the design and implementation of DSP systems* (eds. Wayne Burleson, Konstantinos Konstantinides) of *Journal of VLSI Signal Processing*, Kluwer, Boston, Feb. 1998.

[26] S.Nakagawa et al., "A single chip 5 GOPS macroblock-level pixel processor for MPEG2 real-time encoding", *Proc. IEEE Custom Integrated Circuits Conf.*, Santa Clara CA, pp.397-400, May 1995.

[27] K.Nakamura, et al., "A 500 MHz 4Mb CMOS pipeline-burst cache SRAM with point-to-point noise reduction coding I/O", *Proc. IEEE Int. Solid-State Circ. Conf.*, San Francisco CA, pp.406-407, Feb. 1997.

[28] P.R.Panda, N.D.Dutt, A.Nicolau, " Memory data organization for improved cache performance in embedded processor applications", *Proc. 9th ACM/IEEE Intnl. Symp. on System-Level Synthesis*, La Jolla CA, pp.90-95, Nov. 1996.

[29] D.A.Patterson et al., "Intelligent RAM (IRAM): chips that remember and compute", *Proc. IEEE Int. Solid-State Circ. Conf.*, San Francisco CA, pp.224-225, Feb. 1997.

[30] P.Pirsch, N.Demassieux, W.Gehrke, "VLSI architectures for video compression - a survey", *Proc. of the IEEE*, invited paper, Vol.83, No.2, pp.220-246, Feb. 1995.

[31] S.Przybylski, "New DRAM architectures", tutorial at *IEEE Int. Solid-State Circ. Conf.*, San Francisco CA, Feb. 1997.

[32] J.Rabaey, "System-level power estimation and optimization - challenges and perspectives", *Proc. IEEE Intnl. Symp. on Low Power Design*, Monterey, pp.158-160, Aug. 1997.

[33] K.Roenner, J.Kneip, "Architecture and applications of the HiPar video signal processor", to appear in *IEEE Trans. on Circuits and Systems for Video Technology*, special issue on "VLSI for video signal processors" (eds. B.Ackland, T.Nishitani, P.Pirsch), 1996.

[34] I.Saeed, "Integrated memory ups speed, saves power", *Electronic Engineering Times*, pp.94, April 28, 1997.

[35] T.Seki, E.Itoh, C.Furukawa, I.Maeno, T.Ozawa, H.Sano, N.Suzuki, "A 6-ns 1-Mb CMOS SRAM with Latched Sense Amplifier", *IEEE J. of Solid-state Circuits*, Vol.SC-28, No.4, pp.478-483, Apr. 1993.

[36] P.Slock, S.Wuytack, F.Catthoor, G.de Jong, "Fast and extensive system-level memory exploration for ATM applications", *Proc. 10th ACM/IEEE Intnl. Symp. on System-Level Synthesis*, Antwerp, Belgium, pp.74-81, Sep. 1997.

[37] T.Sugibayashi et al., "A 30 ns 256 Mb DRAM with a multi-divided array structure", *IEEE J. Solid-state Circ.*, Vol.SC-28, No.11, pp.1092-1096, Nov. 1993.

[38] V.Tiwari, S.Malik, A.Wolfe, "Power analysis of embedded software: a first step towards software power minimization", *Proc. IEEE Int. Conf. Comp. Aided Design*, Santa Clara CA, pp.384-390, Nov. 1994.

[39] I.Verbauwhede, F.Catthoor, J.Vandewalle, H.De Man, "Background memory management for the synthesis of algebraic algorithms on multi-processor DSP chips", *Proc. VLSI'89, Int. Conf. on VLSI*, Munich, Germany, pp.209-218, Aug. 1989.

[40] S.Wuytack, F.Catthoor, L.Nachtergaele, H.De Man, "Power Exploration for Data Dominated Video Applications", *Proc. IEEE Intnl. Symp. on Low Power Design*, Monterey, pp.359-364, Aug. 1996.

[41] T.Yamagata et al., "Circuit design techniques for low-voltage operating and/or giga-scale DRAMs", *Proc. IEEE Int. Solid-State Circ. Conf.*, San Francisco CA, pp.248-249, Feb. 1995.