Towards Energy Efficient Parallel Computing on Consumer Electronic Devices

Karl Fürlinger, Christof Klausecker, and Dieter Kranzlmüller

Department of Computer Science, MNM-Team
Ludwig-Maximilians-Universität (LMU)
München, Germany

Abstract. In the last two decades supercomputers have sustained a remarkable growth in performance that even out-performed the predictions of Moore's law, primarily due to increased levels of parallelism [19]. As industry and academia try to come up with viable approaches for exascale systems, attention turns to energy efficiency as the primary design consideration. At the same time, energy efficiency has always been the main concern in the mobile computing area. Additionally, mobile and consumer electronic devices are becoming ever more powerful as the use cases (e.g., Web 2.0 applications, video encoding, virtual and augmented reality) become more computationally demanding. It is therefore an interesting question to ask if these devices are the possible building blocks of future HPC systems. It was the workstation and server market in the past that provided the CPUs that power supercomputers and it might be the consumer electronic market that provides the underlying technology in the future.

In this paper we try to analyze the current state of energy efficient parallel and distributed computing on mobile and consumer electronic devices. We provide an overview of performance characteristics of some current and announced future devices for scientific computation and we build a small proof-of-concept cluster from Apple's second generation "Apple TV" devices and evaluate its performance on standard benchmark applications. We discuss the limiting factors, and analyze the industry trajectory that we believe could make consumer electronic-based design a feasible technology basis for future HPC system designs.

1 Introduction

Total power consumption and energy efficiency of each component have become important considerations for the design of data-centers and high performance computing facilities. For the next generation of supercomputers (exascale systems), energy efficiency will even become the single most important constraining factor [13]. At the same time, consumer electronic and handheld devices have always been designed with energy efficiency in mind, with efficient ARM-based CPUs powering the majority of devices in the recent history.

Driven by use cases such as HD video streaming, rich Web 2.0 browser applications, virtual reality, and 3D gaming, there are strong market forces driving the future development of theses mobile devices towards more powerful compute capabilities [15]. For example, dual and even quad core mobile CPU designs with GPU integration have been announced for the next generation of tablet computers.

In the light of these developments it is therefore interesting to analyze whether consumer electronic devices could become building blocks of future HPC systems. The supercomputing market is (for the most part) too small to sustain its own CPU ecosystem and in the past has leveraged workstation and server CPUs. In the future, the underlying technology might very well come from consumer electronic space.

With our AppleTV Cluster[1] we try to provide a data point on the current state of energy efficient parallel and distributed computing on ARM powered consumer electronic devices. The second generation Apple TV (ATV2) is meant to be used as a video streaming client and shares most of its hardware internals with the first generation iPad. The Apple A4 processor combines an ARM Cortex-A8 running at 1 GHz with a PowerVR SGX535 GPU and 256 MB RAM. The device is small (3.9 × 3.9 × 0.9 inches / 9.8 × 9.8 × 2.3 cm) relatively inexpensive (about 100 USD) and consumes only about 2-3 Watts). We have built a small (currently containing four nodes) proof-of-concept cluster out of ATV2s and evaluate it with respect to its power and performance characteristics.

The rest of this paper is organized as follows: In Sect. 2 we give a short overview of the current state of the ARM-based computing ecosystem. In Sect. 3 we describe the hardware and software setup of our cluster and in Sect. 4 we evaluate both the single node characteristics as well as the power and performance of the whole system. We discuss related work in Sect. 5 and conclude and provide an outlook on future work in Sect. 6.

2 ARM Computing

ARM holdings is a company headquartered in Cambridge, UK that develops and licenses CPU designs but does not manufacture CPUs themselves. The company was founded in 1990 as a joint venture to continue development of the Acorn RISC Machine (later known as the Advanced Risc Machine), and reports that its technology is used in over one-quarter of all electronic devices [1]. Almost every modern mobile phone is based on ARM architecture CPUs [1] and multi-core mobile devices have started to appear recently [3].

Server solutions based on ARM chips have also recently been announced. The startup company Calxeda has announced plans for a 2U server with 120 Cortex-A9 quad-core chips and a fast interconnect network [11]. Marvell has announced a quad-core server SoC design based on the Cortex-A9 and with support for DDR3 memory and PCI-Express 2.0 interface [4]. Most devices use the ARM Cortex-A8 (single core) or Cortex-A9 (up to quad-core) CPU designs.

[1] `http://www.applevtcluster.com`

3 The AppleTV Cluster

We built a small cluster out of four second generation AppleTV (ATV2) devices. To enable the installation of custom software, it was necessary to perform a *jailbreak*. Once jailbroken, the device comes with a ssh server pre-installed that allows interactive shell access to the BSD-based iOS operating system (iOS 4.2.1, Darwin 10.4.0 kernel). The installation of a fully functional development toolchain (gcc 4.2.1) and editor is relatively straightforward from then on. Please consult our project web page at `http://www.appletvcluster.com` for a more detailed HOWTO guide.

The ATV2 nodes in our cluster are connected to an Ethernet switch. Access to the cluster nodes is organized through a gateway system (a conventional x86-based PC in our case) which is also connected to the switch. To use the cluster to run parallel jobs, we installed an MPI (Message Passing Interface) distribution as the next step. We chose MPICH 2 from Argonne National Lab and configured it using the hydra process manager and TCP transport mechanism.

4 Evaluation

In this section we provide some results of our ongoing evaluation of the characteristics of our ATV2 cluster. We divide the evaluation into a section on single node performance and aspects pertaining to the whole system. Throughout this section we either provide absolute performance numbers and a rough (order of magnitude) comparison to the performance level of contemporary server hardware or compare the ATV2 directly to a popular ARM-based computing platform, the BeagleBoard (`http://www.beagleboard.org`).

We have used a BeagleBoard xM for this comparison which was provided courtesy of collaborators at TU Munich and is also based on the Cortex-A8 CPU. It has an Texas Instruments DM3730 SoC with a 1 GHz[2] Cortex-A8 CPU, 64 KB L1 cache, 256 KB L2 cache, and 512 MB low power DDR RAM. Several benchmarking efforts have previously used the BeagleBoard to compare the Cortex-A8 with with other low-power CPU solutions (cf. Sect. 5), and we analyze the relative performance of the BeagleBoard and the ATV2 in order to make our data comparable to those published results.

4.1 Single Node

Table 1 lists several key characteristics of the ATV2's Apple A4 SoC as reported by the operating system through the iOS "sysctl" command.

To independently confirm these parameters, we have used membench [8]. This benchmark tests the memory system performance by accessing arrays of increasing sizes with varying strides. A detailed discussion of the benchmark is beyond the scope of this paper but the data in Figures 1 and 2 confirms the data in Table 1 and additionally shows that the latency to L1 cache is 1 cycle and the latency to L2 cache is 8 cycles.

[2] Our board was configured to run at 800 MHz instead of 1 GHz.

Property	Value	sysctl entry
L1I cache size	32 KB	hw.l1icachesize
L1D cache size	32 KB	hw.l1dcachesize
L2 cache size	512 KB	hw.l2cachesize
Cacheline size	64 B	hw.cachelinesize
Bus frequency	100 MHz	hw.busfrequency
Memory size	247 MB	hw.memsize
Page size	4KB	hw.pagesize

Table 1. Characteristics of the Apple A4 chip as listed by the "sysctl" command.

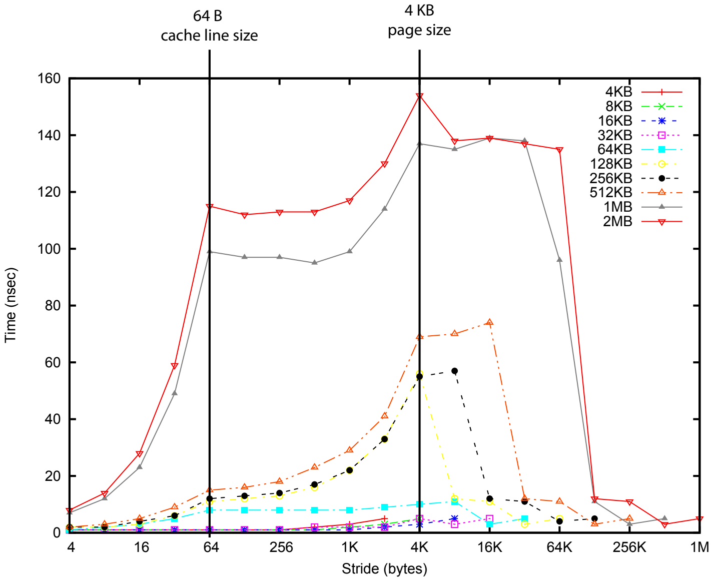

Fig. 1. Membench runs with array sizes ranging from 4 KB to 2 MB.

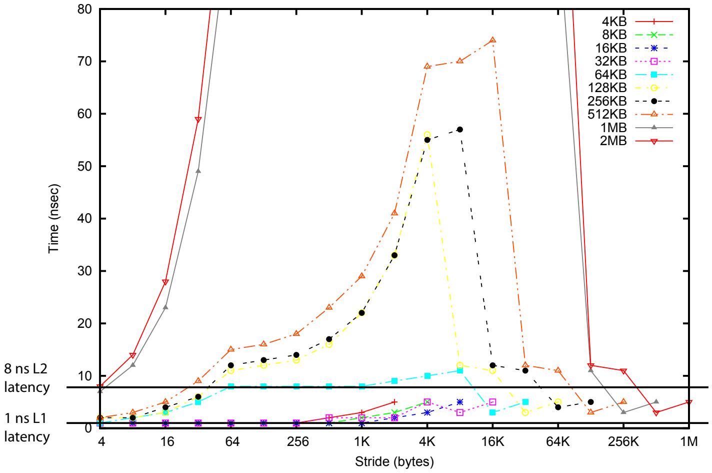

Fig. 2. Detail of the Membench data shown in Fig. 1, with a focus on array sizes from 4 KB to 512 KB.

CPU Performance: We ran the Coremark benchmark[3] to measure the performance of our individual ATV2 nodes. Coremark is an open source benchmark released by the Embedded Microprocessor Benchmark Consortium (EEMBC) with the goal to provide isolated CPU performance tests. Its focus is on benchmarking CPU cores of embedded systems, however, the website also contains results of high-end desktop and server CPUs. Coremark is based on commonly used algorithms and while its isolated approach does not necessarily reflect real applications, the single value results enable to easily compare performance of different processors. Table 2 shows our benchmark results on the BeagleBoard xM and the ATV2. The results show very comparable performance on a per-MHz basis, the small differences are most likely due to different compiler versions and different cache sizes on the two platforms.

By comparison, we also measured an Intel Atom N270 running at 1.6 GHz at 4674 (2.92 per MHz) with two threads and 3029 (1.89 per MHz) with only one thread. Modern sever nodes achieve much higher total Coremark scores (An Intel Xeon L5640 is listed on the Coremark webpage with a score of more than 110000). However, on a per-thread and per-MHz basis, the best coremark score listed is around 4.2, indicating that the Cortex-A8 achieves comparable performance according to this metric.

While Coremark largely only test the integer performance, Linpack is a popular floating point intensive benchmark. Linpack solves a dense linar system of equations using LU factorization. Table 3 shows the results we have obtained by running the (sequential) Linpack benchmark on a single node. Linpack can

[3] `www.coremark.org`

Device	Coremark score	Coremark per MHz
BeagleBoard xM (800 MHz)	1928	2.41
ATV2 (1 GHz)	2316	2.32

Table 2. Coremark benchmark results.

either be run for single (SP) or double precision (DP) floating numbers. In the case of SP, the Cortex-A8 CPU is able to utilize SIMD registers using NEON instructions. NEON registers are 128 bits wide and allow up to four SP floats to be operated on at once, however NEON is not available for double precision. Unfortunately the compiler version we used on the ATV2 is not able to generate NEON code and we where thus not able to improve the Linpack performance past the 57.2 MFlops reported in Table 3. We are still investigating the BeagleBoard's comparatively low performance and the anomaly of faster DP than SP performance.

	BeagleBoard xM (800 MHz)	ATV2 (1 GHz)
Linpack SP (w/o NEON)	22.6	57.5
Linpack SP (w/ NEON)	33.8	–
Linpack DP (w/o NEON)	29.3	40.8

Table 3. Linpack benchmark results.

Memory Subsystem: To test the memory system performance we ran the standard stream [5] benchmark and the results are displayed in Table 4. The data shows the results for four variants and compares the results we obtained on the AppleTV2 with the BeagleBoard xM running at 800 MHz. The ATV2 reportedly uses 200 MHz DDR2 RAM with a 64 bit wide memory bus, which would explain the relatively large performance advantage over the BeagleBoard xM (with 166 MHz and 32 bit memory bus). By comparison, a modern Intel Core i7 CPU with 800 MHz DDR2 RAM delivers more than ten times this bandwidth.

Operation	BeagleBoard xM (800 MHz)	ATV2 (1 GHz)
copy	481.1	749.8
scale	492.9	690.0
add	485.5	874.7
triad	430.0	696.1

Table 4. Results of the Stream benchmark on the ATV2 and the BeagleBoard xM (in MB/s).

4.2 Whole System

MPI Performance: We measured the basic performance characteristics of the interconnect using the OSU microbenchmark suite [16]. Figure 3 shows a plot of the achieved bandwidth and latency. Note that the ATV2 only supports 100MBit Ethernet and that our MPI implementation uses TCP as the transport mechanism. The 100MBit Ethernet is about two orders of magnitude slower than today's modern high performance interconnects (10GigE, Infiniband QDR) and this is reflected in the comparatively poor bandwidth and latency results.

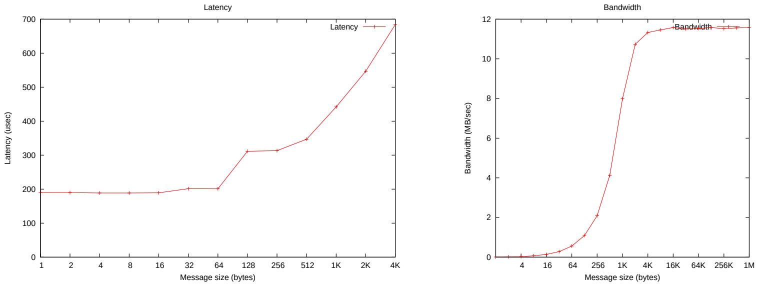

Fig. 3. Latency (half round-trip latency in Microseconds) and Bandwidth (in MB/Second) as measured by the OSU Microbenchmark suite.

High Performance Linpack (HPL) We ran the High Performance Linpack benchmark (HPL) in parallel on all four nodes of the cluster. The best performance we achieved was 160.4 MFlops in double precision arithmetic. Taking into account a power consumption of 10 Watts for the whole cluster (see the next section) this results in a performance of about 16 MFlops/Watt. For comparison, the currently (April 2011) top-ranked system in the Green500 list is a BlueGene/Q prototype with over 1600 MFlops/Watt.

To put this result in perspective one has to keep in mind a couple of things though. First, in double precision the Cortex-A8 does not support SIMD parallelism using NEON and Linpack greatly benefits from wider SIMD width. Experiments [6] showed that doubling the SIMD width in contemporary Intel CPUs from SSE to AVX can improve the performance of Linpack (on a single node) by about 72%. Second, the VFP unit implementing the DP arithmetic is not pipelined and each floating point operation takes a full 9-17 cycles (depending on operation and operands). Assuming an average latency of 15 cycles, the peak double precision floating point rate of the Cortex-A8 would only be 66.7 MFlops (and our Linpack run achieves 40 MFlops).

Third, each ATV2 contains a relatively powerful PowerVR SGX535 GPU that contributes to its power consumption and could improve the floating point performance significantly, if it could be utilized for the computation. Unfortunately, OpenCL programming is not supported on the AppleTV though and so far we found no simple way to exploit the capabilities of the GPU. Lastly, the Green500 list also contains systems that are a lot less energy efficient than the BlueGene/Q prototype. The least energy efficient system (ranked at 500) is a Dell PowerEdge Infiniband Cluster with only 21 MFlops/Watt which is close to the ATV2 cluster result.

Power Consumption: We estimated power consumption using a simple power meter. As the measurement accuracy of our setup is limited, and we can only provide an approximation of the power consumption of the cluster (all four nodes). We tested power consumption with three different baseline loads and our results indicate that all the whole cluster (all four ATV2 nodes, but not including the switch) consume about four Watts when idle and and about eight to ten Watts when fully loaded running the Linpack benchmark.

5 Related Work

Several projects have previously built and benchmarked clusters composed of unconventional nodes, such as the Sony Playstation 3 [10],[17], Set-Top-Boxes [14], and the Microsoft XBox [20].

ARM-powered computer devices have also been benchmarked before. A course paper from the University of Texas [18] contains an architectural and benchmark comparison of the Cortex-A8 CPU and the Intel Atom N330. A detailed comparison of the Cortex-A8 with various x86-based solutions is also presented in an online article [2] by Van Smith. The author finds the ARM based solution to be competitive regarding their integer performance but lacking in terms of floating point and memory performance.

In supercomputing, the successful IBM BlueGene line of systems can be seen as an early example of supercomputer design inspired by embedded processors. BG/L was based on a relatively low performance dual-core PowerPC 440 processor which lead to massively parallel but well balanced systems [7]. BG/P increased the number of cores to four and eased some of the programmability restrictions of the overall system. A prototype of BlueGene/Q is currently (April 2011) the most energy efficient system on the Green500 list [12] with over 1600 MFlops/Watt. While the BlueGene architecture has been specifically optimized for HPC workloads, this is not the case for ARM-based systems. For example there is currently no support for double precision floating point SIMD operations in NEON. Instead, ARM CPUs often come packaged with fairly powerful GPUs.

6 Outlook and Conclusion

In this paper we have presented the hardware and software setup of our AppleTV cluster. The second generation AppleTV (ATV2) is an example of a growing number of consumer electronic devices that combine an ARM CPU and a powerful GPU in a single package. These devices are designed for energy efficiency from the ground up and are increasingly driven by computationally intensive use cases such as video encoding and augmented reality.

While the raw performance and the performance per Watt we have observed with the AppleTV cluster so far is not competitive with conventional desktop CPU (or GPU) solutions, we believe that technology from the consumer electronic sector can play an increasing role in server and high-performance computer installations.

A number of technical issues will have to be addressed for this to happen though. First, NEON has to be extended to fully support double precision floating point arithmetic. While fast single precision floating point hardware has successfully been exploited before [9], most applications require a fully compliant IEEE-754 double precision implementation and a large fraction of the performance of HPC systems can be attributed to wide data parallelism in the form of GPUs or wide SIMD registers. Second, ECC memory is required for any mission critical system, and third, communication intensive applications benefit tremendously from a fast interconnect. Some of these issues are already being addresses in SoC designs such as the Marvell Armada XP [4].

We plan to continue the evaluation of the AppleTV cluster along several directions. First, we would like to conduct a more detailed study of the ATV2 in comparison to the single-core BeagleBoard and the dual-core PandaBoard platforms in order to the evaluate the properties of the ARM multicore approach. Second, a large fraction of the compute power of future consumer electronic devices will come from integrated GPUs and we plan to investigate the capabilities of these GPUs for general purpose computing on the ATV2 and other devices. Finally, the ATV2 is much more competitive in integer than in floating point intensive calculations and we plan to investigate the use its application to non-traditional HPC workloads.

References

1. ARM company profile `http://www.arm.com/about/company-profile/index.php`.
2. The coming war: ARM versus x86 `http://vanshardware.com/2010/08/mirror-the-coming-war-arm-versus-x86/`.
3. LG optimus 2x `http://www.lg.com/global/press-release/article/lg-launches-world-first-and-fastest-dual-core-smartphone.jsp`.
4. Marvell Armada XP multicore series `http://www.marvell.com/products/processors/embedded/armada_xp/`.
5. STREAM: Sustainable memory bandwidth in high performance computers `http://www.cs.virginia.edu/stream/`.
6. Vektorisierungskünstler `https://www.heise.de/artikel-archiv/ct/2011/4/156_Compiler-Intel-Composer-XE-2011-mit-AVX-Optimierung`.
7. NR Adiga et al. An overview of the BlueGene/L supercomputer, 2002.
8. Remzi H. Arpaci, David E. Culler, Arvind Krishnamurthy, Steve G. Steinberg, and Katherine Yelick. Empirical evaluation of the CRAY-T3D: a compiler perspective. *SIGARCH Comput. Archit. News*, 23:320–331, May 1995.

9. Alfredo Buttari, Jack Dongarra, Julie Langou, Julien Langou, Piotr Luszczek, and Jakub Kurzak. Mixed precision iterative refinement techniques for the solution of dense linear systems. *Int. J. High Perform. Comput. Appl.*, 21:457–466, November 2007.
10. Alfredo Buttari, Piotr Luszczek, Jakub Kurzak, Jack Dongarra, and George Bosilca. SCOP3: A rough guide to scientific computing on the PlayStation 3. version 0.1. Technical Report UT-CS-07-595, Innovative Computing Laboratory, University of Tennessee Knoxville, April 2007.
11. Calxeda 5 watt ARM server `http://insidehpc.com/2011/03/14/calxeda-boasts-of-5-watt-arm-server-node/`.
12. The Green500 List, web page: `http://www.green500.org`.
13. Peter M. Kogge et al. Exascale computing study: Technology challenges in achieving exascale systems, 2008. DARPA Information Processing Techniques Office (IPTO) sponsored study.
14. Richard Neill, Alexander Shabarshin, and Luca P. Carloni. A heterogeneous parallel system running OpenMPI on a broadband network of embedded set-top devices. In *Proceedings of the 7th ACM international conference on Computing frontiers*, CF '10, pages 187–196, New York, NY, USA, 2010. ACM.
15. NVIDIA. The benefits of multiple CPU cores in mobile devices (whitepaper). `http://goo.gl/g3MXo`, 2010.
16. OSU Micro-Benchmarks `http://mvapich.cse.ohio-state.edu/benchmarks`.
17. PS3 cluster at NCSU `http://moss.csc.ncsu.edu/~mueller/cluster/ps3/`.
18. Katie Roberts-Hoffman and Pawankumar Hedge. ARM Cortex-A8 vs. Intel Atom: Architectural and benchmark comparisons, 2009.
19. The Top 500 Supercomputer Sites, web page: `http://www.top500.org`.
20. Unmodified Xbox Cluster `http://www.bgfax.com/xbox/home.html`.