Journal of Physics: Conference Series **664** (2015) 092024 doi:10.1088/1742-6596/664/9/092024

Evaluating the power efficiency and performance of multi-core platforms using HEP workloads

P Szostek and V Innocente
CERN, Geneva 23, CH-1211, Switzerland
E-mail: pawel.szostek@cern.ch, vincenzo.innocente@cern.ch

Abstract. As Moore's Law drives the silicon industry towards higher transistor counts, processor designs are becoming more and more complex. The area of development includes core count, execution ports, vector units, uncore architecture and finally instruction sets. This increasing complexity leads us to a place where access to the shared memory is the major limiting factor, resulting in feeding the cores with data a real challenge. On the other hand, the significant focus on power efficiency paves the way for power-aware computing and less complex architectures to data centers. In this paper we try to examine these trends and present results of our experiments with *Haswell-EP* processor family and highly scalable HEP workloads.

1. Introduction

In this paper we assess Intel *Haswell-EP* - a new dual-socket computing platform launched by Intel in September 2014. We perform a comparison in which we include three Haswell-EP processors with varying number of cores: E5-2699v3, E5-2698v3 and E5-2683v3 with 18, 16 and 14 cores respectively. Moreover, we include parts from two previous generations of Intel Processors, namely Sandy Bridge-EP E5-2690 (hereafter abbreviated to "SNB-EP") and its successor, Ivy Bridge-EP E5-2695v2 (abbreviated to "IVB-EP").

Haswell-EP comes with a set of major features on which we wanted to focus: it is the first dual-socket platform implementing support for AVX2 instruction set. It is also the first platform introducing Cluster-on-die, which allows the L3 cache and the main memory attached to this socket to be divided into nodes inside the socket. Last but not least, it offers a wide range of new power-saving features.

2. New architectural features in *Haswell-EP*

With the *Haswell-EP* processors the cores, depending on the core count, can be arranged in three different ways. They can have form of two to four columns of cores split physically

into one (4-8 cores) or two groups (10-18 cores). In the latter case the L3 cache and the main memory attached to a socket might form one or two entities, each of them being a separate NUMA (Non-Uniform Memory Access) node - the latter configuration is called *Cluster-on-die* (hereafter abbreviated as "COD"). These two groups of cores are connected by bidirectional switches to ensure communication between each other. Cluster-on-die, according to Intel, allows to decrease the cache hit latency for a cost of slightly decreased L3 cache hit rate. This feature can be turned on and off at the BIOS level. In our experiments we tried to compare its effectivness by running every benchmark on every SKU (Stock Keeping Unit) with COD being enabled and disabled.

Haswell-EP is the first dual socket CPU providing support for the AVX2 instruction set, which among other things introduces 3-operand fused multiply-accumulate instruction, being a common operation when doing matrix computations. In our tests we try to assess efficiency of a vectorized code with respect to the previously available instruction sets and microarchitectures.

Haswell-EP offers a handful of new power efficiency-related features. Firstly, the frequency and voltage are scaled on a per-core basis, as opposed to a per-socket basis in the previous generations. This feature might come very handy when running AVX code. When an Ivy Bridge or a Haswell CPU runs AVX or AVX2 code, it automatically scales down its core clock to a lower frequency. Thanks to the per-core voltage and frequency regulator, this clock frequency penalty will not propagate to other cores than the one actually executing the AVX code. In addition, the uncore frequency and voltage scaling is independent from the cores. All together the two features allow to better adapt to the kind of workload (CPU-bound or memory-bound) that is being executed.

3. Hardware configuration

In our tests we employed three different SKUs of *Haswell-EP* processors, all being in the higher range of core counts within this family. All the generations of CPUs included in our tests share the same cache configuration: each core has 32kB of each L1 data and instruction caches, 256kB of L2 cache and 2.5MB of shared L3 cache per core. As already explained in section 2, the uncore architecture, i.e. the way that the cores and caches are connected with each other, has been modified in *Haswell-EP*. Table 1 shows essential parameters of the tested CPUs. Thermal Design Power of the processors, which is the maximal power that the cooling system has to dissipate, is abbreviated to TDP.

SNB-EP and IVB-EP processors were fitted on an Intel S2600JF motherboard and *Haswell-EP* parts were mounted on Intel S2600KP. Every machine was equipped with 8 DIMMs of 8GB main memory, DDR3 for SNB-EP and IVB-EP, DDR4 for *Haswell-EP*. Each system was fitted with two Intel Solid State Drives DC S3500 240GB, configured with LVM stripping. The machines had TurboBoost disabled, Simultaneous Multi-Threading (called SMT hereafter) enabled, P- and C-states were disabled. The operating system chosen for our tests was Scientific Linux CERN 6.6 with the 2.6.32-504.12.2 kernel. This system might be considered outdated, but it is still run in the CERN Data Centre and in most sites of the WLCG (Worlwide LHC Computing Grid). Since in our paper we focus on the performance of HEP workloads, we decided to stick to an environment which they most likely would use. Considering that it provides support for NUMA, we don't expect the most recent release of SLC (CERN CentOS 7) to change performance numbers

SKU	platform	cores	frequency	L3 cache	TDP	Feature size
E5-2690	*Sandy Bridge-EP*	8	2.9GHz	20MB	135W	32nm
E5-2695v2	*Ivy Bridge-EP*	12	2.4GHz	30MB	115W	22nm
E5-2683v3	*Haswell-EP*	14	2.0GHz	35MB	120W	22nm
E5-2698v3	*Haswell-EP*	16	2.3GHz	40MB	135W	22nm
E5-2699v3	*Haswell-EP*	18	2.3GHz	45MB	145W	22nm

Table 1: Parameters of the Intel processors involved in the tests. In *Haswell-EP* number of cores was doubled with respect to Sandy Bridge-EP, while the TDP is maintained at the same level.

importantly.

4. HEP-SPEC06

4.1. Description of the benchmark

HEP-SPEC06 has been chosen by a work group affiliated by HEPiX as a standard HEP benchmarks. It is a set of single-threaded scientific real-life applications available on a commercial basis. Since it has proven to correlate significantly with HEP applications, the community decided to adopt it as a reference benchmarks. It is widely use to define pledges at data centres worlwide.

There are several factors that make us consider this benchmark outdated and not suitable for testing modern mulit-core and vector hardware: since the workloads are single-threaded, in order to fully utlize many cores one has to run multiple workload instances in parallel. Furthermore, the workloads are not vectorized and show poor scalability. Nevertheless we include their results, but they should not be considered as the main metrics to assess performance of the hardware.

The benchmark was compiled with gcc 4.4.7 and was run using the standard runspec command. This ancient compiler was chosen following the recommendations for using HEPSPEC06, as published by the HEPiX Benchmarking Working Group.

4.2. Results

Below we present scalability results of HEP-SPEC06 (thereafter abbreviated as HS06). Scores returned by HS06 can be interpreted as a speedup obtained with respect to a reference run, therefore they are inversely linear to the actual exection time. In the figure 1 the are HS06 scores for the tested platforms. These results are not scaled and show the actual performance of every system. The figure 2 presents HS06 results scaled down to a common frequency of 2.7GHz and divided by the number of workload instances run at a time. These results can be interpreted as the input of every loaded core to the total score obtained with the given number of parallel instances. In the figure we visualize results obtained for HS06 by turning the COD feature on and off. On every tested platform COD provided extra performance: 7.82%, 7.27% and 10.87% for E5-2683v3, E5-2698v3 and E5-2699v3 respectively.

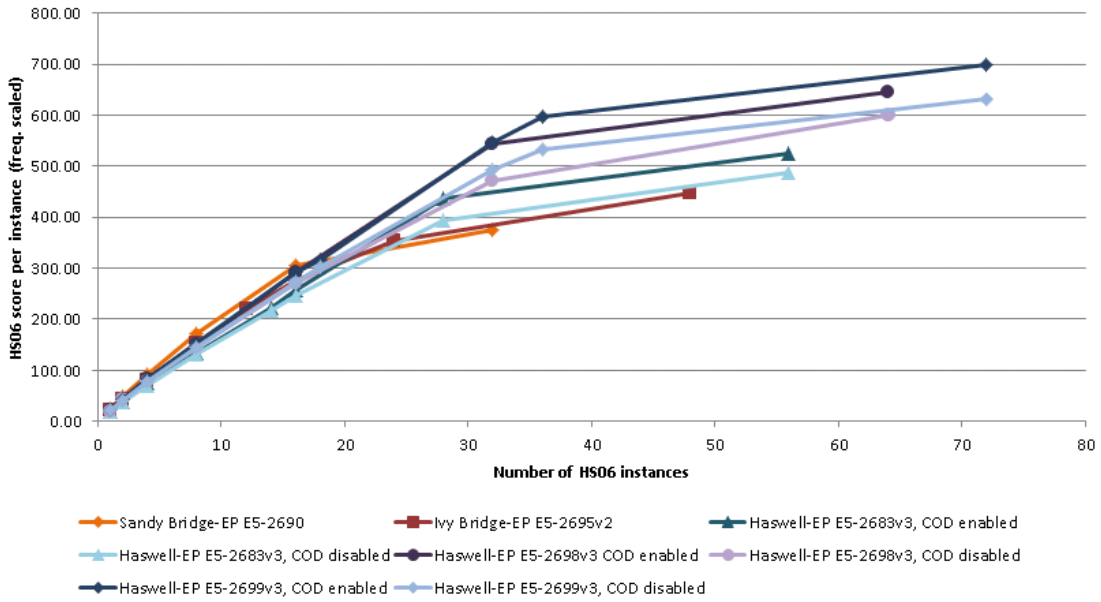

Figure 1: HEP-SPEC06 scalability

	COD disabled	COD enabled	COD gain
Haswell-EP E5-2683v3	486.72	524.78	7.82%
Haswell-EP E5-2698v3	600.66	644.30	7.27%
Haswell-EP E5-2699v3	631.57	700.23	10.87%

Table 2: HEP-SPEC06 unscaled total scores and performance gains with COD disabled/enabled

5. ParFullCMS with Multi-threaded Geant4

5.1. Description of the benchmark

Geant4 ([1], [2]) is the major toolkit used in the simulation of the detectors in Large Hadron Collider (LHC). It is a principal representative of the workloads run in the WLCG and constitutes a significant portion of its CPU time. While being multi-threaded, it can exploit multi-threaded parallelism in event processing - once the geometry and the processes are initialized, threads are spawned and events are simulated in parallel. In order to reduce memory footprint, the read-only memory chunks are shared among threads. The version employed in the tests was `geant4-10-01-patch-01`, released on 27.03.2015, built with the default `-O2` optimization flag.

ParFullCMS is a benchmark built on the top of the Geant4 library. It is a stand-alone application implementing a realistic geometry of the CMS detector with full physics simulation, but a simplified setup for data collection, and operating a uniform magnetic-field. It simulates the full geometry of the CMS detector at LHC. To run it, the default

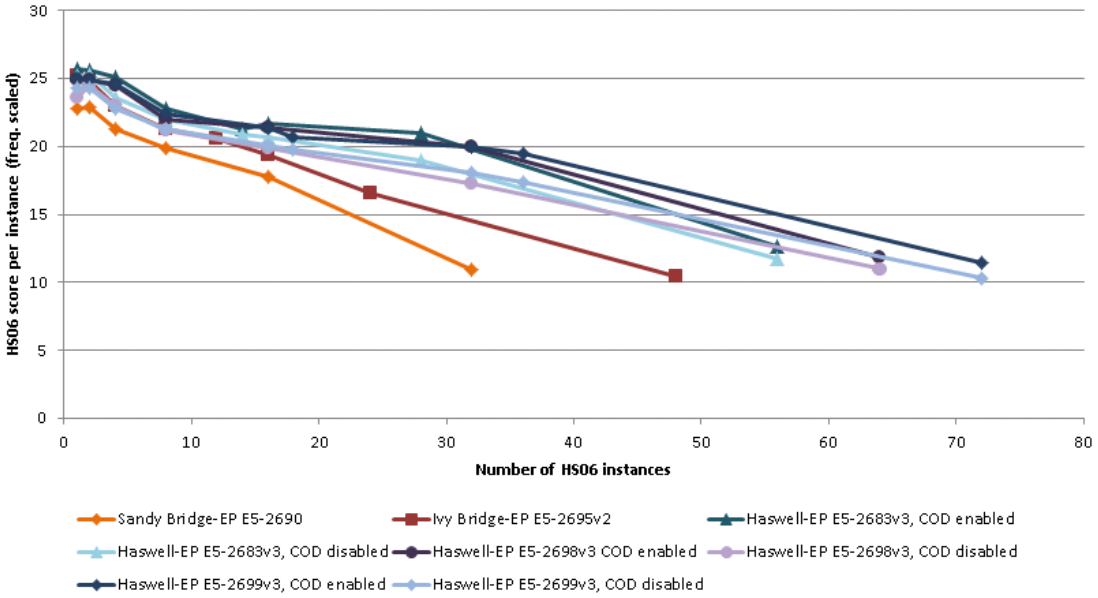

Figure 2: HEP-SPEC06 single core performance scalability

physics list QGSP_BERT was used. The benchmark was built using gcc 4.9.2 with the default compilation flags.

5.2. Description of the experiment

The ParFullCMS benchmark was used to measure data throughput scalability (i.e. events per second) and power efficiency scalability (i.e. events per joule). In order to collect respective numbers we attached the machines to a power meter before then power suppply, i.e. the measurements include the power consumed by the power supply itself, and ran ParFullCMS with varying number of threads. The application was run in a weak scaling fashion, i.e. while the number of loaded cores was being changed, the workload per core was kept constant making the problem size linear to the number of cores used. At the end of each run the total power consumption over the running time was read out from the device.

5.3. Results

Figure 3 shows throughput scalability of the tested platforms. The tested machines provide very similar scalability, with the input of HSW-EP cores being slightly bigger than for their predecessors. When moving to the SMT SNB-EP provides 25% extra total throughput. IVB-EP and HSW-EP platforms offer very similar gains, being equal to 30%, 29%, 32% and 32% for E5-2695v2, E5-2683v3, E5-2698v3 and E5-2699v3 respectively. Figure 4 shows the power efficiency scalability for the tested platforms, i.e. how many events can be processed per joule with a given number of threads running in parallel. This measurement proves significant in power efficiency in *Haswell-EP* platform. While IVB-EP was able to process

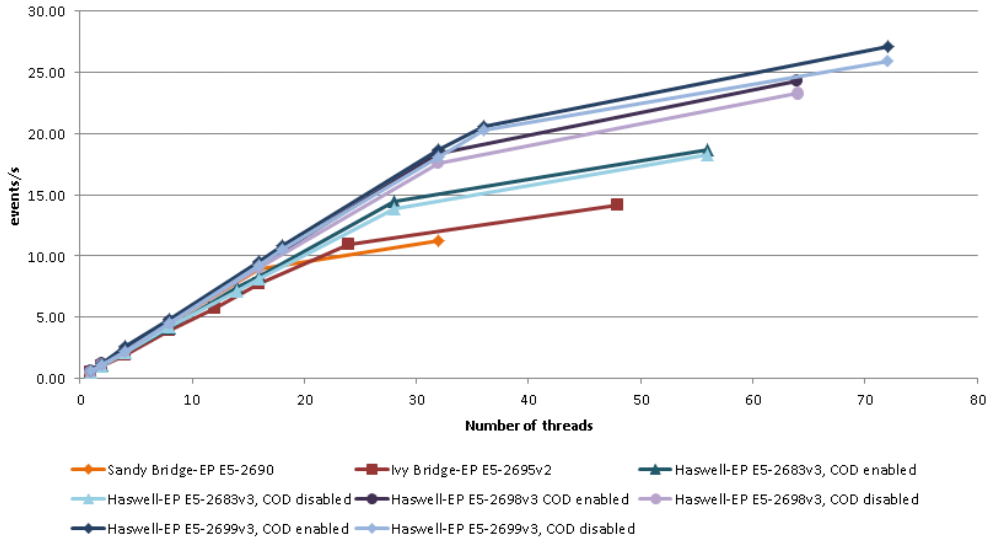

Figure 3: ParFullCMS throughput scalability

46% more events per the same amount of energy, *Haswell-EP* is capable of processing up to 102% more events than IVB-EP and 196% more than SNB-EP per unit of energy.

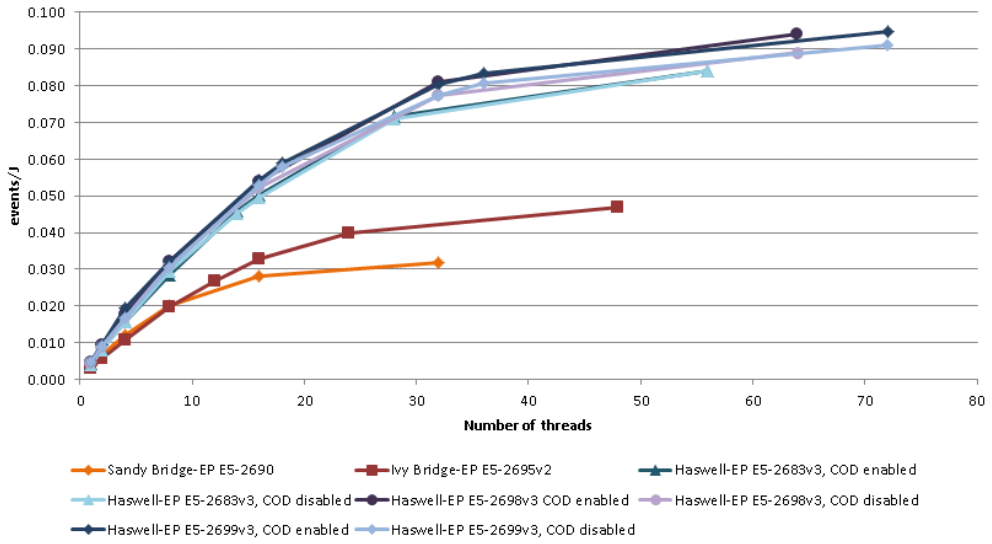

Figure 4: ParFullCMS power efficiency scalability

6. VIFit

6.1. Description of the benchmark

VIFit is an evolved version of MLFit [3]. It is a multi-threaded and fully vectorized benchmark, performing a fitting procedure in order to adjust PDF (Probability Density Function) parameters to best reproduce data from the experiment numerical data. To ensure stability of the results VIFit does not perform full minimization - instead the objective function and its gradient is evaluated for a fixed number of points and iterations. It was used to test vectorization performance of the tested plaforms.

VIFit introduces a handful of improvements over its predecessor: runtime is limited to the order of a few seconds, its profile is not dominated by the exp function, the allocated memory can be split into chunks, whose number is defined as a command line parameter, making it NUMA and COD-friendly. Morevore, its performance is intentionally biased by the memory subsystem efficiency. The benchmark was compiled with gcc 4.9.2 using the following compilation flags: `-funroll-loops`, `-finline-functions` , `-Ofast`, `-ftree-loop-if-convert-stores`, `-fvisibility-inlines-hidden`, `-std=gnu++11`, `-ftree-vectorizer-verbose=1`, `-fopt-info-vec`, `-fipa-pta`. On the top of them we used various vectorization flags in order to produce machine code suitable for different instruction set extenstions: `-mavx2`, `-mavx`, `-msse4.2`, `-fno-tree-vectorize` targeting AVX2, AVX, SSE4.2 and no vectorization, respectively.

The benchmark was run by using as many threads as hardware threads (with SMT) available. The threads were pinned down to the cores.

6.2. Results

In the figure 5 vectorization speedup obtained with various instruction sets is presented. For each platform non vectorized variant is the baseline. *Haswell-EP* servers provide up to 2x speedup with AVX2 and 1.7x with AVX instruction sets. For every SKU the speedup was higher for COD enabled by 2-2.4%. The figure 6 shows speedups of all the tested platforms with respect to SNB-EP, calculated by scaling the run time proportionally to the clock frequency. While the IVB-EP provided 1.5x more performance than SNB-EP (while having 1.5x more cores), the HSW-EP platforms provide 2.91x, 2.99x and 3.14x for E5-2683v3 (1.75x more cores), E5-2698v3 (2x more cores) and E5-2699v3 (2.25x more cores) respectively. This means that, opposed to IVB-EP, we see an additional speedup that is an effect of not only core count increase, but also improvements in the core and uncore architecture.

7. Conclusions

The *Haswell-EP* platform provides an across-the-board improvement in performance with respect to the previous generations. The overall performance adventage of an 18-core HSW-EP (E5-2699v3) over a 12-core Ivy Bridge-EP (E5-2695v2) is established at 1.57x speedup for HEP-SPEC06 benchmark, 1.92x better throughput and 2.02x and 2.07x better power efficiency for ParFullCMS and VIFit respectively. *Haswell-EP* is a platform offering a massive core-count increase with respect to its predecessor, Ivy Bridge-EP. While it maintains the power consumption at a level comparable to IVB-EP, the number of available cores grew by 50%. We did not see a significant improvement in the single core performance with a legacy single-threaded and not vectorized software. When running HS06 in a

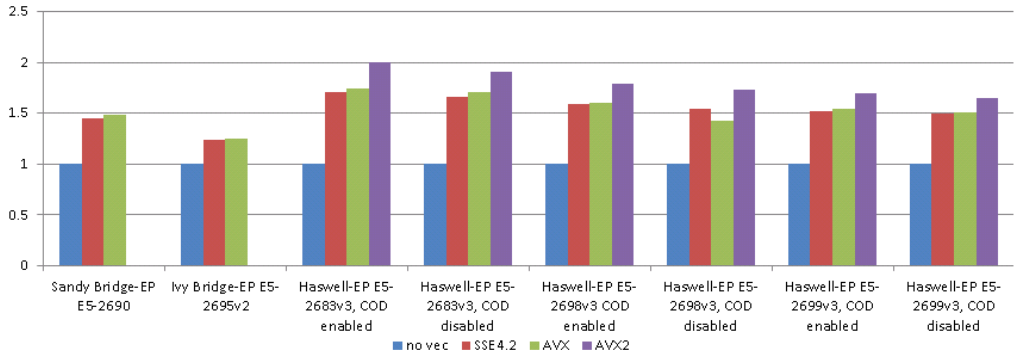

Figure 5: Speedups obtained with various target instruction sets. Not vectorized run time was used as a baseline for each platform.

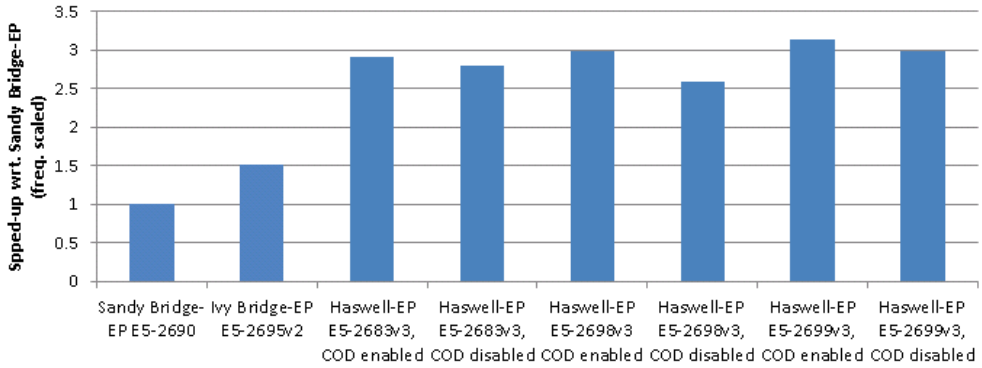

Figure 6: Speedups obtained with respect to Sandy Bridge-EP. Runtimes were scaled proportionally to the clock frequencies.

weak scaling test we were to achieve higher throughput - proportional to the increase in number of cores. *Haswell-EP* shows important improvement in the power efficiency. Our experiments with Geant4 and ParFullCMS showed that it can process more than twice more events per unit of energy compared to IVB-EP. With *Haswell-EP* the highest total speedup obtained in the tests was achieved for VIFit, which is a multi-threaded and highly vectorized workload. *Haswell-EP* also provides higher vectorization gains than previous dual-socket platforms for the same instruction sets.

References

[1] Agostinelli S et al. 2003 Geant4 - a simulation toolkit *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors* **506** 250-303
[2] Dong X, Cooperman G and Apostolakis J 2010 Multithreaded Geant4: Semi-automatic Transformation into Scalable Thread-Parallel *Lecture Notes in Computer Science* **6272** 287-303
[3] Lazzaro A, Jarp S, Leduc J, Nowak A and Valsan L 2012 Report on the parallelization of the MLfit benchmark using OpenMP and MPI *CERN openlab report*
[4] Cosmo G 2015 private communication